Bioinformatique et analyse de données multiomiques : principes et applications chez les levures pathogènes Candida glabrata et Candida albicans

Thomas Denecker

université PARIS-SACLAY

Bioinformatique et analyse de données multiomiques : principes et applications chez les levures pathogènes *Candida glabrata* et *Candida albicans*

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°577
Structure et dynamique des systèmes vivants (SDSV)
Spécialité de doctorat : sciences de la vie et de la santé
Unité de recherche : Université Paris-Saclay, CEA, CNRS, Institute for Integrative Biology of the Cell (I2BC), 91198, Gif-sur-Yvette, France
Référent : Faculté des sciences d'Orsay

Thèse présentée et soutenue à Orsay, le 16/09/2020, par

Thomas DENECKER

Composition du Jury

Sarah COHEN BOULAKIA Professeure, LRI, Paris Saclay	Présidente
Bertrand COSSON Professeur, Paris Diderot	Rapporteur & Examinateur
Marie-Agnès DILLIES Ingénieure de recherche, Pasteur	Rapportrice & Examinatrice
Stéphane LE CROM Professeur, Sorbonne université	Rapporteur & Examinateur
Hélène CHIAPELLO Ingénieure de recherche, INRAE	Examinatrice
Gaëlle LELANDAIS Professeure, I2BC, Paris Saclay	Directrice de thèse
Jean-Michel CAMADRO Directeur de recherche, IJM	Invité

Thèse de doctorat

NNT : 2020UPASL010

Remerciements

Chères lectrices, chers lecteurs,

Je vous remercie d'avoir ouvert cette thèse et de prendre le temps de la lire. Si aujourd'hui, vous pouvez l'avoir entre les mains, c'est grâce à une multitude de personnes exceptionnelles qui m'ont accompagnées pendant cette aventure et je tiens à les remercier très chaleureusement dans les prochains paragraphes.

Aux membres de mon jury

Sarah Cohen Boulakia, Bertrand Cosson, Marie-Agnès Dillies, Stéphane Le Crom, Hélène Chiapello, je vous remercie d'avoir pris le temps de lire mon manuscrit et de me conseiller. Vous avez toujours été très bienveillants et ça toujours été un plaisir d'échanger avec vous.

Au Professeur Gaëlle Lelandais, "ma Boss et mon mentor"

Il arrive parfois de rencontrer des personnes qui vous donne envie de vous dépasser et donner le meilleur de vous-même et tu fais partie de ces personnes. Je me souviens parfaitement de notre première rencontre dans une salle de TD à la Halle au Farine. Tu m'as fait découvrir les statistiques, j'étais complètement perdu et pourtant tu n'as pas baissé les bras ! Depuis, nous ne nous sommes plus quittés. Tu as cru en moi, tu as pris des risques pour moi et tu as toujours su trouver les mots justes en toutes circonstances. Je te remercie infiniment pour ces 3 merveilleuses années de thèse et pour toutes ces émotions ! #FineEquipe !

À l'équipe Camadro, "la dream team de la spectrométrie"

Françoise Auchere, Jean-Michel Camadro, Véronique Legros, Emmanuel Lesuisse, Laurent Lignières, Pierre Poulain, Nicolas Senecaut, Samuel Terrier et Valérie Serre, je vous remercie pour votre accueil. J'ai passé une 3ème année de thèse passionnante auprès de vous. Mon cerveau a souvent été K.O. avec toutes ces nouvelles notions mais vous avez toujours été patients et compréhensifs. Jean-Michel, je te remercie de m'avoir apporté des fondations solides en protéomique. Tu as toujours trouvé du temps pour répondre à mes interrogations et ça toujours été très agréable d'échanger avec toi. Samuel, nous avons bien fait de discuter de MoNET qui n'aurait pas vu le jour sans ton impulsion ! Tu as été le déclic qui a lancé un projet depuis longtemps dans les cartons ! Et un grand merci à toi ainsi qu'à Véronique et Laurent

pour tout l'aspect technique expérimentale. Pierre, j'espère que nous aurons encore l'occasion de nous faire nos "salons de thé / discussions de recherche" et un grand merci pour tous tes conseils ces dernières années !

À l'équipe Cadoret

L'équipe qui m'a vu faire mes premiers pas en bioinformatique ! Je vous remercie pour cette superbe collaboration jalonnée de belles réussites ! Je vous remercie aussi pour toutes les découvertes culinaires (le stollen, Gianduja, les Tortinas,...) et toutes nos discussions Top chef qui donnent faim dès 9h ! Jean-Charles, je te remercie de m'avoir donné ma chance (et promis, cette fois-ci j'ai compris, la réponse n'est pas le séquençage !). Je vous remercie Giuseppe de m'avoir accueilli dans votre équipe. Les droïdes protocolaires, quel plaisir de travailler au quotidien avec vous ! Djihad, un grand merci pour tes « et tu penses qu'on pourrait faire ça ? » qui ont poussé START-R à toujours aller plus loin. Enfin, je remercie Su-Jung et Anne-Lise pour leur gentillesse. Vous garderez toujours une place spéciale.

À l'équipe Fairhead

Nous l'avons fait ! Quand, j'ai commencé ma thèse nous rêvions d'un NAR et nous l'avons fait ! Un grand merci pour cette émotion ! Je te remercie Cécile de m'avoir prêté un bureau pour travailler avec vous, merci Adela de m'avoir replongé dans le monde médical et m'avoir permis de donner une autre dimension au projet, merci Monique d'avoir partagé avec moi ta vision globale et transverse de la biologie, merci Youfang pour tes conseils et Laetitia pour toutes nos discussions de société.

À l'équipe Lespinet

Je vous remercie pour votre accueil et vos conseils tout au long de ma thèse. Je te remercie Olivier de m'avoir accueilli dans ton équipe, Mélina et Marie-Hélène pour nos discussions statistiques, les thésards de l'équipe (Christos, Hugo, Jean-Noël et Nicolas) toujours de bonne humeur et solidaires dans l'épreuve qu'est la thèse et plus généralement tous les membres de l'équipe qui ont été de très bons conseils notamment lors de la préparation du concours de l'école doctorale.

À l'équipe Malagnac

Même si nous n'avons pas eu l'occasion de beaucoup travailler ensemble, vous avez toujours été à l'écoute et intéressés par ce que je faisais. Je vous remercie d'avoir toujours cherché à comprendre ce qui m'a permis d'améliorer ma façon de présenter mes projets.

Et aussi

À Claire Toffano-Nioche, tu te souviens de ce déjeuner de Noël, où je t'ai dit « Viens, on crée une formation sur la reproductibilité » ? Même si ça semblait fou, tu m'as répondu « Banco ! On y va ! ». 1 an et demi plus tard, notre formation a été reprise par l'Institut Français de Bioinformatique ! Que de chemin parcouru ! Un grand merci d'avoir cru en mon idée et de m'avoir toujours poussé plus loin. Ce partenariat a été pour moi une super expérience ! Un grand merci pour nos nombreuses discussions, pour tous tes tests et ton implication pour défendre notre projet !

À Chrystelle, tu as toujours été là dans les bons moments comme dans les mauvais. J'ai toujours trouvé en toi une oreille attentive et attentionnée. Merci pour ta joie, tes conseils et tes nombreuses découvertes culinaires (les Tortinas en tête !).

À Stéphane, pour nos discussions autour d'un café ! Merci pour tous les films et séries que tu m'as fait découvrir, pour l'initiation à la magie, pour nos nombreux débats, tous ces bons chocolats et cette incroyable motivation que tu m'as donnée dans le sport ! A nous, le *summer body* toute l'année !

À Hadrien, pour ton soutien lors du concours de l'école doctorale et les moments difficiles qui ont entouré mon début de thèse.

Aux Marines (et leurs compagnons **Pierre** et **Joris**), à **Armelle**, à **Emilie**, à **Laure-Hélène** et **Jérémy** pour les merveilleux moments d'évasions que j'ai pu passer avec vous.

À la team DRC23, à toutes les émotions que nous avons pu avoir les Juliens et Erwan ! Quand j'y repense j'ai encore des palpitations ! « Je suis trop vieux pour ces conneries ! ».

À mes grands-parents qui ont toujours été là pour moi. A nos parties de tarots, à la bûche et tous les merveilleux moments passés ensemble. Merci pour tout.

À mes frères, Jérôme, Jean-Yves et Tom et à ma sœur, Morgane. Je vous remercie pour tes ces bons moments passés ensembles et les nombreux à venir ! Trop peu souvent mais tellement précieux !

À Monica, ma "belle-mère", mais qui en réalité bien plus que ça. Même avec des simples constructions en Kapla que nous réalisions avec Tom, nous avions l'impression d'avoir construit le Tour Eiffel ! Merci pour tous tes encouragements, nos discussions et tes conseils depuis toutes ces années.

À la famille Schmitt, que de bons moments passés avec vous enrichissants sur le plan intellectuel et humain, à nos parenthèses bucoliques, aux magnifiques découvertes. Un grand merci Olivier pour nos superbes parties d'échecs, d'avoir partagé ta passion pour la magie et de m'avoir fait découvrir le yoga.

À ma famille d'adoption, les Chasport et sa valeur ajoutée : Alain, les Juliens, Jacqueline, Marie-Claude et Marion. J'ai passé presque la moitié de ma vie à vos côtés et j'ai toujours eu l'impression d'être un des vôtres. Vous m'avez vu et fait grandir, vous m'avez fait découvrir de nouveaux horizons, vous avez été là dans les grandes étapes de ma vie et toujours d'un soutien indéfectible. Bisous cœur !

À mon père, un homme, un ami et père formidable. Que de bons souvenirs avec toi. Tu as toujours cru en moi et tu m'as toujours soutenu. Je te serai reconnaissant pour l'homme que je suis devenu grâce à toi. J'aurai tellement adoré parler de ce manuscrit avec toi… Merci pour tout papou.

À ma mère, « Houston, ici la lune ! » Merci de m'avoir transmis tes valeurs, ta détermination et ta ténacité qui m'ont été d'une aide précieuse pendant cette thèse. Merci pour tes encouragements et ta confiance. Tu es toujours à l'écoute et disponible pour une franche rigolade. Et pour tout le reste, tu le sais déjà !

À Nanou, mon grand amour, ma meilleure amie. Nous nous sommes construits ensemble et aujourd'hui, si j'en suis là, c'est grâce à toi. Et oui, nous y sommes ! Depuis tant d'années que nous plaisantons sur les Dr Nanous. Nous pouvons enfin le dire ! Tu as toujours été une très grande source d'inspiration, une très grande force. Tu es la seule qui sait m'apaiser dans les moments de doutes et de stress par ta gentillesse et ta patience. Je suis tellement fier et comblé de t'avoir à mes côtés. Je t'aime.

Table des matières

Remerciements 3
Table des matières 7
Liste des abréviations 11
Liste des figures 13
Liste des tableaux 26
Avant - propos 28
Introduction générale 32
I. Les fondements de la bioinformatique 33
1. Les définitions de la bioinformatique 33
2. Les usages de la bioinformatique 37
3. L'importance de la mutualisation des savoir-faire et des ressources 41
II. Les fondements de l'analyse de données 42
1. L'importance de différencier « donnée », « information » et « connaissance » 42
2. L'influence de la visualisation : le couplage œil – cerveau 47
3. Les différentes étapes d'une analyse de données 56
III. Les spécificités de l'analyse de données multi-omiques 62
1. À l'ère de la génomique fonctionnelle et de la biologie des systèmes 62
2. Un besoin de reproductibilité 68
3. Un besoin de statistiques 76
4. La bioinformatique face à des nouveaux défis 82
5. Pour conclure, sortir du « cloud mental » avec agilité 91
Contributions aux efforts mutualisés en bioinformatique : développements de logiciels et formations des chercheurs en biologie 94
I. L'application Web « bPeaks App » pour l'analyse de données ChIPseq 95
1. Le contexte du projet 95
2. Le logiciel bPeaks 95
3. Le logiciel bPeaks App 95

4. Le PDF de la publication dans la revue « BMC Research Notes » (Denecker et al. 2018) 96

II. L'application Web « Pixel » pour l'annotation, le partage et l'exploration de données multi-omiques 97

1. Le contexte du projet 97

2. Un développement en deux temps 97

3. Les logiciels Pixel et Pixel2 97

4. Le PDF de la publication dans la revue « PeerJ » (Denecker et al. 2019) 98

III. Le partage d'expérience pour aider à la création d'applications WEB avec Shiny 99

1. Une rencontre sur les réseaux sociaux 99

2. Un projet pour débuter 99

3. Le PDF de l'article publié sur le site Internet « Bioinfo-fr.net » (Denecker, 2019) 100

IV. L'application Web « MONet » pour l'intégration et la visualisation de réseaux multi-omiques 101

1. Le contexte du projet 101

2. Le logiciel MONet 101

3. La présentation détaillée (travail non publié) 101

V. La formation « FAIR_Bionfo » pour l'apprentissage des pratiques informatiques qui soutiennent la reproductibilité des résultats 113

1. Le contexte du projet 113

2. La documentation pédagogique 113

3. Le futur de la formation FAIR_Bioinfo 116

4. PDF de l'article déposé dans HAL (Denecker et al. 2020) 118

Contributions aux projets d'analyse de données multi-omiques : génomique fonctionnelle des levures pathogènes *Candida glabrata* et *Candida albicans* 119

I. Les levures pathogènes *Candida glabrata* et *Candida albicans* 120

1. Des risques importants pour la santé publique 120

2. Des spécificités importantes, malgré l'appellation partagée de levures « *Candida* » 122

3. Ce qu'il faut retenir 127

II. Étude transcriptomique pour l'exploration des mécanismes d'homéostasie du fer chez *Candida glabrata* 128

1. Le contexte de l'étude 128

2. La mise en application de la technologie des puces à ADN 139

3. Le PDF de l'article publié dans la revue « *NAR Genomics and Bioinformatics* » 149

4. Pour aller plus loin : ouverture sur les levures du clade des *Nakaseomyces* 150

5. Pour conclure, de nouvelles stratégies thérapeutiques impliquant le fer ? 158

III. Étude systématique des modifications post-traductionnelles des protéines chez *Candida albicans* 160

1. L'introduction générale 160

2. Le matériel et les méthodes 166

3. Les résultats 173

4. Les conclusions et perspectives 186

IV. Contributions à d'autres projets 193

1. Étude de la reproductibilité des résultats obtenus par spectrométrie de masse 193

2. Collaboration avec l'équipe de Jean-Charles Cadoret : le logiciel START-R 194

3. Collaboration avec l'équipe de Cécile Fairhead : caractérisation des orthologues au sein du clade des *Nakaseomyces* 196

Bilan de thèse et Conclusion 205

I. Ce que nous avions prévu 206

II. Ce que nous avons fait 209

III. Ce que nous aurions pu ou pourrions faire 212

1. Le big data en biologie 212

2. L'explosion de la « data science » 220
3. L'intelligence artificielle a-t-elle une place en biologie ? 221
Annexes 228
I. Les aperçus des différents formats évoqués dans cette thèse 229
1. BAM 229
2. BED 229
3. FASTA 230
4. FASTQ 230
5. GFF 231
6. HTML 232
7. idXML 233
8. JSON 233
9. mzId (ou mzIdentML) 234
10. mzML 235
11. pepXML 235
12. PDF 236
13. RAW 236
14. SAM 238
15. XML 238
II. Recette du fameux gâteau au chocolat 240
1. Liste des réactifs 240
2. Le protocole expérimental 240
III. Le cadre de travail 242
Références bibliographiques 243

Liste des abréviations

A	
ADN	Acide Désoxyribonucléique
ANR	Agence Nationale de la Recherche
API	*Application programming interface*
ARN	Acide Ribonucléique
C	
C. albicans	*Candida albicans*
C. bracarensis	*Candida bracarensis*
CGD	*Candida Genome Database*
C. glabrata	*Candida glabrata*
C. nivariensis	*Candida nivariensis*
I	
IA	Intelligence Artificielle
	Intelligence Augmentée
iHKG	*iron Homeostasis Key Genes*
IP	Immuno-Précipitation
	Intellectual Property
	Intervention Pharmaceutique
	Internet Protocol
G	
GRYC	*Genome Resources for Yeast Chromosomes*
L	
LC-MS/MS	*Liquid Chromatography - Mass Spectrometry / Mass Spectrometry*

N	
N. delphensis	*Nakaseomyces delphensis*
NGS	*Next Generation sequencing*
O	
OS	*Operating System*
P	
PCR	*Polymerase Chain Reaction*
S	
S. cerevisiae	*Saccharomyces cerevisiae*
SGD	*Saccharomyces Genome Database*

Liste des figures

Figure 1 – Nuages des mots utilisés dans les titres des articles publiés sur une période de cinq années (2000 – 2005 à gauche et 2015 – 2020 à droite) dans les revues de Bioinformatique : Bioinformatics, BMC Bioinformatics, Brefings in Bioinformatics et Journal of bioinformatics and computational biology. Plus la taille du mot est grande, plus son utilisation est fréquente. La période 2000 – 2005 est la période de la thèse de Gaëlle Lelandais tandis que la période 2015 – 2020 est la période de la thèse de Thomas Denecker. Cette idée de représentation est inspirée d'un séminaire de recherche donné par Jean-Michel Camadro à l'IJM en Février 2019. La figure a été réalisée avec l'outil WEB « wordart.com »..... 30

Figure 2 – Carte conceptuelle réalisée en 2016 pour clarifier les concepts en relation avec la « Modélisation des systèmes biologiques » (encadré en violet). Quatre notions majeures sont encadrées en rouge : « Expérimentation in silico », « Prédiction », « Synthèse des connaissances » et « Fouille de données ». Les couleurs regroupent des concepts similaires ou fortement reliés. Cette carte a été réalisée avec le logiciel Xmind8F..... 36

Figure 3 – Évolution des prix du séquençage d'un génome humain. À partir de 2008, les coûts ont été très fortement réduits. Cela correspond à l'arrivée des technologies des séquençages hautement parallélisées. Les séquences sont plus courtes, mais plus nombreuses. La source de ces données est Wetterstrand KA. DNA Sequencing Costs: Data provenant du NHGRI Genome Sequencing Program (GSP). Ces données sont mises en regard de la loi de Moore (ligne bleue) qui décrit une tendance en informatique impliquant le doublement de la « puissance de calcul » tous les deux ans. Les améliorations technologiques qui suivent la loi de Moore sont considérées comme très performantes. Ici, nous observons une évolution encore plus rapide. 38

Figure 4 – Principaux domaines d'application de la bioinformatique. Les trois grands domaines abordés lors de cette thèse sont indiqués en rouge : « Études omiques », « Biologie des systèmes » et « Annotation fonctionnelle ». Cette figure est inspirée de l'article de M. Bilotta (Bilotta et al. 2018)..... 40

Figure 5 – Représentation pyramidale du modèle DIC inspiré de l'article de J. David (David 2019). Les données sont associées aux observations, les informations résultent des analyses des données et enfin les connaissances émergent des informations. Pour cela il est nécessaire de

prendre en compte un contexte aussi vaste que possible (problématique de l'intégration d'informations issues de données hétérogènes). 47

Figure 6 – Nombre d'importations et d'exportations en Angleterre au XVIII^{e} siècle. Cette figure est extraite du livre de William Playfair publié en 1801. Elle permet, très simplement, d'observer la corrélation positive entre ces deux variables, ainsi que leur augmentation au cours du temps. Cette image est librement accessible en ligne24F 48

Figure 7 – Les premières infographies de l'histoire apparues successivement au XVII^{e} siècle. (A) Le diagramme de Coxcomb de Florence Nightingale représentant les pertes britanniques pendant la guerre de Crimée (1858). (B) La carte figurative des pertes successives en hommes de l'armée française lors de la campagne de Russie 1812-1813, dessinée par Charles Minard, (1869).....49

Figure 8 – Évolution de l'intérêt porté aux problématiques de la visualisation depuis 2004. Cette figure a été réalisée à partir de données extraites de Google Trends29F. Le code pour réaliser cette figure est disponible sur GitHub Gist30F. La tendance de recherche d'un mot-clé correspond au nombre de recherches réalisées à un moment donné et rapporté entre 0 et 100 en fonction du plus grand nombre de recherches pour ce même mot-clé. 50

Figure 9 – Infographie comparant la vitesse de traitement des informations reçues par nos sens en les rapportant à des flux informatiques. Les performances de l'œil sont comparables à la bande passante d'un câble Ethernet, les performances du touché sont comparables à la vitesse de lecture d'une clé USB et enfin les performances de l'odorat et de l'audition sont comparables à celles d'un disque dur. Cette infographie est proposée par David McCandless (McCandless 2014).....51

Figure 10 – Enchaînement des concepts présentés dans cette première partie du manuscrit de thèse. Les concepts de « données », « analyses de données », « informations » et « connaissances » sont présentés dans le chapitre précédent. Les concepts de « visualisation de données » et « infographie » sont présentés dans ce chapitre.....52

Figure 11 – Interprétations associées aux 4 types de graphes proposées par S. Berinato dans son livre Good Charts (Berinato 2016). Ces graphiques sont associés à des tâches différentes : illustration d'une idée, visualisation de données, découverte visuelle ou création d'idées.....52

Figure 12 – Variables visuelles selon Jacques Bertin. La figure a été extraite de son ouvrage Sémiologie graphique. Les diagrammes. Les réseaux. Les cartes (1968). 2 DP pour plan à deux dimension, G pour Grain, V pour valeur, T pour taille, F pour forme, OR pour orientation et C pour couleur. 53

Figure 13 – Signification des couleurs en fonction des cultures. Cette visualisation est extraite du site informationisbeautiful.net34F. Cette infographie est aussi utilisée sur la page de garde de son livre Information is beautiful (McCandless 2000). 54

Figure 14 – Mise en application des 4 types de visualisations « déclarative », « conceptuelle », « exploratoire » et « basée sur les données ». Ces principes sont détaillés Figure 11, page 52. Ce travail a été réalisé dans le contexte du projet d'étude de l'homéostasie du fer de la levure pathogène Candida glabrata (voir page 128). 56

Figure 15 – Illustration du processus cyclique d'une analyse de données. Six grandes étapes sont présentées sur cette figure, telles que décrites dans l'ouvrage Introduction to Statistics & Data analysis de R. Peck (Peck et al. 2016).....57

Figure 16 – Répartition du temps de travail lors des différentes étapes d'une analyse de données. La partie « nettoyage » des données est de loin la plus longue (60% du temps). Cette illustration est extraite du sondage de CrowdFlower (CrowdFlower 2016). 59

Figure 17 – Bilan des stratégies expérimentales permettant d'étudier les mécanismes de fonctionnement des gènes. Cette figure est adaptée d'une figure du cours de B. Cosson, donné en 2019 dans de DU « Création, analyse et valorisation de données omiques » de l'Université de Paris. La figure était initialement extraite de l'article de W. Soon (Soon et al. 2013). Près d'une demi-douzaine de niveaux de régulation de l'expression des gènes sont représentés ici. Ils illustrent la complexité et la multitude des mécanismes qui peuvent être considérés dans une étude de génomique fonctionnelle.....65

Figure 18 – Illustration d'un facteur de transcription spécifique (protéine X) de type « activateur ». La protéine X existe sous deux formes, une forme inactive et une forme active (notée X*). Sous sa forme active, la protéine X* se fixe au niveau des séquences promotrices de ses gènes cibles et permet ainsi une augmentation de la transcription. Le signal S_x est à l'origine du déplacement de l'équilibre de X vers sa forme active X*. Cette figure est extraite du livre de U. Alon (page 5).....66

Figure 19 – Comparaison des termes les plus associés dans les titres de publications concernant la génomique fonctionnelle et la biologie des systèmes. Les listes des journaux explorés sont basées sur les recommandations de Pubmed pour la recherche « Functional Genomics » et « System Biology ». Les titres de ces journaux ont été ensuite extraits à partir de l'outil de recherche avancée de Pubmed et un nuage de mots a été généré sur le site WordArt.com. Cette figure a été réalisée le 01/04/2020. 68

Figure 20 – Illustration du principe « FAIR data » qui a été dérivé pour créer la formation FAIR_bioinfo. Cette formation a été proposée en 2019 aux chercheurs de l'I2BC. 74

Figure 21 – Graphique des probabilités associées au nombre de fois ou le côté FACE d'une pièce de monnaie (équilibrée) est observé sur 10 lancers indépendants de la pièce. Observer 9 ou 10 fois le côté FACE est très peu probable, toutefois cela n'est pas impossible. 79

Figure 22 – Probabilité de rejeter au moins une fois l'hypothèse nulle lors de la réalisation de plusieurs tests statistiques de manière indépendante. Pour chaque test, le risque de 1ère espèce est fixé à 5%. 80

Figure 23 – Analyse de 791 articles provenant de 5 journaux qui montre que plus de la moitié des études présentées concluent à tort que l'absence de significativité (valeur P > 0.05) signifie l'absence d'effet. Ne pas rejeter l'hypothèse H_0 est considéré comme une preuve que l'hypothèse H_0 est vraie. 82

Figure 24 – Comparaison des "profondeurs" associées à deux termes GO. À droite la description du terme est plus précise que celle présentée à gauche. Les DAG ont été extraits d'Amigo75F. 84

Figure 25 – Nombre de termes GO référencés dans la base de données, en fonction du temps (a) et nombre de termes GO déclarés obsolètes dans la base de données, en fonction du temps (b). Cette figure est extraite de l'article de F. Nakano (Nakano et al. 2019) 85

Figure 26 – Évolution du prix du megaoctet au cours du temps. Les données ont été extraites le 03/04/2020 en ligne78F. Le code pour réaliser cette figure est disponible sur GitHub Gist79F 87

Figure 27 – Figure extraite de l'article de B. Fecher (Fecher et al. 2013) présentant les 5 écoles de pensée pour tendre vers l'Open Science. 89

Figure 28 – La planification, entre fiction et réalité (illustration de phdcomics). 92

Figure 29 – Représentation des différentes étapes d'un sprint lors d'un projet agile (@pawan-pawar) 92

Figure 30 – Exemple de réseau obtenu lors de la recherche d'une liste de protéines de la levure Saccharomyces cerevisiae. À gauche une capture d'écran de la page WEB de MONet, dans laquelle la liste de gènes/protéines peut être saisie. À droite le réseau obtenu, après interrogation de la base de données STRING.....106

Figure 31 – Exemple de dashboard obtenu lors de la recherche d'une liste de protéines de la levure S. cerevisiae. Il s'agit de la même liste que celle présentée dans la figure précédente. 107

Figure 32 – Représentation schématique d'un réseau d'éléments co-exprimés. Cette figure est extraite de l'article présenté page 149. 108

Figure 33 – Exemple de graphe obtenu à partir d'un fichier de données d'expression de gènes de la levure C. glabrata. 109

Figure 34 – Exemple de la page Overview après l'import d'un fichier. 109

Figure 35 – Extrait du rapport exporté du gène TP53 de l'Humain. 110

Figure 36 – Résumé schématique des différentes étapes de fonctionnement du logiciel MONet à partir d'un tableau de données d'expression de gènes.....112

Figure 37 – Les 7 étapes proposées dans la formation FAIR_bioinfo pour rendre une analyse de données reproductible. 115

Figure 38 – Les différentes solutions proposées par B. Grüning pour augmenter la reproductibilité des analyses bioinformatiques (Grüning et al. 2018). 117

Figure 39 – Evolution au cours du temps (période 1997 à 2007) des pourcentages d'infections causées par les levures C. albicans (vert), C. glabrata (rose), C. parapsilopsis (rose claire), C. tropicalis (bleu clair) et C. krusei (magenta). Ce graphique est extrait de l'article de Guinea et al. et a été réalisé à partir des données ARTEMIS DISK (Guinea 2014).....121

Figure 40 – Représentation schématique de la formation de biofilms par les levures C. albicans et C. glabrata. C. albicans forme des biofilms plus épais, avec beaucoup de biomasse à la fin de la formation du biofilm et produit plus de matrice extracellulaire que C. glabrata. Les biofilms matures de C. albicans sont composés d'un réseau dense de pseudohyphes, hyphes et cellules de levure, tandis que les biofilms de C. glabrata sont composés uniquement de cellules de levure compactes, formant un biofilm mince mais dense (Galocha et al. 2019). 123

Figure 41 – Représentation schématique des stratégies d'infections causées par C. albicans et C. glabrata. C. albicans (en haut) est capable de changer de forme et devenir un hyphe capable de détruire les tissus, provoquant une forte réponse immunitaire. De nombreux aspects de la pathogénicité de C. glabrata sont encore inconnus, comme le mécanisme précis de l'invasion (en bas). Les lésions tissulaires actives de l'hôte sont faibles, tout comme la réponse immunitaire. Le franchissement de la barrière épithéliale peut se faire par le biais de matériels médicaux (Brunke et al. 2013). 124

Figure 42 – Aperçu des informations fonctionnelles disponibles chez les levures S. cerevisiae, C. albicans et C. glabrata. La figure a été réalisée à partir de données mises à jour le 06/05/2020134F sur les sites internet des bases de données SGD et CGD. 125

Figure 43 – Évolution du nombre de publications au cours du temps associées aux espèces de levures S. cerevisiae, C. albicans et C. glabrata. Les données ont été obtenues de la base de données PubMed après une recherche du nom de des espèces. La figure a été réalisée le 07/05/2020. Le code pour générer cette figure est disponible sur GitHub Gist135F. 125

Figure 44 – Relations phylogénétiques entre C. glabrata, les espèces du clade des Nakaseomyces pour lesquelles le génome est entièrement séquencé et les autres espèces du clade des Saccharomycotina. Les espèces pathogènes sont indiquées en rouge. CTG indique un changement du code génétique ; WGD indique la duplication du génome entier ancestral ; et EPA indique la lignée où les deux expansions indépendantes des gènes EPA ont eu lieu. Ces gènes sont importants pour la pathogénicité de C. glabrata. Cette figure est extraite de l'article T. Gabaldón (Gabaldón et al. 2016). 126

Figure 45 – Photographies des levures C. glabrata et C. albicans. À gauche des cellules de levure C. glabrata cultivées sur milieu Sabouraud obtenue par A. Angoulvant. À droite, des cellules de levure C. albicans avant et après la conversion morphologique (Sudbery 2011)..... 127

Figure 46 – Illustration schématique des apports et des pertes en fer dans l'organisme humain. 129

Figure 47 – L'homéostasie du fer et sa modulation par érythropoïèse et inflammation. L'hepcidine bloque les principaux flux du fer dans le plasma (provenant principalement des macrophages spléniques qui recyclent les érythrocytes mais aussi de l'absorption duodénale et des réserves dans les hépatocytes) en provoquant la dégradation de son récepteur, la ferroportine, exportateur de fer. La production d'hepcidine par le foie est régulée à la hausse par l'augmentation des niveaux de fer dans le plasma et des réserves de fer dans le foie. L'infection et l'inflammation stimulent également la transcription du gène codant pour l'hepcidine (Ganz et al. 2015). 132

Figure 48 – Représentation schématique de la stratégie de C. glabrata pour s'échapper d'un macrophage. Cette levure est surtout connue pour une stratégie de persistance, survivant et prospérant à l'intérieur des macrophages, conduisant finalement à la lyse des cellules immunitaires en raison de la charge fongique (Galocha et al. 2019). 133

Figure 49 – Homéostasie du fer chez les champignons. La régulation de l'homéostasie du fer (côté gauche de la figure) est indiquée pour différentes espèces de champignons dont la levure C. glabrata (triangle pointant vers le haut orange). Les principaux facteurs de transcription régulés à la hausse pendant la privation de fer pour initier l'absorption du fer fongique (à droite) sont écrits en gras (Gerwien et al. 2018). 134

Figure 50 – Figures récapitulatives des gènes impliqués dans l'homéostasie du fer chez C. glabrata et décrits dans la littérature. La figure A est en condition de carence en fer et la figure B en condition de surcharge en fer. La partie gauche correspond à des diagrammes de Venn. Chaque cercle représente un ensemble de gènes réagissant à la condition étudiée (carence ou surcharge) dans un jeu de données disponible dans la littérature. Le croisement entre deux cercles (deux expériences) correspond à un ensemble de gènes identifiés comme réagissant dans la condition étudiée dans deux expériences. La partie droite des figures correspond aux différents facteurs de transcription identifiés et les gènes avec lesquels ils interagissent. Ces deux figures sont extraites et expliquées en détails dans l'article de F. Devaux (Devaux et al. 2019). 138

Figure 51 – Représentation du plan d'expérience utilisé dans l'étude de l'homéostasie du fer de la levure C. glabrata, à partir d'une référence commune. La condition C1 correspond à une étude dans un milieu pauvre en fer à 30 °C, C2 à un milieu pauvre en fer à 37 °C, C3 à un milieu riche en fer à 30 °C, C4 à un milieu riche en fer à 37 °C et C5 l'influence de la température (30 °C vs 37 °C). Ces conditions sont décrites en détail dans notre article inséré ci-dessous (page 149) (Denecker et al. 2020). 141

Figure 52 – Analyse en composante principale de l'ensemble des données de puces à ADN utilisées dans cette étude de l'homéostasie du Fer chez la levure pathogène C. glabrata. Le code pour réaliser cette figure est disponible sur GitHub Gist148F.....142

Figure 53 – Représentation schématique des différentes conditions expérimentales utilisées pour comparer l'abondance de l'ARNm avec la technologie des puces à ADN. Cinq conditions nommées C1, C2, C3, C4 et C30-37 ont été définies. Des valeurs de Z-Score ont été calculées pour chaque condition, en comparant les échantillons écrits en rouge (respectivement, "BPS", "FeSO4" et "37°") aux échantillons écrits en vert (respectivement, "Control" et "30°").....143

Figure 54 – Définitions et hypothèse de travail extraites de notre article (Denecker et al. 2020). (A) Représentation schématique des changements extracellulaires de la disponibilité du fer, auxquels est confrontée la levure pathogène C. glabrata. « Low » signifie que la disponibilité en fer est faible, la cellule doit s'adapter à la carence en fer. « High » signifie que la disponibilité en fer est élevée, la cellule doit s'adapter à la surcharge en fer. L'homéostasie du fer est représentée ici comme le processus physiologique central qui permet de maintenir un environnement intracellulaire dans un état constant d'équilibre en fer, malgré les changements extérieurs. (B) Trois classes de gènes ont été étudiées dans cet article. La première classe "all genes" se réfère à tous les gènes pour lesquels nous avions des données. La deuxième classe, les "iron responsive genes", désigne les gènes pour lesquels des changements d'expression sont observés dans au moins une des expériences de transcriptomique. Enfin, la troisième classe "iHKG" fait référence à un nouvel ensemble de gènes ayant des fonctions particulières importantes pour la cellule afin de contrebalancer les fluctuations externes de la disponibilité du fer, dans n'importe quelle direction (faible ou forte). (C) Représentation schématique des deux types de iHKG sur la base des dérégulations observées dans notre ensemble de données, respectivement dans des conditions de fer "Low" (-) et "High" (+). Les "types I" sont des iHKG avec des dérégulations opposées en cas de fer faible et élevé alors que les "types II" sont des

iHKG avec une dérégulation constante (ou parallèle) dans des conditions de fer faible et élevé. 145

Figure 55 – Réseau de co-expression des 637 gènes réagissant au fer dans les conditions de fer faibles (gauche) et de fer élevées (droite). Chaque nœud correspond à un gène coloré en rouge s'il est surexprimé par rapport à une condition standard ou en vert s'il est sous exprimé. Les carrés sont les iHKG de Type I, les gros ronds de Types II et les petits ronds les autres gènes. 146

Figure 56 – Exploration des termes GO pour définir un nombre limité de fonctions générales auxquelles tous les gènes réagissant au fer peuvent être assignés. (A) Représentation schématique du processus appliqué pour traiter les termes GO dans cette étude. À partir d'une liste de termes GO (dans notre cas, tous les termes dans "Processus biologique"), nous avons reconstruit le DAG fourni par la base de données GO. Les niveaux sont associés à chaque terme des GO en fonction de leur position dans le DAG (pour plus de détail, voir la section Matériels et Méthodes et la Figure supplémentaire S10 de l'article). Ceci est illustré ici par des cases de couleur. Les termes GO sont ensuite regroupés en fonction d'un niveau (niveau 3 dans cet exemple) dans la hiérarchie des GO. Nous obtenons ainsi des "Meta-GO", c'est-à-dire des groupes de termes GO qui partagent un ancêtre commun dans le DAG. Dans l'analyse présentée dans l'article et dans ce manuscrit, le niveau 4 a été utilisé pour créer les Meta-GO (voir Matériel et Méthodes et les données supplémentaires S3). (B) Fonctions générales définies dans cette étude pour mettre en évidence les rôles physiologiques des gènes réagissant au fer. Elles sont appelées "Metabolism". (F1), "Regulation" (F2), "Redox Signaling" (F3), "Transport / Trafficking" (F4), "Iron sulfur cluster synthesis and assembly" (F5) et "Others". 147

Figure 57 – Exploration des facteurs de transcription dans le sous réseau fonctionnel des gènes co-exprimés "Regulation" (Fer faible à gauche et fer élevé à droite). Le gène codant pour le facteur de transcription Hap1 est au voisinage immédiat du facteur de transcription Aft1....149

Figure 58 – Arbre phylogénétique composé des 4 levures d'intérêt : C. glabrata, C. bracarensis, C. nivariensis et N. delphensis extrait du site phylomeDB. 153

Figure 59 – Nombre d'orthologues trouvés entre les gènes réagissant au fer chez C. glabrata et une des 3 espèces : C. bracarensis, C. nivariensis et N. delphensis (diagramme en secteur de

gauche). Si aucun orthologue, nous avons regardé le type (diagramme en secteur de droite). 154

Figure 60 – Résultats du croisement des listes de gènes ne possédant pas d'orthologues chez C. glabrata et faisant partie de la liste des gènes réagissant au fer chez C. glabrata. Les abréviations CNIVA, CBRAC et NDELP correspondent respectivement aux espèces C. nivariensis, C. bracarensis et N. delphensis. Chaque barre correspond au nombre de gènes communs aux listes sélectionnées (identifiées en dessous par des points noirs). Par exemple, la barre la plus à gauche correspond au croisement des listes des 3 espèces qui partagent 30 gènes. Les couleurs présentes sur les barres correspondent aux types des gènes réagissant au fer chez C. glabrata (bleu pour les gènes de types I, orange pour les gènes de types II et gris pour les autres). 155

Figure 61 – Nuages de points avec sur l'axe des ordonnées les logFC de gènes réagissant au fer chez C. glabrata et en abscisse les logFC des orthologues dans l'une des 3 espèces étudiées : C. bracarensis, C. nivariensis et N. delphensis. Le point bleu à la forme carrée sont les gènes de Type I dans notre étude, les oranges à la forme ronde les type II et gris à la forme ronde les gènes qui ne sont ni Type I, ni Type II. Les gènes différentiellement exprimés chez C. glabrata et dans la levure étudiée (abs(logFC) > 1) sont nommés sur les nuages de points. 157

Figure 62 – Répartition des gènes réagissant au fer chez C. glabrata, C. nivariensis et C. bracarensis dans les grandes fonctions créées chez C. glabrata. 158

Figure 63 – De nombreuses modifications se produisent entre le génome et le protéome. L'accumulation de toutes ces modifications augmente considérablement la complexité du protéome. Cette illustration est extraite du site de Thermo Fishier scientific156F et concerne l'Homme. 160

Figure 64 – Résumé des différentes étapes pour identifier des protéines par l'approche Bottom up. 162

Figure 65 – Spectres de masses pour un peptide avec une modification post-traductionnelle (oxydation de la méthionine - en haut) et le même peptide sans modification post-traductionnelle (en bas). Ces spectres sont issus de l'article de Xu et al (Xu et al. 2019). De nombreux autres exemples sont disponibles en figures supplémentaires de cet article. 165

Figure 66 – Exemple des informations renseignées pour la modification post-traductionnelle Glutathionylation dans RAId. 169

Figure 67 – Vue d'ensemble des différents fichiers générés lors de notre analyse. 172

Figure 68 – Nombre de protéines identifiées par fichier RAW. Comparé aux autres expériences, les expériences 1913001 conduisent à moins d'identification que les autres (les 6 lignes du bas). Il s'agit pourtant des réplicats biologiques de l'expérience 1847003. La seule différence entre les deux est un passage dans un congélateur à -80°C pendant un an. Après discussion avec l'équipe, il est connu que la congélation peut entraîner des pertes de protéines. 174

Figure 69 – Distribution du nombre de peptides différents conduisant à l'identification d'une protéine. 174

Figure 70 – Répartition des protéines identifiées entre les formes hyphe et levure dans différents cas : à gauche, en étudiant uniquement les protéines identifiées à partir de peptides ne contenant pas de modification post-traductionnelle, au centre en n'étudiant que les protéines identifiées à partir de peptides contenant des modifications post-traductionnelles et à droite en étudiant toutes les protéines sans distinction des peptides. 175

Figure 71 – Répartition des protéines identifiées avec ou sans modifications post-traductionnelles dans différents cas : à gauche, en étudiant les protéines identifiées dans la forme levure, au centre en étudiant que les protéines identifiées dans la forme hyphe et à droite en étudiant toutes les protéines sans distinction de forme. 176

Figure 72 – Nombre de protéines en fonction du nombre de modifications post-traductionnelles détectées. 177

Figure 73 – Nombre de protéines identifiées pour une modification post-traductionnelle donnée. Les flèches rouges pointent les modifications post-traductionnelles qui sont actuellement recherchées en routine (Oxidation (Met), phosphorylation (Ser, Thr, Tyr), acetylation (N-term of protein) et carbamidomethylation (Cys)). Les flèches bleues pointent des modifications post-traductionnelles qui ont permis de mettre en évidence de nouvelles protéines uniquement grâce à elles. La flèche verte pointe la modification post-traductionnelle Glutathionylation très étudiée dans le laboratoire de Jean-Michel Camadro. 178

Figure 74 – Nombre de protéines identifiées en fonction du nombre de fichiers utilisés pour les identifier. 179

Figure 75 – Distribution des logFCs permettant de mettre en évidence des protéines principalement identifiées dans la forme hyphe (à droite) ou dans la forme levure (à gauche). 180

Figure 76 – Comparaison des protéines identifiées par la méthode classique (Proteome discoverer) et notre protocole basé sur RAId. 191

Figure 77 – Comparaison de la liste des 637 gènes réagissant au fer avec les protéines identifiées par une méthode classique (Proteome discoverer) et notre protocole basé sur RAId. 192

Figure 78 – Illustration schématique du principe de la méthode BHR et BHR adaptée. 198

Figure 79 – Confirmation d'un lien d'orthologie entre deux éléments. Ici, nous savons que les ORF1, ORF2, ORF4 et ORF5 sont orthologues entre l'espèce A et B. Nous souhaitons confirmer l'orthologie entre les ORF3. Comme ORF1, ORF2, ORF4 et ORF5 sont conservés et dans le même ordre, nous pouvons conclure que les ORF3 sont orthologues entre les espèces A et B 199

Figure 80 – Annotation associée aux orthologues trouvés entre C. glabrata et les 3 autres levures étudiées. La majorité des ORFs ont un orthologue associé. 200

Figure 81 – Annotation des orthologues associés aux gènes qui n'avaient pas d'orthologue sur phylomeDB 201

Figure 82 – Amélioration de la recherche d'orthologues dans le clade des Nakaseomyces ... 201

Figure 83 – Croisement des listes de gènes ne possédant toujours pas d'orthologues chez C. glabrata et faisant partie de la liste des gènes réagissant au fer. Les couleurs des barres présentes sur les barres correspondent aux types des gènes réagissant au fer chez C. glabrata (bleu pour les gènes de types I, orange pour les gènes de types II et gris pour les autres). 202

Figure 84 – Programme de thèse envisagé lors de l'audition pour l'obtention d'une bourse ministérielle à l'école doctorale "Structure et Dynamique des Systèmes Vivants" (diapositive extraite de la présentation). 207

Figure 85 – Illustration de ce que nous considérions comme le défi le plus complexe de la thèse lors de l'audition pour l'obtention d'une bourse ministérielle à l'école doctorale "Structure et Dynamique des Systèmes Vivants" (la figure est extraite de la présentation).....208

Figure 86 – Les différents projets menés au cours de cette thèse (2017-2020).....211

Figure 87 – Les différentes disciplines composant la data science. Il s'agit d'un domaine de recherche interdisciplinaire à l'intersection entre une douzaine de thématiques. Cette figure est extraite du livre de W. Van der Aalst (Van der Aalst 2016).....220

Figure 88 – Représentation schématique du lien entre les domaines de l'intelligence artificielle, le Marchine learning et le Deep learning.....221

Figure 89 – Evolution du nombre de publications parues sur Pubmed concernant le Deep learning. Les données ont été obtenues grâce à l'outil timeline de Pubmed lors de la recherche « Deep learning » réalisée le 17/04/2020.....225

Figure 90 – Exemple de classifications de CellProfiler (Bray et al. 2015). À partir d'une photographie initiale du milieu de culture (A), CellProfiler est capable de classer en fonction de l'aire (E) et la couleur de la colonie (F).....225

Figure 91 – Illustration du remplissage et de la composition d'un fichier RAW généré lors d'une analyse de spectrométrie de masse en tandem LC-MS/MS par l'approche Bottom up.....237

Liste des tableaux

Tableau 1 – Quelques types de données emblématiques en bioinformatique. Les données sont à la fois hétérogènes et représentent des volumes importants. 40

Tableau 2 – Tableau fictif illustrant la notion de « données structurées ». Les gènes sont présentés en ligne, tandis que les colonnes correspondent à différentes variables étudiées. Les valeurs numériques pourraient correspondre à des mesures de différentiels de l'expression des gènes, observées entre deux conditions différentes de cultures de cellules (ce sont des logFC). 43

Tableau 3 – Tableau récapitulatif de la notion de risque en fonction des hypothèses mises en concurrence lors d'un test statistique. L'hypothèse H_0 est l'hypothèse nulle, tandis que l'hypothèse H_1 est l'hypothèse alternative. L'erreur de Type 1 correspond au risque de 1ère espèce, tandis que l'erreur de Type 2 correspond au risque de 2ème espèce. 78

Tableau 4 – Programme proposé lors de la première formation FAIR_bioinfo. Cette formation a été proposée aux chercheurs de l'I2BC en 2019. 115

Tableau 5 – Répartition du fer dans un organisme humain adulte de 60 kg (Abbaspour et al. 2014). 129

Tableau 6 – Tableau récapitulatif d'une liste de gènes décrits dans la littérature comme impliqués dans l'homéostasie du fer de C. glabrata. 137

Tableau 7 – Nombre de caractéristiques chromosomiques pour lesquelles des sondes sont trouvées sur les puces à ADN utilisées dans cette étude. "Uncharacterized" signifie que les ORF ont été prédits sur la base de l'analyse des séquences mais qu'elles manquent de caractérisation expérimentale et "Verified" signifie qu'il existe des preuves expérimentales pour un produit fonctionnel. 140

Tableau 8 – Informations génomiques disponibles sur le site Internet de la base de données GRYC concernant C. bracarensis, C. nivariensis et N. delphensis. 151

Tableau 9 – Liste des modifications post-traductionnelles renseignées dans le programme d'identification RAId. En fonction de la modification post-traductionnelle, la différence de

masse est plus ou moins importante. Ces différences sont détectables par le spectromètre de masse. Les masses sont exprimées en Da. 164

Tableau 10 – Liste des fichiers RAW étudiés avec différentes caractéristiques expérimentales. Les lignes rouges correspondent à la forme hyphe et les lignes bleues à la forme levure. 168

Tableau 11 – Liste des informations principales disponibles après une identification avec RAId 170

Tableau 12 – Protéines les plus spécifiques de la forme hyphe (logFC les plus élevés). Elles sont beaucoup plus identifiées dans la forme hyphe que dans la forme levure. 182

Tableau 13 – Protéines les plus spécifiques de la forme levure (logFC les plus élevés). Elles sont beaucoup plus identifiées dans la forme levure que dans la forme hyphe. 184

Tableau 14 – Listes de quelques protéines collectées dans la littérature connues comme spécifiques d'une forme (hyphe ou levure). Cette spécificité est comparée au ratio R (Levure/Hyphe). 186

Tableau 15 – Liste de gènes n'ayant pas d'orthologue chez C. bracarensis, C. nivariensis et N. delphensis et faisant partie des gènes réagissant au fer chez C. glabrata 203

Avant - propos

Pourquoi cette thèse ?

J'ai commencé mon parcours scientifique comme technicien dans les laboratoires de l'Assistance Publique des Hôpitaux de Paris (période 2011 – 2014). J'y ai observé l'arrivée des premiers séquenceurs à très haut débit (technologie Illumina) et pour mes collègues, une problématique majeure de l'utilisation de ces nouveaux appareils était la gestion, l'organisation et le traitement de la grande quantité de données générées (plusieurs Giga octets pour une seule utilisation). C'est dans ce contexte que j'ai décidé de poursuivre mes études. Je souhaitais en effet acquérir les compétences nécessaires à l'analyse de ces nouvelles données biologiques. J'ai suivi une année de Licence 3 puis deux années de Master en Bioinformatique. Gaëlle Lelandais (ma directrice de thèse) enseignait les statistiques et la génomique fonctionnelle dans ces formations et c'est ainsi que je l'ai rencontrée. Elle a encadré un de mes stage de recherche en Master 1, mon stage de recherche en Master 2 et nous avons préparé ensemble le concours pour l'obtention d'une bourse de thèse auprès l'école doctorale SDSV de l'Université Paris-Saclay. Mes trois années de thèses se sont ensuite déroulées à l'I2BC (Institut de Biologie Intégrative de la Cellule), au sein du département Biologie des Génomes. J'ai travaillé dans un contexte interdisciplinaire, collaborant avec des chercheurs en bioinformatique, mathématiques, statistiques, informatique et biologie. Ainsi, j'ai acquis des savoir-faire et une expérience en « bioinformatique et analyse de données multi-omiques ». Dans ce manuscrit, mon objectif est de vous présenter les principes de ce type d'analyses et les mises en applications qui ont été réalisées au cours de mon travail de thèse.

Évolution de la bioinformatique face au défi de l'analyse de données

Aujourd'hui, l'analyse de données est une problématique de recherche à part entière et ces dix dernières années, le domaine de la bioinformatique s'est en conséquence totalement réinventé. La Figure 1 illustre cette évolution. Lors de la thèse de Gaëlle (période 2002 – 2005), les mots les plus fréquemment utilisés dans les titres des publications en bioinformatique étaient *protein*, *gene* ou *sequence*. Lors de ma thèse (période 2017 – 2020), le mot le plus fréquemment utilisé était celui de *data*. Il est clair que l'exploitation des données est un enjeu majeur des projets de recherche en biologie actuels. Disponibles en très grandes quantités dans des bases de données publiques et librement réutilisables, elles sont hétérogènes (données numériques, de textes, images, séquences biologiques, etc.) et conservées sur des supports d'information également très hétérogènes (papiers ou numériques).

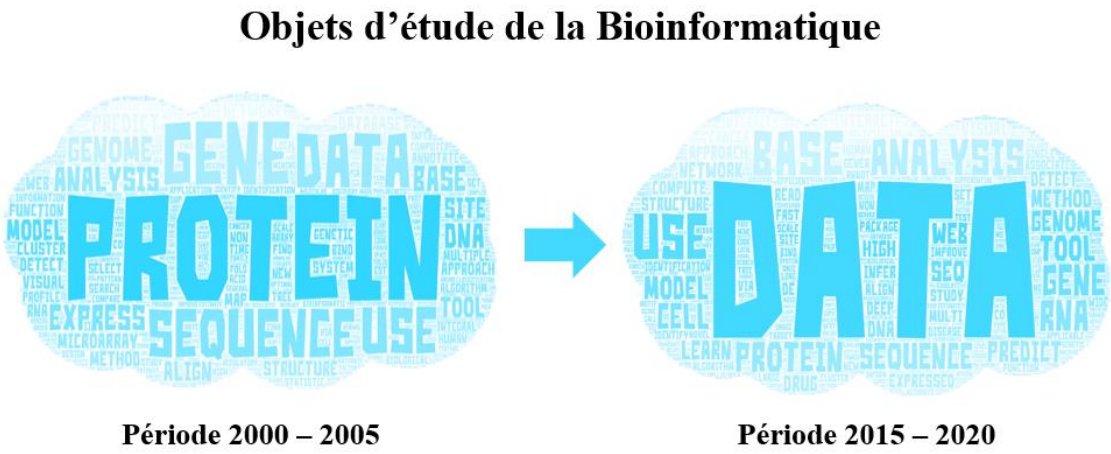


Figure 1 – Nuages des mots utilisés dans les titres des articles publiés sur une période de cinq années (2000 – 2005 à gauche et 2015 – 2020 à droite) dans les revues de Bioinformatique : Bioinformatics, BMC Bioinformatics, Brefings in Bioinformatics *et* Journal of bioinformatics and computational biology. *Plus la taille du mot est grande, plus son utilisation est fréquente. La période 2000 – 2005 est la période de la thèse de Gaëlle Lelandais tandis que la période 2015 – 2020 est la période de la thèse de Thomas Denecker. Cette idée de représentation est inspirée d'un séminaire de recherche donné par Jean-Michel Camadro à l'IJM en Février 2019. La figure a été réalisée avec l'outil WEB « wordart.com ».*

Ainsi, disposer d'une grande quantité de données pour répondre à un questionnement biologique n'est souvent plus le défi principal. La vraie difficulté réside dans la capacité à convertir les données en informations, puis en connaissances. Je reviendrai sur ce cheminement dans la suite du manuscrit (page 42) et je l'illustrerai à travers les deux grandes problématiques scientifiques étudiées lors cette thèse. La première concerne l'étude de l'homéostasie du fer chez la levure pathogène *Candida glabrata* (années 1 et 2 de la thèse, page 128). La seconde concerne l'étude systématique des modifications post-traductionnelles des protéines chez la levure pathogène *Candida albicans* (année 3 de la thèse, page 160). Nous verrons que pour ces deux projets, j'ai exploité des données « omiques » : transcriptomiques et protéomiques. J'ai développé des outils bioinformatiques et des outils d'analyses qui ont permis l'émergence de nouvelles hypothèses de recherche en biologie. J'ai également constamment porté une attention particulière aux problématiques de reproductibilité et de partage des résultats avec la communauté scientifique.

Organisation du mémoire

Ce manuscrit se décompose en trois grandes sections. La première section « Introduction générale » (page 53) a pour objectif de préciser les notions théoriques et pratiques, nécessaires

au positionnement de cette thèse dans un contexte scientifique général. Les mots clés qui décrivent le mieux mon travail sont « bioinformatique », « analyse de données » et « technologies multi-omiques ». Dans cette section, je présente différentes définitions et principes associés à ces terminologies, extraits de la littérature ou issues de mon expérience de thèse. La deuxième section intitulée « Contributions aux efforts mutualisés en bioinformatique » (page 95) présente d'une part les outils bioinformatiques que j'ai développés au cours de ma thèse (bPeaks App, Pixel et MONet), et d'autre part une formation (FAIR_Bioinfo) que j'ai créé et animée à l'I2BC. La troisième section intitulée « Contributions aux projets d'analyses de données multi-omiques » (page 120) présente les travaux que j'ai réalisés en génomique fonctionnelle des levures pathogènes de l'homme, *Candida glabrata* et *Candida albicans*. J'ai également eu l'opportunité de travailler pour différents projets développés par des collègues. Ces contributions seront présentées dans une dernière partie (page 193). Chaque section est accompagnée des versions originales des publications associées.

Pour accompagner ce manuscrit de thèse, une ressource numérique a été mise en ligne https://thomasdenecker.github.io/thesisWebsite/ , permettant au lecteur qui le souhaite d'approfondir et de compléter certains éléments de ce texte.

Introduction générale

I. Les fondements de la bioinformatique

1. Les définitions de la bioinformatique

a. Dans la littérature

Si vous demandez à deux bioinformaticiens de définir leur discipline, ils auront certainement des réponses différentes. Il en est de même dans la littérature où de multiples définitions ont été présentées, depuis l'apparition du terme dans les années 70 (Hogeweg 2011). Ainsi, d'après T. Pons, la bioinformatique est *« un domaine de connaissance supra disciplinaire qui associe la biologie, la chimie, la physique et l'informatique en une science à socle large, importante pour approfondir notre compréhension des sciences de la vie »*[1] (Pons et al. 2007). D'après cette définition, le scientifique « bioinformaticien » n'existe pas car il est impossible d'être informaticien, biologiste, médecin, mathématicien et chimiste simultanément. La bioinformatique résulte de l'association entre des chercheurs de disciplines complémentaires pour former une unique « *science* »[2] dite « *hybride* »[3]. C'est un domaine interdisciplinaire qui a pour objectif d'appréhender l'étude des systèmes complexes (ici les systèmes biologiques).

D'après N. Luscombe, « *la bioinformatique consiste à conceptualiser la biologie en termes de macromolécules (d'un point de vue physico-chimique), puis à appliquer des techniques "informatiques" (dérivées de disciplines telles que les mathématiques appliquées, l'informatique et les statistiques) pour comprendre et organiser les informations associées à ces molécules, à grande échelle* »[4] (Luscombe et al. 2001). Ici, la bioinformatique apparaît comme une discipline centrée sur l'étude des objets biologiques (gènes, transcrits, protéines, etc.) en utilisant l'outil informatique (ordinateur au sens large). La notion de large échelle, ou

[1] « *A supradisciplinary field of knowledge that merges biology, chemistry, physics, and computer science into a broad-based science that is important to furthering our understanding of the life sciences*».

[2] « *field of science in which biology, computer science, and information technology merge to form a single discipline* » - National Center for Biotechnology Information (NCBI).

[3] « *Bioinformatics, a hybrid science that links biological data with techniques for information storage, distribution, and analysis to support multiple areas of scientific research, including biomedicine* » - Encyclopædia Britannica.

[4] « *Bioinformatics is conceptualizing biology in terms of macromolecules (in the sense of physical-chemistry) and then applying "informatics" techniques (derived from disciplines such as applied maths, computer science, and statistics) to understand and organize the information associated with these molecules, on a large-scale*».

omique[5], apparaît cette fois explicitement. D'après E. Pevsner, la bioinformatique est « *l'application d'outils informatiques pour organiser, analyser, comprendre, visualiser et stocker des informations en relation avec des macromolécules biologiques* »[6] (Pevsner 2015), et d'après the National Human Genome Research Institute (NHGRI), « *la bioinformatique est la branche de la biologie qui s'occupe de l'acquisition, du stockage, de l'affichage et de l'analyse des informations trouvées dans les données de séquences d'acides nucléiques et de protéines* »[7]. Ces deux dernières définitions sont intéressantes, car elles prennent en compte les évolutions vers la « *data* » de la bioinformatique (voir Figure 1, page 30). La gestion d'informations à partir de données est présentée comme l'objectif principal de la bioinformatique.

Ces définitions sont complétées par la présentation de trois niveaux de gestion de l'information (Pevsner 2015) : à l'échelle (1) de la cellule par mise en application du dogme central de la biologie moléculaire (ADN, ARN, protéine), (2) de l'organisme par la prise en compte de la différentiation cellulaire et de l'organisation des cellules en tissus et (3) de l'arbre de la vie, en prenant en compte les spécificités et la diversité des espèces. Ces différents niveaux d'études de l'information biologique sont aussi rapportés dans la revue de W. Diniz (Diniz et al. 2017). Dans son ouvrage, E. Pevsner présente la bioinformatique comme une discipline en constante évolution, en fonction de l'avancement des connaissances et des dogmes associés en biologie (Pevsner 2015). Enfin, R. Altman et J. Dugan défendent une définition explicitement centrée sur les « omes » : génome, transcriptome, protéome (Altman et al. 2003). Ils évoquent également des problématiques techniques comme la conception, la validation et la diffusion de logiciels en bioinformatique, le stockage et le partage des données, la reproductibilité et l'interprétation des résultats.

b. Au regard de la biologie computationnelle

En raison de la difficulté de définir unanimement le terme de bioinformatique, des terminologies voisines sont fréquemment utilisées. C'est le cas par exemple de « biologie

---

[5] Pour aller plus loin, une ressource numérique est disponible : https://thomasdenecker.github.io/thesisWebsite/annexes/omiques/ [Accessible le 10/08/2020].

[6] « T*he application of computational tools to organize, analyze, understand, visualize and store information associated with biological macromolecules ».*

[7] « *Bioinformatics is the branch of biology that is concerned with the acquisition, storage, display, and analysis of the information found in nucleic acid and protein sequence data* ».

computationnelle ». Je me suis demandé si bioinformatique et biologie computationnelle étaient la même chose. Selon une définition du NIH (*National Institutes of Health*), la biologie computationnelle est définie comme « *le développement et l'application de méthodes analytiques et théoriques de données, la modélisation mathématique et les techniques de simulation informatique pour l'étude de la biologie, les systèmes comportementaux et sociaux* »[8] (Downing et al. 2000). Les notions de « modélisation » et « simulation » sont originales dans cette définition et permettent de faire le *distinguo* entre la « bioinformatique » et la « biologie computationnelle ». La bioinformatique a pour objectifs d'explorer et de décrire les phénomènes biologiques, tandis que la biologie computationnelle a pour objectifs de modéliser et de simuler les phénomènes biologiques. Pour cela, une étape de retranscription des observations biologiques en équations mathématiques est nécessaire. C'est la modélisation des systèmes biologiques.

À la page suivante, je présente une carte conceptuelle que nous avions réalisée avec G. Lelandais en 2016, lors de la préparation du concours pour l'obtention de ma bourse de thèse à l'école doctorale. Notre objectif était de clarifier les apports de la modélisation en biologie à mon projet de thèse (Figure 2). Plus d'une vingtaine de concepts avaient été reliés et regroupés selon différentes couleurs. Ils avaient été choisis en fonction de nos expériences et nos lectures de l'époque. À noter que certains concepts nous avaient semblés particulièrement intéressants et avaient été entourés en rouge. Nous avions ainsi d'une part les concepts de « prédictions » et « expérimentation *in silico* » qui se rattachent au domaine de la biologie computationnelle, et d'autre part, nous avions les concepts de « fouille de données » et « synthèse (ou résumé) des connaissances ». Ces derniers sont en relation plus directe avec mes activités de thèse et se rattachent au domaine de la bioinformatique.

a. Ce qu'il faut retenir

En conclusion, il ne me semble pas exister « une » bioinformatique, mais plutôt « des » bioinformatiques. Ces bioinformatiques dépendent de la formation initiale du scientifique qui les pratiques (informatique, mathématiques, statistiques ou biologie), de ses compétences techniques (utilisation de logiciels existants, capacité à utiliser un langage de programmation

[8] « *The development and application of data-analytical and theoretical methods, mathematical modeling and computational simulation techniques to the study of biological, behavioral, and social systems.* »

ou développer de nouveaux logiciels à façon) et enfin de l'échelle à laquelle il étudie les objets biologiques (étude des propriétés individuelles ou bien des propriétés globales des systèmes).

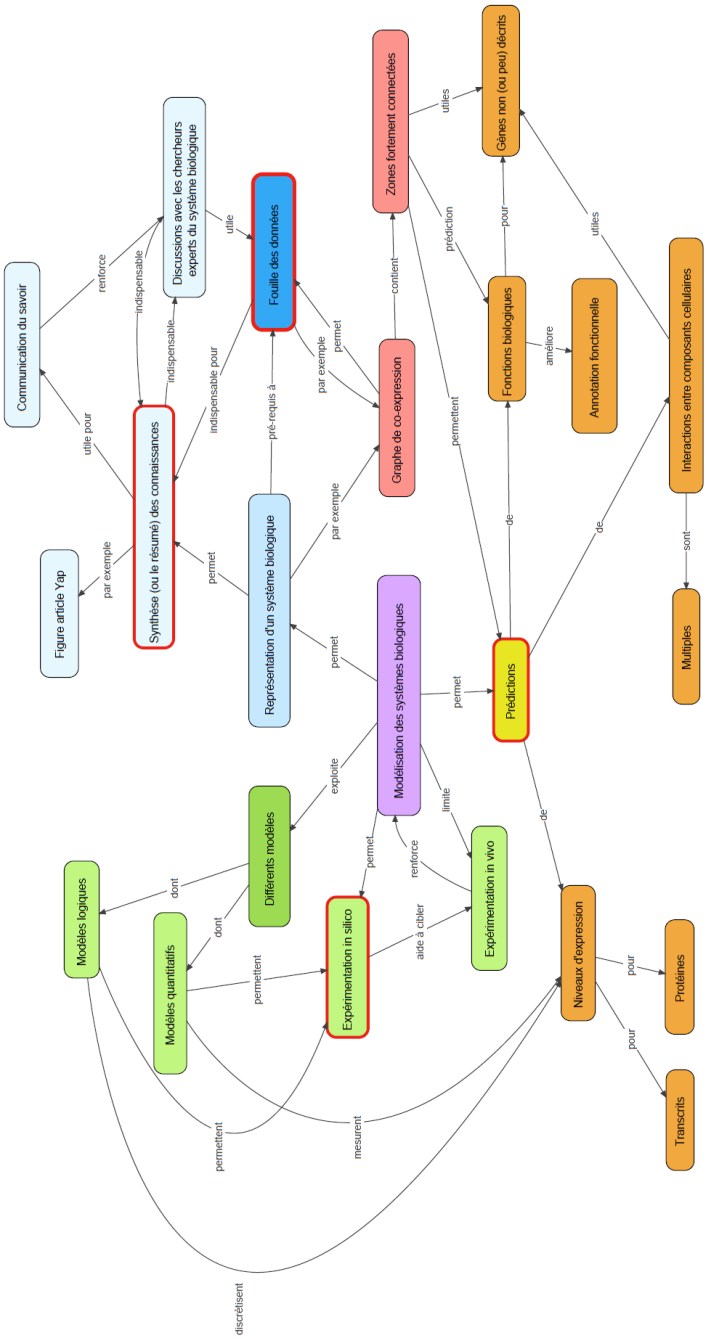


Figure 2 – Carte conceptuelle réalisée en 2016 pour clarifier les concepts en relation avec la « Modélisation des systèmes biologiques » (encadré en violet). Quatre notions majeures sont encadrées en rouge : « Expérimentation in silico », « Prédiction », « Synthèse des connaissances » et « Fouille de données ». Les couleurs regroupent des concepts similaires ou fortement reliés. Cette carte a été réalisée avec le logiciel Xmind[9].

[9] https://www.xmind.net/fr/ [Accessible le 04/08/2020]

2. Les usages de la bioinformatique

a. Accompagner les évolutions des technologies de séquençage des acides nucléiques

Médecine personnalisée et prédictive, identification et caractérisation de pathogènes, suivis épidémiologiques, etc., ces sujets ont fait les grands titres de l'actualité mondiale au printemps 2020[10]. La médecine change grâce à des ensembles de données de plus en plus larges et diversifiées, à l'échelle des patients ou des populations. Dans ce contexte, les technologies expérimentales « omiques » occupent une position privilégiée. Il s'agit des séquençages de génomes, des études de l'expression des gènes, des quantifications des protéines, des recherches de biomarqueurs, etc. De nouvelles problématiques d'études émergent et la bioinformatique est une des clés de la réussite de ces nouveaux défis. Elle fournit les outils computationnels nécessaires pour la gestion, le traitement et la compréhension de ces masses de données. L'exemple le plus emblématique est celui du projet Génome Humain[11], initié en 1990 et dont les objectifs étaient de :

- Déterminer la séquence de l'ensemble des bases qui composent l'ADN humain (environ 3 milliards) ;
- Annoter l'ensemble des gènes dans le génome humain (~ 20 000 – 25 000) ;
- Stocker toutes ces informations dans des bases de données ;
- Créer de nouveaux outils d'analyses fondés sur l'exploitation de ces nouvelles bases de connaissances.

Pour mener à bien ce projet, des améliorations techniques sur le plan expérimental ont été développées, accompagnées d'une diminution importante des temps de séquençage et des coûts financiers associés. Ainsi, s'il a fallu 4 ans pour séquencer le premier milliard de nucléotides qui compose le génome humain, seulement 4 mois ont été nécessaires pour le suivant. Au début du projet, le séquençage d'un nucléotide coutait de l'ordre de 10 $, alors que cela ne coutait plus que 0.10 $ à la fin (diminution d'un facteur 100, Figure 3).

[10] Cette introduction de thèse a été rédigée pendant la première période de confinement liée à la COVID-19.
[11] https://fr.wikipedia.org/wiki/Projet_g%C3%A9nome_humain [Accessible le 04/08/2020]

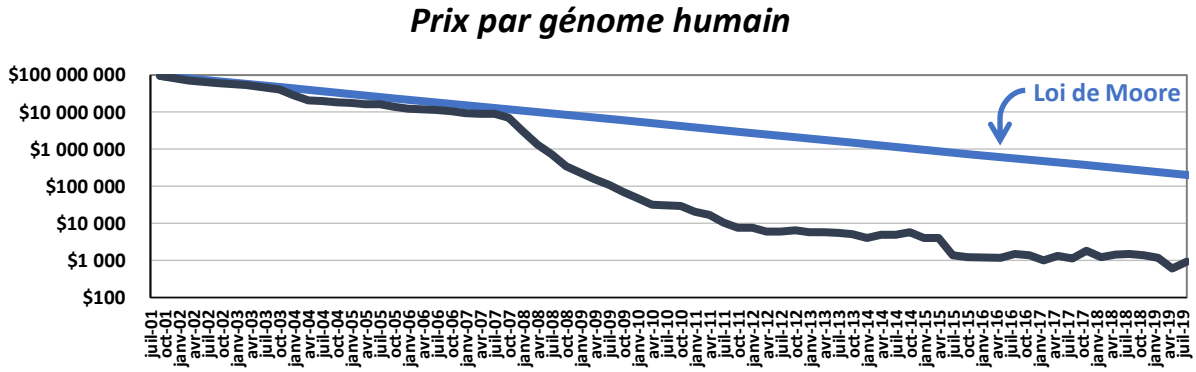


Figure 3 – Évolution des prix du séquençage d'un génome humain. À partir de 2008, les coûts ont été très fortement réduits. Cela correspond à l'arrivée des technologies des séquençages hautement parallélisées. Les séquences sont plus courtes, mais plus nombreuses. La source de ces données est Wetterstrand KA. DNA Sequencing Costs: Data provenant du NHGRI Genome Sequencing Program (GSP)[12]. Ces données sont mises en regard de la loi de Moore (ligne bleue) qui décrit une tendance en informatique impliquant le doublement de la « puissance de calcul » tous les deux ans. Les améliorations technologiques qui suivent la loi de Moore sont considérées comme très performantes. Ici, nous observons une évolution encore plus rapide.

À la fin du projet en 2003, la mise à disposition de la communauté scientifique d'une très grande quantité de données a été accompagnée d'un besoin de ressources pour les exploiter. Il n'était plus raisonnable pour les laboratoires de travailler de façon individuelle et la mutualisation des ressources informatiques et techniques est devenue indispensable. Cela a permis à d'autres projets d'être engagés comme par exemple :

- Dès 2003, le projet HapMap qui avait pour objectif de proposer un haplotype du génome humain. Pour cela, plus de 5 millions d'échantillons biologiques ont été analysés ;
- En 2006, le projet *Cancer Atlas* dont l'objectif était de proposer une description des cancers à l'échelle moléculaire. Aujourd'hui, plus de 20 000 cancers primaires ont été caractérisés pour 33 types cellulaires différents ;
- En 2008, le projet 1000 génomes dont l'objectif était de séquencer 1000 génomes provenant de populations humaines hétérogènes et ainsi de mettre en évidence des variations génétiques. Depuis, d'autres projets du même type ont été initiés comme le projet britannique 100 000 génomes ou le projet 100 000 génomes asiatiques[13].

[12] www.genome.gov/sequencingcostsdata [Accessible le 23/03/2020]

[13] https://www.genomicsengland.co.uk/about-genomics-england/the-100000-genomes-project/ et https://www.asianscientist.com/2016/02/topnews/genomeasia-100k-initiative-sequence-100000-asian-genomes/ [Accessibles le 21/04/2020]

À nouveau, le point commun de ces projets est la quantité considérable de données qui en résultent, provoquant un phénomène nommé « *data deluge* » par les scientifiques.

b. Accompagner le déluge des données en biologie

La terminologie « *data deluge* » est devenue très populaire dans les publications scientifiques à partir de 2009 (Bell et al. 2009). En effet, lors du séquençage d'un génome humain par exemple, il faut prévoir un espace informatique de stockage de plus de 100 Go. Une estimation des ressources nécessaires uniquement pour le projet 100 000 génomes (voir ci-dessus) est ainsi supérieure à 50 pétaoctets (soit 50 000 000 Gigaoctets !). En 2016, la terminologie « *big data deluge* » a commencé à être employée (RR 2016)[14]. Cette évolution a été en relation avec une augmentation de l'hétérogénéité des données produites (Tableau 1). La diversification des techniques expérimentales à haut débit multipliant les formats de fichiers (images de microscopie, spectres de masses, mesures d'intensités lumineuses, vidéos, etc.) et complexifiant les problématiques d'analyse de données.

Type de données	Base de données	Volumes de données
Mesures de l'expression des gènes	ArrayExpress	72 981 expériences, 2 432 182 tableaux, 56.67 TB de données archivées [1]
Mesures de l'expression des gènes	GEO	3 527 170 échantillons réparties dans 127 633 expériences [2]
Structures 3D des protéines	PDB	162 269 structures [3]
Séquences nucléotidiques	GenBank	399 376 854 872 de bases et 216 214 215 séquences [4]
Données d'identification ou de quantification des protéines	Pride	9 991 jeux de données, 457 490 390 spectres de masse [5]

[14] Une définition du Big Data est proposée dans la ressource complémentaire numérique : https://thomasdenecker.github.io/thesisWebsite/annexes/bigData/ [Accessible le 08/08/2020].

Références des données [Accessibles le 07/04/2020]

[1] https://www.ebi.ac.uk/arrayexpress/

[2] https://www.ncbi.nlm.nih.gov/geo/

[3] https://www.rcsb.org/

[4] https://www.ncbi.nlm.nih.gov/genbank/statistics/

[5] https://www.ebi.ac.uk/pride/

Tableau 1 – Quelques types de données emblématiques en bioinformatique. Les données sont à la fois hétérogènes et représentent des volumes importants.

c. Ce qu'il faut retenir

En conclusion, si les domaines d'application de la bioinformatique sont multiples (Figure 4), il existe des missions communes à cette discipline. Elles peuvent se résumer de la façon suivante : (1) assurer la disponibilité pérenne et organisée des données, (2) maintenir les outils d'analyse fonctionnels et utilisables par le plus grand nombre des scientifiques et (3) garantir les ressources informatiques suffisantes (capacités de calcul et de stockage) à l'ensemble des laboratoires de biologie. À l'échelle des différents domaines d'application, il s'agit d'un travail critique pour lequel une synchronisation des initiatives menées entre les chercheurs de différentes disciplines en biologie, représente une plus-value évidente.

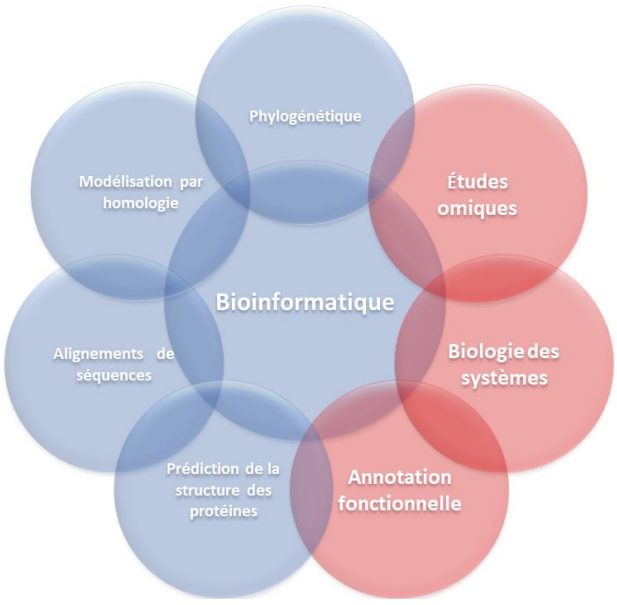


Figure 4 – Principaux domaines d'application de la bioinformatique. Les trois grands domaines abordés lors de cette thèse sont indiqués en rouge : « Études omiques », « Biologie des systèmes » et « Annotation fonctionnelle ». Cette figure est inspirée de l'article de M. Bilotta (Bilotta et al. 2018).

3. L'importance de la mutualisation des savoir-faire et des ressources

Conscients de l'intérêt de partager leurs ressources, les chercheurs en bioinformatique se sont organisés. À l'échelle nationale, la Société Française de BioInformatique (SFBI) a été créée en 2005. Son objectif est de promouvoir la recherche interdisciplinaire en bioinformatique, en rassemblant les chercheurs, les enseignants, les ingénieurs et les étudiants francophones qui travaillent sur des questions relevant de la bioinformatique. Ce rassemblement repose sur l'organisation de rencontres, opportunités d'échanges scientifiques (conférence JOBIM par exemple). La communauté des bioinformaticiens est aussi aidée par l'Institut Français de Bioinformatique (IFB)[15]. Il s'agit d'une infrastructure française de service en bioinformatique créée dans le cadre du programme national des « Investissements d'Avenir » (2010). L'IFB a pour missions de :

- Mutualiser, soutenir et coordonner le développement des ressources et des activités de support à la recherche par le biais de plateformes. Celles-ci dépendent d'organismes publics de recherche (CNRS, INRAE, INRIA, CEA et INSERM), des universités, etc. ;
- Proposer un accès à des services indispensables à la recherche, un accompagnement des projets nécessitant un fort niveau d'expertise en informatique, et la possibilité de participer à des projets collaboratifs au niveau national et international ;
- Proposer des formations pour les chercheurs souhaitant se former à la bioinformatique (comme l'école de bioinformatique de Roscoff) mais aussi pour les bioinformaticiens souhaitant actualiser et renforcer leurs connaissances (comme le DU Bioinformatique Intégrative[16] de l'Université de Paris).

L'IFB s'inscrit dans une initiative à grande échelle en étant le nœud français de l'infrastructure européenne de bioinformatique ELIXIR (ESFRI). ELIXIR réunit les principales organisations européennes des sciences de la vie pour la gestion et la sauvegarde du volume croissant de données générées par la recherche financée par des fonds publics. À l'aide de ces infrastructures, la bioinformatique moderne a les moyens de répondre au défi de l'analyse de données massives.

[15] https://www.france-bioinformatique.fr/ [Accessible le 04/08/2020]

[16] https://ressources.france-bioinformatique.fr/fr/diplome-universitaire-en-bioinformatique-integrative-du-bii-2020 [Accessible le 05/08/2020]

II. Les fondements de l'analyse de données

Si la bioinformatique est définie de façon assez variable, en fonction des disciplines d'application et des époques, l'analyse de données est, quant à elle, définie de manière très homogène dans la littérature. Il s'agit du « *processus d'inspection, de nettoyage, de transformation et de modélisation des données, dans le but de découvrir des informations utiles, d'éclairer la conclusion et d'appuyer la prise de décision* » (Wikipédia, avril 2020). Cette définition est cohérente avec les définitions proposées par exemple dans le dictionnaire *Cambrige*[17] ainsi que celle du cours Internet GURU99[18]. Il est intéressant de souligner que les notions de « prise de décision » ou de « conseils » sont présentées systématiquement comme les objectifs finaux de l'analyse de données.

1. L'importance de différencier « donnée », « information » et « connaissance »

Que ce soit oralement ou par écrit, les termes « donnée », « information » et « connaissance » sont souvent utilisés de façon concomitante. Mais sont-ils interchangeables pour autant ? Personnellement, j'ai appréhendé récemment une réflexion sur leur signification et les implications associées dans un processus d'analyse de données. Dans cette partie, je vous présente le résultat de mes réflexions. Afin d'être le plus clair possible, je vais m'aider d'une autre de mes passions : la cuisine (exemple de la confection d'un gâteau au chocolat).

a. Donnée

Définition générale

Une donnée est « *un élément brut qui n'a pas encore été interprété, mis en contexte* » (Chaudet 2009). Les données « *sont collectées par observations* » (*Glossary of statistical terms*)[19]. Il s'agit d'une « *description élémentaire d'une réalité, le résultat d'une comparaison entre deux*

[17] « *Un processus d'examen de l'information, particulièrement à l'aide d'un ordinateur, dans l'objectif de trouver quelque chose de nouveau ou d'aider à la prise de décision* ».

[18] *« L'analyse des données est définie comme un processus de nettoyage, de transformation et de modélisation des données pour découvrir des informations utiles pour la prise de décision commerciale. Le but de l'analyse des données est d'extraire des informations utiles des données et de prendre la décision en fonction de l'analyse des données* ».

[19] https://stats.oecd.org/glossary/detail.asp?ID=532 [Accessible le 05/08/2020]

événements du même ordre soit en d'autres termes une observation ou une mesure » (Wikipédia, avril 2020). Ainsi, les données peuvent prendre différentes formes[20] : nombre, texte, audio, vidéo, etc. Cette diversité permet de distinguer deux types de données : les données structurées et les données non structurées.

Données structurées

Ces données sont généralement organisées dans des bases de données. Il est possible de les récupérer facilement, soit grâce à une interface WEB comme c'est le cas pour les bases de données biologiques les plus célèbres (GEO[21], SRA[22], Pride[23], etc.), soit en écrivant des requêtes en langage informatique. Ci-dessous, un exemple de données structurées (Tableau 2). Il s'agit d'un tableau avec en ligne un ensemble de gènes et en colonne des variables quantitatives.

	logFC 1	...	*logFC m*
Gène 1	2.05	…	1.85
Gène 2	1.85	…	0.57
Gène 3	0.02	…	-0.06
...	…	…	…
Gène n	-3.59	…	-2.46

Tableau 2 – Tableau fictif illustrant la notion de « données structurées ». Les gènes sont présentés en ligne, tandis que les colonnes correspondent à différentes variables étudiées. Les valeurs numériques pourraient correspondre à des mesures de différentiels de l'expression des gènes, observées entre deux conditions différentes de cultures de cellules (ce sont des logFC).

Données non structurées

Ce type de données est plus complexe à utiliser. Il faut réaliser un travail préalable afin de les organiser. Un exemple de données non structurées est le texte des articles scientifiques ou celui

[20] Des contenus supplémentaires proposés dans la ressource numérique de la thèse : https://thomasdenecker.github.io/thesisWebsite/annexes/data/ [Accessible le 08/08/2020]

[21] https://www.ncbi.nlm.nih.gov/geo/ [Accessible le 22/08/2020]

[22] https://www.ncbi.nlm.nih.gov/sra [Accessible le 14/04/2020]

[23] https://www.ebi.ac.uk/pride/archive/ [Accessible le 14/04/2020]

des pages WEB. Ci-dessous, je présente une donnée non structurée. Il s'agit de la description du gène *FET3* de la levure *Saccharomyces cerevisiae*, issue de la base de données SGD[24]. Nous pouvons observer la description d'une relation fonctionnelle avec le gène *FTR1* :

> " *Ferro-O2-oxidoreductase; multicopper oxidase that oxidizes ferrous (Fe2+) to ferric iron (Fe3+)* ***for subsequent cellular uptake by transmembrane permease Ftr1p****; required for high-affinity iron uptake and involved in mediating resistance to copper ion toxicity, belongs to class of integral membrane multicopper oxidases; protein abundance increases in response to DNA replication stress*"

Dans ce contexte, comment observer d'autres relations fonctionnelles entre des gènes ? Par exemple comment collecter toutes les relations de régulation transcriptionnelle entre des protéines régulatrices (facteurs de transcription) et des gènes cibles, décrites dans les articles scientifiques ? En informatique, les algorithmes de *text mining* ont pour objectif de rechercher automatiquement ce type de relations. Une curation manuelle est également possible, même si cela représente un travail très fastidieux. J'ai beaucoup utilisé l'outil Web PathoYeastract [25] qui propose les résultats de ce type de travail, c'est-à-dire une structuration de données de régulations transcriptionnelles à partir des articles scientifiques.

Conclusion

Une caractéristique majeure de la donnée est qu'elle correspond à une observation, ni plus ni moins. Quelle soit structurée ou non structurée, elle n'a pas encore de sens biologique. C'est ainsi que pour créer une base de données, un informaticien n'a pas besoin de connaître le sens des données qu'il manipule, mais seulement leur structure. Lors de la confection d'un gâteau au chocolat, les données seraient, issues d'un livre de cuisine quelconque, « *il vous faut des œufs, de la farine, du beurre et du chocolat* ». Pour quoi faire ? On ne sait pas, mais cela n'est pas un problème pour aller faire ses courses !

b. Information

Le mot information vient du latin *formare* qui signifie « mettre en forme » (Larousse, avril 2020). Une information est ainsi une donnée associée à un sens, une interprétation. Si les

[24] https://www.yeastgenome.org/locus/S000004662 [Accessible le 27/03/2020]
[25] http://pathoyeastract.org/ [Accessible le 27/03/2020]

différents ingrédients nécessaires à la confection d'un gâteau au chocolat sont associés correctement, et que l'ensemble est cuit à la bonne température pendant un temps adapté, un excellent gâteau au chocolat est obtenu. Les données, ici les ingrédients, ont été organisées et associées en suivant une méthodologie d'analyse précise (la recette). De façon similaire, une expérience RNAseq permet d'obtenir des « données de comptage[26] » pour chaque gène de l'organisme étudié. Ces données sont ensuite converties en « information », par exemple en information de comparatif d'expression (le gène A est plus exprimé dans la condition 1 que dans la condition 2). Cette information est plus ou moins fiable en fonction des données sous-jacentes (niveau de reproductibilité des observations entre réplicats par exemple), à l'instar du gâteau au chocolat pour lequel des ingrédients de qualité permettent d'obtenir un meilleur goût.

c. Connaissance

Définition

Une connaissance est « *une information comprise, c'est-à-dire assimilée et utilisée qui permet d'aboutir à une action* » (Chaudet 2009). Mes lectures m'ont permis de distinguer deux types de connaissances (Nonaka et Takeuchi 1995) : la connaissance tacite et la connaissance explicite.

Connaissance tacite

Il s'agit de la connaissance que possèdent les individus. Elle n'est pas formalisée et elle est difficilement transmissible. C'est par exemple la recette secrète de la buche de Noël faite par ma grand-mère : « - *Combien de grammes de farine ? - Aucune idée, tu le sens au poignet lorsqu'il y en a assez* ». Ce sont des connaissances qui se transmettent par imitation et imprégnation. Nous le savons sans le savoir, nous les utilisons sans vraiment nous en rendre compte. Ces connaissances tacites sont utiles en analyse de données. Associées à l'expérience des chercheurs, les connaissances tacites orientent les choix des méthodes, des outils et des représentations.

[26] Les données de comptage correspondent généralement au nombre de séquences (ou *reads*) alignées au niveau des gènes.

Connaissance explicite

Il s'agit de la connaissance formalisée et transmissible sous forme de documents réutilisables. Pour reprendre l'exemple de la connaissance tacite, la connaissance explicite correspond à la recette détaillée du gâteau au chocolat. L'ingrédient « secret » est nommé, quantifié, et une explication de son intérêt dans la recette est connue. Par exemple, ajouter de la fleur de sel au mélange de la pâte renforcera le gout du chocolat. Ainsi, à l'issue d'une expérience de RNAseq, une fois l'information « le gène A est plus exprimé dans la condition 1 que dans la condition 2 » obtenue, le défi consiste à utiliser cette information pour créer de la connaissance. Cette connaissance est, par exemple, l'existence d'un régulateur transcriptionnel, actif sur le gène A uniquement dans la condition 1.

Plusieurs informations pour une seule connaissance

Il est important de souligner qu'à cette étape du processus, de très nombreuses informations, souvent hétérogènes, sont associées pour produire une connaissance certaine, qu'elle soit tacite ou explicite. Ainsi, démontrer le rôle d'un facteur de transcription consiste à étudier une souche délétée pour le gène codant ce facteur de transcription (expériences de puces à ADN ou RNAseq), étudier la fixation du facteur de transcription au niveau du promoteur du gène (expérience de ChIPseq), etc. Un exemple de méthodologie pour confirmer l'existence d'un facteur de transcription est proposée dans la publication de J. M. Vaquerizas (Juan M. Vaquerizas, Sarah A. Teichmann et al. 2012).

d. Ce qu'il faut retenir

Le plus souvent inconsciemment, les termes « donnée », « information » et « connaissance » sont utilisés en décalage avec leur sens initial. J'ai pu noter, au cours de ma thèse, l'importance de définir avec précision les mots employés. Les acronymes en sont un parfait exemple. Pour vous, qu'est-ce qu'une IP ? J'ai posé la question autour de moi. Personnellement, la première chose qui me vient à l'esprit c'est « *Internet Protocol* », mais pour mes collègues biologistes, il s'agira d'une « Immuno-Précipitation ». Pour mes proches dans le domaine de la Santé, IP signifie « Intervention Pharmaceutique » et enfin il signifie « *Intellectual Property* » pour mes amis qui travaillent dans l'industrie. Cet exemple peut sembler anecdotique, mais qu'en est-il de terminologies plus complexes telles que « modèle » ou « gène ». Discuter les représentations mentales associées à ces termes avec un biologiste et un statisticien, peut aboutir à des échanges vifs. À nouveau, la prise en compte du contexte est essentielle : un organisme modèle en

biologie, un modèle statistique, un modèle en base de données, etc. Cette notion de pluralisme est évoquée comme une difficulté majeure dans l'article *The challenges of big data biology* de S. Leonelli (Leonelli 2019). Son exemple repose sur la définition des termes « métabolisme » et « pathogène ».

En résumé, une donnée qui a un sens est une information. Une information qui est comprise est une connaissance. Ces notions peuvent être représentées à l'aide d'une pyramide comme celle présentée Figure 5. L'association de ces mots est nommée le modèle DIC (Boubaker et al. 2010).

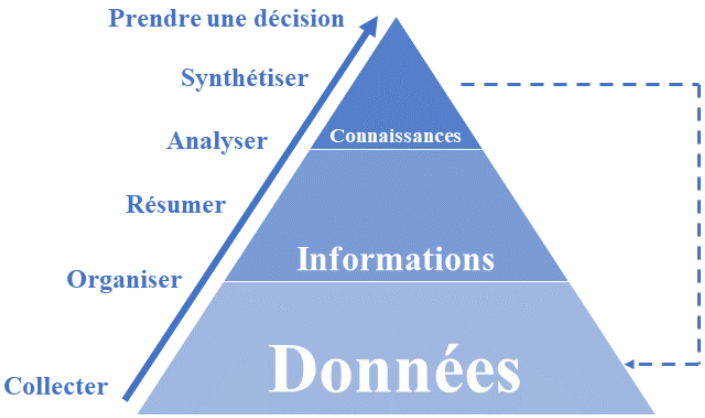


Figure 5 – Représentation pyramidale du modèle DIC inspiré de l'article de J. David (David 2019). *Les données sont associées aux observations, les informations résultent des analyses des données et enfin les connaissances émergent des informations. Pour cela il est nécessaire de prendre en compte un contexte aussi vaste que possible (problématique de l'intégration d'informations issues de données hétérogènes).*

2. L'influence de la visualisation : le couplage œil – cerveau

a. Car « une image vaut mille mots »

Comme dit le proverbe : "*une image vaut mille mots*" (Confucius). En effet, l'esprit humain est très visuel et l'illustration des savoirs est un besoin très ancien (Aparicio et al. 2014). Les supports ont changé au cours du temps (papyrus, toile, photographie, vidéo, etc.), mais nous sommes toujours animés d'un désir de création d'images. Nous communiquons nos émotions avec des « *smileys* » et nous montons nos meubles à partir de notices ne contenant plus que des illustrations. La visualisation aide à comprendre, à ressentir ou à penser. G. Lelandais et J.M. Camadro ont souvent évoqué devant moi l'importance du « couplage œil cerveau ». Parmi les

moyens de communications existants, je vais en présenter deux qui sont particulièrement utilisés : la visualisation de données et l'infographie.

b. Différencier visualisation de données et infographie

Visualisation de données

La visualisation de données (ou *data visualisation* ou encore en abrégé *dataviz*) désigne la représentation graphique de données. Autrement dit, comment convertir des ensembles de données structurées (voir page 42), quantitatives ou qualitatives, en des objets visuels, c'est à dire des points, des barres, des courbes, des cartographies, etc. La visualisation de données permet de simplifier l'exploration et l'analyse de données structurées. C'est également un outil de communication puissant. La visualisation de données telle que nous la connaissons aujourd'hui, est apparue au XVII[e] siècle, par le biais de W. Playfair. Il a créé différents types de diagrammes pour illustrer ses recherches dans le domaine de l'économie. Par la suite, il a écrit un livre appliquant ces techniques de représentations (Playfair 1801) (voir l'exemple de la Figure 6).

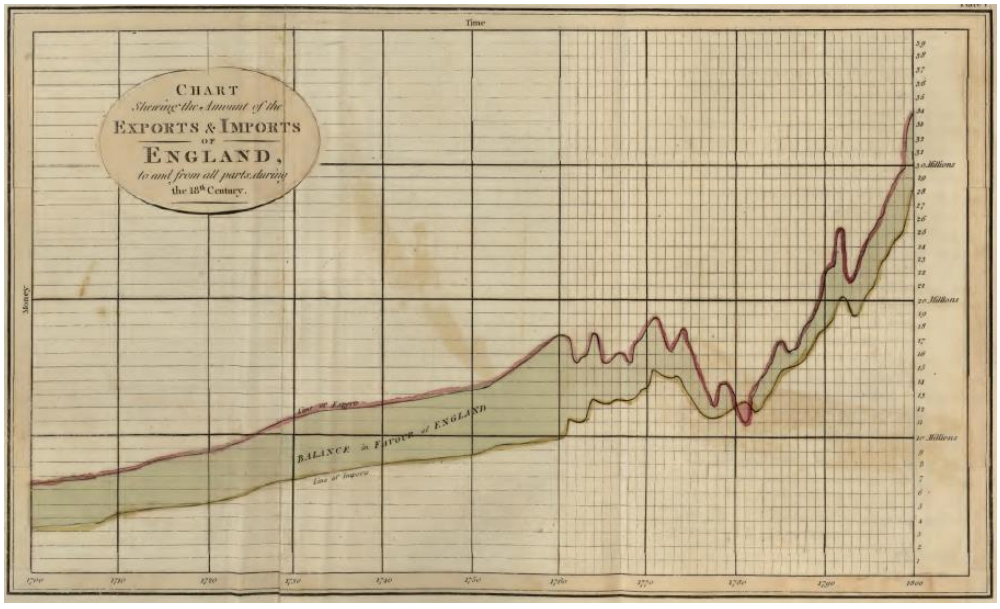


Figure 6 – Nombre d'importations et d'exportations en Angleterre au XVIII[e] siècle. Cette figure est extraite du livre de William Playfair publié en 1801. Elle permet, très simplement, d'observer la corrélation positive entre ces deux variables, ainsi que leur augmentation au cours du temps. Cette image est librement accessible en ligne.[27]

27 https://archive.org/details/PLAYFAIRWilliam1801TheCommercialandPoliticalAtlas/page/n13/mode/2up [Accessible le 28/04/2020]

Infographie

Une infographie est une représentation visuelle d'informations (et pas uniquement des données comme précédemment). Une infographie se compose de plusieurs éléments comme des graphiques (visualisation de données), des photos ou dessins, ainsi que des éléments de texte (Figure 7). L'objectif d'une infographie est de proposer une vue d'ensemble d'un sujet, rapidement compréhensible malgré la complexité éventuelle du sujet. Les premières infographies sont apparues au cours du XVII[e] siècle avec le diagramme de Coxcomb de F. Nightingale et la carte figurative de C. Minard (considéré comme le pionnier de l'infographie)[28].

A **B**

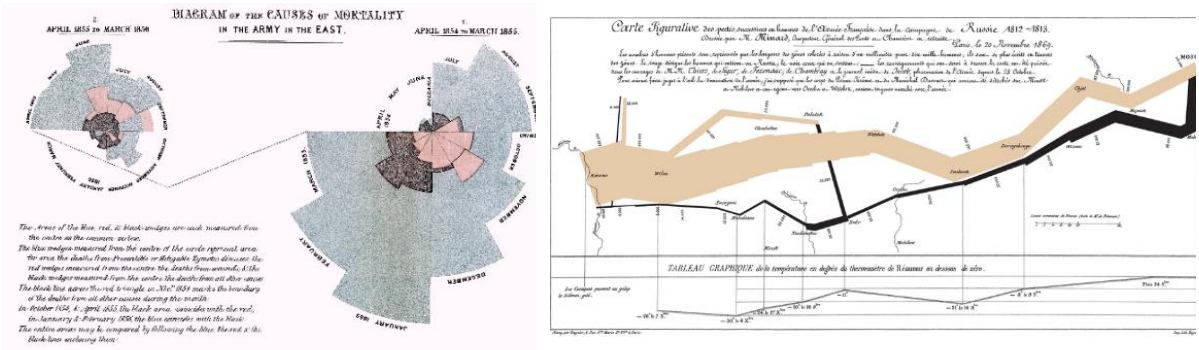


Figure 7 – Les premières infographies de l'histoire apparues successivement au XVII[e] siècle. (A) Le diagramme de Coxcomb de Florence Nightingale représentant les pertes britanniques pendant la guerre de Crimée (1858)[29]. (B) La carte figurative des pertes successives en hommes de l'armée française lors de la campagne de Russie 1812-1813, dessinée par Charles Minard, (1869[30]).

Intérêt de différencier les deux

Ces deux procédés sont largement confondus. Cette confusion est compréhensible car dans les deux cas, les objectifs sont communs : il s'agit de communiquer un message, une idée. L'influence de la visualisation est grandissante ces dernières années, grâce notamment aux problématiques engendrées par le *Big data* (Figure 8). D'après le mathématicien C. Humby,

[28] Le site https://visionscarto.net/charles-joseph-minard-cinquante-cartes regroupe les principales cartes crées par Charles Minard. [Accessible le 04/08/2020]

[29] https://commons.wikimedia.org/wiki/File:Nightingale-mortality.jpg [Accessible le 29/04/2020]

[30] https://commons.wikimedia.org/w/index.php?curid=297925 [Accessible le 29/04/2020]

« *les données sont le nouveau pétrole* »[31], et ces 5 dernières années, plus d'une centaine de livres ont été publiés sur les problématiques de visualisation. Alors quelles sont les différences ? Je dirais que la visualisation de données est une procédure exploratoire. Elle incite à la réflexion et à la formulation de nouvelles hypothèses. Elle est facilement automatisable parce qu'elle exploite des données structurées. Elle est composée de peu d'illustrations (du type photos ou schémas), mais elle peut être interactive. Un spécialiste dans le domaine est H. Rosling. Il a imaginé une visualisation dynamique de l'évolution des pays au cours du temps, en fonction de l'espérance de vie et des revenus annuels par habitant[32] (Rosling et al. 2018). Comme il le conseille « *laissez le jeu de données vous faire changer d'avis* »[33].

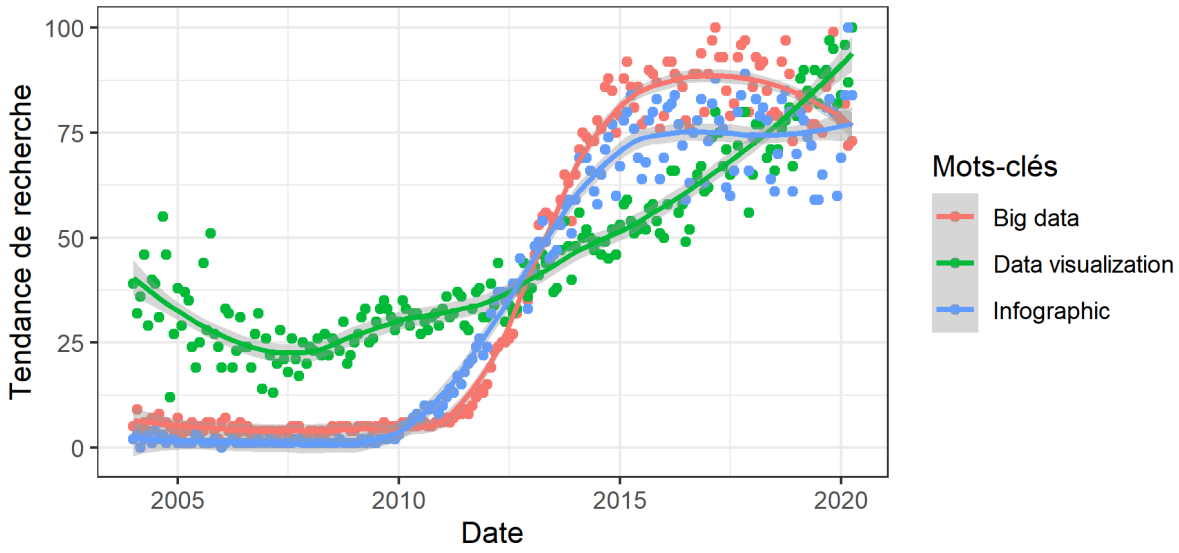


Figure 8 – Évolution de l'intérêt porté aux problématiques de la visualisation depuis 2004. Cette figure a été réalisée à partir de données extraites de Google Trends[34]. Le code pour réaliser cette figure est disponible sur GitHub Gist[35]. La tendance de recherche d'un mot-clé correspond au nombre de recherches réalisées à un moment donné et rapporté entre 0 et 100 en fonction du plus grand nombre de recherches pour ce même mot-clé.

L'infographie, quant à elle, a un objectif de synthèse, de vulgarisation des connaissances. Une place plus importante est laissée à la créativité et à l'originalité, comme pour la conception d'une toile de peinture. Elle est riche en illustrations et exploite la faculté impressionnante des

[31] "*Data is the new oil.*" (2006)
[32] https://www.youtube.com/watch?v=jbkSRLYSojo [Accessible le 30/04/2020]
[33] "*Let the dataset change your mindset*"
[34] https://trends.google.fr/trends/?geo=FR [Accessible le 01/05/2020]
[35] https://gist.github.com/thomasdenecker/b1996dfdc8ec2c2b71ca747c96c872e6 [Accessible le 01/05/2020]

humains à traiter des informations visuelles (Figure 9). La réalisation d'une infographie est difficilement automatisable et en conséquence, est généralement figée dans le temps. Un spécialiste en la matière est D. McCandless qui a créé le site informationisbeautiful.net[36] et a publié 3 livres regroupant de très nombreuses infographies qu'il explique notamment lors de conférences TEDx[37] (McCandless 2000; 2009; 2014). Vous trouverez ci-dessous deux de ses infographies (Figure 9 et Figure 13).

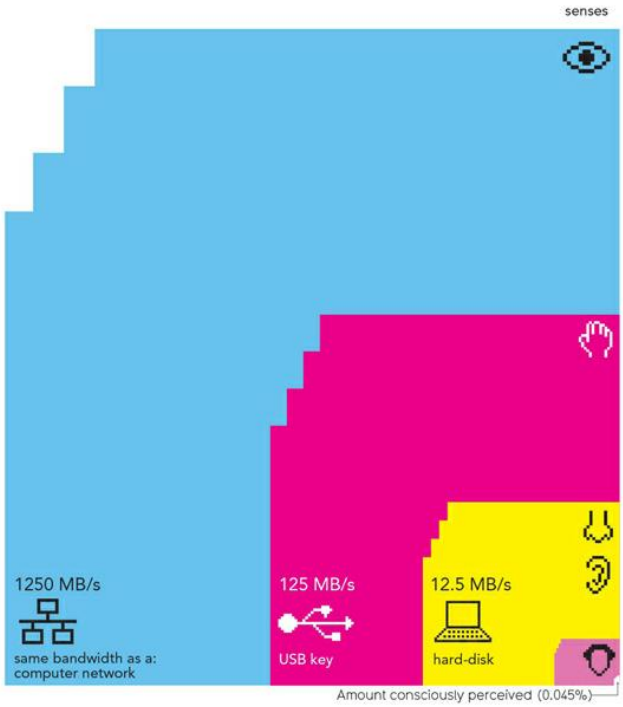


Figure 9 – Infographie comparant la vitesse de traitement des informations reçues par nos sens en les rapportant à des flux informatiques. Les performances de l'œil sont comparables à la bande passante d'un câble Ethernet, les performances du touché sont comparables à la vitesse de lecture d'une clé USB et enfin les performances de l'odorat et de l'audition sont comparables à celles d'un disque dur. Cette infographie est proposée par David McCandless (McCandless 2014).

En conclusion, il existe une forte notion d'interprétation en infographie. La raison est que les infographies reposent sur des informations et des connaissances. Nous avons vu dans le chapitre précédent que ces notions doivent être différenciées des données initiales. Ce sont les données que visualisation de données a pour objectif d'étudier, « *de faire parler* » comme certains

[36] https://informationisbeautiful.net/ [Accessible le 30/04/2020]

[37] https://www.youtube.com/watch?v=pLqjQ55tz-U [Accessible le 30/04/2020]

collègues le disent. Je propose ci-dessous un récapitulatif de l'enchaînement de différents concepts présentés jusque-là (Figure 10).

Figure 10 – Enchaînement des concepts présentés dans cette première partie du manuscrit de thèse. Les concepts de « données », « analyses de données », « informations » et « connaissances » sont présentés dans le chapitre précédent. Les concepts de « visualisation de données » et « infographie » sont présentés dans ce chapitre.

c. Règles de l'art

La visualisation de données et l'infographie reposent sur un outil commun : le graphe (historiquement nommé « graphique » ou plus récemment nommé « visualisation »). Dans son livre *Good Charts* (Berinato 2016), S. Berinato distingue quatre types de graphes (Figure 11). Ceux qui exploitent une visualisation déclarative, une visualisation conceptuelle, une visualisation exploratoire et enfin une visualisation fondée sur les données.

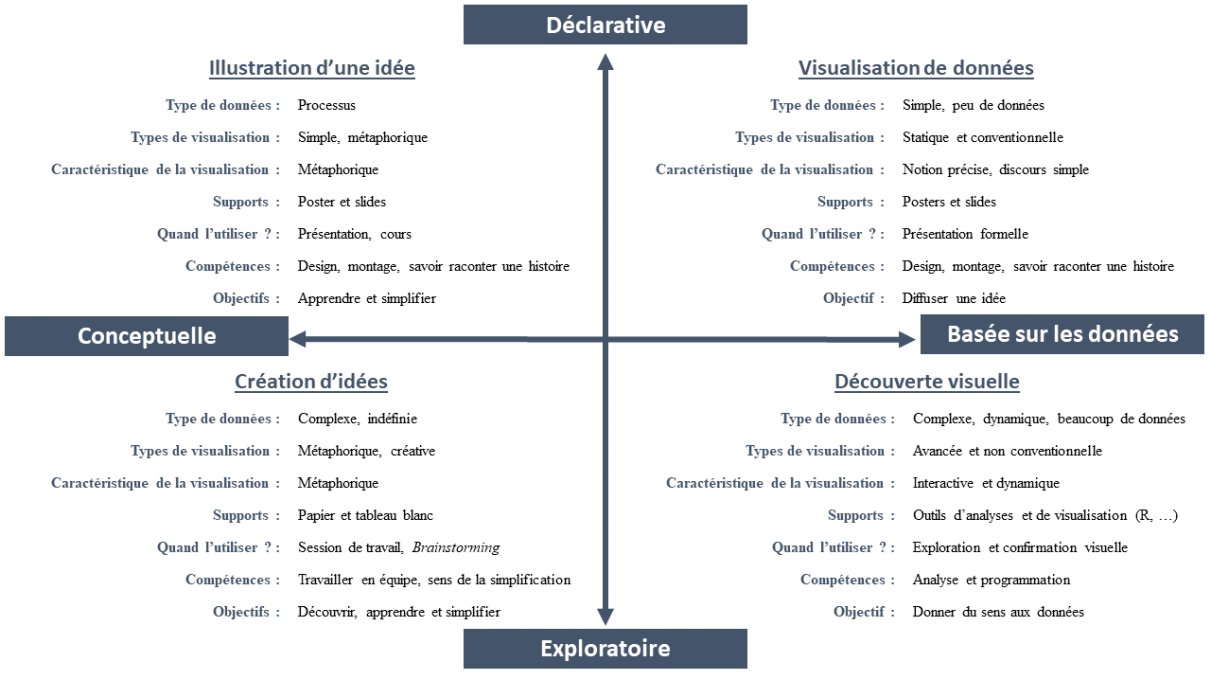


Figure 11 – Interprétations associées aux 4 types de graphes proposées par S. Berinato dans son livre Good Charts (Berinato 2016). *Ces graphiques sont associés à des tâches différentes : illustration d'une idée, visualisation de données, découverte visuelle ou création d'idées.*

Dans ce contexte, il définit un « bon graphique » comme la combinaison d'une très bonne connaissance du contexte avec une excellente organisation graphique (Berinato 2016). Cela souligne une fois de plus, l'importance de définir le contexte dans lequel les données exploitées ont été produites tout d'abord, puis le contexte dans lequel elles ont été transformées en informations et connaissances. Concernant la conception de l'organisation graphique, le défi est important. Il repose en effet sur des connaissances souvent implicites (voir page 45) : « - *Pourquoi cette figure est belle ? – Je ne sais pas, mais elle belle…* ».

Un certain nombre de règles de l'art sont toutefois présentées dans la littérature. Dans son livre Sémiologie graphique (Bertin 1968), J. Bertin définit huit "variables visuelles" avec lesquelles les données peuvent être représentées. Il s'agit de la position (plan en deux dimension), la taille, la forme, la couleur, la valeur (que nous pouvons comparer à l'intensité de la couleur), l'orientation et le grain (Figure 12).

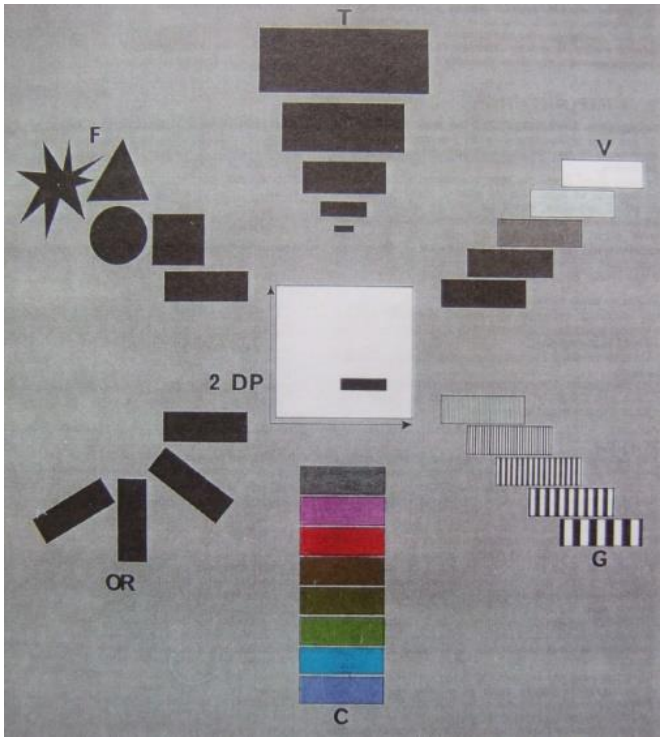


Figure 12 – Variables visuelles selon Jacques Bertin. La figure a été extraite de son ouvrage Sémiologie graphique. Les diagrammes. Les réseaux. Les cartes (1968). 2 DP pour plan à deux dimension, G pour Grain, V pour valeur, T pour taille, F pour forme, OR pour orientation et C pour couleur.

En fonction des choix réalisés, l'impact du message sur notre pensée ne sera pas le même. Ce phénomène est bien documenté par exemple pour les choix de couleurs (Figure 13). La chance et la bonne fortune sont associées à la couleur verte dans les pays de l'ouest de l'Europe et aux

États-Unis, alors qu'en Chine, elles sont associées à la couleur rouge. Le rouge est en Europe une couleur plutôt associée à la colère et au danger.

Ainsi, il existe un réel « art de la visualisation ». Pour optimiser l'impact d'un message, J. Bertin conseille de suivre un principe d'efficacité. L'idée est simple : « *si pour obtenir une réponse correcte et complète à une question donnée, à partir des mêmes choses, une construction requiert un temps d'observation plus court qu'une autre construction, on dira qu'elle est plus efficace pour cette question* ». Cette définition est basée sur la notion de « coût mental » proposée par G. K. Zipf (Zipf 1935)[38]. Plus ce coût est faible plus l'information est communiquée facilement. La prise en compte des spécificités de notre cerveau est un élément important en visualisation de données, c'est le couplage œil-cerveau.

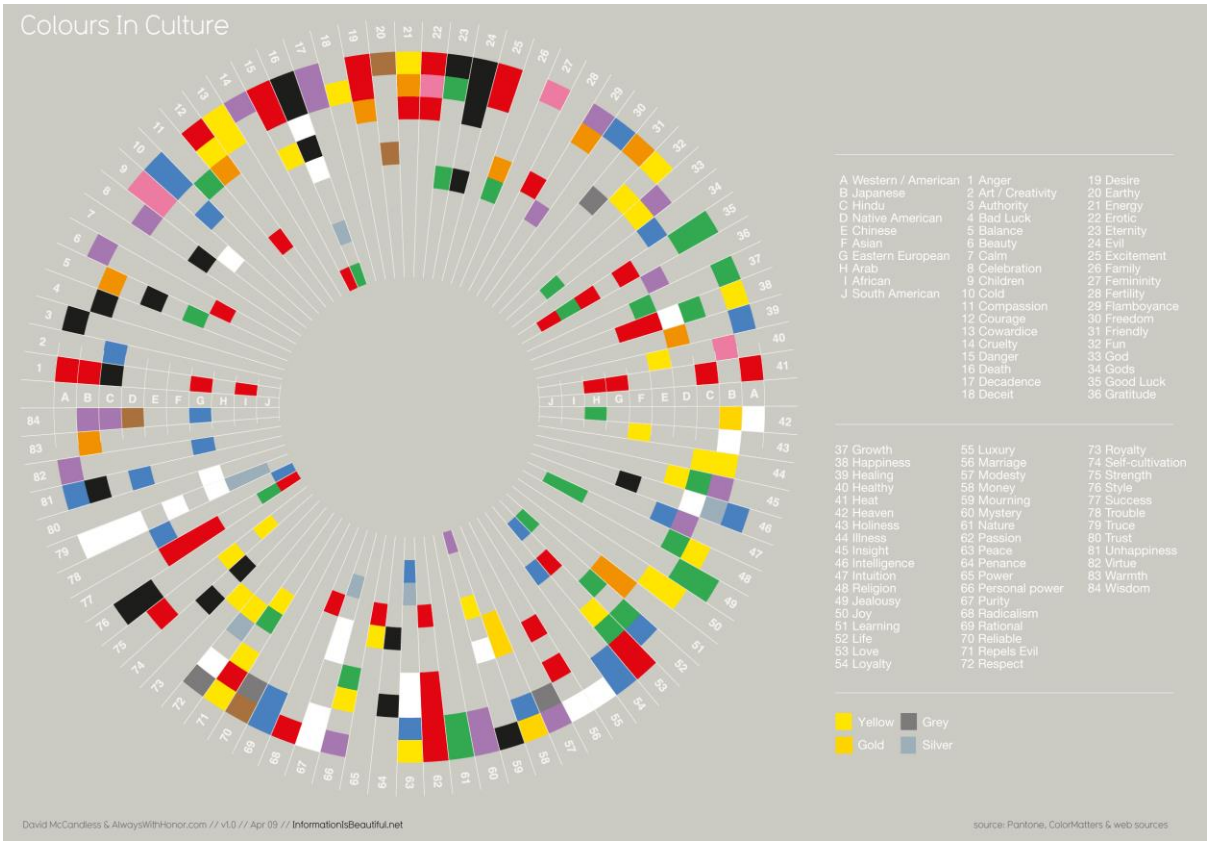


Figure 13 – Signification des couleurs en fonction des cultures. Cette visualisation est extraite du site informationisbeautiful.net.[39]*. Cette infographie est aussi utilisée sur la page de garde de son livre Information is beautiful* (McCandless 2000).

[38] Le livre est accessible ici : https://hdl.handle.net/2027/mdp.39015008729983 [Accessible le 04/08/2020]

[39] https://informationisbeautiful.net/visualizations/colours-in-cultures/ [Accessible le 29/04/2020]

Enfin E. Tufte, considéré comme le père moderne de la visualisation de données, établit 5 grands principes[40] (Tufte 2001) : (1) La représentation des nombres doit correspondre aux vraies proportions ; (2) L'étiquetage doit être clair et détaillé; (3) Il faut éviter les confusions entre les variations de *design* graphique et les variations de données; (4) Le nombre de dimensions représentées doit être le même que le nombre de dimensions dans les données ; (5) Les graphiques ne doivent pas citer des données hors contexte.

d. Ce qu'il faut retenir

Pour résumer, l'excellence graphique est atteinte lorsque « *des idées complexes sont communiquées clairement, précisément et efficacement […] Les graphiques révèlent les données* »[41]. Au cours de ma thèse, la visualisation de données et les infographies ont eu un rôle important dans les projets auxquels j'ai participé. Pour réaliser les figures de notre article en relation avec l'étude de l'homéostasie du fer (voir page 149), nous avons mis en application les 4 types de visualisations proposés par S. Berinato (Figure 14). La visualisation de données et l'infographie ont ainsi été les piliers pour la mise en forme des résultats, mais aussi pour raconter une histoire cohérente[42].

[40] Ces principes sont ici seulement cités. Dans la ressource complémentaire numérique, nous montrons par divers exemples, comment en détournant ces principes, les graphiques peuvent tromper les personnes qui les regardent. https://thomasdenecker.github.io/thesisWebsite/annexes/graphiques/ [Accessible le 08/08/2020]

[41] "*complex ideas communicated with clarity, pression and efficiency […] Graphics reveal data*" (page 13).

[42] Notion de *data storytelling.*

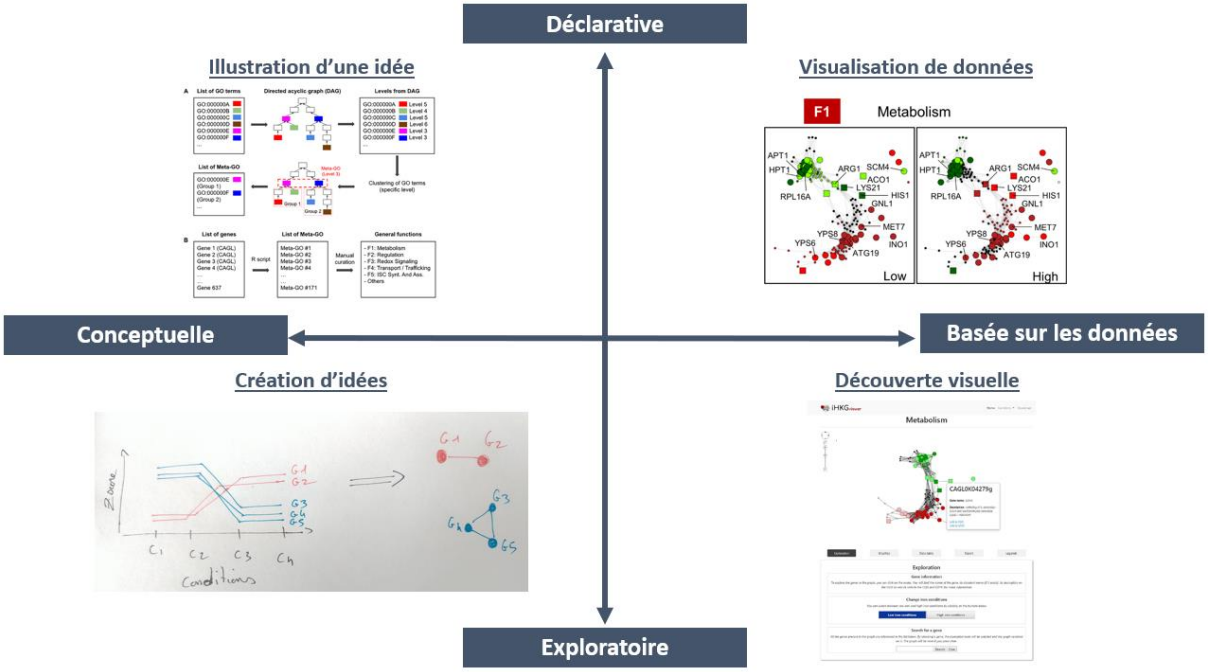


Figure 14 – Mise en application des 4 types de visualisations « déclarative », « conceptuelle », « exploratoire » et « basée sur les données ». Ces principes sont détaillés Figure 11, page 52. Ce travail a été réalisé dans le contexte du projet d'étude de l'homéostasie du fer de la levure pathogène Candida glabrata (voir page 128).

3. Les différentes étapes d'une analyse de données

Je retiens de ces trois années de thèse qu'une analyse de données ne se passe généralement pas comme prévu. Plus complexe ou bien plus longue qu'anticipé, il manque fréquemment un élément. Comment seraient les résultats si tel ou tel paramètre était modifié ? Faut-il continuer ? Les résultats ne sont pas ceux initialement imaginés, que faire ? Faut-il recommencer ? Voici des questions qui ne m'ont pas quitté. La rédaction de ce chapitre a été l'opportunité d'essayer de prendre du recul vis-à-vis de mes premières expériences en tant que « analyste de données ». J'ai essayé de décrire et discuter ce que devraient être les différentes étapes d'un travail raisonné. Ma principale prise de conscience est qu'une analyse de données n'est pas un processus linéaire, mais plutôt un processus cyclique (Peck et al. 2016). Le défi pour un chercheur qui débuterait dans le domaine est ainsi de ne pas rester bloqué dans un cycle perpétuel.

a. Cycle d'analyse

Idéalement, un travail en analyse de données se décompose en 6 grandes étapes (Figure 15) : (1) Formulation de la question scientifique ; (2) Recherche, collecte des jeux de données disponibles, utiles pour répondre à la question scientifique ; (3) Préparation des données,

vérification de l'intégrité de leur structure ; (4) Étude exploratoire et analyses préliminaires ; (5) Formulation d'hypothèses statistiques, étude de validation ; (6) Interprétation et conclusion.

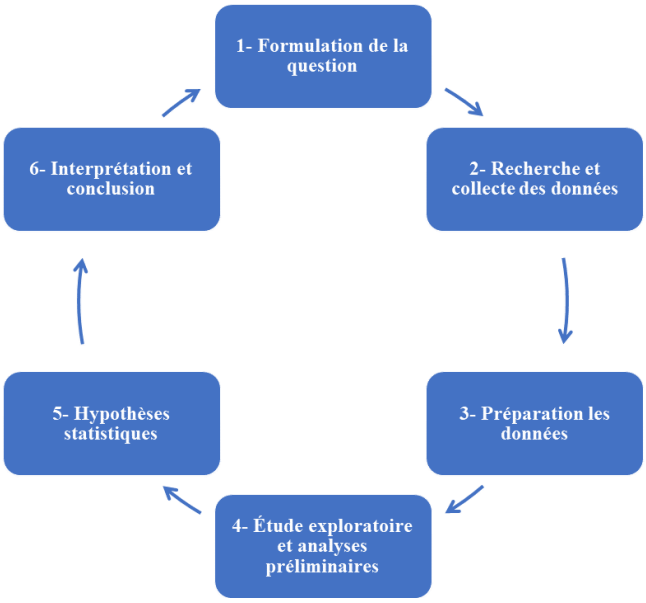


Figure 15 – Illustration du processus cyclique d'une analyse de données. Six grandes étapes sont présentées sur cette figure, telles que décrites dans l'ouvrage Introduction to Statistics & Data analysis de R. Peck (Peck et al. 2016).

b. Détails des différentes étapes

1- Formulation de la question scientifique

C'est sans doute l'étape la plus importante du projet. Une formulation précise de la question scientifique permet de définir plus clairement les objectifs visés, les résultats attendus et ainsi d'anticiper les difficultés associées. Il est intéressant de garder en mémoire la définition d'une analyse de données (page 42). Pour quelles problématiques une « prise de décision » ou des « conseils » sont-ils nécessaires ? Ainsi, dire « *je cherche à comprendre les différences de fonctionnement de ces cellules dans un milieu qui contient ou ne contient pas de fer* » (question vague, réponses possibles multiples, à quel niveau se situe la prise de décision ?) est différent de « *je cherche tous les gènes dont les mesures normalisées de logFC obtenues par puces à ADN entre les conditions A et B sont différentes, avec un risque d'erreur associé de 1%* » (question précise, réponses possibles limitées, prise de décision fondée sur un test statistique). Alors bien sûr, une question générale comme la première est plus attractive sur le plan scientifique. L'idée est ici clairement de créer de nouvelles connaissances : le but ultime du chercheur. Mais c'est à ce niveau que réside une erreur typique en analyse de données. En effet, il ne me semble pas possible de créer de la connaissance directement à partir des données.

Il est nécessaire de produire dans un premier temps des informations[43]. Or, les informations sont justement obtenues en répondant à des questions plus « terre à terre », telles que la deuxième question énoncée ci-dessus. Ainsi, la juxtaposition de cycles d'analyses initiés chacun sur des questions simples, précises, permet de produire de la connaissance.

2- Recherche et collecte des jeux de données

Être bioinformaticien en 2020, est pour cette étape un grand avantage. Nous l'avons vu précédemment, de nombreuses données sont disponibles dans les bases de données publiques. Il est ainsi possible d'envisager de répondre à de très nombreuses questions scientifiques sans avoir à créer de nouvelles données, seulement en réexploitant celles existantes. Bien sûr, il existe un risque notamment concernant la qualité des données mais cela importe peu finalement, l'essentiel étant de pouvoir produire des informations à partir de ces données. Associées les unes avec les autres, ces informations permettront d'imaginer de nouvelles expériences, qui elles (une fois analysées) permettent de démontrer de nouvelles connaissances. Une erreur typique en analyse de données, consiste pour cette étape à récupérer un maximum de données « au cas où ». Il est au contraire important de ne choisir que les données qui sont pertinentes vis-à-vis de la question scientifique initiale, il peut s'agir de données structurées ou non (notions définies page 42).

3- Préparation des données

Cette étape est plus ou moins complexe, selon les données choisies. Comme évoqué précédemment, il est plus difficile de travailler avec des données non structurées. Toutefois, même les données structurées doivent être préparées et vérifiées avant de débuter leur analyse. Il s'agit par exemple de combiner plusieurs tableaux, éliminer les valeurs manquantes, repérer des valeurs aberrantes ou redondantes, etc. Peu gratifiante, cette étape est pourtant essentielle. Les taches importantes pour préparer un jeu de données sont expliquées par O. Elgabry, dans un guide très complet illustré par des exemples (Elgabry 2019). Un sondage présenté *dans Data Science report* (CrowdFlower 2016) quantifie à 60% le temps d'une analyse de données qui est consacré à cette étape (Figure 16).

[43] Les différences entre « donnée », « information » et « connaissance » sont décrites page 42.

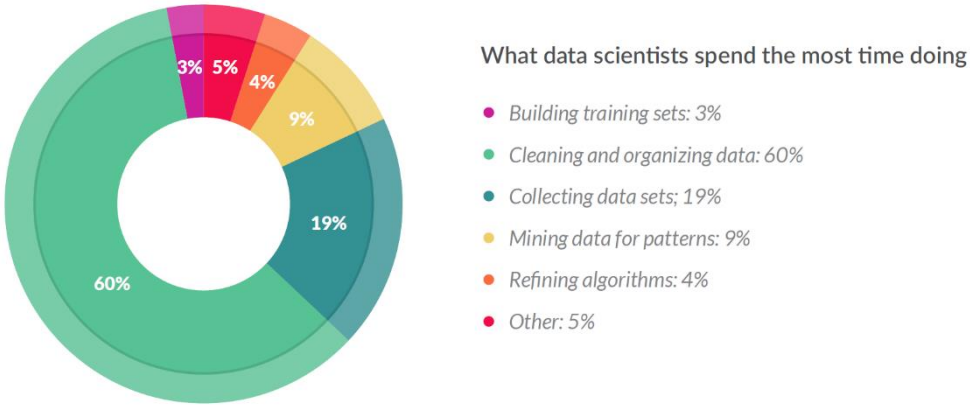


Figure 16 – Répartition du temps de travail lors des différentes étapes d'une analyse de données. La partie « nettoyage » des données est de loin la plus longue (60% du temps). Cette illustration est extraite du sondage de CrowdFlower (CrowdFlower 2016).

4- Étude exploratoire et analyses préliminaires

Ce sont les premières vraies manipulations des données. Il s'agit littéralement d'explorer les données, en traçant de multiples graphiques, calculant de nombreux paramètres, etc. J'aime bien l'idée que cette exploration a pour objectif de « faire connaissance » avec les données. Les méthodes de visualisation décrites précédemment sont à cette étape d'un grand intérêt. Attention toutefois à l'erreur qui consiste à vouloir « faire tout bien » de façon « robuste et reproductible ». Une exploration est d'autant plus efficace qu'elle est simple, sans retenue vis-à-vis des standards de bonnes pratiques en informatique (présentés dans la suite du manuscrit, page 68). Tous les outils logiciels sont alors bons à utiliser (même Excel !), à condition d'être bien maîtrisés par la personne qui effectue l'exploration. L'idée étant de diminuer au maximum la réflexion sur le « comment faire ? » pour rester bien concentré sur le « quoi faire ? ». Ainsi dans sa formation, R. Peng[44] explique que l'exploration est une étape « *quick and dirty* ». C'est une spécificité notable vis-à-vis des étapes suivantes du cycle d'analyse.

5- Contraste entre hypothèse statistique et scientifique

À la suite des observations faites lors de l'étape d'exploration, un plan d'analyse peut être établi. Il s'agit, à l'avance, de décider des méthodologies à employer, en particulier des tests

[44] https://www.coursera.org/learn/exploratory-data-analysis [Accessible le 21/04/2020]

statistiques à appliquer et des règles de décisions associées (choix des seuils de risques). Cette étape du travail est marquée par une conversion du questionnement scientifique en hypothèses ou modèles statistiques. Par exemple, une problématique d'identification de gènes dont les niveaux d'expression sont différents entre différentes conditions peut être appréhendée par la réalisation d'un test statistique de comparaison de moyennes. Une erreur typique consiste à confondre « hypothèses scientifiques » et « hypothèses statistiques » (Boussier 2019). Ainsi une hypothèse scientifique qui consiste à dire « je pense que ce gène est important pour l'adaptation des cellules de levures à une carence en fer, son expression doit être augmentée dans un milieu pauvre en fer, par rapport à une situation contrôle » peut avoir comme hypothèse statistique associée « la moyenne des logFC est différente de 0 ». J'ai vu souvent des collègues déjà convaincus que leur hypothèse scientifique était vraie (pour de très bonnes raisons sans aucun doute), rejeter avec force les résultats d'un test statistiques qui n'allaient pas dans le sens attendu. Il s'agit de situations où les statistiques sont mal comprises. Invalider une hypothèse statistique ne signifie pas invalider une hypothèse scientifique. C'est une information comme une autre, qui est à prendre en compte dans la globalité du raisonnement scientifique. Enfin à cette étape, les problématiques de plan d'expérience, de répétabilité, réplicabilité et reproductibilité doivent également être prises en compte soigneusement (voir page 68). Si la phase d'exploration ne doit être aucunement contrainte, cette étape doit être rigoureuse et bien documentée. Les informations produites sont d'autant plus fiables et à même d'être utilisées pour créer des nouvelles connaissances.

6- Interprétations et conclusions

C'est l'étape du bilan final : une réponse à la question scientifique a-t-elle été obtenue ? Un temps de mise en forme des résultats, de rédaction de rapports et de réalisation d'infographies[45] est nécessaire. L'expertise dans le domaine scientifique pour lequel l'analyse de données est réalisée est très importante. En effet, elle permet l'identification et la discussion de liens avec des connaissances déjà établies pouvant conduire à de nouveaux questionnements scientifiques. Ainsi, « *la boucle est bouclée* » : retour à la première étape du processus d'analyse de donnés (formulation de la question scientifique) et début d'un nouveau cycle. Le projet avance un pas après l'autre.

[45] La différence entre « visualisation de données » et « infographie » est présentée page 48.

c. Ce qu'il faut retenir

L'analyse de données nécessite un savoir-faire qui s'acquière au fil des expériences. Il est très facile de tomber dans un certain nombre de pièges. Il en existe beaucoup que je n'ai pas encore identifiés, mais ceux évoqués dans ce paragraphe, je les ai expérimentés personnellement. Ils ont été la source parfois d'incompréhensions avec G. Lelandais. Il nous semblait donc important de vous les communiquer.

III. Les spécificités de l'analyse de données multi-omiques

1. À l'ère de la génomique fonctionnelle et de la biologie des systèmes

a. De la génétique à la génomique

Génétique

Le terme « génétique » a été utilisé pour la première fois en 1905, dans une lettre de W. Bateson[46]. Il a défini la génétique comme la « *science de l'hérédité et de la variation* ». La génétique était donc l'étude des caractères transmissibles entre les générations. La discipline associée était née « officiellement » quelques années avant, lors d'un comité international rassemblant une communauté adepte des Lois de Mendel (Mendel 1865). Depuis, les définitions des termes « génétique » et « gène » ont beaucoup évoluées. En effet, elles étaient initialement associées à la connaissance selon laquelle un gène est transcrit en un ARNm qui est ensuite traduit en une protéine (dogme de la biologie moléculaire). Aujourd'hui, cette représentation de l'expression d'un gène apparaît très naïve. J'ai lu un jour sur Twitter « qu'un gène est un élément qui génère de la gêne quand on veut le définir précisément ». C'est assez vrai ! Surtout pour le bioinformaticien que je suis, à qui ses simplifications en biologie sont facilement reprochées. Sur ce thème, j'ai apprécié la lecture des textes du philosophe J. Gayon. Il explique très bien que même si la notion de gène est complexe, utiliser des définitions simplifiées reste utile, en particulier dans un contexte global (Gayon 2016). Ce contexte global pour moi, est représenté par la « génomique ».

Génomique

Dans le dictionnaire Larousse, la génomique est définie comme la discipline qui étudie tout ce « *qui se rapporte au génome, c'est-à-dire à l'ensemble du matériel génétique porté par les êtres vivants* ». Cette définition est complétée sur le site Wikipédia[47] de la manière suivante : « *discipline de la biologie moderne […] qui étudie le fonctionnement d'un organisme, d'un organe, d'un cancer, etc. à l'échelle du génome, au lieu de se limiter à l'échelle d'un seul gène.* ». Les évolutions de la génomique sont intimement liées aux évolutions des technologies de

[46] https://dnalc.cshl.edu/view/16195-Gallery-5-William-Bateson-Letter-page-1.html [Accessible le 24/03/2020]
[47] https://fr.wikipedia.org/wiki/G%C3%A9nomique [Accessible le 24/03/2020]

séquençage. Elles ont été étonnantes ces 50 dernières années et les génomes de plus de 300.000 organismes[48] (archées, bactéries, eucaryotes) sont actuellement disponibles dans les bases de données publiques. Mais ensuite ? Que faire de ces séquences ?

De nouvelles questions

Dans son texte autobiographique (Dujon 2019), B. Dujon raconte comment, à l'issue de l'effort colossal qu'avait représenté le séquençage du génome de *S. cerevisiae* (1996), les directeurs des laboratoires impliquées dans le *consortium* se sont retrouvés pour discuter du « *What next* ? ». Il exprime ses interrogations ressenties à l'époque, face à des chercheurs qui souhaitaient au plus vite travailler à nouveau sur les problématiques biologiques qui les intéressaient initialement, plutôt que de poser de nouvelles « questions génomiques ». Son analyse est que la plupart des chercheurs de l'époque n'étaient pas intéressés réellement par le génome en tant que tel. Celui-ci était considéré seulement comme un grand dictionnaire des séquences des gènes.

b. Les défis de la génomique actuelle

La plupart des introductions à la génomique que j'ai pu lire dans la littérature, par exemple le cours d'introduction à la génomique du NIH[49] ainsi que celles de différents ouvrages (World Health Organization 2002; Bunnik et al. 2013; Pevsner 2015), identifient deux grands axes de recherche en génomique : la génomique structurale[50] d'une part et la génomique fonctionnelle d'autre part.

Génomique structurale

La génomique structurale est dans la continuité de ce qui était nommé « génétique moléculaire », à l'époque où le séquençage des acides nucléiques était fondé sur la méthode de Sanger[51]. Si la génomique structurale suit les évolutions des technologies de séquençage et les coûts financiers associés (voir page 38), ses objectifs restent les mêmes au cours du temps.

---

[48] https://gold.jgi.doe.gov/ [Accessible le 01/05/2020]

[49] https://www.genome.gov/About-Genomics/Introduction-to-Genomics [Accessible le 22/04/2020]

[50] Ici, la dénomination de « génomique structurale » est à différencier de « génomique structurelle » (en anglais, *structural genomics*) correspondant à l'étude de la conformation 3D des acides nucléiques.

[51] La figure 2 disponible sur la ressource numérique https://thomasdenecker.github.io/thesisWebsite/annexes/omiques/ [accessible le 11/11/2020] retrace les grandes lignes de l'évolution des techniques expérimentales de séquençage des acides nucléiques.

Schématiquement, il s'agit de déterminer la séquence d'un génome puis de l'annoter en identifiant le long de la séquence génomique des régions fonctionnelles caractéristiques telles que les régions codantes, les éléments régulateurs, les éléments répétés, etc. Sur ce sujet une revue intéressante est proposée dans le livre de E. Koonin (Koonin et al. 2003).

Génomique fonctionnelle

La génomique fonctionnelle, quant à elle, a pour objectif de caractériser les mécanismes de fonctionnement des gènes, à partir de l'étude de leurs produits d'expression (transcrits ou protéines). Il est ainsi utile pour travailler en génomique fonctionnelle de disposer d'un génome bien annoté, comme c'est le cas pour les levures *Candida glabrata* et *Candida albicans* que j'ai étudiées. Ces 15 dernières années, le spectre des analyses réalisables en génomique s'est considérablement élargi (Figure 17) et de nombreuses plateformes de recherche se sont développées (par exemple celles de l'ENS[52], l'I2BC[53], Curie[54], etc.) pour soutenir les laboratoires.

[52] https://www.ibens.ens.fr/spip.php?rubrique11 [Accessible le 24/04/2020]
[53] https://www.i2bc.paris-saclay.fr/spip.php?article399 [Accessible le 24/04/2020]
[54] https://science.institut-curie.org/platforms/next-generation-sequencing/ [Accessible le 24/04/2020]

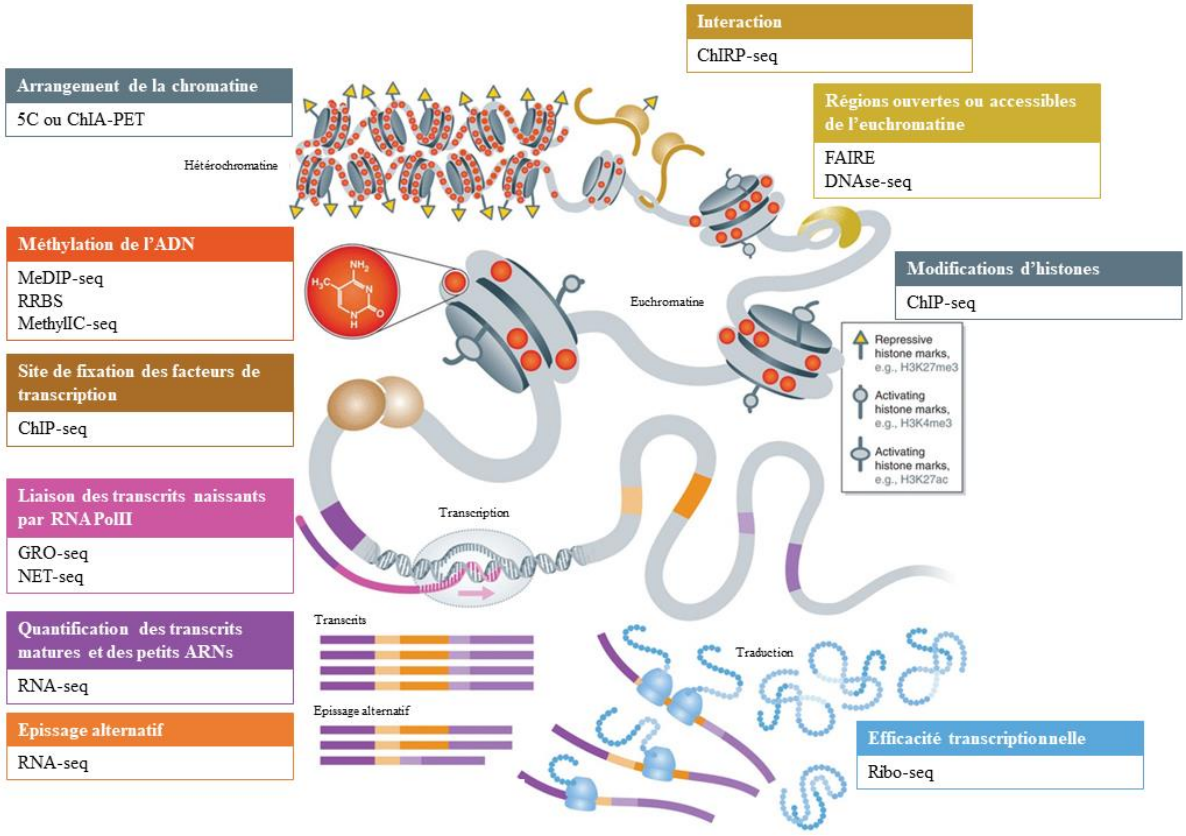


Figure 17 – Bilan des stratégies expérimentales permettant d'étudier les mécanismes de fonctionnement des gènes. Cette figure est adaptée d'une figure du cours de B. Cosson, donné en 2019 dans de DU « Création, analyse et valorisation de données omiques » de l'Université de Paris. La figure était initialement extraite de l'article de W. Soon (Soon et al. 2013)*. Près d'une demi-douzaine de niveaux de régulation de l'expression des gènes sont représentés ici. Ils illustrent la complexité et la multitude des mécanismes qui peuvent être considérés dans une étude de génomique fonctionnelle.*

L'encyclopédie en ligne enseqlopedia[55] propose des descriptions utiles pour une liste très complète de technologies « omiques ». Également, le projet Encode a délivré un ensemble de lignes directrices[56] qu'il est conseillé de suivre pour mettre au point les meilleurs protocoles.

c. Simplifications possibles ?

Dans un tel contexte de génomique fonctionnelle, il est difficile de ne pas se sentir submergé par la diversité des données produites (notion de *Big data deluge*, page 39), mais également par la complexité des phénomènes biologiques étudiés (Figure 17). Aussi, des simplifications sont-elles possibles ? La lecture des travaux de U. Alon laisse penser que oui.

[55] http://enseqlopedia.com/enseqlopedia/ [Accessible le 24/04/2020]

[56] https://www.encodeproject.org/about/experiment-guidelines/ [Accessible le 24/04/2020]

Exemple du processus d'activation de la transcription d'un gène

Dans son livre « *An Introduction To Systems Biology* », U. Alon présente le mécanisme de transcription d'un gène dans le schéma ci-dessous (Figure 18). Ce schéma a l'intérêt de lui permettre de souligner ce qu'il considère être les étapes principales du processus d'activation de la transcription d'un gène : (1) une protéine régulatrice X est modifiée par un signal S_x capté par la cellule, (2) la protéine modifiée X* se fixe sur un motif de régulation, localisé dans le promoteur du gène Y, (3) cette fixation permet le recrutement de l'ARN polymérase et la synthèse des molécules d'ARNm qui sont (4) traduites en protéines.

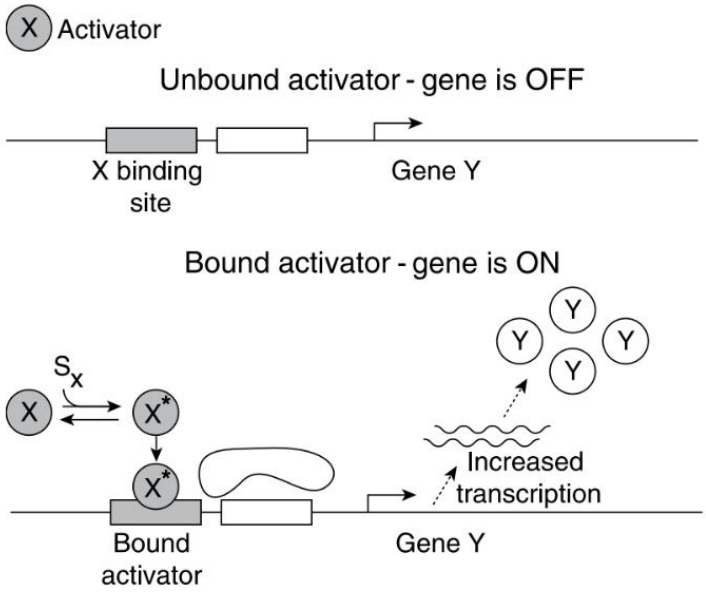


Figure 18 – Illustration d'un facteur de transcription spécifique (protéine X) de type « activateur ». La protéine X existe sous deux formes, une forme inactive et une forme active (notée X). Sous sa forme active, la protéine X* se fixe au niveau des séquences promotrices de ses gènes cibles et permet ainsi une augmentation de la transcription. Le signal S_x est à l'origine du déplacement de l'équilibre de X vers sa forme active X*. Cette figure est extraite du livre de U. Alon (page 5).*

Sur la base de ce schéma, il expose de façon très didactique[57] comment il est possible d'étudier la dynamique de réponse d'une simple régulation (notée X → Y) et explique pourquoi un mécanisme d'auto-régulation négatif du facteur de transcription X permet d'accélérer grandement la dynamique de réponse du système. Ainsi, il démontre un avantage pour la cellule à mettre en place ces auto-régulations, et de fait, celles-ci sont observées dans les réseaux de

[57] https://www.weizmann.ac.il/mcb/UriAlon/movies/Systems%20Biology%20Course%202011 [Accessible le 01/05/2020]

régulations transcriptionnelles de façon beaucoup plus fréquente que ce qui est attendu par le hasard (Shen-Orr et al. 2002).

Point de vue d'un physicien

En tant que physicien, U. Alon fait l'analyse que les systèmes biologiques se caractérisent par une surprenante simplicité. Ils emploient et combinent un petit nombre de ce qu'il nomme les « *basic building-block circuits* » (Alon 2020). Bien que les changements évolutifs se produisent de manière aléatoire, il montre que ceux-ci convergent systématiquement vers les mêmes solutions, qui obéissent à des règles génériques. C'est ainsi qu'il réussit avec des équations mathématiques très simples à capturer des propriétés de phénomènes qui semblent très complexes de premier abord. Une prouesse !

d. Ce qu'il faut retenir

Ainsi, les simplifications sont possibles, dans la mesure où l'objet d'intérêt devient le système biologique dans son ensemble, et non ses composants individuels. C'est littéralement de la « biologie des systèmes ». Cela est très intéressant et finalement cohérent avec les réflexions de J. Gayon et B. Dujon, précédemment évoquées au sujet de la génomique. Comme nous le rappelle C. Wanjek[58] ainsi que R. Breiling, il existe plusieurs définitions de la biologie des systèmes (Breitling 2010). À travers mes différentes lectures sur le sujet, certaines notions s'accordent. Pour commencer, la biologie des systèmes se fonde systématiquement sur l'idée que « *le tout est plus que la somme de ses parties* » (Aristote). Une fonction biologique est constituée de multiples éléments hétérogènes. Identifier ces éléments et surtout comprendre comment ils interagissent les uns avec les autres est ainsi l'intérêt de la biologie des systèmes. L'idée de l'importance des interactions n'est pas nouvelle. Dès 1968, L. von Bertalanffy propose une « théorie générale des systèmes » centrée sur les relations entre les éléments constitutifs plutôt que sur leurs spécificités individuelles (Von Bartalanffy 1969). Depuis, de nombreuses publications et travaux ont permis de renforcer le concept que des structures spatio-temporelles très riches peuvent émerger à partir d'éléments simples qui interagissent localement. A. Lesne va jusqu'à parler de pléonasme en évoquant la « biologie des systèmes » (Lesne 2009). Elle exprime l'idée que la biologie est finalement un immense système composé

[58] https://irp.nih.gov/catalyst/v19i6/systems-biology-as-defined-by-nih [Accessible le 21/05/2020]

de différents éléments qui interagissent entre eux dans l'espace et dans le temps. Pour aller plus loin dans les définitions de la biologie des systèmes et sur son évolution, la lecture de l'article *What is systems biology ?* de R. Breitling (Breitling 2010) est intéressante.

En conclusion, je me suis demandé quel est le lien entre la génomique fonctionnelle et la biologie des systèmes. Je répondrais que la génomique fonctionnelle est le socle qui identifie les éléments. Elle est fondée essentiellement sur la notion de génome (Figure 19). La biologie des systèmes est la prise en compte des interactions entre ces éléments. Elle est ainsi fondée essentiellement sur la notion de réseaux (Figure 19).

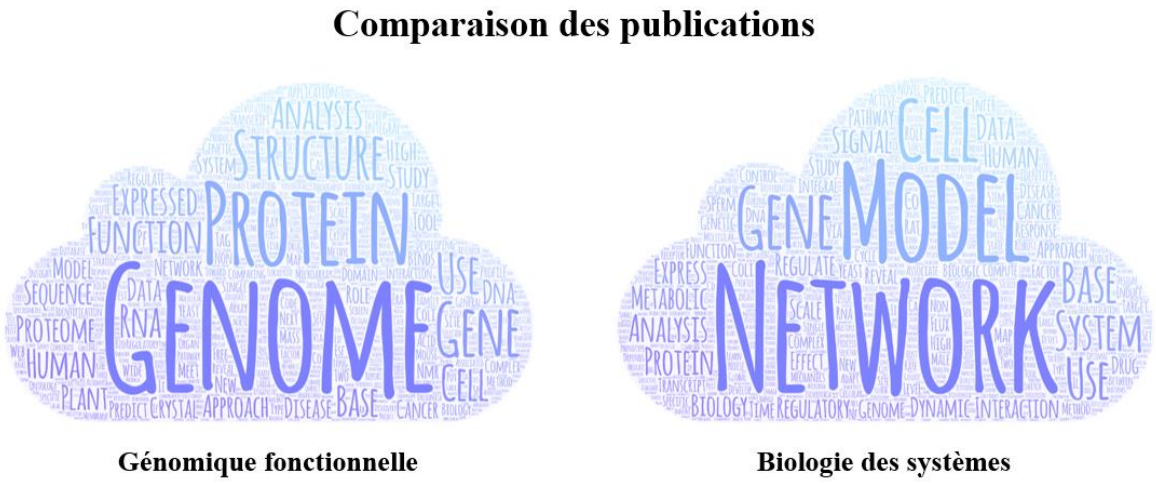


Figure 19 – Comparaison des termes les plus associés dans les titres de publications concernant la génomique fonctionnelle et la biologie des systèmes. Les listes des journaux explorés sont basées sur les recommandations de Pubmed pour la recherche « Functional Genomics » et « System Biology ». Les titres de ces journaux ont été ensuite extraits à partir de l'outil de recherche avancée de Pubmed et un nuage de mots a été généré sur le site WordArt.com. Cette figure a été réalisée le 01/04/2020.

2. Un besoin de reproductibilité

a. Répétabilité, réplicabilité et reproductibilité : les 3R de la confiance

En science, la « répétabilité », la « réplicabilité » et la « reproductibilité » sont des termes qui sont très utilisés mais souvent confondus. Avant de donner les définitions disponibles dans la littérature, je vais réutiliser mon illustration du gâteau au chocolat.

Dans votre cuisine, vous avez créé LA recette du gâteau au chocolat et vous l'écrivez avec toutes les indications nécessaires pour la refaire. Quelques temps plus tard, vous avez refait ce

gâteau dans votre cuisine en suivant la recette et il était toujours aussi bon. Dans ce cas, nous parlons de « répétabilité ». Des convives sont venus pour le diner et comme ils étaient nombreux, vous avez fait deux gâteaux en réalisant la recette successivement (toujours dans votre cuisine). Vos convives étaient ravis, les gâteaux étaient bons et avaient le même goût. Dans ce cas, nous parlons de « réplicabilité ». Un de vos convives avait apprécié particulièrement votre gâteau et vous avait demandé la recette. Quelques temps plus tard, vous avez reçu une photo de la part de votre convive, ravi, parce qu'il avait réussi à refaire le gâteau dans sa cuisine et qu'il était aussi bon. Dans ce cas, nous parlons de « reproductibilité ». L'extrapolation de ces concepts en biologie expérimentale est directe, en remplaçant la recette du gâteau au chocolat par un protocole expérimental.

Maintenant que nous avons une idée imagée de la signification de ces termes, posons précisément les définitions. Il est important de garder à l'esprit que des confusions peuvent être liées à la langue (français / anglais) comme nous le rapporte H. Plesser (Plesser 2018). Chaque terme est ainsi défini par deux sources : (1) la norme ISO 5725-4 :2020 : exactitude (justesse et fidélité) des résultats et méthodes de mesure[59] et (2) des publications scientifiques en langue anglaise.

Reproductibilité

La norme ISO 5725-4:2020 définit la reproductibilité comme « *l'étroitesse de l'accord entre les résultats individuels obtenus sur le même échantillon soumis à l'essai dans des laboratoires différents et dans les conditions suivantes : analyste différent, appareil différent, jour différent ou même jour* ». En 2019, un rapport américain définit précisément pour la première fois la notion de reproductibilité en bioinformatique (The National Academies of Sciences, Engineering, Medicine 2019) comme « *l'obtention de résultats de calcul cohérents en utilisant les mêmes données d'entrée, étapes de calcul, méthodes, code et conditions d'analyses* »[60].

[59] Les définitions sont issues de la partie 4 : Méthodes de base pour la détermination de la justesse d'une méthode de mesure normalisée.

[60] "*obtaining consistent computational results using the same input data, computational steps, methods, code, and conditions of analysis*".

Réplicabilité

La norme ISO 5725-4:2020 définit la réplicabilité comme « *l'étroitesse de l'accord entre les résultats individuels successifs obtenus sur le même échantillon soumis à l'essai dans le même laboratoire et dans les conditions suivantes : même analyste, même appareil, même jour* ». Le rapport *Reproducibility and Replicability in Science* la définit comme « *l'obtention de résultats cohérents entre les études visant à répondre à une même question scientifique, chacune d'entre elles ayant obtenu ses propres données* »[61].

Répétabilité

La norme ISO 5725-4:2020 définit la répétabilité comme « *l'étroitesse de l'accord entre les résultats individuels obtenus sur le même échantillon soumis à l'essai dans le même laboratoire et dont au moins l'un des éléments suivants est différent : l'analyste, l'appareil, le jour* ». L'*Association for Computing Machinery* la définit comme (même équipe, même dispositif expérimental) « *la mesure pouvant être obtenue avec une précision déterminée par la même équipe en utilisant la même procédure de mesure, le même système de mesure, dans les mêmes conditions de fonctionnement, au même endroit et sur plusieurs essais. Pour les expériences de calcul, cela signifie qu'un chercheur peut répéter de manière fiable son propre calcul. (Même équipe, même dispositif expérimental)* »[62].

b. Crise de la reproductibilité

En informatique

Je trouve qu'il est difficile de concevoir que deux ordinateurs ne donnent pas exactement le même résultat, quand ils font la même analyse. Dans l'imaginaire collectif, c'est l'humain qui fait des erreurs, pas les machines. Et pourtant, une crise de la reproductibilité existe en informatique. Le résultat d'une étude montre par exemple que 65 % des logiciels testés ne sont pas réutilisables (Collberg et al. 2015). Comment expliquer cette problématique de

[61] "*Obtaining consistent results across studies aimed at answering the same scientific question, each of which has obtained its own data*".

[62] « *The measurement can be obtained with stated precision by the same team using the same measurement procedure, the same measuring system, under the same operating conditions, in the same location on multiple trials. For computational experiments, this means that a researcher can reliably repeat her own computation* ». Issue de la page web https://www.acm.org/publications/policies/artifact-review-badging [Accessible le 27/04/2020].

reproductibilité en informatique ? Une des raisons majeures concerne les infrastructures informatiques, qui sont de plus en plus mouvantes. D'un ordinateur à l'autre, les OS peuvent ne pas être les mêmes (Windows, Mac OS, Linux) et les versions se succèdent à un rythme important. Même constat pour les logiciels. Ainsi, on croit qu'un programme informatique est exécuté de la même façon sur deux ordinateurs, alors que des différences existent. Également, il ne faut pas oublier qu'un niveau supplémentaire de variabilité peut exister au sein même de certains logiciels ou algorithmes qui utilisent des procédures avec des composantes aléatoires. C'est ainsi que le programme d'identification RAId, que j'ai beaucoup utilisé (voir la section page 160), choisit aléatoirement 10 000 spectres de masse lors d'une identification de protéines. Il est compréhensible dans ces conditions, d'obtenir des résultats différents, même si tous les calculs informatiques ont été exécutés de la même manière.

En bioinformatique

La bioinformatique est ainsi une discipline concernée par les enjeux de la reproductibilité. Toute analyse de données comporte de nombreux degrés de liberté qu'il est nécessaire de figer afin de garantir la capacité de n'importe quel chercheur à reproduire les résultats obtenus. J'ai été confronté lors de la réalisation de mes travaux de thèse à de multiples problématiques techniques dont quelques exemples sont présentés ci-dessous.

- #### Impossibilité d'installer les programmes et outils informatiques

Plusieurs causes peuvent expliquer l'échec de l'installation. Par exemple, l'utilisation d'un OS (Windows, MacOS ou Linux) incompatible avec le programme. Encore aujourd'hui, nous retrouvons des outils uniquement disponibles sur Windows. Des solutions sont disponibles (Wine sur Linux, une machine virtuelle, etc.) mais ne sont pas toujours efficaces. L'utilisation par les logiciels de dépendances qui ne sont plus disponibles. L'exemple le plus marquant que j'ai rencontré est lorsque j'ai cherché à utiliser un outil d'ontologie plébiscité dans la littérature. Il était malheureusement basé sur une ancienne version de *blast* que le NCBI indique comme obsolète et l'a rendue inaccessible. Je n'ai donc pas réussi à l'utiliser.

- #### Mise à jour d'une des dépendances qui rend les scripts informatiques non fonctionnels

Il s'agit ici du changement des noms des arguments d'une fonction. J'ai rencontré ce cas plusieurs fois avec le logiciel R. La fonction est la même, les arguments sont les mêmes, mais ils ne sont pas présentés avec le même nom. Il peut s'agir également de l'utilisation de différentes versions d'un langage de programmation informatique, un exemple est Python.

Aujourd'hui, la version officielle est de Python est la 3. Il reste encore de nombreux codes informatiques qui n'ont pas été traduits de Python 2 à Python 3, les rendant difficilement utilisables (nécessité d'un environnement où Python 2 est installé).

- Impossibilité de reproduire des résultats identiques d'une analyse de données

Le problème peut résulter de la version stable d'un langage qui n'est pas la même en fonction de l'OS utilisé. R est encore une fois le parfait exemple. Sur Linux, la version stable de R a plusieurs versions de retard par rapport à celle préconisée par le CRAN (site officiel de R). Ce changement de version cause des différences dans les résultats et peut conduire à l'impossibilité d'installer certains *packages*. Le problème peut également résulter de la version des *packages*. Pour poursuivre sur l'exemple précédent, les *packages* de R sont souvent mis à jour, proposant plus de stabilité, de fonctionnalité ou de meilleurs résultats. Comme les *packages* ne sont pas strictement identiques, alors il n'y a pas de reproductibilité parfaite possible.

Ainsi, quotidiennement, nous sommes confrontés en bioinformatique à ces problèmes, mais des solutions sont disponibles pour éviter autant que possible les sources de variabilités informatiques. Mais qu'en est-il des variabilités biologiques qui ont lieu lors de la génération des données ?

En biologie

En biologie expérimentale, même si les manipulations semblent parfaitement identiques, il n'est pas si surprenant d'observer des variations (dans une limite raisonnable). Les raisons sont nombreuses. Par exemple, l'expérimentateur peut changer, le matériel biologique être différent, les conditions de travail au laboratoire peuvent varier (température, humidité, ventilation, etc.). Lorsque ces variations sont trop importantes, l'expérience est considérée comme non reproductible. D'après l'article de M. Baker, 70% des études publiées ne seraient pas reproductibles (Baker et al. 2016). Plusieurs raisons sont évoquées incluant la très forte pression à la publication associée à la difficulté d'obtention de financements, la fragilité de l'analyse de données (négliger un éventuel effet *batch*[63], utiliser des tests statistiques non-adaptés, etc.) ou la faiblesse du plan expérimental (faible nombre de réplicats, non prise en compte des biais techniques, etc.). Pour 90 % des chercheurs interrogés dans cet article (plus de 1000), les clés

[63] Ensemble des effets conduisant des changements dans les mesures lors de la réalisation d'une même expérience : par le changement de laboratoire, de personnel, des lots des réactifs, des appareils, …

pour une biologie plus reproductible sont : un plan expérimental plus robuste et une meilleure utilisation des statistiques. Pour obtenir un plan expérimental plus robuste, plusieurs pistes sont envisageables :

- S'appuyer sur les recommandations proposées par des sites de références comme ENCODE[64] ou sur des publications (Tarca et al. 2006) ;
- Pratiquer la répétition d'expériences[65], une notion abordée lors de la journée d'étude proposée par Ouvrir la Science "Repenser la robustesse et la fiabilité en recherche : les chercheurs face à la crise de la reproductibilité". S'appuyer également systématiquement sur des plateformes de recherche et solliciter les conseils (en amont du projet) de statisticiens avertis ;
- Anticiper un maximum les biais[66], en particulier les biais de confusion. Il s'agit des erreurs engendrées par la non prise en compte d'une variable (variable de confusion) qui influence à la fois la cause supposée et l'effet supposé.

c. Liste de recommandations pour un contexte de travail « FAIR »

En théorie …

Le rapport *Reproducibility and Replicability in Science* recommande de suivre 4 grandes lignes directrices pour garantir la reproductibilité :

- Description de la partie expérimentale – L'ensemble des méthodes, des instruments, des procédures, des mesures et des conditions expérimentales devront être détaillées et expliquées ;
- Description de la partie informatique – Toutes les étapes de l'analyse des données, les choix techniques ainsi que les codes informatiques devront être partagés ;
- Description de l'analyse de données – Il faut être capable à tout moment de répondre au 3 questions : Quand ? Comment ? Pourquoi ? ;

---

[64] https://www.encodeproject.org/about/experiment-guidelines/ [Accessible le 07/08/2020]

[65] « *Pratiquer la répétition d'une même expérience (dans des conditions similaires ou très voisines) apporte également un gage de confiance dans l'obtention des résultats, et constitue même un passage obligé pour certaines disciplines.* ». https://www.ouvrirlascience.fr/repenser-la-robustesse-et-la-fiabilite-en-recherche-les-chercheurs-face-a-la-crise-de-la-reproductibilite-2/ [Accessible le 07/08/2020]

[66] De nombreuses ressources listent tous les biais existants comme https://data36.com/statistical-bias-types-explained/ [Accessible le 07/08/2020]

- Discussion des choix et résultats obtenus – Il s'agira de commenter les conclusions obtenues à partir des résultats.

En résumé, pour être reproductible, une équipe de recherche doit avoir imaginé un plan expérimental robuste, et avoir décrit et justifié tous ses choix. C'est à cette condition que les informations obtenues par d'autres seront les mêmes.

… et en pratique, comment les mettre en place ?

Ayant le désir d'être aussi reproductible que possible, je me suis posé beaucoup de questions sur le sujet. J'ai constaté un manque de contenus pédagogiques francophones, et j'ai observé que mes collègues de l'I2BC étaient en demande de mes retours d'expérience. J'ai ainsi créé une formation sur le sujet nommée FAIR_bioinfo (voir page 113). L'objectif de cette formation (qui a eu lieu en 2019) était de les sensibiliser aux différentes techniques de reproductibilité, en appliquant des principes dérivés du « *FAIR data* » (GO FAIR 2020; Wilkinson 2016). Des données dites « FAIR » sont des données qui valident les 4 critères suivants (Wilkinson et al. 2016) :

- ***F**indable* – Les données sont faciles à trouver aussi bien par les humains que par les systèmes informatiques ;
- ***A**ccessible* – Les données doivent être stockées à long terme, facilement accessibles et/ou téléchargeables. Même si les données ne sont pas directement accessibles (politique, étrangères, accès restreint, etc.), elles peuvent être considérées comme FAIR si les métadonnées sont accessibles ;
- ***I**nteroperable* – Les données doivent être faciles à combiner avec d'autres jeux de données, être dans un format standard et accompagnées de métadonnées ;
- ***R**eusable* – Les descriptions accompagnant les données doivent être suffisamment claires pour pouvoir les reproduire. Les données doivent être prêtes à être réutilisées.

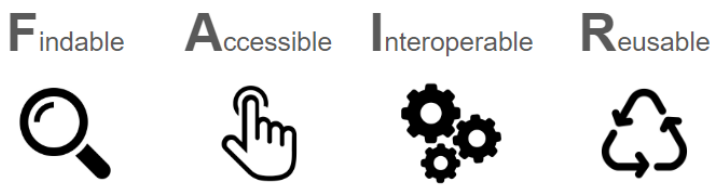


Figure 20 – Illustration du principe « FAIR data » qui a été dérivé pour créer la formation FAIR_bioinfo. Cette formation a été proposée en 2019 aux chercheurs de l'I2BC.

Ainsi, j'ai eu l'idée de transposer ces principes aux outils et processus d'analyse de données en bioinformatique. Nous obtenons ainsi les principes FAIR_bioinfo :

- ***F**indable* – Les outils utilisés pour effectuer l'analyse de données sont les outils les plus courants (si possible) dans le domaine et simples à trouver ;
- ***A**ccessible* – Toutes les ressources sont disponibles et les outils sont un maximum *Open source* ;
- ***I**nteroperable* – Les différents outils doivent être capables de communiquer entre eux et fonctionner aussi bien sur un ordinateur personnel que sur un serveur partagé ;
- ***R**eusable* – Le protocole doit être rejouable simplement, à l'identique dans un environnement contrôlé. Les résultats obtenus pour un même jeu de données doivent être parfaitement identiques.

En combinant des données FAIR avec des scripts et des protocoles d'analyse FAIR, nous aboutissons à des informations FAIR. Les contenus pédagogiques que j'ai mis en place pour la formation FAIR_Bioinfo sont présentés dans la section suivante.

d. Ce qu'il faut retenir

En conclusion, il est tout à fait possible de « maîtriser les quatre cavaliers de l'irréproductibilité » (Bishop 2019), en s'appuyant sur de nouvelles forces :

- Les principes FAIR data – Les données doivent être simples à trouver, simples d'accès, interopérables et réutilisables ;
- Les principes FAIR_bionfo - Tout doit être mis en place pour qu'à partir d'un même jeu de données, les résultats obtenus à partir d'une même analyse de données soient les mêmes ;
- La science ouverte – Aujourd'hui, nous avons à notre disposition de nombreux outils pour partager et rendre accessible nos données et nos résultats. Ainsi, il sera possible d'avoir de nombreux retours (aussi bien positifs que négatifs pouvant entrainer un débat comme par exemple lorsqu'il y a un manque de robustesse ou de reproductibilité au cours d'une étude clinique) ;
- Les financements – Les financements étant de plus en plus difficiles à obtenir, le niveau d'exigence sur la qualité de la génération des données et la reproductibilité des résultats a

augmenté. Par exemple, pour obtenir un financement de l'Agence Nationale de la Recherche (ANR), il faut impérativement fournir un Plan de Gestion des données[67].

3. Un besoin de statistiques

a. Problématique des grands jeux de données

Le « *Big data* » est une dénomination à la mode actuellement. Cette terminologie désigne « *des ensembles de données devenus si volumineux qu'ils dépassent l'intuition et les capacités humaines d'analyse et même celles des outils informatiques classiques de gestion de base de données ou de l'information* » (Wikipédia)[68]. À noter que la notion de *Big data* ne repose pas uniquement sur la quantité totale des données mais plus généralement sur les 3 V de Gartner (Laney 2001) : Volume, Vitesse / Vélocité et Variété (diversité)[69].

Dans ce contexte, le *Big data* constitue-t-il une source infinie de nouvelles informations ? *A priori* oui, mais à la condition de mettre en place des procédures d'analyse adaptées. Sinon le *Big data* peut devenir une source importante de « *Fake News* ». Un matin, G. Lelandais est arrivée dans mon bureau et m'a montré un passage du livre « Déchéance de rationalité » de G. Bronner. Elle avait souligné la phrase suivante : « *lorsque l'esprit est motivé à chercher dans une masse de données, il finit toujours par découvrir un chemin - généralement une ligne droite - vers le récit qu'il désire trouver* ». Elle trouvait cette réflexion intéressante car totalement transposable à la problématique d'étude des grands jeux de données en biologie.

En effet, si le chercheur est déjà totalement persuadé que son hypothèse scientifique[70] est vraie (voir page 57), il pourrait ne retenir de ces explorations d'un jeu de données de grande taille que les informations qui supporteraient son hypothèse, tout en rejetant (de façon consciente ou inconsciente) les informations qui contrediraient sa conviction initiale. Ainsi les statistiques ont un rôle important à jouer, permettant de démêler des observations, celles qui ont une grande probabilité d'être dues « au hasard » de celles qui au contraire ont une faible probabilité de

---

[67] https://anr.fr/fr/actualites-de-lanr/details/news/lanr-met-en-place-un-plan-de-gestion-des-donnees-pour-les-projets-finances-des-2019/ [Accessible le 30/04/2020]

[68] https://fr.wikipedia.org/wiki/Big_data [Accessible le 15/04/2020]

[69] Ces 3V sont illustrés dans la ressource complémentaire numérique : https://thomasdenecker.github.io/thesisWebsite/annexes/bigData/ [Accessible le 08/08/2020]

[70] Pour rappel, une « hypothèse scientifique » est à différencier d'une « hypothèse statistique ».

l'être. Ces dernières sont intéressantes et permettent d'imaginer l'existence d'un effet biologique particulier.

b. Vigilance au « p-hacking »

Étapes d'un test statistique

Schématiquement, la réalisation d'un test statistique se décompose en 5 grandes étapes. La première consiste à poser les hypothèses (hypothèse nulle et hypothèse alternative). Comme nous l'avons vu lors de la présentation du cycle d'analyse de données (page 57), cette étape est délicate car elle nécessite de convertir une hypothèse scientifique en hypothèse statistique. La deuxième étape consiste à collecter les données (mesures de la variable aléatoire étudiée) au sein d'un ou plusieurs échantillons. La troisième étape consiste à définir une valeur pour le risque de première espèce (généralement noté α, voir ci-dessous). Ce risque correspond à la probabilité de rejeter l'hypothèse nulle (H_0) sous un modèle dans lequel l'hypothèse nulle est vraie. La quatrième étape consiste à calculer une statistique de test, à partir des observations des échantillons. De façon plus générale, cette statistique de test est une variable aléatoire (c'est-à-dire dépendante des échantillons), dont la loi de probabilité est connue si l'hypothèse H_0 est vraie. Enfin la cinquième étape consiste à calculer une « valeur P » (ou *P-value*). C'est cette valeur P qui retient le plus souvent l'attention, avec l'idée que « *plus elle est petite, mieux c'est* ». En effet, une petite valeur P encourage à décider de rejeter l'hypothèse nulle et donc supporter l'idée qu'un effet biologique existe.

Origine de la valeur P

La valeur P est une notion ancienne que nous pouvons trouver dès 1900, avec Pearson qui propose de calculer « *la probabilité que la valeur observée de la statistique du chi carré soit dépassée sous l'hypothèse nulle* ». Cette notion a été réintroduite par J. Gibbons et J. Pratt dans l'article *P-values: Interpretation and Methodology* (Gibbons et al. 1975). La valeur P est « *la probabilité pour un modèle statistique donné sous l'hypothèse nulle H0 d'obtenir la même valeur ou une valeur encore plus extrême que celle observée* ». L'objectif est de montrer que les observations ne sont pas compatibles avec l'hypothèse nulle (la valeur P est faible, H_0 est rejetée) car dans le cas contraire, le résultat observé serait fortement improbable. Ce mode de raisonnement est souvent comparé au principe de la preuve par l'absurde (puisque les observations sont peu probables sous le modèle de l'hypothèse nulle, l'hypothèse alternative est conservée).

Mise en relation avec les risques de 1ère et 2ème espèces

Comme vu précédemment, le risque de première espèce est la probabilité de rejeter l'hypothèse nulle (H_0) alors que celle-ci est vraie. Une valeur de 5% (probabilité 0.05) est une valeur classique, mais de plus en plus de recommandations sont données en biologie, afin de réduire ce risque à 0.5% soit une probabilité à 0.005 (Benjamin et al. 2018). Le risque de première espèce est en effet souvent associé à la notion de « Faux positif » (Tableau 3). Le risque de seconde espèce est quant à lui associé à l'hypothèse alternative (H_1). C'est la probabilité de ne pas rejeter l'hypothèse H_0, alors que H_1 est vraie. En biologie, ce risque est associé à la notion de « Faux négatif » (Tableau 3). Il est toutefois très peu utilisé en raison de la complexité de définir un modèle sous lequel l'hypothèse H_1 est vraie.

	Hypothèse H_0 vraie	Hypothèse H_1 vraie
Hypothèse H_0 acceptée	Vrai négatif	Faux négatif Erreur de type 2 : β
Hypothèse H_1 acceptée	Faux positif Erreur de type 1 : risque α	Vrai positif

Tableau 3 – Tableau récapitulatif de la notion de risque en fonction des hypothèses mises en concurrence lors d'un test statistique. L'hypothèse H_0 est l'hypothèse nulle, tandis que l'hypothèse H_1 est l'hypothèse alternative. L'erreur de Type 1 correspond au risque de 1ère espèce, tandis que l'erreur de Type 2 correspond au risque de 2ème espèce.

Dans ce contexte, la valeur P (*P-value*) est comparée au risque de première espèce. La règle appliquée est simple : si la valeur P calculée est inférieure au risque de première espèce, alors l'hypothèse H_0 est rejetée, tandis que si la valeur P est supérieure au risque de première espèce, alors l'hypothèse H_0 n'est pas rejetée.

Problématique de tests multiples

La problématique des tests multiples dérive directement des définitions des risques présentées ci-dessus. Ainsi imaginons que nous sommes dans une situation où l'hypothèse H_0 est vraie. Les données collectées dans les échantillons sont compatibles avec ce modèle H_0. Ces données sont utilisées pour réaliser un test statistique. Cela signifie qu'un risque de première espèce est choisi (par exemple 5%), qu'une statistique de test est calculée et utilisée pour obtenir une valeur P. Si la valeur P est inférieure à 0,05 alors l'hypothèse H_0 est rejetée. Cette situation revient à commettre une erreur puisque nous savons que H_0 est vraie. Ce qui est intéressant, c'est que par définition du risque de première espèce, il est prédéterminé que cette situation à

5% de chances de se produire. Ainsi, je sais que si je joue avec une pièce équilibrée, mon analyse statistique du comportement de la pièce à 5% de chance de conclure que la pièce n'est pas équilibrée alors qu'en réalité, la pièce est équilibrée. Par le hasard, un nombre atypique d'une des deux faces aura été observé. Ce sont des choses qui peuvent arriver, même si l'hypothèse H_0 est vraie (voir Figure 21).

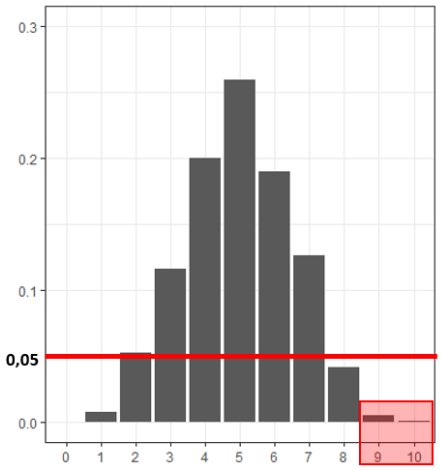


Figure 21 – Graphique des probabilités associées au nombre de fois ou le côté FACE d'une pièce de monnaie (équilibrée) est observé sur 10 lancers indépendants de la pièce. Observer 9 ou 10 fois le côté FACE est très peu probable, toutefois cela n'est pas impossible.

Donc 5% est le risque d'erreur associé à la réalisation d'un test statistique. Mais que se passe-t-il si le nombre de tests est multiplié ? Par exemple, si l'équilibrage d'un grand nombre de pièces est étudié. Dans un cours réalisé au DU « Création, analyse et valorisation de données -omiques »[71], J. Boussier nous a présenté les choses de la manière suivante : « *Imaginez que vous jouez à un jeu. Le jeu de rejeter H_0 alors que H_0 est vraie. Quelles sont vos chances de gagner, si vous jouez une fois ?* ». Nous avons tous répondu 5%, la valeur du risque de première espèce. Il a continué : « *Très bien, vous avez donc 5% de chances de gagner et 95% de chances de perdre. Imaginez maintenant que vous jouez 3 fois ? Quelles sont vos chances de gagner, c'est-à-dire de rejeter au moins une fois l'hypothèse H_0 alors que l'hypothèse H_0 est toujours vraie ?* ». Une manière simple de répondre est de calculer le complémentaire de l'événement « perdre tout le temps », c'est-à-dire ne jamais rejeter H_0 alors que H_0 est vraie. Nous obtenons alors : Probabilité(gagner au moins une fois) = 1 - Probabilité(perdre tout le temps) = 1 – (0.95 x 0.95 x 0.95) = 14%. C'est beaucoup, et pourtant je n'ai joué que trois fois. La Figure 22

[71] J'ai été diplômé de cette formation en 2018.

présente l'évolution du risque de rejeter à tort au moins une fois l'hypothèse H_0 en fonction du nombre de tests statistiques réalisés.

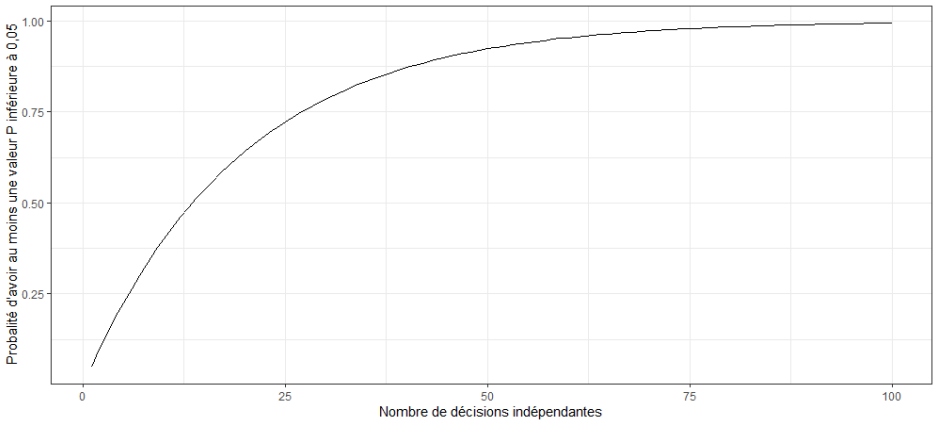


Figure 22 – Probabilité de rejeter au moins une fois l'hypothèse nulle lors de la réalisation de plusieurs tests statistiques de manière indépendante. Pour chaque test, le risque de 1ère espèce est fixé à 5%.

La problématique des tests multiples est donc la suivante : en multipliant le nombre de tests, le risque d'erreur est augmenté. Une solution consiste à corriger les valeurs P pour que globalement sur l'ensemble des tests réalisés le risque d'erreur reste contrôlé à 5%. De nombreuses méthodes de correction de la valeur P sont disponibles comme celle de Bonferroni ou de Benjamini-Hochberg. Pour approfondir le sujet, l'article *Why, When and How to Adjust Your P Values?* (Jafari et al. 2019) est très complet.

Qu'est-ce que le p-hacking ?

Le *p-hacking* est la réalisation d'une étude statistique qui s'appuie sur des tests multiples, mais sans le montrer. À nouveau, cela peut être réalisé de façon consciente ou inconsciente. Il existe plusieurs situations de *p-hacking* :

- **Situation 1** : Sur un même jeu de données, plusieurs tests sont réalisés (par exemple un test paramétrique et un test non paramétrique), et seuls les résultats du test le plus significatif (avec la valeur P la plus faible) sont montrés.
- **Situation 2** : Pour que la valeur P calculée à partir des observations d'un échantillon soit inférieure à 0.05, des observations sont ajoutées (ou retirées).
- **Situation 3** : Sur un unique jeu de données composé de plusieurs variables, de nombreuses analyses sont réalisées. À force de multiplier les tests sur des variables différentes, il y en a un qui fonctionne et seul celui-ci est montré (nous

avons vu qu'avec seulement 3 tests, la probabilité de gagner au moins une fois est supérieure à 10 %)[72].

Il est très facile de se retrouver dans une configuration de *p-hacking*, surtout dans la situation 3 qui est très proche de l'étape d'exploration que nous avons décrite dans le cycle d'analyse de données (page 57). C'est la raison pour laquelle cette étape d'exploration doit être suivie d'une étape de validation rigoureuse. Une solution serait de séparer systématiquement les grands jeux de données en deux : un jeu d'entrainement (exploration) et un jeu test (validation). Cette idée est toutefois très peu mise en application. Pour approfondir le sujet, l'article *The Extent and Consequences of P-Hacking in Science* (Head et al. 2015) est très complet.

c. Utiliser les statistiques pour convaincre les autres

Dans un article au titre provocateur « Faut-il brûler les tests de signification statistique ? » (Mbengue 2010), A. Mbengue explique que la bonne utilisation des statistiques est un sujet controversé. Il conseille d'appliquer un principe de prudence concernant les statistiques et surtout souhaite alerter sur le fait que les chercheurs se cachent parfois derrière les résultats d'un tests statistiques pour minimiser leur responsabilité dans la prise de décision finale. Mon avis est qu'en bioinformatique, nous ne sommes pas tous spécialistes en statistiques, moi le premier. Le choix du bon test pour les bonnes données est difficile. Dans la rubrique commentaire de Nature, certains chercheurs invitent même « *les scientifiques à se dresser contre la signification statistique* » (Amrhein et al. 2019). De nombreuses autres publications accompagnent cet article et confirment cette tendance actuelle (Amrhein et al. 2018; Amrhein, Trafimow, et al. 2019; Hurlbert et al. 2019; Debrouwere et al. 2014; Greenland 2017; McShane et al. 2019). Nous y apprenons par exemple, que sur 791 articles étudiés, 51 % se trompent sur l'interprétation de la non-significativité comme une absence d'effet (Figure 23). S'ajoute à cela une vraie problématique de reproductibilité statistique (Amrhein et al. 2017).

d. Ce qu'il faut retenir

En résumé, les statistiques sont un outil plus adapté à l'étape de validation que celle d'exploration des données. Elles permettent de tester des hypothèses statistiques et non

[72] Comme par exemple trouver un lien entre les bonbons et l'acné en testant les couleurs du bonbon une par une (https://www.explainxkcd.com/wiki/index.php/882:_Significant [Accessible le 07/08/2020]

scientifiques. Travailler avec de grands jeux de données en biologie est une situation particulière et une vigilance est nécessaire pour ne pas présenter des résultats erronés. Il est important de toujours privilégier des résultats nuancés par les statistiques plutôt que d'essayer de les catégoriser par les statistiques (significatif / non significatif par rapport à un seuil fixe).

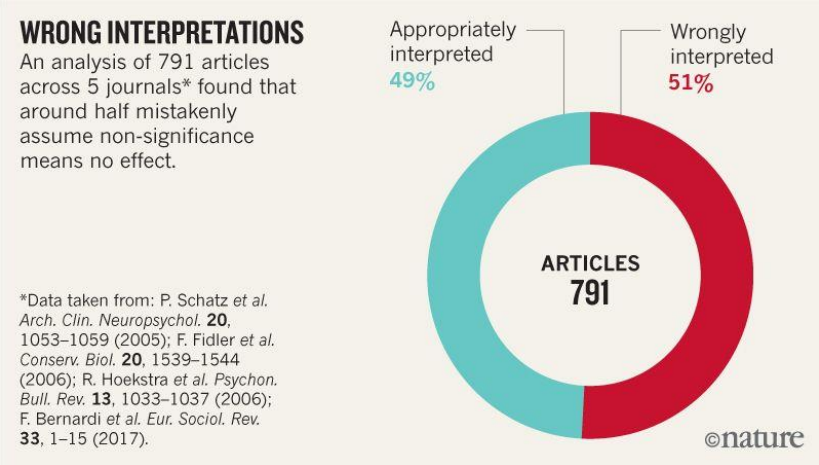


Figure 23 – Analyse de 791 articles provenant de 5 journaux qui montre que plus de la moitié des études présentées concluent à tort que l'absence de significativité (valeur P > 0.05) signifie l'absence d'effet. Ne pas rejeter l'hypothèse H_0 est considéré comme une preuve que l'hypothèse H_0 est vraie.

4. La bioinformatique face à des nouveaux défis

En 2005 (Rhee 2005), le principal défi de la bioinformatique était présenté comme en relation avec « *l'hétérogénéité de la façon dont les données sont analysées, annotées et affichées ainsi que le manque de lien entre les données disponibles* ».[73]. S. Rhee imputait ces problématiques au jeune âge de la discipline. Quinze ans plus tard, sont-elles toujours d'actualité ?

a. Problématique de l'hétérogénéité des données

L'hétérogénéité est partout en bioinformatique : diversité des organismes étudiés, des techniques expérimentales appliquées, des solutions informatiques proposées. Les différentes sources de données sont maintenues par des communautés avec des organisations différentes. Je vais dans cette partie présenter les défis associés.

[73] "*The heterogeneity of how data are analyzed, annotated, and displayed and the lack of connectivity among the available data*"

Hétérogénéité des données

L'article de R. Elmasri présente plusieurs raisons de cette hétérogénéité en lien direct avec la donnée (Elmasri et al. 2010) :

- La complexité – Cette problématique est liée à la définition même de la génomique fonctionnelle. Nous cherchons à comprendre les liens et interactions entre les différents composants cellulaires à différents niveaux (gènes, ARN, protéines, etc.).
- La diversité – Les données disponibles sont de natures très diverses : numériques, images, textes, coordonnées, etc.
- La non-exhaustivité – La biologie est une science où beaucoup de choses restent encore à découvrir. L'ensemble des données n'est donc pas complet.
- La taille – Gérer les données de l'Humain ne représente pas le même défi que les données de levures. Les stratégies ne sont pas les mêmes. À titre d'exemple, l'espace sur un disque dur informatique d'un fichier au format FASTA contenant la séquence du génome humain GRCh38[74] est de 3.1 Go alors que celui du génome de la levure pathogène *C. glabrata* CBS138[75] n'est que de 12.5 Mo (quasiment 250 fois moins) ;
- Le manque de standard – Aujourd'hui, des bases de données de référence existent mais ne stockent pas les mêmes informations.

Hétérogénéité de la qualité des données

Un exemple typique d'une hétérogénéité de la qualité des données est l'annotation fonctionnelle des gènes. Celle-ci peut avoir été obtenue après une vérification expérimentale (annotation très fiable) ou bien par une procédure informatique automatique (annotation plus ou moins fiable). Lorsqu'un génome est nouvellement séquencé, il est très fréquent d'utiliser l'annotation par transfert automatique, sur la base d'homologie entre les séquences des gènes ou des protéines. Le principe est simple. Si les séquences se ressemblent très fortement entre deux espèces, nous pouvons supposer que les gènes ont conservé la même fonction. Dans ce cas, si un des deux gènes est annoté et pas l'autre, l'annotation est transférée au gène jusqu'alors inconnu ou sans fonction.

[74] https://www-ncbi-nlm-nih-gov.insb.bib.cnrs.fr/genome/guide/human/ [Accessible le 14/04/2020]
[75] http://www.candidagenome.org/download/sequence/C_glabrata_CBS138/current/ [Accessible le 14/04/2020]

La *Gene Ontology*[76] est un projet bioinformatique avec pour objectif de structurer la description des fonctions des gènes dans le cadre d'une ontologie[77] commune à toutes les espèces. Les chercheurs précisant les définitions en fonction de leur domaine d'étude et leurs besoins, certains domaines de la biologie sont plus précisément annotés et mieux décrits que d'autres. Un biais peut être observé également au sein d'une même espèce pour laquelle certains gènes très étudiés sont bien annotés, tandis que certains autres gènes sont peu étudiés et ont uniquement une annotation automatique, voire aucune annotation. Étudier l'enrichissement fonctionnel au cours d'une analyse de données dans ces situations aboutit à des résultats complexes à interpréter. La Figure 24 illustre la différence de profondeur que peuvent avoir deux termes GO (exemple GO :0034755[78] et GO :0000472[79]). De plus, nous pouvons observer qu'un terme GO peut avoir plusieurs parents ce qui rend l'exploitation plus complexe.

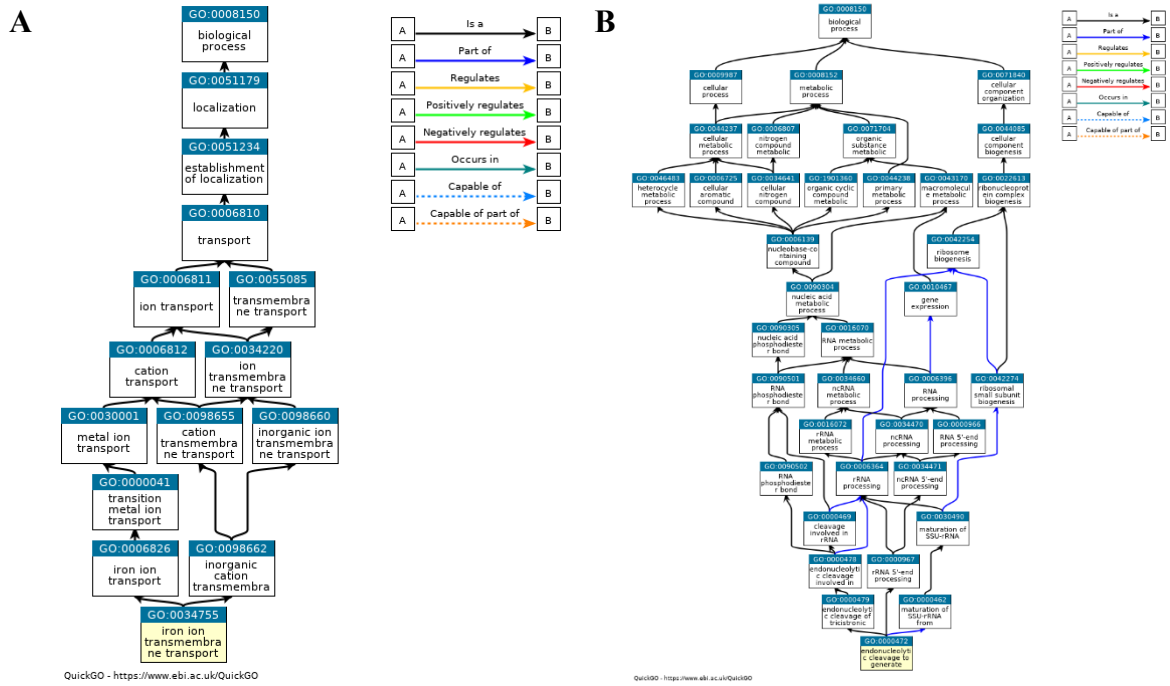


Figure 24 – Comparaison des "profondeurs" associées à deux termes GO. À droite la description du terme est plus précise que celle présentée à gauche. Les DAG ont été extraits d'Amigo.[80].

[76] http://geneontology.org/ . Pour en savoir sur la *Gene Ontology*., nous proposons une ressource numérique : https://thomasdenecker.github.io/thesisWebsite/annexes/geneOntology/ [Accessible le 10/08/2020].

[77] Ensembles structurés de termes représentant le sens d'un champ d'informations acceptés par une communauté pour modéliser un ensemble de connaissances.

[78] http://amigo.geneontology.org/amigo/term/GO:0034755 [Accessible le 14/04/2020]

[79] http://amigo.geneontology.org/amigo/term/GO:0000472 [Accessible le 14/04/2020]

[80] http://amigo.geneontology.org/amigo [Accessible le 14/04/2020]

Une dernière difficulté possiblement rencontrée avec les termes GO est l'accumulation de termes obsolètes (Figure 25).

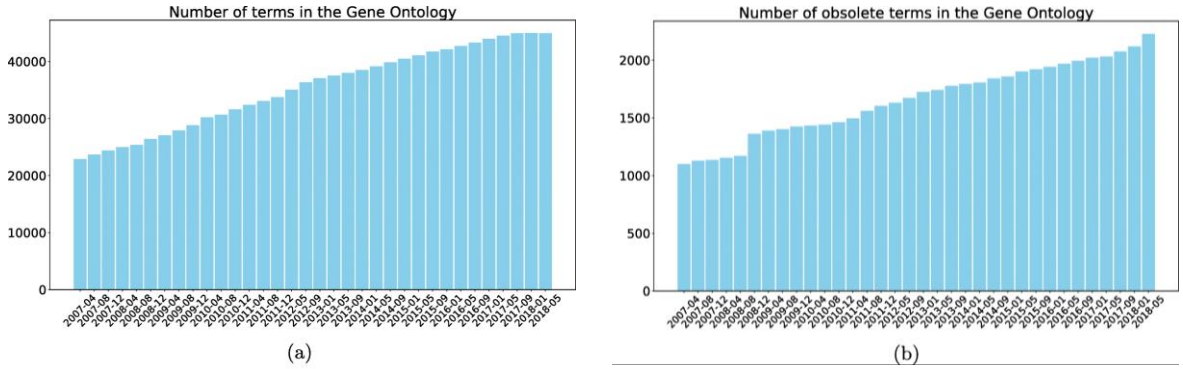


Figure 25 – Nombre de termes GO référencés dans la base de données, en fonction du temps (a) et nombre de termes GO déclarés obsolètes dans la base de données, en fonction du temps (b). Cette figure est extraite de l'article de F. Nakano (Nakano et al. 2019)

b. Problématique de l'accessibilité et du partage des données

La collecte des données est critique pour le travail d'analyse. Dans la réalisation de mon travail de thèse, j'ai été confronté à différentes problématiques en relation avec l'accessibilité des données.

Accessibilité partielle aux données

C'est le cas typique des données jointes à des publications. Le tableau de données en libre accès ne comporte que les informations mises en relief dans la publication (par exemple seulement les gènes différentiellement exprimés). Dans certains articles, les données ne sont accessibles que sur demande à un des auteurs de l'article. Un point marquant lorsque nous recherchons des données dans la littérature est que toutes les expériences ont abouti à des résultats concluants. M. Munafò, professeur de psychologie biologique à l'université de Bristol, nous indique que « *le biais de publication contre les résultats nuls est un problème majeur connu qui limite la fonction d'autocorrection de la science - tout ce qui s'y attaque devrait aider la science à progresser plus rapidement* »[81]. Dans une colonne parue dans Nature, D. Mehta écrit : « *Les*

[81] *"Publication bias against null results is a known major problem that limits the self-correction function of science – anything that addresses this should help science progress more rapidly".*

éditeurs, les évaluateurs et les autres membres de la communauté scientifique doivent lutter contre la préférence de la science pour les résultats positifs - pour le bénéfice de tous »[82]. Elle en profite pour nous partager son expérience concernant la difficulté de publier des résultats négatifs en évoquant la pression des publications positives (Mehta 2019). Dans ce contexte, des initiatives ont vu le jour comme les BMC Research notes. Leur constat ? Des données ne sont souvent pas publiées alors qu'elles pourraient contribuer à faire progresser d'autres projets. Ce nouveau format permet aux chercheurs de décrire leurs données, de les partager avec la communauté et d'avoir une valorisation de leurs travaux. Nous avons ainsi publié des résultats en spectrométrie de masse (Lelandais et al. 2019a) (voir page 193).

Obstacles au partage de données

En 2009, le comité *Ensuring the Utility and Integrity of Research Data in a Digital Age* a proposé un rapport dans lequel il a listé deux principaux obstacles au partage des données (National academy of sciences 2009). Le premier était le temps. Les chercheurs ont besoin de temps pour contrôler et analyser les données afin de conclure et corriger des erreurs potentielles. Les données ne sont donc pas partagées immédiatement. Le second obstacle était la volonté de garder les données privées, pour des raisons de réglementations ou par peur de la concurrence. Une pratique courante est le maintien des données en privé lors du processus de *reviewing* d'un article. La rétention de données et leur restriction d'accès ont des conséquences importantes. Une limitation du flux d'informations conduit à un ralentissement de l'innovation. Enfin, il conduit à un ralentissement de la vérification de la recherche. Toutes les données peuvent conduire à l'affirmation ou la réfutation de conclusions et ainsi permettre de faire avancer les connaissances.

Impact sur l'environnement

Aujourd'hui, le partage de données « omiques » est gratuit, grâce aux infrastructures telles que GEO, SRA, PRIDE, etc. Cela encourage le partage des données. Mais combien de temps ce modèle économique pourra perdurer ? Je me suis demandé quels étaient les coûts associés à ces centres de stockage. En effet, si le prix du méga octet ne cesse de diminuer (Figure 26), cette baisse de prix à des conséquences. Tout d'abord le stockage semble gratuit (ou très peu cher)

[82] "*Publishers, reviewers and other members of the scientific community must fight science's preference for positive results — for the benefit of all"*

et surtout, il apparaît illimité. Nous stockons ainsi de plus en plus, sans vigilance vis-à-vis de la redondance et de l'utilité des données conservées.

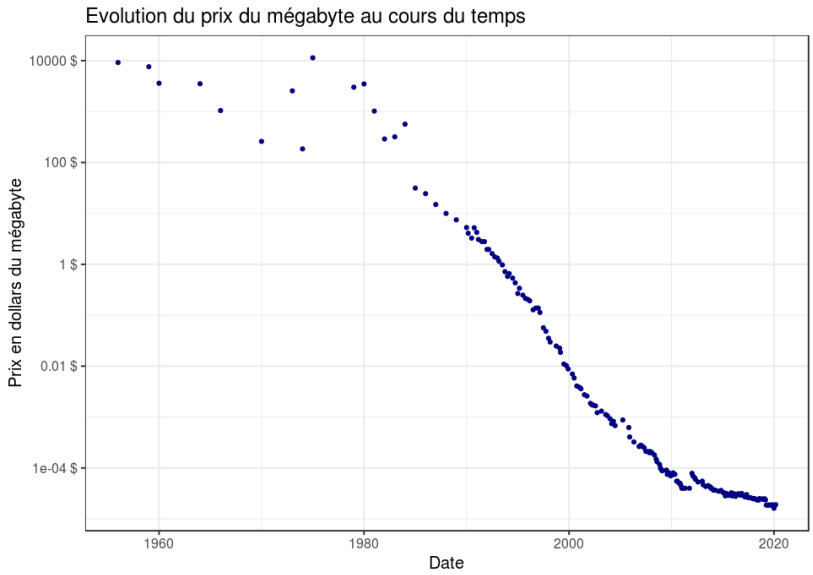


Figure 26 – Évolution du prix du megaoctet au cours du temps. Les données ont été extraites le 03/04/2020 en ligne.[83]. Le code pour réaliser cette figure est disponible sur GitHub Gist.[84]

Dans l'article *How Much Does Storage Really Cost?* (Dutta et al. 2013), les auteurs expliquent que plusieurs paramètres sont à prendre en compte pour évaluer un coût global :

- Le coût initial – C'est le coût de l'achat du matériel informatique : les disques durs, l'équipement internet, les serveurs, etc ;
- Le prix du sol – C'est le coût des bâtiments qui hébergent le matériel informatique. Ce prix est très variable en fonction des localisations géographiques ;
- Le prix de l'énergie – C'est le coût de l'alimentation des bâtiments et du matériel informatique. Nous retrouvons parmi les grands domaines de dépenses l'électricité, la connexion internet, la régulation de la température, etc ;
- Le service – Il s'agit de toutes les dépenses d'entretien et de maintenance de l'infrastructure, des serveurs, etc ;
- L'élimination des déchets – Le traitement des disques durs usagés par exemple ;
- Le coût environnement – En effet, notre activité numérique à un impact sur la planète.

[83] https://jcmit.net/diskprice.htm [Accessible le 03/04/2020]

[84] https://gist.github.com/thomasdenecker/da2122d442ebc14a65bfedc3b6664449 [Accessible le 03/04/2020]

Dans La face cachée du numérique (Quotidien 2019), il est indiqué que 25 % des émissions de gaz à effet de serre sont générées par le numérique à cause des *data centers* (c'est à dire par le stockage de données). Dans *Le numérique : quel impact pour la planète ?*[85], M. Maillard rappelle que si les solutions de *cloud* se démocratisent car elles permettent un accès facilité aux données, la conséquence directe est un achat plus important de matériel (tablettes, smartphones, etc.) entrainant un impact sur l'environnement. En bioinformatique, nous sommes d'importants consommateurs de matériels et d'énergie. Faut-il arrêter de travailler pour autant ? Je pense que non, mais une fois conscients de la situation, les bioinformaticiens peuvent changer leurs habitudes pour réduire leur impact sur l'environnement. L'ouvrage *Bits of Power: Issues in Global Access to Scientific Data* résume parfaitement cette idée dans cet extrait : « *La valeur des données réside dans leur utilisation* ». Par conséquent, mon avis est qu'il faut partager les données autant que possible, mais il faut le faire de façon réfléchie.

c. Science ouverte (*Open Science*)

Qu'est-ce que la Science ouverte (*Open Science*) ?

La plupart des résultats de recherche sont partagés à la communauté *via* des articles publiés dans des revues scientifiques. Celles-ci sont en règle générale payantes et l'accès aux nouvelles connaissances en est donc limité. À titre d'exemple, les abonnements aux revues payantes pour les chercheurs de l'INSERM a représenté une dépense annuelle de 3.5 millions d'euros en 2017[86]. Dans ce contexte, une autre philosophie émerge, c'est l'*Open Science*. L'*Open Science* est un terme parapluie regroupant de nombreux autres termes : *Open data*, *Open access*, *Open source*, etc. Les objectifs communs sont de démocratiser les connaissances et donc le savoir et d'en simplifier l'accès en appliquant par exemple les principes FAIR (voir page 73). Dans l'article de B. Fecher (Fecher et al. 2013), il est proposé 5 écoles de pensées pour soutenir l'*Open Science* et changer les habitudes dans la recherche (Figure 27). Les auteurs proposent de mettre en place des plateformes de création de connaissances collaboratives, des plateformes avec un accès libre aux outils d'analyses, de rendre les résultats et les données libres d'accès, de trouver un nouveau moyen d'évaluer la recherche et de démocratiser l'accès au savoir. En

[85] https://www.linfodurable.fr/technomedias/la-pollution-invisible-du-numerique-632 [Accessible le 23/08/2020]

[86] https://www.inserm.fr/recherche-inserm/science-ouverte. D'après le site https://presse.inserm.fr/service-presse/inserm-en-chiffres/, l'Inserm regroupe près de 14 000 chercheurs, ingénieurs, techniciens, gestionnaires, hospitalo-universitaires, post-doctorants… L'Inserm paye donc 250 € d'abonnements aux revues payantes par personne !

2020, des avancées importantes sont constatées. Des outils tels que Github (pour partager les codes informatiques et travailler ensemble) ou Conda (pour la mise à disposition d'outils gratuitement) sont très populaires. Le dépôt des données « omiques » dans les bases de données publiques (SRA, GEO, PRIDE) est systématique avant leur publication dans les journaux scientifiques. La revue Nature propose également une page web regroupant les différents dépôts de données classés par type de données (Scientific Data 2017). Un exemple appliquant les principes FAIR est proposé pour chaque dépôt cité.

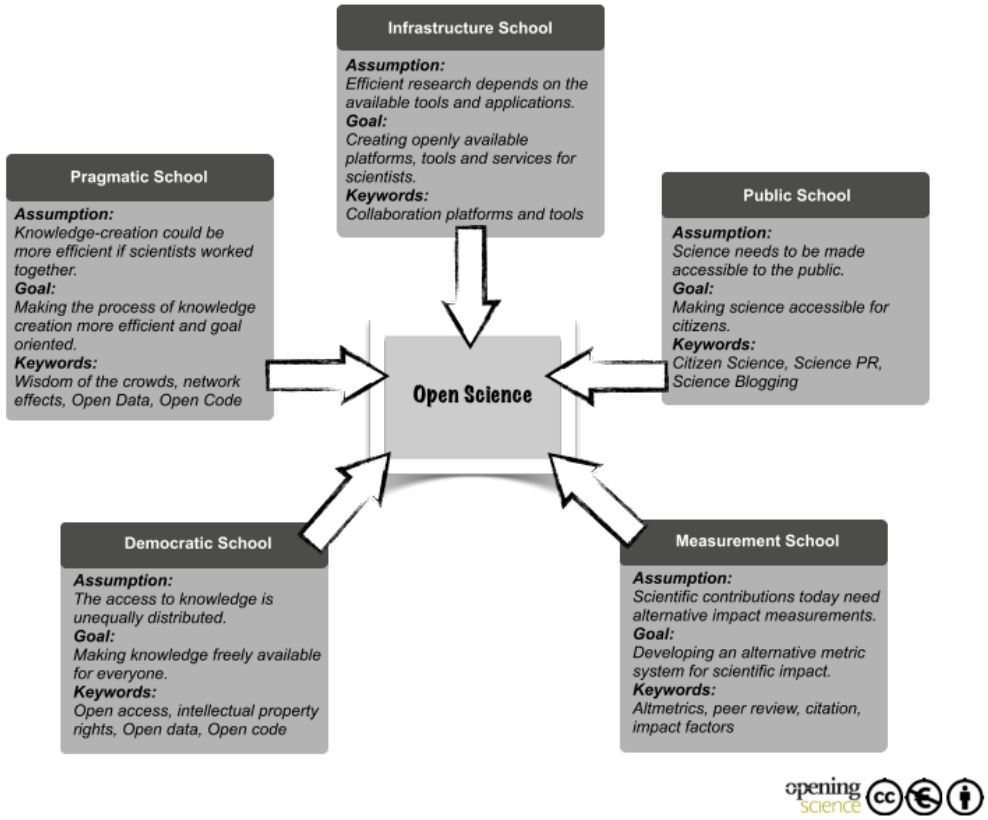


Figure 27 – Figure extraite de l'article de B. Fecher (Fecher et al. 2013) *présentant les 5 écoles de pensée pour tendre vers l'Open Science.*

La publication des articles scientifiques reste un élément délicat vis-à-vis de l'*Open Science*. Plusieurs solutions émergent telles que :

- Publier dans des revues en libre accès – Dans ces revues, l'auteur paye pour rendre l'article accessible à tous. Les articles publiés au cours de cette thèse sont tous accessibles librement dans des revues qui soutiennent l'*Open science*. Généralement,

une licence sur les outils et les données est ajoutée pour protéger les droits (comme par l'utilisation de Zenodo[87]).

- Déposer dans une archive ouverte – Les archives ouvertes sont des plateformes web. Les chercheurs peuvent y déposer gratuitement leurs travaux mêmes ceux publiés dans une revue. En France, la plus célèbre et pluridisciplinaire archive ouverte est HAL[88].
- Déposer une prépublication (*preprint*) – Une dernière solution est de déposer l'article qui n'est pas encore publié sur des sites spécialisés en prépublication. L'article est ainsi accessible à tous, même s'il ne s'agit pas encore de la version définitive. L'avantage de cette solution est qu'elle apporte beaucoup de visibilité à l'article. En biologie, la plateforme la plus utilisée est bioRxiv[89]

À quand une recherche 100% ouverte ?

Même si de nombreux chercheurs sont conscients de la nécessité d'un changement, la transition reste difficile. Ainsi, le gouvernement français a mis en place en 2018 un Plan national pour la science ouverte (Ministère de l'enseignement supérieur de la recherche et de l'innovation 2018) prévoyant les mesures suivantes :

1. « *Rendre obligatoire la publication en accès ouvert des articles et livres issus de recherches financées par appel d'offres sur fonds publics* » ;
2. « *Créer un fond pour la science ouverte* » ;
3. « *Soutenir l'archive ouverte nationale HAL et simplifier le dépôt par les chercheurs qui publient en accès ouvert sur d'autres plateformes dans le monde* ».

Ces actions ne se limitent pas à la recherche française. En 2016, un plan européen a été proposé (Zaken 2016) et des conférences internationales ainsi que des formations ont lieu dans le monde entier. Un rapport des progrès est proposé et fréquemment mis à jour sur le site Open Access 2020[90]. Enfin, l'Organisation for Economic Co-operation and Development (OECD) préconise un nouvel effort visant à améliorer l'accès du public aux données de recherche (OECD 2007).

[87] https://zenodo.org/ [Accessible le 14/04/2020]
[88] Hyper Article en Ligne – https://hal.archives-ouvertes.fr/ [Accessible le 27/03/2020]
[89] https://www.biorxiv.org/ [Accessible le 27/03/2020]
[90] https://oa2020.org/progress-report/ [Accessible le 27/03/2020]

Selon cette approche, « *l'ouverture signifie un accès sur un pied d'égalité pour la communauté internationale de la recherche au coût le plus bas possible, de préférence pas plus que le coût marginal de la diffusion. Le libre accès aux données de recherche grâce au financement public devrait être facile, rapide, convivial et de préférence basé sur Internet* »[91]. Les principes proposés couvrent 13 grands domaines : l'ouverture, la flexibilité, la transparence, la conformité juridique, la protection de la propriété intellectuelle, la responsabilité formelle, le professionnalisme, l'interopérabilité, la qualité, la sécurité, l'efficacité, la responsabilité et la durabilité.

5. Pour conclure, sortir du « cloud mental » avec agilité

Comme nous l'avons vu au début de cette introduction générale, la bioinformatique s'est, en 15 ans, transformée pour répondre aux défis de l'analyse des données massives (Figure 1, page 30). Les enjeux sont techniques, mais aussi méthodologiques. L'analyse de données requiert un savoir-faire particulier, peu reconnu et peu présenté dans les ouvrages. J'ai souhaité dans cette introduction, présenter les problématiques auxquelles j'ai été confronté ces dernières années.

Pour terminer, j'aimerais évoquer une conférence donnée par U. Alon en 2013[92]. Il évoque, avec beaucoup de bienveillance la notion de « cloud ». Il ne s'agit pas du cloud informatique, mais d'un cloud « mental ». Celui-ci correspond à un état psychologique dans lequel tout étudiant se retrouve à un moment donné de ses projets. Il est perdu, bloqué entre son point de départ et son objectif visé. « *Comment avancer ? Faut-il continuer ? Me suis-je trompé ?* ». Personnellement, j'ai passé beaucoup de temps dans ce cloud mental. À chaque fois, notre stratégie de sortie a été la planification (Figure 28). La méthode de travail fondée sur la notion de cycle d'analyse de données (voir page 60) peut être comparée à la méthode Scrum utilisée en informatique pour un développement agile. Il s'agit de « *déterminer des objectifs atteignables et élaborer un plan d'action pour y arriver rapidement et efficacement* ». Cette méthode repose sur des approches dites agiles (Figure 32). Nous commençons par établir les besoins représentés par les questions scientifiques (*1- Requirements*). Nous réfléchissons aux

---

[91] *"access on equal terms for the international research community at the lowest possible cost, preferably at no more than the marginal cost of dissemination. Open access to research data from public funding should be easy, timely, user-friendly and preferably Internet-based."*

[92] Conférence TEDx du 6 février 2013 : https://www.youtube.com/watch?v=RVoz_pEeV8I [Accessible le 14/04/2020]

approches à suivre pour y répondre (*2- Design*) puis nous réalisons l'exploration de données (*3- Development*). Nous testons des hypothèses statistiques (*4- Testing*) puis partageons nos résultats (*5- Deployment*) pour obtenir un retour de la communauté experte (*6- Review*). Une fois ce « sprint » terminé, nous pouvons passer à un nouveau cycle.

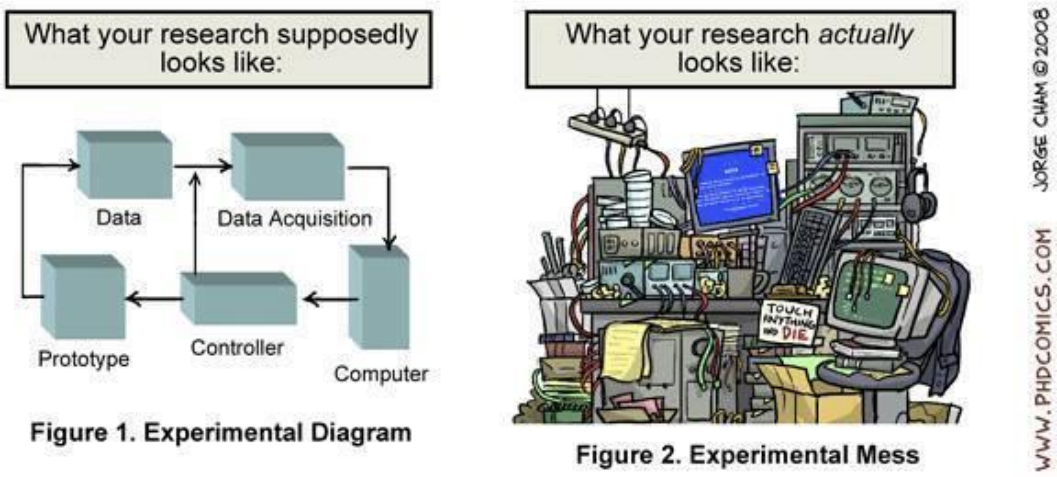


Figure 28 – La planification, entre fiction et réalité (illustration de phdcomics).

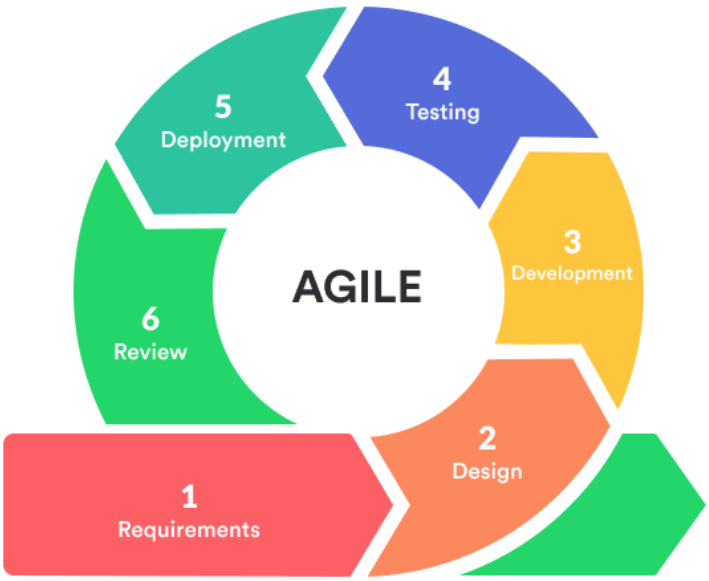


Figure 29 – Représentation des différentes étapes d'un sprint lors d'un projet agile (@pawan-pawar)

Un ensemble de chercheurs en informatique a proposé un manifeste pour le développement Agile de logiciels :

Nous découvrons comment mieux développer des logiciels par la pratique et en aidant les autres à le faire. Ces expériences nous ont amenés à valoriser :

Les individus et leurs interactions *plus que les processus et les outils*

Des logiciels opérationnels *plus qu'une documentation exhaustive*

La collaboration avec les clients *plus que la négociation contractuelle*

L'adaptation au changement *plus que le suivi d'un plan*

Nous reconnaissons la valeur des seconds éléments, mais privilégions les premiers.

Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland et Dave Thomas

Nous pourrions les transposer pour tendre vers une analyse de données plus agiles. Nous nous sommes prêtés à l'exercice pour obtenir les préceptes suivants :

Nous découvrons comment mieux réaliser une analyse de données par la pratique et en aidant les autres à la faire. Ces expériences nous ont amené à valoriser :

Les individus et leurs interactions *plus que les processus et les outils*

Des logiciels opérationnels *plus qu'une documentation exhaustive*

La collaboration avec les équipes expérimentales *plus qu'une simple récupération de données*

L'adaptation au changement lié à la variabilité des données *plus que le suivi d'un plan*

Nous reconnaissons la valeur des seconds éléments, mais privilégions les premiers.

Thomas Denecker et Gaëlle Lelandais

Contributions aux efforts mutualisés en bioinformatique : développements de logiciels et formations des chercheurs en biologie

I. L'application Web « bPeaks App » pour l'analyse de données ChIPseq

1. Le contexte du projet

bPeaks App est une application WEB développée au printemps de l'année 2018. À cette période, j'avais la responsabilité de l'encadrement du stage d'une étudiante de 2ème année de Licence de biologie de l'Université Paris-Saclay, Éléonore Pillot-Lucas. Elle souhaitait approfondir ses connaissances en programmation informatique, acquises dans le cadre de sa formation. Dans ce contexte, nous avons réfléchi à un projet qui (1) serait utile aux activités du laboratoire ; (2) pourrait être réalisé sur une courte durée (quelques semaines) et (3) nous permettrait de tester l'utilisation d'une librairie R nommée Shiny (Chang et al. 2019). Cette librairie avait en effet attiré mon attention par la possibilité de créer des graphiques interactifs et des pages WEB dynamiques, facilitant ainsi les analyses de données avec R.

2. Le logiciel bPeaks

bPeaks est un package R développé par G. Lelandais et J. Merhej en 2012 disponible sur le site du CRAN[93]. Il permet la détection des sites de fixation de protéines à l'ADN, à partir de données ChIPseq. Depuis sa publication en 2014 (Merhej et al. 2014), bPeaks est utilisé en routine au laboratoire. Son utilisation requiert toutefois des compétences techniques pour son installation et l'exécution des scripts R.

3. Le logiciel bPeaks App

Dans ce projet, notre souhait était de développer une interface graphique au logiciel bPeaks à l'aide de la librairie Shiny. bPeaks App est présenté en détails dans la publication disponible en ligne[94] et insérée à la section suivante. Le code source est disponible sur Github[95] et un site

[93] https://cran.r-project.org/web/packages/bPeaks/index.html [Accessible le 02/06/2020]
[94] https://pubmed.ncbi.nlm.nih.gov/30286789 [Accessible le 02/06/2020]
[95] https://thomasdenecker.github.io/bPeaks-application/ [Accessible le 02/06/2020]

internet de présentation est accessible en ligne[96]. L'application est actuellement utilisée sur le serveur de l'équipe[97].

4. Le PDF de la publication dans la revue « BMC Research Notes » (Denecker et al. 2018)

[96] https://github.com/thomasdenecker/bPeaks-application [Accessible le 02/06/2020]

[97] https://bpeaks.data-fun.io/ [Accessible le 02/06/2020]

Denecker and Lelandais *BMC Res Notes (2018) 11:698*
https://doi.org/10.1186/s13104-018-3802-y

BMC Research Notes

RESEARCH NOTE

Open Access

Empowering the detection of ChIP-seq "basic peaks" (bPeaks) in small eukaryotic genomes with a web user-interactive interface

Thomas Denecker* and Gaëlle Lelandais

Abstract

Objective: bPeaks is a peak calling program to detect protein DNA-binding sites from ChIPseq data in small eukaryotic genomes. The simplicity of the bPeaks method is well appreciated by users, but its use via an R package is challenging and time-consuming for people without programming skills. In addition, user feedback has highlighted the lack of a convenient way to carefully explore bPeaks result files. In this context, the development of a web user interface represents an important added value for expanding the bPeaks user community.

Results: We developed a new bPeaks application (bPeaks App). The application allows the user to perform all the peak-calling analysis steps with bPeaks in a few mouse clicks via a web browser. We added new features relative to the original R package, particularly the possibility to import personal annotation files to compare the location of the detected peaks with specific genomic elements of interest of the user, in any organism, and a new organization of the result files which are directly manageable via a user-interactive genome browser. This significantly improves the ability of the user to explore all detected basic peaks in detail.

Keywords: ChIP-seq, Peak calling, Protein DNA-binding sites, Small eukaryotic genomes, bPeaks

Introduction

ChIP-seq, i.e. chromatin immunoprecipitation sequencing, is an experimental approach to analyze protein interactions with DNA [1]. Peak detection (also referred as "peak calling") consists of identifying all the genomic regions in which a significant enrichment of DNA sequences (or reads) in a ChIP sample is observed compared to a control sample. These regions are expected to represent DNA-binding sites for the studied protein [2]. A considerable number of peak calling software packages have been developed (for instance [3, 4], etc.) and choosing the appropriate software, optimized for a specific biological system of interest, is a prerequisite for successful ChIP-seq data interpretation.

In this context, we proposed a methodology to identify "basic Peaks" (bPeaks) in small eukaryotic genomes [5]. The general idea was to take advantage of simpler peak calling for species with small genome sizes (< 20 Mb). The program bPeaks thus performs an exploration of ChIP-seq results at the nucleotide scale. It uses a sliding window, which compares the read distributions between the immunoprecipitation (IP) sample and a control sample. We implemented the bPeaks program with the R language and the associated package is available at the CRAN website [6]. Since its original publication, the bPeaks R package has been downloaded more than 11,360 times (July 2018) and successfully used to identify DNA-binding sites for different proteins in several yeast species [7–10].

Our colleagues, essentially experimental biologists, appreciate the simplicity of the bPeaks methodology. The program uses a combination of only four thresholds to mimic "good peak" properties, as described by investigators who visually inspect ChIP-seq results on a genome browser [5]. However, they highlighted several difficulties. First, working with an R program is a challenging task for people with only limited bioinformatic

*Correspondence: thomas.denecker@u-psud.fr
Institut de Biologie Intégrative de la Cellule (I2BC), Centre National de la Recherche Scientifique: UMR9198, Université Paris-Saclay, Université Paris-Sud - Paris 11, 11 - Bâtiment 400, Orsay, France

skills. Initial installation of the necessary software and libraries, importing of the ChIP-seq data in R, and running a bPeaks search can be excessively difficult for simply technical reasons and may thus be an obstacle to the in-depth analysis of the results. Also, bPeaks generates a large number of output files (several dozen), which are all automatically written and stored in a single operating system (OS) folder. These files were meant to be helpful for further investigation (for example, detection of regulatory motifs), but feedback from users brought to our attention the need for their better organization and documentation. Finally, the bPeaks R package only comprises pre-registered annotations of genes for yeast species because our group's research activities are focused on functional genomics in yeast. This is an important limitation for researchers interested in other organisms.

We developed a new application to overcome these limitations. Referred hereafter as the "bPeaks App" (bPeaks application), it is used via a web browser and makes it possible to perform all the analysis steps required to identify protein DNA-binding sites with ChIP-seq results in a few mouse clicks. We developed new functionalities in the bPeaks App, relative to the original R package, to (i) evaluate the overall quality of the analyzed ChIP-seq data (Lorenz curve and PBC calculation), (ii) facilitate the exploration of output files (with a user-interactive genome browser), and (iii) upload annotation files to compare the location of the detected peaks with particular genomic elements of interest to the user, in any organism. bPeaks App is an open source program available on Github [11]. It has the advantage that it can be deployed locally (on a personal workstation) or on a server. Here, we present the technical solutions that were chosen and explain the main functionalities of the bPeaks App.

Main text

Methods

General overview

The bPeaks App is a web application which uses the original bPeaks R package [6]. The backend of the application is based on three mainstream open source technologies: Github [12], Docker [13], and PostgresSQL [14] (see Fig. 1a). The frontend solutions of the application were chosen to provide users a particularly easy-to-use experience using Shiny, the Web Application Framework for R [15]. The Plotly R package [16]

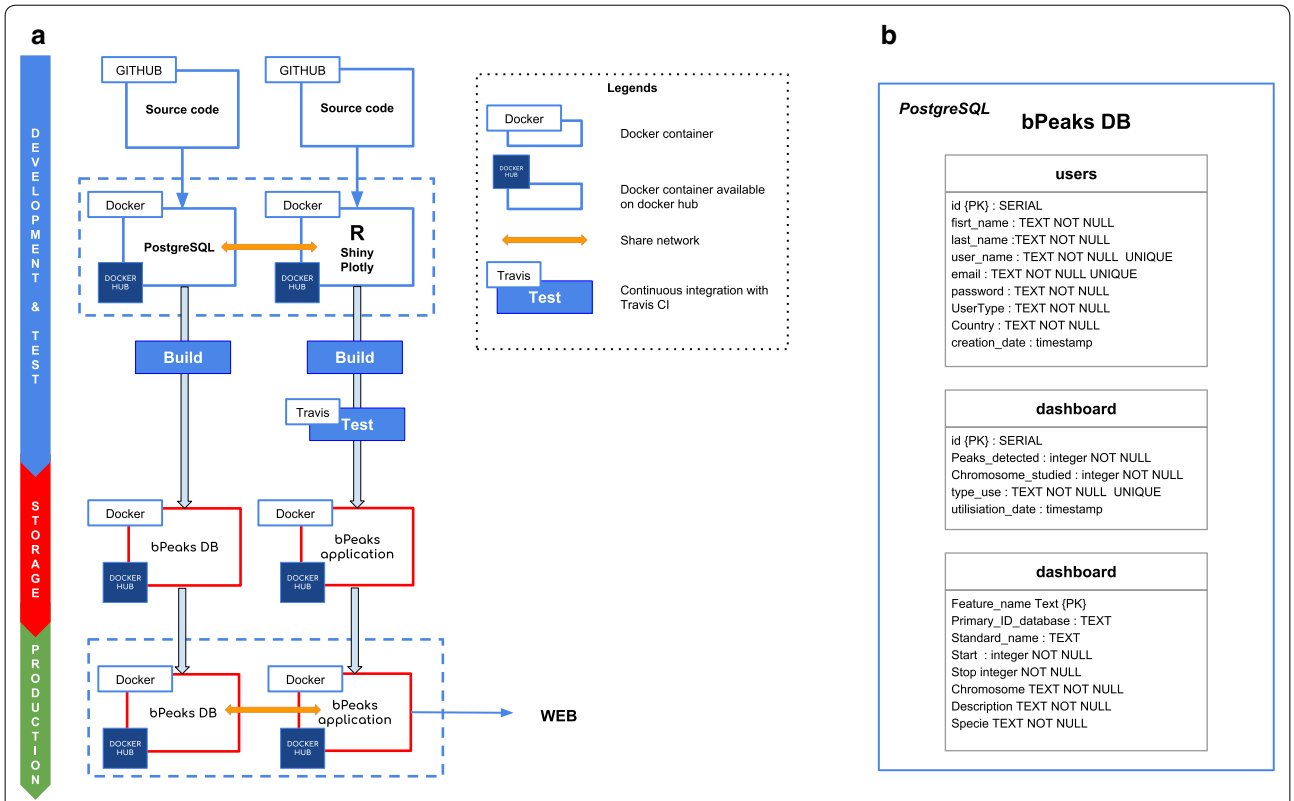


Fig. 1 Technical solutions to develop the bPeaks App. **a** Open source softwares used to develop the application are shown here. It is divided into three sections: (1) development and test (in blue), (2) database storage (in red), and (3) final production (in green). **b** Tables defined in the PostgreSQL database that highlight the functioning of the bPeaks App

was used to obtain dynamic graphical representations, together with Google chart [17]. The application requires a database to control user access. The solution proposed by Shiny requires payment. Thus, we preferred another approach, based on a PostgreSQL database. Put very simply, the database is comprised of three tables: one to manage user information, one to manage an information dashboard and one for gene annotations (Fig. 1b). The connection between R and PostgreSQL was accomplished using RPostgreSQL [18] and the protection of user passwords achieved with the pgcrypto extension.

Strategy for versioning the application

Two complementary axes were considered to ensure appropriate versioning of the bPeaks App: (i) R package dependencies and (ii) OS dependencies. The package manager packrat [19] was used to precisely follow the latest versions of all the packages used for development of the bPeaks App. It saves libraries locally and generates a packrat.lock file. This file lists the detailed package versions that were used, including all dependencies. The bPeaks App was built on a containerization paradigm (see Fig. 1a) with Docker, since R software is also dependent on the OS. Our objective was to entirely pack the application and its dependencies in a virtual container. Thus, it was possible to build images that contain everything required for the bPeaks App to function. These images are downloaded on the host system from the Docker Hub ([20, 21]).

Installation

The bPeaks App was meant to be installed either on a personal workstation or a laboratory web server. The application can manage several users and multiple simultaneous connections. The bPeaks App can be deployed on Linux, MacOS X, or Windows 10. Detailed information to deploy the bPeaks App can be found in Github README [22]. Minimal requirements are:

- 64 bits OS.
- Docker community edition > v18 (with a minimum of 3 GB of RAM allocated).
- Access to the internet (required for Docker image download).

For deployment on a local workstation, installation scripts are available to create a launcher script to facilitate the use of the bPeaks App. This launcher starts all the components needed to run the application without entering a single command line.

Criterion to evaluate ChIP-seq data quality

Lorenz curves and PCR Bottleneck Coefficients (PBCs) are classical criteria to evaluate ChIP-seq data quality [23] and were both implemented into the bPeaks App. Details of the calculations are presented in Additional file 1.

Results

bPeaks App

With the new bPeaks App, our aim was to (1) simplify the use of the bPeaks peak calling method for those with no bioinformatics skills, (2) add several ChIP-seq data representations to assess the overall quality of the initial experiment, and (3) facilitate the exploration of peak calling results. Also, we wanted to guarantee the reproducibility of any results obtained with bPeaks App, systematically tracing all the analysis steps and computational tool versions. We decided to divide the application into two parts referred hereafter as "bPeaks analyzer" and "bPeaks explorer". bPeaks analyzer focuses solely on the peak calling step. Output files are automatically renamed and reorganized in different OS folders. These files represent the starting point for the bPeaks explorer part, which allows dynamic and user-interactive visualization of the detected peaks, as well as peak localization relative to particular genomic elements (coding or promoter sequences, DNA repeated regions, etc.). All these features in bPeaks explorer are novel compared to previous R package outputs, which were only static files. These sub-applications are accessible after an authentication phase on the home page (see Additional file 2). A study case is shown in Additional file 3 to the illustrate the use of the bPeaks App.

bPeaks analyzer to run the detection of peaks

bPeaks analyzer is a web interface to apply the bPeaks method. We paid particular attention to not modify the original R package, so that identical results will be obtained whether bPeaks analyzer is used or not. Figure 2 shows how information required to run bPeaks can be specified in several dedicated areas. Note that help and documentation can be systematically obtained (see Fig. 2b). We maximally simplified the configuration. This is well-illustrated with the functionality of the bPeaks App in reading ChIP-seq data files (IP and control, see Fig. 2a). It is possible to import files with different separators, with the presence or not of a header line, and with the presence or not of quoting characters. Once the selected files are uploaded, a preview is shown, allowing the user to verify that the data importation is correct (see Fig. 2a). Default values for the four thresholds are provided (the same as

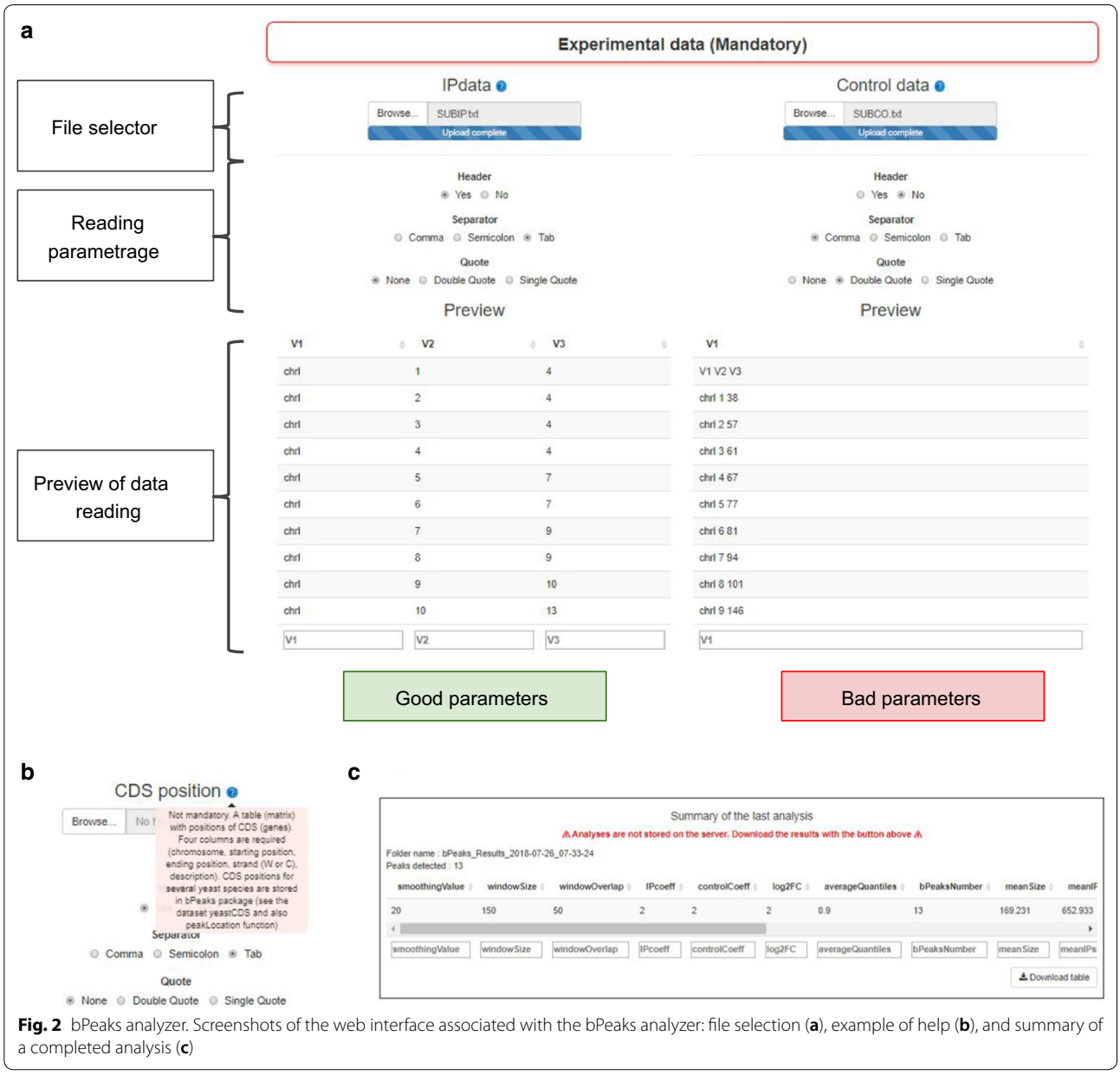


Fig. 2 bPeaks analyzer. Screenshots of the web interface associated with the bPeaks analyzer: file selection (**a**), example of help (**b**), and summary of a completed analysis (**c**)

in the original bPeaks R package). The user can modify the values and run the analysis. Once the analysis is complete, the results are summarized in a table (see Fig. 2c), which can be downloaded. Notably, no file is kept on the server after user sign out. Indeed, during user authentication, a temporary folder is created. All the user analyses and explorations will be saved in this temporary folder. When the user session is over, the folder is deleted. Thus, the bPeaks App does not saturate the workspace memory of the computer on which it is installed. However, the user can download his results at any time (as a zipped file) and save them for future exploration.

bPeaks explorer to inspect detected peaks with an interactive genome browser

bPeaks Explorer is used to generate a graphical and user-interactive overview of the results obtained with bPeaks Analyzer. The web page is divided into five parts (Fig. 3 and more detailed information in Additional file 4):

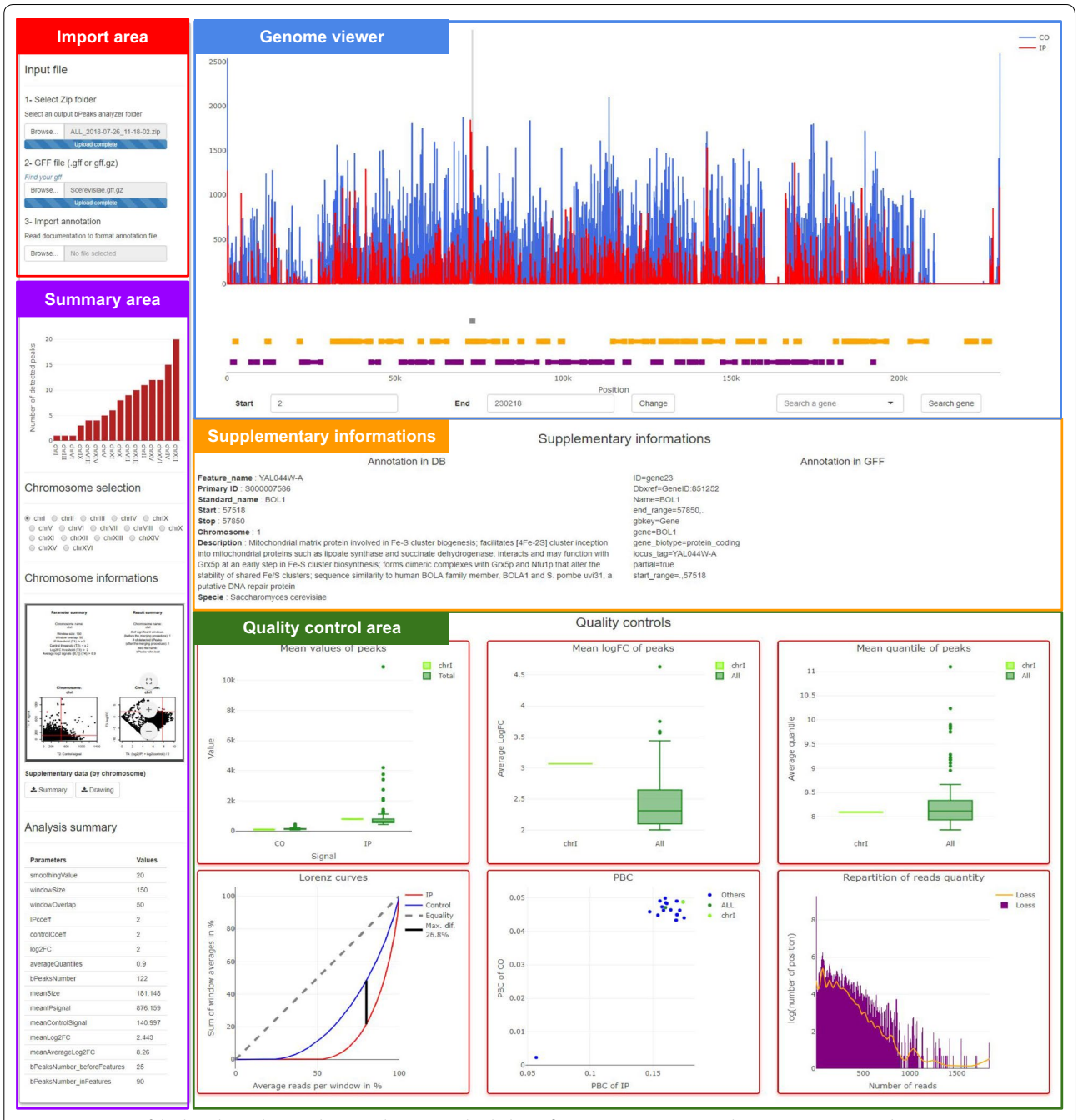


Fig. 3 Structure of the bPeaks explorer web page. The page is divided into five parts: import area (red), summary area (purple), genome viewer (blue), supplementary information (orange), and quality control area (green)

- An import area, in which the user can upload three files: (1) a zip file generated by the bPeaks Analyzer (see the previous section), (2) an annotation file (GFF format) from the NCBI web service [24] to annotate genomic regions of interest, and (3) a gene annotation file from reference databases (format detailed in Additional file 4).
- A summary area, which is comprised of a table containing a summary of the bPeaks analysis parameters, a barplot showing the number of detected peaks per chromosome, and the graph summary per chromosome.
- A user-interactive genome browser (see below) to explore detected peaks.

- A supplementary information area to obtain information about selected genes or selected peaks.
- A quality control area where six graphs are available (average number of reads detected in the peak, average logFC, average quantile, Lorenz curves, and PBC and repartition of read quantity).

Discussion

The objective of the bPeaks App is to empower the use of bPeaks, an efficient peak-caller in small eukaryotic genomes. With its docker, there is no need to worry about installing R and the necessary packages. Indeed, all packages and their dependencies are installed in the image available on Docker Hub. We used a package manager to allow reproducibility of the results. Thus, it is possible to reproduce an analysis with the same packages. Through the use of Shiny, the user does not need any computer or programming skills. The user is guided to enter the various parameters. To help him, information bubbles are available at each step. We provide quality controls (Lorenz curve, PBC, etc.) to validate the experimental part. Thus, the user will know whether the analyses are of good quality or not before exploration. Finally, the exploration of results is greatly simplified through the use of Plotly and its dynamic graphics. The user can browse the genome, chromosome by chromosome, and explore the various detected peaks. In conclusion, we propose a completely open source, free, and user-friendly solution for the detection of binding sites between protein and DNA in eukaryotes with small genomes.

Limitations

The implementation strategies and packages used in R limit the use of bPeaks explorer to organisms with small genomes (< 20 Mb). Moreover, there is no choice in the peak calling strategy. The application only uses and manages results from bPeaks.

Additional files

Additional file 1. Criteria to evaluate ChIP-seq data quality. Illustrated calculation method of the quality criteria: Lorenz curves and PBC.

Additional file 2. Connection to the web interface. Authentication details and starting the bPeaks App.

Additional file 3. A use case in the yeast *Saccharomyces cerevisiae* with transcription factor Pdr1.

Additional file 4. Detailed description of the main parts of the bPeaks explorer application.

Abbreviations

ChIP-seq: chromatin immunoprecipitation sequencing; CO: control; DNA: deoxyribonucleic acid; IP: immunoprecipitation; OS: operating system; PBC: PCR bottleneck coefficient.

Authors' contributions

TD implemented the bPeaks App and GL tested the application. All authors have written, read, and approved the paper.

Acknowledgements

This work was funded by the Agence Nationale pour la Recherche (CANDIHUB project, Grant Number ANR-14-CE14-0018-02). We thank Eleonore Pillot-Lucas for her participation in the project during her 2nd licence year internship, Charles Hébert and Pierre Grognet for helpful discussions.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

Datasets. Example datasets for bPeaks App are available in the Github repository (data folder: https://github.com/thomasdenecker/bPeaks-application/tree/v1.0.0/Data and database folder: https://github.com/thomasdenecker/bPeaks-application/tree/v1.0.0/Database). Raw sequencing data files (FASTQ files) associated to the ChIP-seq analyses of Pdr1 transcription factor (detailed in Additional file 3) in *S. cerevisiae* are available in SRA (https://www.ncbi.nlm.nih.gov/sra) under accession SRX1441673 and SRX1441642. Detailed information regarding the ChIPseq data processing can be found in the original bPeaks article (see https://doi.org/10.1002/yea.3031).

Software. Project name: bPeaks Application; Project home page: https://github.com/thomasdenecker/bPeaks-application/tree/v1.0.0; Archived version: https://doi.org/10.5281/zenodo.1324933; Operating system(s): Windows, Mac Os X, Linux; Programming language: R, HTML, CSS, Javascript; Other requirements: Docker v18 +; License: BSD-3 License.

Consent for publication

Not applicable.

Ethics approval and consent to participate

Not applicable.

Funding

This work was funded by the Agence Nationale pour la Recherche (CANDIHUB project, grant number ANR-14-CE14-0018-02).

Publisher's Note

Received: 11 August 2018 Accepted: 27 September 2018
Published online: 04 October 2018

References

1. Kim TH, Ren B. Genome-wide analysis of protein-DNA interactions. Annu Rev Genomics Hum Genet. 2006;7:81–102.
2. Johnson DS, Mortazavi A, Myers RM, Wold B. Genome-wide mapping of in vivo protein-DNA interactions. Science. 2007;316:1497–502.
3. Thomas R, Thomas S, Holloway AK, Pollard KS. Features that define the best ChIP-seq peak calling algorithms. Brief Bioinform. 2017;18:441–50.
4. Steinhauser S, Kurzawa N, Eils R, Herrmann C. A comprehensive comparison of tools for differential ChIP-seq analysis. Brief Bioinform. 2016;17:953–66.
5. Merhej J, Frigo A, Le Crom S, Camadro J-M, Devaux F, Lelandais G. bPeaks: a bioinformatics tool to detect transcription factor binding sites from ChIPseq data in yeasts and other organisms with small genomes. Yeast. 2014;31:375–91.

6. bPeaks: an intuitive peak-calling strategy to detect transcription factor binding sites from ChIP-seq data in small eukaryotic genomes. Cran. https://cran.r-project.org/web/packages/bPeaks/index.html. Accessed 31 Jul 2018.
7. Thiébaut A, Delaveau T, Benchouaia M, Boeri J, Garcia M, Lelandais G, et al. The CCAAT-binding complex controls respiratory gene expression and iron homeostasis in Candida Glabrata. Sci Rep. 2017;7:3531.
8. Merhej J, Thiebaut A, Blugeon C, Pouch J, Ali Chaouche MEA, Camadro J-M, et al. A network of paralogous stress response transcription factors in the human pathogen Candida glabrata. Front Microbiol. 2016;7:645.
9. Merhej J, Delaveau T, Guitard J, Palancade B, Hennequin C, Garcia M, et al. Yap7 is a transcriptional repressor of nitric oxide oxidase in yeasts, which arose from neofunctionalization after whole genome duplication. Mol Microbiol. 2015;96:951–72.
10. Lelandais G, Blugeon C, Merhej J. ChIPseq in yeast species: from chromatin immunoprecipitation to high-throughput sequencing and bioinformatics data analyses. Methods Mol Biol. 2016;1361:185–202.
11. bPeaks-application. Github. https://github.com/thomasdenecker/bPeaks-application. Accessed 31 Jul 2018.
12. The world's leading software development platform Github. GitHub. https://github.com/. Accessed 27 Jul 2018.
13. Docker-Build, Ship, and Run Any App, Anywhere. Docker. https://www.docker.com/. Accessed 27 Jul 2018.
14. PostgreSQL: the world's most advanced open source database. Postgres. https://www.postgresql.org/. Accessed 27 Jul 2018.
15. Chang W, Cheng J, Allaire JJ, Xie Y, McPherson J. shiny: Web application framework for R. 2018. https://CRAN.R-project.org/package=shiny. Accessed 27 Jul 2018.
16. Sievert C. plotly for R. 2018. https://plotly-book.cpsievert.me. Accessed 27 Jul 2018.
17. Gesmann M, de Castillo D. googleVis: interface between R and the Google Visualisation API. R J. 2011;3:40–4.
18. Conway J, Eddelbuettel D, Nishiyama T, Prayaga SK, Tiffin N. RPostgreSQL: R Interface to the "PostgreSQL" Database System. 2017. https://CRAN.R-project.org/package=RPostgreSQL. Accessed 27 Jul 2018.
19. Ushey K, McPherson J, Cheng J, Atkins A, Allaire JJ. packrat: a dependency management system for projects and their R package dependencies. 2018. https://CRAN.R-project.org/package=packrat. Accessed 27 Jul 2018.
20. bpeaks_db. Dockerhub. https://hub.docker.com/r/tdenecker/bpeaks_db/. Accessed 31 Jul 2018.
21. bpeaks_docker. Dockerhub. https://hub.docker.com/r/tdenecker/bpeaks_docker/. Accessed 31 Jul 2018.
22. bPeaks application—Read me. Github. https://github.com/thomasdenecker/bPeaks-application/blob/master/README.md. Accessed 31 Jul 2018.
23. Landt SG, Marinov GK, Kundaje A, Kheradpour P, Pauli F, Batzoglou S, et al. ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. Genome Res. 2012;22:1813–31.
24. NCBI—Genome. NCBI. https://www.ncbi.nlm.nih.gov/genome. Accessed 31 Jul 2018.

II. L'application Web « Pixel » pour l'annotation, le partage et l'exploration de données multi-omiques

1. Le contexte du projet

Le développement du logiciel Pixel a débuté en octobre 2017 en même temps que ma thèse. Le laboratoire était impliqué dans le projet ANR Candihub[98] et était chargé de la mise en place d'une plateforme informatique, permettant l'annotation et le partage des données expérimentales produites par les différents partenaires du projet (les équipes de recherche de C. d'Enfert et de F. Devaux). Ces données étaient de type « omiques » : RNAseq et ChIPseq essentiellement, obtenues chez les levures pathogènes *Candida albicans* et *Candida glabrata*.

2. Un développement en deux temps

Entre octobre 2017 et mars 2018, le développement de Pixel a été réalisé par un prestataire de service, l'entreprise TailorDev[99]. À partir de Septembre 2018, j'ai travaillé sur une amélioration de l'interface graphique permettant l'exploration des données, ce qui m'a conduit à développer le logiciel Pixel2. Celui-ci utilise la technologie Shiny. Je l'avais en effet trouvée très convaincante lors du projet bPeaks App (section précédente, page 95).

3. Les logiciels Pixel et Pixel2

Le logiciel Pixel est présenté en détails dans la publication disponible en ligne[100] et insérée à la section suivante. Le code source de la première version (entreprise TailorDev) ainsi que le code source de la deuxième version (application Web Shiny) sont disponibles sur Github[101]. Une vidéo de présentation est accessible sur YouTube[102]. Trois instances sont actuellement utilisées au laboratoire, *via* leurs installations sur le serveur d'analyse de notre équipe de recherche[103].

---

[98] https://anr.fr/fr/projets-finances-et-impact/projets-finances/projet/funded/project/anr-14-ce14-0018/?tx_anrprojects_funded%5Bcontroller%5D=Funded&cHash=4eaba391ee411fc8174dbbbdd7aa2c8c

[99] https://tailordev.fr/ [Accessible le 02/06/2020]

[100] https://pubmed.ncbi.nlm.nih.gov/30944779 [Accessible le 02/06/2020]

[101] https://github.com/Candihub/pixel, et https://github.com/thomasdenecker/Pixel2 [Accessible le 02/06/2020]

[102] https://youtu.be/yD-nOTgWXp0 [Accessible le 02/06/2020]

[103] https://pixel.data-fun.io/ [Accessible le 02/06/2020]

4. Le PDF de la publication dans la revue « PeerJ » (Denecker et al. 2019)

Pixel: a content management platform for quantitative omics data

Thomas Denecker[1,*], William Durand[2,*], Julien Maupetit[2,*], Charles Hébert[3], Jean-Michel Camadro[4], Pierre Poulain[4,*] and Gaëlle Lelandais[1,*]

[1] CEA, CNRS, Univ. Paris-Sud, Institute for Integrative Biology of the Cell (I2BC), Gif-sur-Yvette, France
[2] TailorDev SAS, Clermont-Ferrand, France
[3] BIOROSETICS, Houilles, France
[4] CNRS, Univ. Paris Diderot, Institut Jacques Monod (IJM), Paris, France
[*] These authors contributed equally to this work.

ABSTRACT

Background. In biology, high-throughput experimental technologies, also referred as "omics" technologies, are increasingly used in research laboratories. Several thousands of gene expression measurements can be obtained in a single experiment. Researchers are routinely facing the challenge to annotate, store, explore and mine all the biological information they have at their disposal. We present here the Pixel web application (Pixel Web App), an original content management platform to help people involved in a multi-omics biological project.

Methods. The Pixel Web App is built with open source technologies and hosted on the collaborative development platform GitHub (https://github.com/Candihub/pixel). It is written in Python using the Django framework and stores all the data in a PostgreSQL database. It is developed in the open and licensed under the BSD 3-clause license. The Pixel Web App is also heavily tested with both unit and functional tests, a strong code coverage and continuous integration provided by CircleCI. To ease the development and the deployment of the Pixel Web App, Docker and Docker Compose are used to bundle the application as well as its dependencies.

Results. The Pixel Web App offers researchers an intuitive way to annotate, store, explore and mine their multi-omics results. It can be installed on a personal computer or on a server to fit the needs of many users. In addition, anyone can enhance the application to better suit their needs, either by contributing directly on GitHub (encouraged) or by extending Pixel on their own. The Pixel Web App does not provide any computational programs to analyze the data. Still, it helps to rapidly explore and mine existing results and holds a strategic position in the management of research data.

Subjects Computational Biology, Genomics, Data Science
Keywords Data cycle analyses, Omics, Open source, Pixel Web App

INTRODUCTION

In biology, high throughput (HT) experimental technologies—also referred as "omics"—are routinely used in an increasing number of research teams. Financial costs associated to HT experiments have been considerably reduced in the last decade (*Hayden, 2014*) and the trend in HT sequencing (HTS) is now to acquire benchtop machines designed for individual research laboratories (for instance Illumina NextSeq500 or Oxford Nanopore Technologies

Submitted 5 October 2018
Accepted 14 February 2019
Published 27 March 2019

Corresponding authors
Pierre Poulain,
pierre.poulain@univ-paris-diderot.fr
Gaëlle Lelandais,
gaelle.lelandais@u-psud.fr

Academic editor
Gerard Lazo

Additional Information and Declarations can be found on page 15

DOI **10.7717/peerj.6623**

MinION, *Blow, 2013*). The number of HT applications in biology has grown so rapidly in the past decade that it is hard to not feel overwhelmed (*Hadfield & Retief, 2018*) ("*The data deluge, 2012*"). It seems possible to address in any organism, any biological question through an "omics" perspective, providing the right HT material and method are found. If HTS is often put at the forefront of "omics" technologies (essentially genomics and transcriptomics, *Reuter, Spacek & Snyder, 2015*), other technologies must be considered. Mass spectrometry (MS) for instance, enables HT identification and quantification of proteins (proteomics). Metabolomics and lipidomics are other derived applications of MS to characterize quantitative changes in small-molecular weight cellular components (*Smith et al., 2014*). Together, they all account for complementary "omics area" with the advantage to quantify distinct levels of cellular components (transcripts, proteins, metabolites, etc.).

Integration of datasets issued from different HT technologies (termed as multi-omics datasets) represents a challenging task from a statistical and methodological point of view (*Huang, Chaudhary & Garmire, 2017*). It implies the manipulation of two different types of data. The first type is the "primary data", which correspond to raw experimental results. It can be FASTQ files for sequencing technology (*Cock et al., 2010*) or mzML files for MS (*Martens et al., 2011*). These files can be stored in public repositories such as SRA (*Leinonen, Sugawara & Shumway, 2011*), GEO (*Clough & Barrett, 2016*), PRIDE (*Martens et al., 2005*) or PeptideAtlas (*Desiere et al., 2006*). Analyses of primary data rely on standard bioinformatics protocols that for instance, perform quality controls, correct experimental bias or convert files from a specific format to another. A popular tool to analyse primary data is Galaxy (*Afgan et al., 2016*), which is an open web-based platform. "Secondary data" are produced upon analysis of primary data. It can be the counts of reads per genes for HTS results or the abundance values per proteins for MS results. In multi-omics datasets analysis, combining secondary data is essential to answer specific biological questions. It can be typically, the identification of differentially expressed genes (or proteins) between several cell growth conditions from transcriptomics (or proteomics) datasets, or the identification of cellular functions that are over-represented in a list of genes (or proteins). In that respect, secondary data can be analysed and re-analysed within a multitude of analytical strategies, introducing the idea of data analysis cycle. The researcher is thus constantly facing the challenge to properly annotate, store, explore and mine all the biological data he/she has at his/her disposal in a multi-omics project. This challenge is directly related to the ability to extract as much information as possible from the produced data, but also to the crucial question of doing reproducible research.

A Nature's survey presented in 2016 indicates that more than 70% of the questioned researchers already experienced an impossibility to reproduce published results, and more than half of them were not able to reproduce their own experiments (*Baker, 2016*). This last point is intriguing. If experimental biology can be subjected to random fluctuations hardly difficult to control, computational biology should not. Running the same software on the same input data is expected to give the same results. In practice, replication in computational science is harder than people generally think (see *Mesnard & Barba, 2017* as an illustration). It requires to adopt good practices for reproducible-research on a daily basis, and not only when the final results are about to be published. Initiatives

to improve computational reproducibility exists (*Peng, 2011*; *Stodden, Guo & Ma, 2013*; *Vasilevsky et al., 2017*; *Rougier et al., 2017*; *Stodden, Seiler & Ma, 2018*), and today it is clear that the data alone are not enough to sustain scientific claims. Comments, explanations, software source codes and tests are prerequisites to ensure that an original research can be replicated by anyone, anytime, anywhere.

We developed the Pixel web application (Pixel Web App) with these ideas in mind. It is a content management platform to help the researchers involved in a multi-omics biological project, to collaboratively work with their HT data. The Pixel Web App does not store the primary data. It is rather focused on annotation, storage and exploration of secondary data (see Fig. 1). These explorations represent critical steps to answer biological questions and need to be carefully annotated and recorded to be further exploited in the context of new biological questions. The Pixel Web App helps the researcher to specify necessary information required to replicate multi-omics results. We added an original hierarchical system of tags, which allows to easily explore and select multi-omics results stored in the system and to use them for new interpretations. The Pixel Web App can be installed on any individual computer (for a single researcher for instance), or on a web server for collaborative work between several researchers or research teams. The entire software has been developed with high quality programming standards and complies to major rules of open-source development (*Taschuk & Wilson, 2017*). The Pixel project is available on GitHub at https://github.com/Candihub/pixel, where full source code and detailed documentation are provided. We present in this article the Pixel Web App design and implementation. We provide a simple case study, emblematic of our daily use of the Pixel Web App, with the exploration of results issued from transcriptomics and proteomics experiments performed in the pathogenic yeast *Candida glabrata*.

MATERIAL AND METHODS

Stack overview

The Pixel Web App provides researchers an intuitive way to annotate, store, explore and mine their secondary data analyses, in multi-omics biological projects. It is built upon mainstream open source technologies (see Fig. 2). Source code is hosted on the collaborative development platform GitHub (https://github.com/) and continuous integration is provided by CircleCI (https://circleci.com/). More precisely, the Pixel Web App uses the Python Django framework. This framework is based on a model-template-view architecture pattern, and data are stored in a PostgreSQL (https://www.postgresql.org/) database. We have built a docker image for the Pixel Web App. Other containers, Nginx (to serve the Django application) and PostgreSQL rely on official docker images. Each installation/deployment will result in the creation/execution of three docker instances: one for the Pixel Web App, one for the PostgreSQL database and one for the Nginx web server. In case of multiple installations, each trio of docker instances is fully isolated, meaning that data are not shared across multiple Pixel Web App installations.

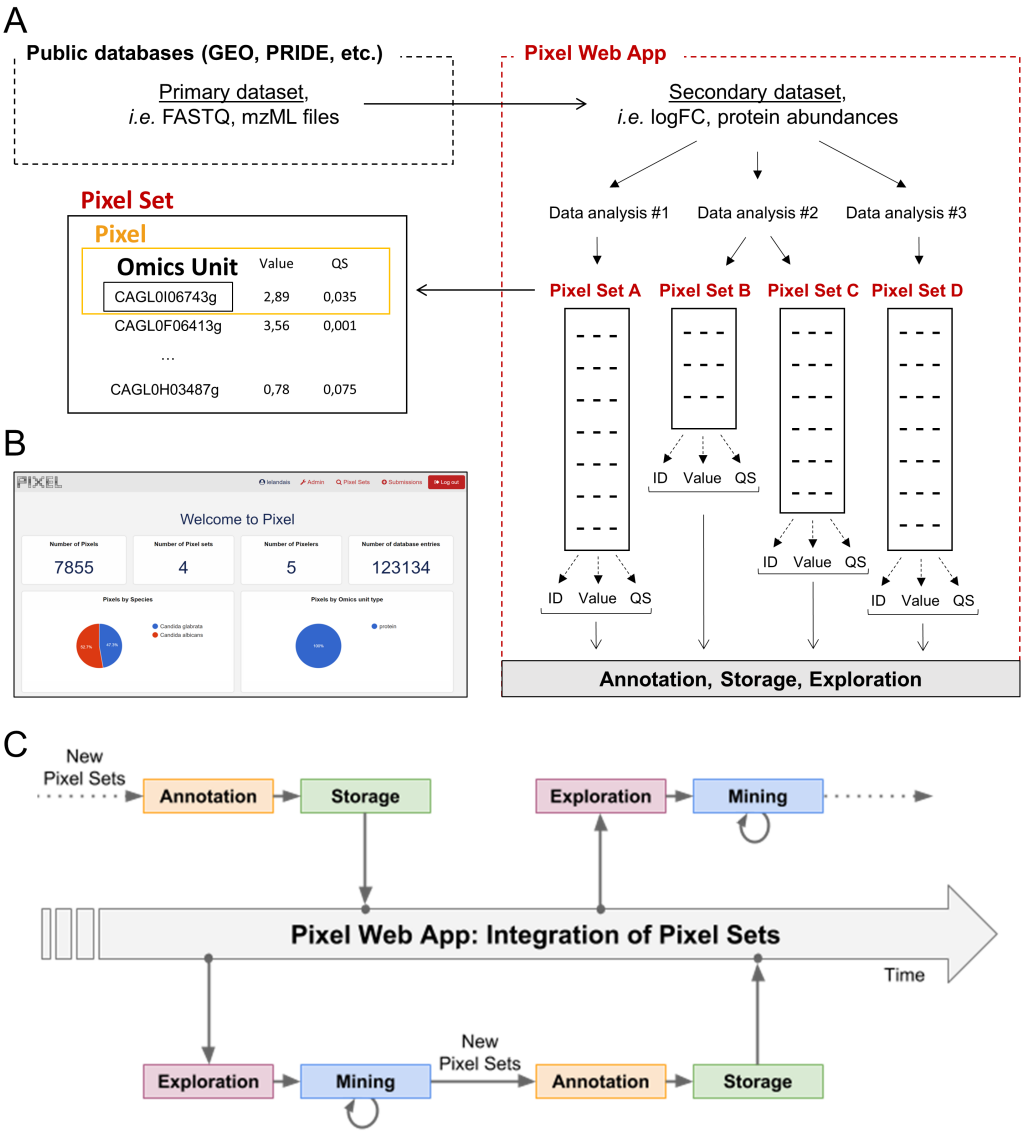


Figure 1 **Dataset flow through the Pixel Web App.** (A) Different types of datasets, which are managed in a multi-omics biological project. Primary and secondary datasets are two types of information arising from HT experimental technologies (see 'Introduction'). Only secondary data and their associated Pixel Sets are stored in the Pixel Web App. Note that several Pixel Sets can emerge from multiple secondary data analyses. They comprise quantitative values (Value) together with quality scores (QS) for several hundred of different "Omics Units" elements (for instance mRNA or proteins, see the main text). Omics Units are identified with a unique identifier (ID). (B) Screenshot of the home page of the Pixel web interface. (C) Schematic representation of the data analysis cycles that surrounds the integration of Pixel Sets in the Pixel Web App (see the main text).

Full-size DOI: 10.7717/peerj.6623/fig-1

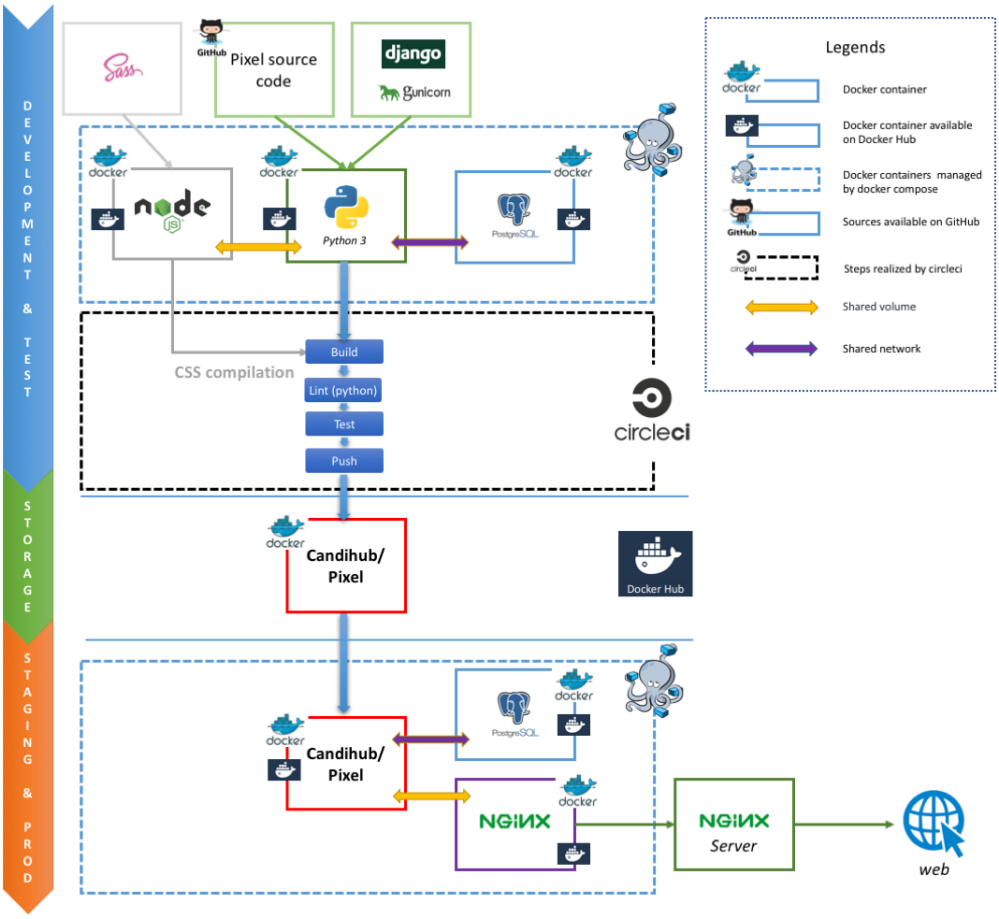


Figure 2 Stack overview of the Pixel Web App. Open source solutions used to develop Pixel are shown here. They are respectively used for the software development and test (blue section), the data storage (green section) and the web application for both staging and production (orange section).
Full-size DOI: 10.7717/peerj.6623/fig-2

Technical considerations

Docker images

The Pixel Web App is built on containerization paradigm (see Fig. 2). It relies on Docker (https://www.docker.com/), i.e., a tool which packages an application and its dependencies in an image that will be run as a container. Docker helps developers to build self-contained images to run a software. These images are downloaded on the host system and used to build the Pixel Web App.

Minimal configuration and dependencies

The Pixel Web App can be deployed on Linux and MacOS operating systems (OS). Deployment on Windows is possible, but this situation will not be described here. Minimal requirements are: *(i)* 64 bits Unix-based OS (Linux/MacOS), *(ii)* Docker community edition >v18, *(iii)* Internet access (required in order to download the Docker images) and *(iv)* [optional] a web server (Apache or Nginx) configured as a reverse proxy.

Installation

A step-by-step tutorial to deploy the Pixel Web App can be found in the project repository (https://github.com/Candihub/pixel/blob/master/docs-install/how-to-install.md) together with a deploy script. To summarize, this script runs the following steps:

- Pull a tagged image of Pixel (web, see docker-composer file),
- Start all instances (web, db and proxy) recreating the proxy and web instances. Collect all static files from the Django app. These files will be served by the proxy instance.
- Migrate the database schema if needed (to preserve existing data).

Note that further technical considerations and full documentation can be found on GitHub repository associated to the Pixel project (https://github.com/Candihub/pixel/tree/master/docs).

RESULTS

Definition of terms: Omics Unit, Pixel and Pixel Set

In the Pixel Web App, the term "Omics Unit" refers to any cellular component, from any organism, which is of interest for the user. The type of Omics Unit depends on the HT experimental technology (transcriptomic, proteomic, metabolomic, etc.) from which primary and secondary datasets were collected and derived (Fig. 1A). In this context, classical Omics Units can be transcripts or proteins, but any other cellular component can be defined as, for instance, genomic regions with "peaks" in case of ChIPseq data analyses (*Merhej et al., 2014*). A "Pixel" refers to a quantitative measurement of a cellular activity associated to a single Omics Unit, together with a quality score (see Fig. 1A). Quantitative measurement and quality score are results of statistical analyses performed on secondary datasets, e.g., search for differentially expressed genes (*Seyednasrollah, Laiho & Elo, 2015*). A set of Pixels obtained from a single secondary data analysis of HT experimental results is referred as a "Pixel Set" (see Fig. 1A). Pixel Sets represent the central information in the Pixel Web App and functionalities to annotate, store, explore and mine multi-omics biological data were designed according to this concept (see below).

Functionalities to annotate, store, explore and mine Pixel Sets

Pixel Sets are obtained from secondary data analyses (see Fig. 1A). Their manipulation with the Pixel Web App consists in *(i)* their annotation, *(ii)* their storage in a database, *(iii)* their exploration and *(iv)* their mining (see Fig. 1C). This represents a cycle of multiple data analyses, which is essential in any multi-omics biological project. These different steps are detailed in the following.

Annotation of Pixel Sets

Annotation of Pixel Sets consists in tracking important details of Pixel Set production. For that, Pixel Sets are associated with metadata, i.e., Supplemental Information linked to the Pixel Sets. We defined minimal information necessary for relevant annotations of Pixel Sets (see Fig. 3). "Species", "Strain", "Omics Unit Type" and "Omics Area" are mandatory information that must be specified *before* a new Pixel Set submission (highlighted in blue, Fig. 3). They refer to general information related to the multi-omics biological project on

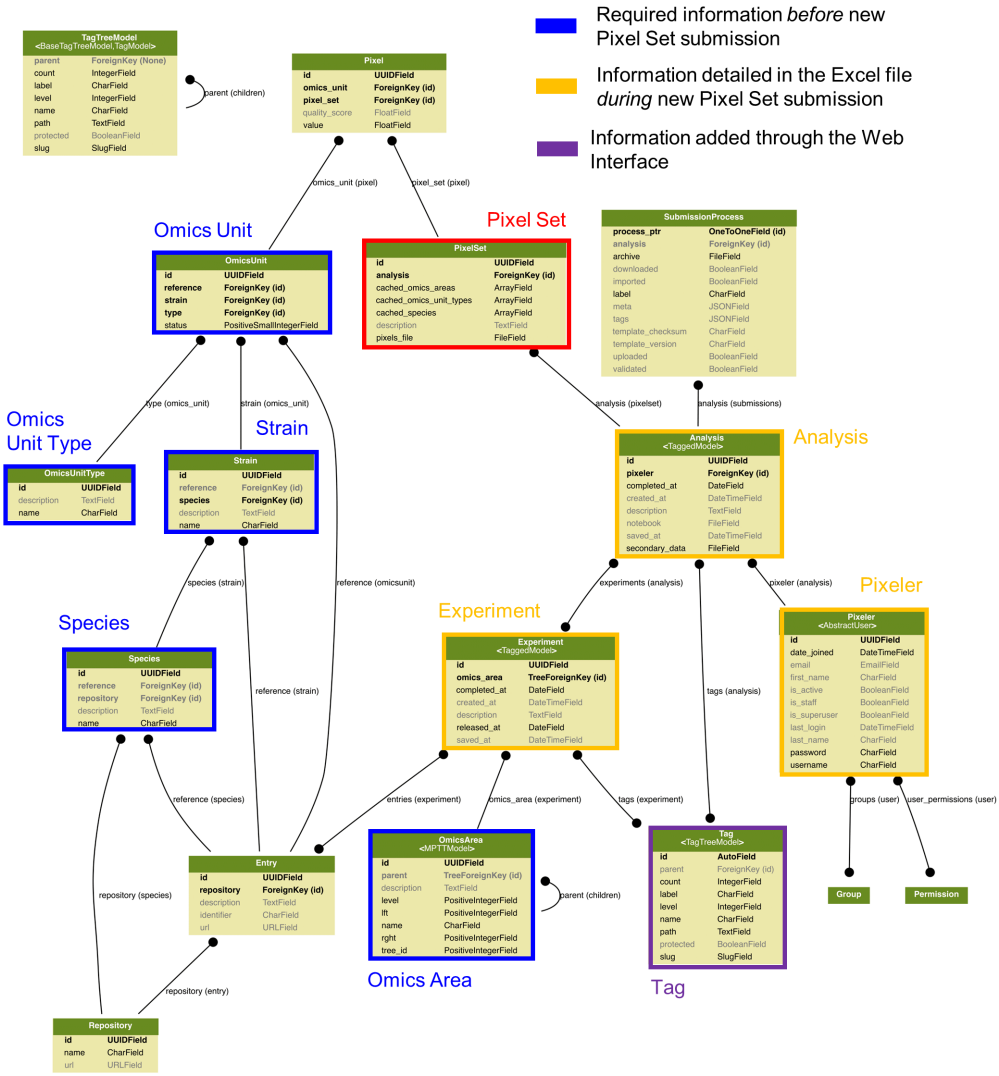


Figure 3 **Data modelling in the Pixel Web App.** The Pixel Set is the central information (see Fig. 1A), the corresponding table in the model is highlighted in red. Information that is required *before* Pixel Set import in the Pixel Web App is surrounded in blue, whereas information required *during* Pixel Set import is highlighted in orange. Other tables are automatically updated during the Pixel Web App data analysis life cycle (see Fig. 1C). An enlarged version of this picture together with full documentation is available online https://github.com/Candihub/pixel/blob/master/docs/pixel-db.pdf.

Full-size DOI: 10.7717/peerj.6623/fig-3

which the researcher is working on: *(i)* the studied organism and its genetic background (Species and Strain, e.g., *Candida glabrata* and ATCC2001), *(ii)* the type of monitored cellular components (Omics Unit Type, e.g., mRNA, protein) and *(iii)* the nature of the experimental HT technology (Omics Area, e.g., RNA sequencing, mass spectrometry). All Omics Units must be declared in the Pixel Web App before new Pixel Set submission. They must be defined with a short description and a link to a reference database. "Experiment" and "Analysis" are Pixel Set mandatory information, input during the submission of new

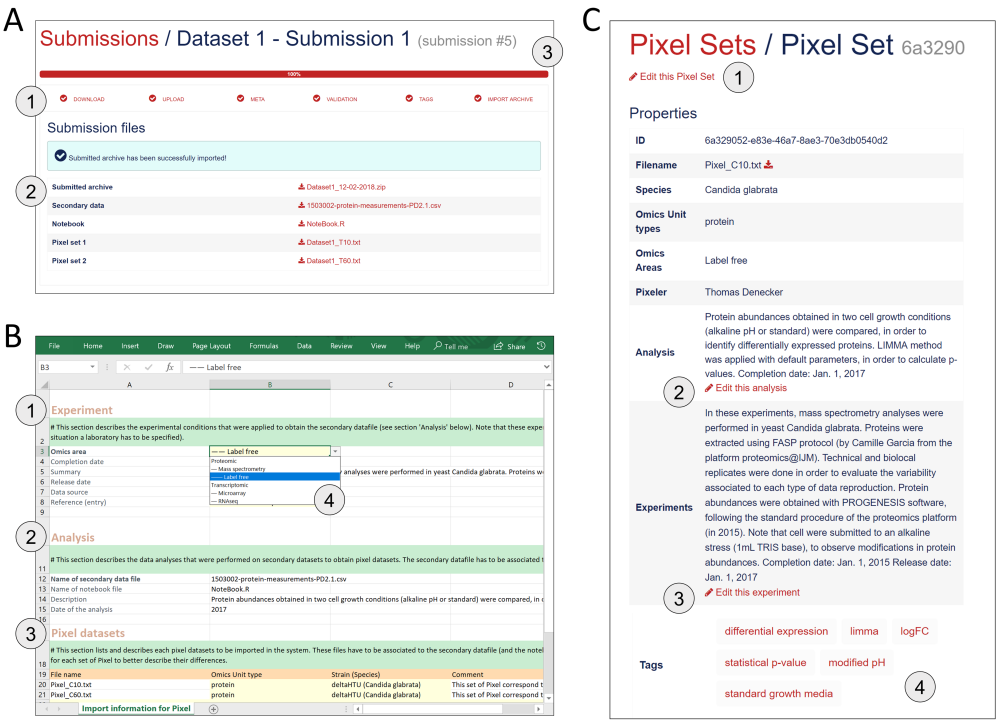


Figure 4 Procedure to import new Pixel Sets in the Pixel Web App. (A) New data-sets are submitted following a dedicated workflow that comprised 6 successive actions named "Download", "Upload", "Meta", "Validation", "Tags" and "Import archive" (see 1). Several files are required (see 2): the secondary data from which the Pixel Sets were calculated, the notebook in which the procedure to compute Pixel Sets from secondary data is described and the Pixel Set files (2 files in this example). A progression bar allows the user to follow the sequence of the submission process. (B) Excel spreadsheet in which annotations of Pixel Sets are written. Information related to the Experiment (see 1), the Analysis (see 2) and the Pixel datasets (see 3) is required. Note that this file must be downloaded at the first step of the submission process ("Download", see A), allowing several cells to be pre-filled with annotations stored in the database (see 4 as an illustration, with Omics area information). (C) All information filled in the Excel file (see B) is extracted and can be modified anytime through a dedicated web page as shown here. User can edit the Pixel Set (see 1), edit the analysis(see 2), edit the experiment (see 3) and add "Tags" (see 4). The Tags are of interest to further explore Pixel Sets in the Pixel Web App.

Full-size DOI: 10.7717/peerj.6623/fig-4

Pixel Sets in the Pixel Web App (highlighted in orange, Fig. 3). They include, respectively, the detailed description of the experimental strategy that was applied to generate primary and secondary data sets (Experiment) and the detailed description of the computational procedures that were applied to obtain Pixel Sets from secondary data set (Analysis). Information regarding the researcher who performed the analyses is referred as "Pixeler".

Storage of Pixel Sets in the database

Import of new Pixel Sets in the Pixel Web App requires the user to follow a workflow for data submission. It corresponds to six successive steps that are explained below (Fig. 4A).

1. The "Download" step consists in downloading a template Excel file from the Pixel Web App (see Fig. 4B). In this file, multiple-choice selections are proposed for "Species", "Strain", "Omics Unit Type" and "Omics Area" fields. These choices

reflect what is currently available in the database and can be easily expanded. User must fill other annotation fields related to the "Experiment", "Analysis" and "Pixeler" information. The Excel file is next bundled into a ZIP archive with the secondary data file (in tab-separated values format), the user notebook (R markdown (https://rmarkdown.rstudio.com/) or Jupyter notebook (http://jupyter.org/) for instance) that contains the code used to produce the Pixel Sets from the secondary data file.

2. The "Upload" step consists in uploading the ZIP file in the Pixel Web App.
3. The step "Meta" consists in running an automatic check of the imported file integrity (md5sum checks are performed, Excel file version is verified, etc.). Note that no information is imported in the database at this stage, but a careful inspection of all Omics Units listed in the submitted Pixel Sets is done. This is why Omics Units need to be pre-registered in the Pixel Web App (see previous section).
4. In "Annotation" step, the annotations of Pixel Sets found in the Excel file (see Fig. 4C) are controlled and validated by the user.
5. Next, the "Tags" step is optional. It gives the opportunity to the user to add tags to the new Pixel Sets (see Fig. 4C), that could be helpful for further Pixel Set explorations (see next section).
6. The final step "Import archive" consists in importing all Pixel Sets in the database, together with annotations and tags.

Note that the procedure of importing metadata as an Excel file has been inspired from the import procedure widely used in GEO (*Clough & Barrett, 2016*).

Exploration of Pixel Sets

The Pixel Web App aims to help researchers to mine and integrate multiple Pixel Sets stored in the system. We developed a dedicated web interface to explore all the Pixel Sets stored in a particular Pixel instance (see Fig. 5). The upper part named "Selection" lists a group of Pixel Sets selected by the user for further explorations (Fig. 5A). The middle part named "Filters" lists the Pixel database contents regarding the Species, Omics Unit Types, Omics Areas and Tags annotation fields. The user can select information (*Candida glabrata* and modified pH here), search and filter the Pixel Sets stored in the database (Fig. 5B). The lower part is a more flexible search field in which keywords can be type. These keywords are searched in the Analysis and Experiment detailed description fields as illustrated here with LIMMA. The web interface also comprised detailed information for the selected subset of Pixel Sets with for instance, distributions of values and quality scores and a list of individual Omics Unit shown at the bottom of the page (Fig. 5C). Note that tags have been implemented to offer to the user a versatile yet robust annotation of Pixel Sets. They are defined during the import process, but they can be modified at any time through the Pixel web interface. Once searched, matching Pixel Sets are gathered in a table that can be exported.

A case study in the pathogenic yeast *Candida glabrata*

The yeast *Candida glabrata* (*C. glabrata*) is a fungal pathogen of human (*Bolotin-Fukuhara & Fairhead, 2014*). It has been reported as the second most frequent cause of invasive

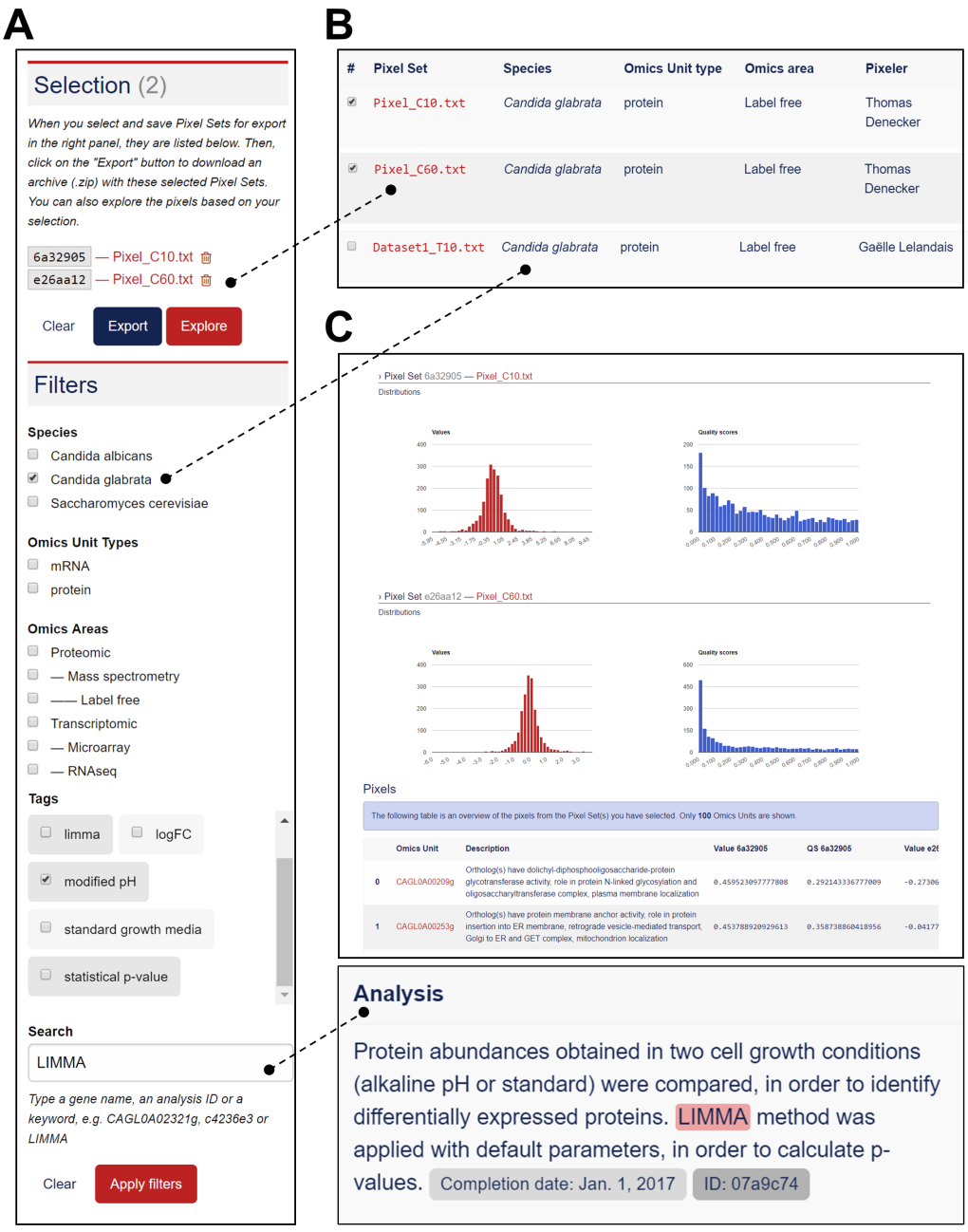


Figure 5 Functionalities to explore the Pixel Sets stored in the Pixel Web App. (A) Screenshot of the exploration menu available *via* the web interface. (B) Screenshot of the table that comprises all Pixel Sets, which match the filter criteria (see A). Particular Pixel Sets can be selected here (for instance "Pixel_C10.txt" and "Pixel_C60.txt"). They will therefore appear in the "Selection" list (see A). (C) Screenshot of the web interface that gives detailed information for the selected subset of Pixel Sets (see A). Distribution of values and quality scores are shown and individual Omics Unit are listed at the bottom of the page.

Full-size DOI: 10.7717/peerj.6623/fig-5

infections due to Candida species, i.e., candidemia, arising especially in patients with compromised immunity (HIV virus infection, cancer treatment, organ transplantation, etc.). Candidemia remains a major cause of morbidity and mortality in the healthcare structures (*Horn et al., 2009*; *Pfaller et al., 2012*). The genome of *Candida glabrata* has been published in 2004 (*Dujon et al., 2004*). Its size is 12.3 Mb with 13 chromosomes and is composed of ~5,200 coding regions. Our research team is familiar with functional genomic studies in *C. glabrata*. In collaboration with experimental biologists, we published in the past ten years half dozen of articles, in which HT technologies were used (*Lelandais et al., 2008*; *Goudot et al., 2011*; *Merhej et al., 2015*; *Merhej et al., 2016*; *Thiébaut et al., 2017*). In our lab, the Pixel Web App is installed locally and store all the necessary genomics annotations to manage any multi-omics datasets in this species.

As a case study, we decided to present how the Pixel Web App can be helpful to answer a specific biological question with only a few mouse clicks. As a biological question, we wanted to identify the genes in the entire *C. glabrata* genome: *(i)* which are annotated as involved in the yeast pathogenicity and *(ii)* for which the expression is significantly modified in response to an environmental stress induced by alkaline pH. Indeed, during a human host infection, *C. glabrata* has to face important pH fluctuations (see *Ullah et al., 2013*; *Brunke & Hube, 2013*; *Linde et al., 2015* for more detailed information). Understanding the molecular processes that allow the pathogenic yeast *C. glabrata* to adapt extreme pH situations is therefore of medical interest to better understand host-pathogen interaction (*Linde et al., 2015*).

In a paper published in 2015, *Linde et al. (2015)* provided a detailed RNAseq based analysis of the transcriptional landscape of *C. glabrata* in several growth conditions, including pH shift experiments. The primary dataset (RNAseq fastq files) is available in the Gene Expression Omnibus (*Clough & Barrett, 2016*) under accession number GSE61606. The secondary dataset (log2 Fold Change values) is available in Table S1 on the journal website (https://academic.oup.com/nar/article/43/3/1392/2411170). A first Pixel Set (labelled A) was created from this secondary dataset, annotated and imported into our Pixel Web App instance, following the procedure previously described. The associated ZIP archive is provided as Supplemental Information, along with all the details related to the experiment set up and the analysis. The Pixel Set A thus illustrates how publicly available data can be managed with the Pixel Web App. In our laboratory, we performed mass spectrometry experiments that also include pH shift (ZIP archive of the data is provided as Supplemental Information). Secondary dataset issued from these experiments leads to the Pixel Set B. Pixel Sets A and B comprise 5,253 Pixels and 1,879 Pixels (Fig. 6).

Transcriptomics (Pixel Set A) and proteomics (Pixel Set B) are interesting complementary multi-omics information that can be easily associated and compared with the Pixel Web App. In that respect, tags allowed to rapidly retrieve them using the web interface, applying the keywords "Candida glabrata" and "alkaline pH" (Fig. 6, Step 1). As we wanted to limit the analysis to the *C. glabrata* genes potentially involved in the yeast pathogenesis, a filter could be used to only retain the Omics Units for which the keyword "pathogenicity" is written in their description field (see Fig. 6, Step 2). As a result, a few numbers of Pixels were thus selected, respectively 17 in Pixel Set A and 6

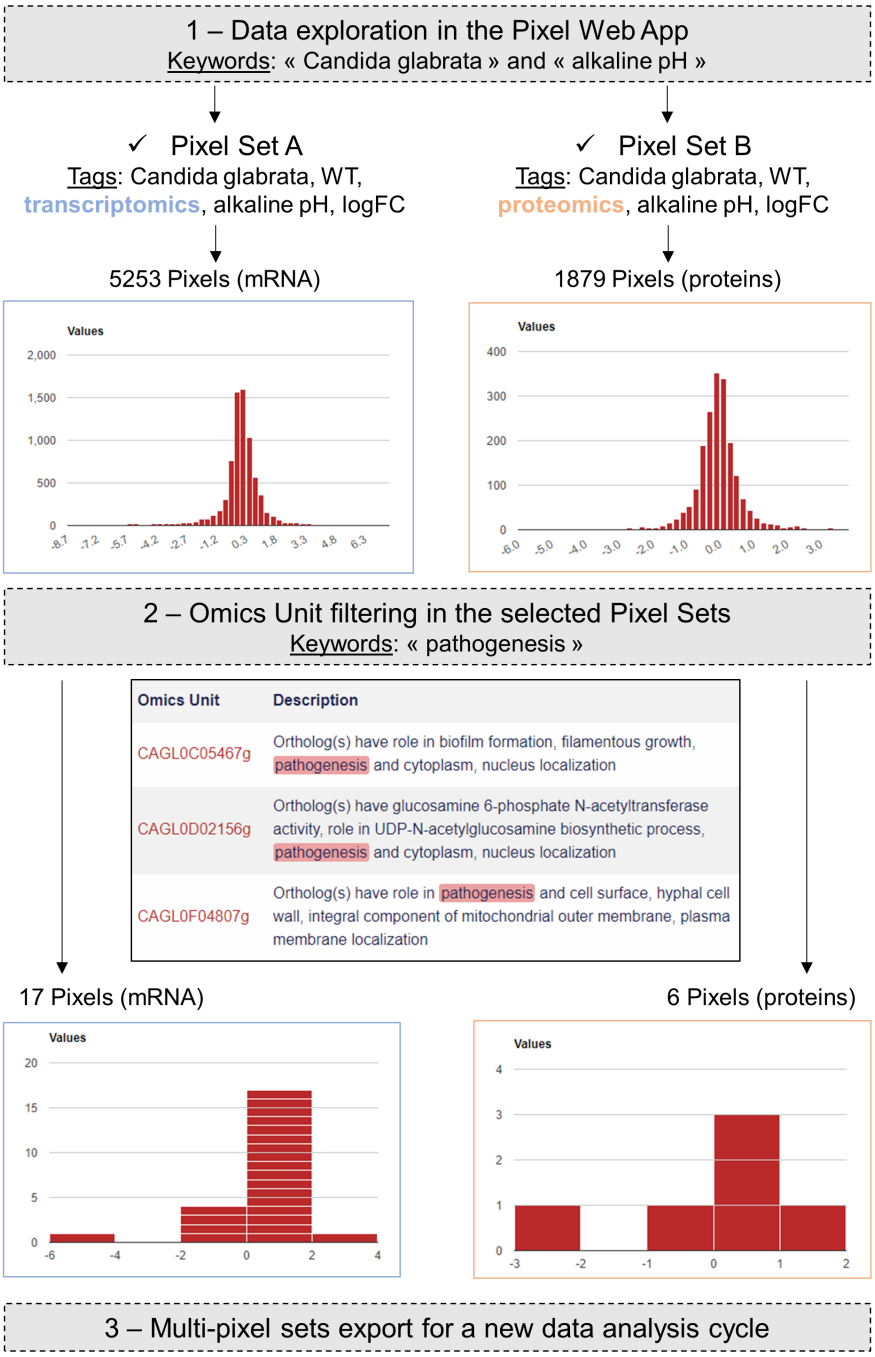


Figure 6 **Case study in the pathogenic yeast *Candida glabrata*.** Our Pixel Web App was explored with the keywords "Candida glabrata" and "alkaline pH". Two Pixel Sets were thus identified because of their tags. Two other tags were identical between the two Pixel Sets ("WT" and "logFC"), indicating that *(i) C. glabrata* strains are the same, *i.e.,* Wild Type, and *(ii)* Pixel values are of the same type, *i.e.,* log Fold Change. Notably Pixel Set A is based on transcriptomics experiments (RNAseq, see the main text), whereas Pixel Set B is based on proteomics experiments (mass spectrometry, see the main text). Omics Unit were next explored using the keyword "pathogenesis" resulting in the identification of 17 Pixels (respectively 6 Pixels) in transcriptomics (respectively proteomics) results. They were combined and exported from the Pixel Web App, hence starting a new data analysis cycle.

Full-size DOI: 10.7717/peerj.6623/fig-6

Table 1 Detailed information regarding the Omics Unit identified in the *C. glabrata* case study. The two first columns give Omics Unit information as described in the Candida Genome Database (*Skrzypek et al., 2017*). All the description fields comprise the keyword "pathogenesis" (in bold). LogFC values measured in transcriptomic (Pixel Set A) and proteomic (Pixel Set B) experiments are shown in the third and fourth columns. Quality scores (QS) are following logFC values. They are *p*-values coming from the differential analysis of logFC replicates. The entire table of multi-pixel sets is available in Supplementary Data.

Omics unit	Description	A	B	A (QS)	B (QS)
1. CAGL0F04807g	Ortholog(s) have role in **pathogenesis** and cell surface, hyphal cell wall, integral component of mitochondrial outer membrane, plasma membrane localization	1,09	1,81	2,23E−19	7,31E−05
2. CAGL0F06457g	Ortholog(s) have role in fungal-type cell wall organization or biogenesis, mitochondrial outer membrane translocase complex assembly, **pathogenesis**, phospholipid transport, protein import into mitochondrial outer membrane	0,30	0,19	4,14E−02	2,65E−01
3. CAGL0I02970g	Ortholog(s) have delta14-sterol reductase activity and role in cellular response to drug, ergosterol biosynthetic process, filamentous growth of a population of unicellular organisms in response to biotic stimulus, **pathogenesis**	0,90	−2,64	4,65E−16	2,19E−05
4. CAGL0I10516g	Ortholog(s) have role in fungal-type cell wall organization, pathogenesis and cytoplasm, eisosome, integral component of plasma membrane, membrane raft localization	1,50	0,57	8,29E−60	1,16E−02
5. CAGL0L08448g	Ortholog(s) have role in actin cytoskeleton organization, eisosome assembly, negative regulation of protein phosphorylation, negative regulation of sphingolipid biosynthetic process and **pathogenesis**	1,67	−0,57	1,77E−75	7,04E−03

in Pixel Set B. The last step consists in combining the mRNA and protein information (see Fig. 6, Step 3). For that a table comprising the multi-pixel sets can be automatically generated and easily exported. We present Table 1: five genes for which logFC values were obtained both at the mRNA and the protein levels, and for which statistical p-values were significant (<0.05). Notably two genes (CAGL0I02970g and CAGL0L08448g, lines 3 and 5 in Table 1) exhibited opposite logFC values, i.e., induction was observed at the mRNA level whereas repression was observed at the protein levels. Such observations can arise from post-translational regulation processes or from possible experimental noise, which could explain approximative mRNA or protein quantifications. In both cases, further experimental investigations are required. The three other genes (CAGL0F04807g, CAGL0F06457g and CAGL0I10516g, underlined in grey in Table 1) exhibited multi-omics coherent results and significant inductions were observed at the mRNA and protein levels. Again, further experimental investigations are required to fully validated these observations. Still, it is worth noting that the gene CAGL0F04807g, is described as "uncharacterized" in the Candida Genome Database (http://www.candidagenome.org/cgi-bin/locus.pl?locus=CAGL0F04807g&organism=C_glabrata_CBS138). Considering that logFC values for this gene are particularly high (>1), such an observation represents a good starting point to refine the functional annotation of this gene, clearly supporting the hypothesis that it has a role in the ability of *C. glabrata* to deal with varying pH situations.

Software availability

Pixel is released under the open-source 3-Clause BSD license (https://opensource.org/licenses/BSD-3-Clause). Its source code can be freely downloaded from the GitHub repository of the project: https://github.com/Candihub/pixel. In addition, the present version of Pixel (4.0.4) is also archived in the digital repository Zenodo (https://doi.org/10.5281/zenodo.1434316).

DISCUSSION

In this article, we introduced the principle and the main functionalities of the Pixel Web App. With this application, our aim was to develop a tool to support on a daily basis, the biological data mining in our multi-omics research projects. It is our experience that research studies in which HT experimental strategies are applied, require much more time to analyse and interpret the data, than to experimentally generate the data. Testing multiple bioinformatics tools and statistical approaches is a critical step to fully understand the meaning of a biological dataset and in this context, the annotation, the storage and the ability to easily explore all results obtained in a laboratory can be the decisive steps to the success of the entire multi-omics project.

The data modelling around which the Pixel Web App was developed has been conceived to find a compromise between a too detailed and precise description of the data (which could discourage the researchers from systematically using the application after each of their analyses) and a too short and approximate description of the data (which could prevent the perfect reproduction of the results by anyone). Also, attention has been paid to allow heterogeneous data, i.e., different Omics Unit Type quantified in different Omics Area, to be stored in a coherent and flexible way. The Pixel Web App does not provide any computational programs to analyse the data. Still, it allows to explore existing results in a laboratory and to rapidly combine them for further investigations (using for instance the Galaxy platform or any other data analysis tool).

Therefore, the Pixel Web App holds a strategic position in the data management in a research laboratory, i.e., as the starting point but also at the final point of all new data explorations. It also helps data analysis reproducibility and gives a constant feedback regarding the frequency of the data analysis cycles; the nature of the import and export data sets as well as full associated annotations. It is thus expected that the content of different Pixel Web App instance will evolve with time, according to the type of information stored in the system and the scientific interests of a research team.

CONCLUSION

The Pixel Web App is freely available to any interested parties. The initial installation on a personal workstation required IT support from a bioinformatician, but once this is done, all administration tasks can be performed through the Web Interface. This is of interest for user with a few technical skills. We chose to work exclusively with open source technologies and our GitHub repository is publicly accessible (https://github.com/Candihub/pixel). We

thus hope that the overall quality of the Pixel Web App source code and documentation will be guaranteed over time, through the shared contributions of other developers.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This work was funded by the Agence Nationale pour la Recherche (CANDIHUB project, grant number ANR-14-CE14-0018-02). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
Agence Nationale pour la Recherche (CANDIHUB project: ANR-14-CE14-0018-02.

Competing Interests

William Durand and Julien Maupetit are employed by TailorDev SAS. Charles Hébert is employed by Biorosetics.

Author Contributions

- Thomas Denecker performed the experiments, contributed reagents/materials/analysis tools, prepared figures and/or tables, authored or reviewed drafts of the paper, approved the final draft.
- William Durand conceived and designed the experiments, performed the experiments, approved the final draft.
- Julien Maupetit and Charles Hébert conceived and designed the experiments, performed the experiments, contributed reagents/materials/analysis tools, approved the final draft.
- Jean-Michel Camadro analyzed the data, approved the final draft.
- Pierre Poulain conceived and designed the experiments, contributed reagents/materials/analysis tools, authored or reviewed drafts of the paper, approved the final draft.
- Gaëlle Lelandais conceived and designed the experiments, analyzed the data, contributed reagents/materials/analysis tools, prepared figures and/or tables, authored or reviewed drafts of the paper, approved the final draft.

Data Availability

The following information was supplied regarding data availability:

Source code can be freely downloaded from the GitHub repository of the project: https://github.com/Candihub/pixel.

The present version of Pixel (4.0.4) is also archived at Zenodo:

Durand, William, Maupetit, Julien, Denecker, Thomas, Hébert, Charles, Poulain, Pierre, & Lelandais, Gaëlle. (2018, September 24). Pixel (v4.0.4): Integration of smart 'omics' data (Version 4.0.4). Zenodo. http://doi.org/10.5281/zenodo.1434316.

Supplemental Information

Supplemental information for this article can be found online at http://dx.doi.org/10.7717/peerj.6623#supplemental-information.

REFERENCES

Afgan E, Baker D, Van den Beek M, Blankenberg D, Bouvier D, Čech M, Chilton J, Clements D, Coraor N, Eberhard C, Grüning B, Guerler A, Hillman-Jackson J, Von Kuster G, Rasche E, Soranzo N, Turaga N, Taylor J, Nekrutenko A, Goecks J. 2016. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. *Nucleic Acids Research* **44**:W3–W10 DOI 10.1093/nar/gkw343.

Baker M. 2016. 1,500 scientists lift the lid on reproducibility. *Nature* **533**:452–454 DOI 10.1038/533452a.

Blow N. 2013. A sequencer in every lab. *BioTechniques* **55**:284 DOI 10.2144/000114107.

Bolotin-Fukuhara M, Fairhead C. 2014. Candida glabrata: a deadly companion? *Yeast* **31**:279–288 DOI 10.1002/yea.3019.

Brunke S, Hube B. 2013. Two unlike cousins: *Candida albicans* and *C. glabrata* infection strategies. *Cellular Microbiology* **15**:701–708 DOI 10.1111/cmi.12091.

Clough E, Barrett T. 2016. The gene expression omnibus database. *Methods in Molecular Biology* **1418**:93–110 DOI 10.1007/978-1-4939-3578-9_5.

Cock PJ, Fields CJ, Goto N, Heuer ML, Rice PM. 2010. The sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Research* **38**:1767–1771 DOI 10.1093/nar/gkp1137.

Desiere F, Deutsch EW, King NL, Nesvizhskii AI, Mallick P, Eng J, Chen S, Eddes J, Loevenich SN, Aebersold R. 2006. The PeptideAtlas project. *Nucleic Acids Research* **34**:D655–D658 DOI 10.1093/nar/gkj040.

Dujon B, Sherman D, Fischer G, Durrens P, Casaregola S, Lafontaine I, De Montigny J, Marck C, Neuvéglise C, Talla E, Goffard N, Frangeul L, Aigle M, Anthouard V, Babour A, Barbe V, Barnay S, Blanchin S, Beckerich J-M, Beyne E, Bleykasten C, Boisramé A, Boyer J, Cattolico L, Confanioleri F, De Daruvar A, Despons L, Fabre E, Fairhead C, Ferry-Dumazet H, Groppi A, Hantraye F, Hennequin C, Jauniaux N, Joyet P, Kachouri R, Kerrest A, Koszul R, Lemaire M, Lesur I, Ma L, Muller H, Nicaud J-M, Nikolski M, Oztas S, Ozier-Kalogeropoulos O, Pellenz S, Potier S, Richard G-F, Straub M-L, Suleau A, Swennen D, Tekaia F, Wésolowski-Louvel M, Westhof E, Wirth B, Zeniou-Meyer M, Zivanovic I, Bolotin-Fukuhara M, Thierry A, Bouchier C, Caudron B, Scarpelli C, Gaillardin C, Weissenbach J, Wincker P, Souciet J-L. 2004. Genome evolution in yeasts. *Nature* **430**:35–44 DOI 10.1038/nature02579.

Goudot C, Etchebest C, Devaux F, Lelandais G. 2011. The reconstruction of condition-specific transcriptional modules provides new insights in the evolution of yeast AP-1 proteins. *PLOS ONE* **6**:e20924 DOI 10.1371/journal.pone.0020924.

Hadfield J, Retief J. 2018. A profusion of confusion in NGS methods naming. *Nature Methods* **15**:7–8 DOI 10.1038/nmeth.4558.

Hayden EC. 2014. The $1,000 genome. *Nature* **507**:294–295 DOI 10.1038/507294a.

Horn DLL, Neofytos D, Anaissie EJJ, Fishman JAA, Steinbach WJJ, Olyaei AJJ, Marr KAA, Pfaller MAA, Chang CC-H, Webster KMM. 2009. Epidemiology and outcomes of Candidemia in 2019 patients: data from the prospective antifungal therapy alliance registry. *Clinical Infectious Diseases* **48**:1695–1703 DOI 10.1086/599039.

Huang S, Chaudhary K, Garmire LX. 2017. More is better: recent progress in multi-omics data integration methods. *Frontiers in Genetics* **8**:1–12 DOI 10.3389/fgene.2017.00084.

Leinonen R, Sugawara H, Shumway M, International Nucleotide Sequence Database Collaboration. 2011. The sequence read archive. *Nucleic Acids Research* **39**:D19–D21 DOI 10.1093/nar/gkq1019.

Lelandais G, Tanty V, Geneix C, Etchebest C, Jacq C, Devaux F. 2008. Genome adaptation to chemical stress: clues from comparative transcriptomics in Saccharomyces cerevisiae and Candida glabrata. *Genome Biology* **9**:Article R164 DOI 10.1186/gb-2008-9-11-r164.

Linde JJ, Duggan SS, Weber M, Horn F, Sieber P, Hellwig D, Riege K, Marz M, Martin R, Guthke R, Kurzai O. 2015. Defining the transcriptomic landscape of Candida glabrata by RNA-Seq. *Nucleic Acids Research* **43**:1392–1406 DOI 10.1093/nar/gku1357.

Martens L, Chambers M, Sturm M, Kessner D, Levander F, Shofstahl J, Tang WH, Römpp A, Neumann S, Pizarro AD, Montecchi-Palazzi L, Tasman N, Coleman M, Reisinger F, Souda P, Hermjakob H, Binz P-A, Deutsch EW. 2011. mzML—a community standard for mass spectrometry data. *Molecular & Cellular Proteomics* **10**:Article R110.000133 DOI 10.1074/mcp.R110.000133.

Martens L, Hermjakob H, Jones P, Adamski M, Taylor C, States D, Gevaert K, Vandekerckhove J, Apweiler R. 2005. PRIDE: the proteomics identifications database. *Proteomics* **5**:3537–3545 DOI 10.1002/pmic.200401303.

Merhej J, Delaveau T, Guitard J, Palancade B, Hennequin C, Garcia M, Lelandais G, Devaux F. 2015. Yap7 is a transcriptional repressor of nitric oxide oxidase in yeasts, which arose from neofunctionalization after whole genome duplication. *Molecular Microbiology* **96**:n/a–n/a DOI 10.1111/mmi.12983.

Merhej J, Frigo A, Le Crom S, Camadro JJ, Devaux F, Lelandais G. 2014. bPeaks: a bioinformatics tool to detect transcription factor binding sites from ChIPseq data in yeasts and other organisms with small genomes. *Yeast* **31**:375–391 DOI 10.1002/yea.3031.

Merhej J, Thiebaut A, Blugeon C, Pouch J, Ali Chaouche MEA, Camadro J-M, Le Crom S, Lelandais G, Devaux F. 2016. A network of paralogous stress response transcription factors in the human pathogen candida glabrata. *Frontiers in Microbiology* **7**:1–16 DOI 10.3389/fmicb.2016.00645.

Mesnard O, Barba LA. 2017. Reproducible and replicable computational fluid dynamics: it's harder than you think. *Computing in Science & Engineering* **19**:44–55 DOI 10.1109/MCSE.2017.3151254.

Peng RD. 2011. Reproducible research in computational science. *Science* **334**:1226–1227 DOI 10.1126/science.1213847.

Pfaller M, Neofytos D, Diekema D, Azie N, Meier-Kriesche HU, Quan SP, Horn D. 2012. Epidemiology and outcomes of candidemia in 3,648 patients: data from the Prospective Antifungal Therapy (PATH Alliance) registry, 2004–2008. *Diagnostic Microbiology and Infectious Diseas* **74**:323–331 DOI 10.1016/j.diagmicrobio.2012.10.003.

Reuter JAA, Spacek DV, Snyder MPP. 2015. High-throughput sequencing technologies. *Molecular Cell* **58**:586–597 DOI 10.1016/j.molcel.2015.05.004.

Rougier NP, Hinsen K, Alexandre F, Arildsen T, Barba LA, Benureau FCYY, Brown CT, De Buyl P, Caglayan O, Davison AP, Delsuc M-AA, Detorakis G, Diem AK, Drix D, Enel P, Girard B, Guest O, Hall MG, Henriques RN, Hinaut X, Jaron KS, Khamassi M, Klein A, Manninen T, Marchesi P, McGlinn D, Metzner C, Petchey OL, Plesser HE, Poisot T, Ram K, Ram Y, Roesch E, Rossant C, Rostami V, Shifman A, Stachelek J, Stimberg M, Stollmeier F, Vaggi F, Viejo G, Vitay J, Vostinar AE, Yurchak R, Zito T. 2017. Sustainable computational science: the ReScience initiative. *PeerJ Computer Science* **3**:1–8 DOI 10.7717/peerj-cs.142.

Seyednasrollah F, Laiho A, Elo LL. 2015. Comparison of software packages for detecting differential expression in RNA-seq studies. *Briefings in Bioinformatics* **16**:59–70 DOI 10.1093/bib/bbt086.

Skrzypek MS, Binkley J, Binkley G, Miyasato SR, Simison M, Sherlock G. 2017. The Candida Genome Database (CGD): incorporation of Assembly 22, systematic identifiers and visualization of high throughput sequencing data. *Nucleic Acids Research* **45**:D592–D596 DOI 10.1093/nar/gkw924.

Smith R, Mathis A, Ventura D, Prince J. 2014. Proteomics, lipidomics, metabolomics: a mass spectrometry tutorial from a computer scientist's point of view. *BMC Bioinformatics* **15**:S9 DOI 10.1186/1471-2105-15-S7-S9.

Stodden V, Guo P, Ma Z. 2013. Toward reproducible computational research: an empirical analysis of data and code policy adoption by journals. *PLOS ONE* **8**:e67111 DOI 10.1371/journal.pone.0067111.

Stodden V, Seiler J, Ma Z. 2018. An empirical analysis of journal policy effectiveness for computational reproducibility. *Proceedings of the National Academy of Sciences of the United States of America* **115**:2584–2589 DOI 10.1073/pnas.1708290115.

Taschuk M, Wilson G. 2017. Ten simple rules for making research software more robust. *PLOS Computational Biology* **13**:e1005412 DOI 10.1371/journal.pcbi.1005412.

The data deluge. 2012. *Nature Cell Biology* **14**:775–775 DOI 10.1038/ncb2558.

Thiébaut A, Delaveau T, Benchouaia M, Boeri J, Garcia M, Lelandais G, Devaux F. 2017. The CCAAT-binding complex controls respiratory gene expression and iron homeostasis in candida glabrata. *Scientific Reports* **7**:Article 3531 DOI 10.1038/s41598-017-03750-5.

Ullah A, Lopes MI, Brul S, Smits GJ. 2013. Intracellular pH homeostasis in Candida glabrata in infection-associated conditions. *Microbiology* **159**:803–813 DOI 10.1099/mic.0.063610-0.

Vasilevsky NA, Minnier J, Haendel MA, Champieux RE. 2017. Reproducible and reusable research: are journal data sharing policies meeting the mark? *PeerJ* **5**:e3208 DOI 10.7717/peerj.3208.

III. Le partage d'expérience pour aider à la création d'applications WEB avec Shiny

1. Une rencontre sur les réseaux sociaux

Convaincu de l'intérêt d'utiliser la solution technique Shiny (Chang et al. 2019), j'ai souhaité partager mon expérience avec la communauté bioinformatique Française. J'ai donc contacté *via* les réseaux sociaux, l'équipe en charge de l'animation du site Bioinfo-fr.net[104]. Cette équipe recherche régulièrement des contributeurs pour rédiger des articles de blog ou proposer des tutoriaux. Je lui ai donc proposé au printemps 2019 la rédaction d'un petit article en Français, permettant de s'initier au développement d'une application WEB à l'aide de Shiny.

2. Un projet pour débuter

L'article présente, pas à pas, les étapes de la création d'une application WEB. Pour faire simple, j'ai décidé d'utiliser un jeu de données très connu des utilisateurs de R : le jeu de données IRIS[105]. Après une introduction présentant l'architecture d'une application WEB, l'article présente les différentes étapes nécessaires pour créer une application complète et l'utiliser pour explorer et visualiser les données. Le code informatique est présenté et commenté. L'application finale permet de lire un fichier en fonction de paramètres enregistrés par l'utilisateur puis d'afficher (1) un tableau avec de la coloration conditionnelle et (2) quatre graphiques obtenus par des approches différentes. Le premier graphique exploite les fonctionnalités de base de représentations de données proposées avec R, le deuxième graphique est réalisé avec la librairie ggplot2 (Wickham 2016), et les deux suivants sont des graphiques dynamiques et interactifs créés à l'aide de Plotly (Sievert 2018) et Google (Gesmann et al. 2011). L'article est disponible en ligne[106] et intégré dans la section suivante. L'ensemble du code permettant de générer l'application est disponible sur GitHub[107].

---

[104] https://bioinfo-fr.net/ [Accessible le 02/06/2020]

[105] https://archive.ics.uci.edu/ml/datasets/iris et https://github.com/bioinfo-fr/bioinfo-fr_Shiny/blob/master/datasetIris.txt [Accessible le 02/06/2020]

[106] https://bioinfo-fr.net/rendre-ses-projets-r-plus-accessibles-grace-a-shiny [Accessible le 02/06/2020]

[107] https://github.com/bioinfo-fr/bioinfo-fr_Shiny [Accessible le 02/06/2020]

3. Le PDF de l'article publié sur le site Internet « Bioinfo-fr.net » (Denecker, 2019)

Découverte :

Rendre ses projets R plus accessibles grâce à Shiny

mer 24 Avr 2019 Thomas Denecker Découverte, Didacticiel 3

Bonjour à tous !

Vous avez un script que vous souhaitez partager avec une équipe expérimentale? Vous ne voulez pas que les utilisateurs modifient le code pour paramétrer votre programme? Vous codez avec R ? Alors cet article est fait pour vous ! Nous allons voir comment créer une application web avec R et permettre à votre utilisateur d'exécuter votre code sans le voir.

Shiny

Le package que nous utiliserons est shiny. Il est proposé par Rstudio (https://shiny.rstudio.com/) et disponible sur le CRAN. Ce package permet de construire des applications web très simplement sans connaissances particulières en HTML et CSS. Les fonctions que nous appellerons dans R vont être traduites en HTML. Par exemple, `h1('Un titre')` sera transformé en `<h1>Un titre</h1>`. Il n'est donc pas indispensable de savoir coder en HTML, mais des connaissances dans les langages web pourront vous être utiles dans des cas particuliers, puisqu'il est possible d'intégrer dans l'application shiny du code HTML brut.

Une application shiny se divise en 2 parties :

- l'UI : Il s'agit de l'interface utilisateur visible dans une page web. Nous pourrons y retrouver des graphes, des tableaux, du texte, etc. L'utilisateur pourra interagir avec cette interface par le biais de boutons, de sliders, de cases, etc.
- le serveur : Il s'agit de la « zone de travail ». Tous les calculs, préparations de données ou analyses que R réalisera seront faits côté serveur.

Nous allons voir dans cet article toutes les étapes pour créer une application complète. Elle sera capable de lire un fichier en fonction de paramètres enregistrés par l'utilisateur puis d'afficher :

- Un tableau avec de la coloration conditionnelle
- 4 graphiques obtenus par des approches différentes
  - Un classique réalisé avec R. Ce graphique sera paramétrable par l'utilisateur (couleur ou titre par exemple).
  - Un réalisé avec la librairie ggplot2
  - Un dynamique réalisé avec plotly
  - Un dynamique réalisé avec Google

L'ensemble du code permettant de réaliser l'application est disponible sur github : https://github.com/bioinfo-fr/bioinfo-fr_Shiny

Pré-requis

Toutes les étapes pour créer une application Shiny seront détaillées dans ce post. Connaître la syntaxe de R simplifiera grandement la lecture de l'article mais n'est pas indispensable.

Pour réaliser cette application, il vous faudra une version à jour de RStudio (plus simple que la console R). Pour l'installer, suivez les étapes suivantes (l'ordre est important) :

1. installer R : https://cran.r-project.org/
2. Installer RStudio : https://www.rstudio.com/products/rstudio/download/

Note : pour les utilisateurs de R les plus avancés, l'application peut être développée dans un environnement virtuel comme docker (sujet de mon prochain post).

Les données

Les données utilisées pour cette application proviennent du tableau IRIS regroupant des mesures sur des fleurs (disponible dans Rdataset et décrit ici https://archive.ics.uci.edu/ml/datasets/iris). Ce jeu de données est très utilisé pour illustrer les fonctions dans R et pour le *machine learning*. Le tableau est composé de 5 colonnes :

- la longueur des sépales ;
- la largeur des sépales ;
- la longueur des pétales ;
- la largeur des pétales ;
- l'espèce de fleurs.

Un fichier au format txt est disponible ici :
https://github.com/bioinfo-fr/bioinfo-fr_Shiny/blob/master/datasetIris.txt .

Les packages R

Les packages utilisés pour réaliser l'application sont disponibles sur le CRAN. Ils s'installent avec la commande : `install.packages()`.
Les packages que nous utiliserons sont :

- **shiny** [1] : Il permettra de construire l'application web
- **shinydashboard** [2]: Il permettra de créer une architecture dynamique à la page web avec une zone de titre, une menu rabattable et une zone principale
- **shinyWidgets** [3] : Il permettra de mettre un message d'alerte pour confirmer la lecture correcte du tableau
- **DT** [4] : Il permettra de créer un tableau dynamique avec de la coloration conditionnelle
- **plotly** [5] , **ggplot2** [6] et **googleVis** [7] : Ils nous permettront de réaliser des graphiques
- **colourpicker** [8] : Il permettra à l'utilisateur de sélectionner une couleur.

Nous utiliserons pour les installer et les charger un autre package : anylib [9]. Ce package est très pratique car il permet d'installer (si besoin) et de charger une liste de package. En plus, il a été créé par un des auteurs de Bioinfo-fr : Aurelien Chateigner. Que demander de plus!

```
install.packages("anyLib")
anyLib::anyLib(c("shiny", "shinydashboard", "shinyWidgets", "DT", "plotly", "ggplot2", "googleVis", "colourpicker"))
```

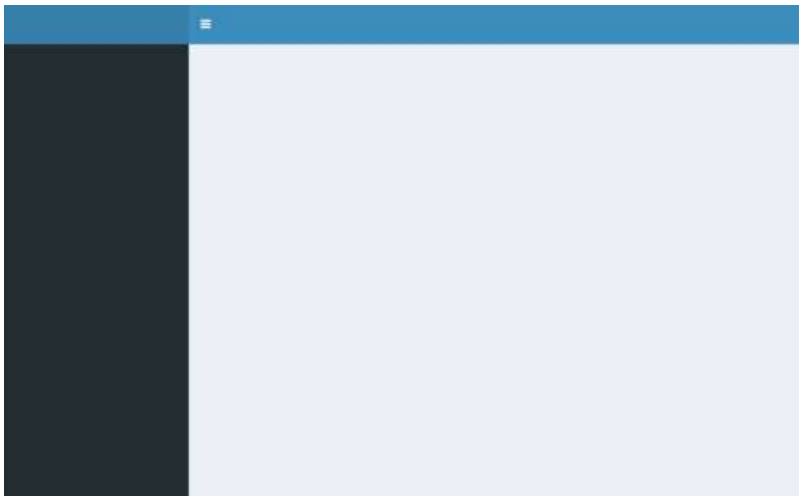
Création de l'architecture

Mise en place d'un dashboard

Pour mettre en forme notre application web (la partie UI visible par l'utilisateur), nous allons utiliser le package shinydashboard. La documentation est présente ici : https://rstudio.github.io/shinydashboard/index.html .

L'architecture minimale avec shinydasbord est zone de titre (bleue), une barre latérale (noir) et une zone principale (grise).

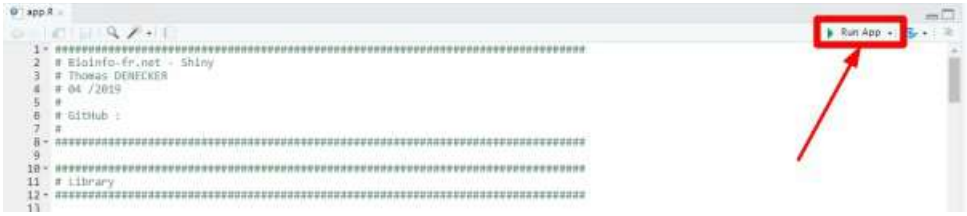
```
library(shiny)
library(shinydashboard)

ui <- dashboardPage(
  dashboardHeader(),
  dashboardSidebar(),
  dashboardBody()
)

server <- function(input, output) { }

shinyApp(ui, server)
```



Visualisation de l'application

Test de l'application

Pour tester l'application, il faut sauvegarder le code puis appuyer sur le bouton `Run App` au dessus à droite de l'éditeur de texte de Rstudio. Un point important, Rstudio reconnaît par défaut les applications qui se nomme app.R. Je vous conseille vivement de nommer votre fichier app.R.



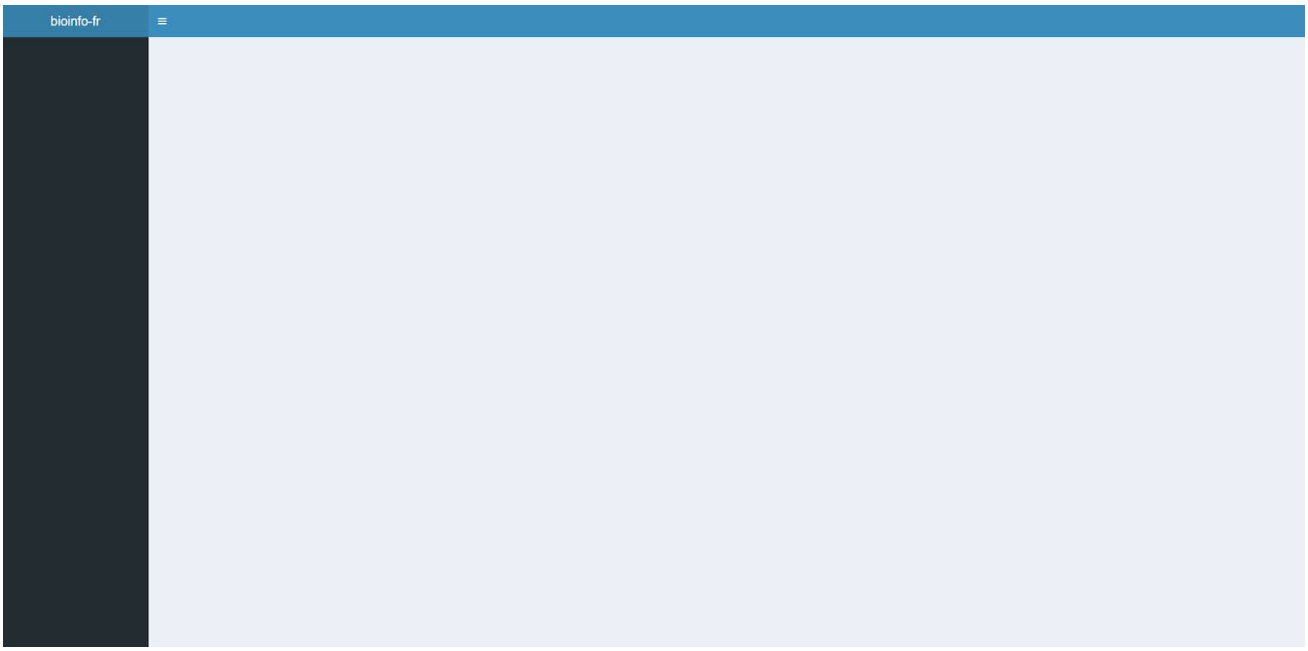
Si vous travaillez avec la console R, vous pouvez lancer la commande suivante :

```
runApp()
```

Ajouter un titre

Dans la fonction dashboardHeader, nous ajoutons un titre à l'application (ici bioinfo-fr). Ce titre sera affiché en haut à gauche.

```
ui <- dashboardPage(
  dashboardHeader(title = "bioinfo-fr"),
  dashboardSidebar( ),
  dashboardBody( )
)
```



Visualisation de l'application

Ajouter des pages

La première étape est d'ajouter des éléments (item) dans la barre de menu latérale (partie noire). Nous utilisons pour cela la fonction `dashboardSidebar`. Nous y ajoutons la fonction `sidebarMenu` qui contient les items du menu.

Ensuite, il faut indiquer que la partie body aura plusieurs pages (des `tabItems`). Chaque `tabItem` correspond à une page accessible par le menu. Le `menuItem` doit avoir le même nom que l'argument `tabName` de la fonction `tabItem` pour y accéder (exemple : readData). Dans chaque page, nous ajoutons un titre de niveau 1 (h1). Vous pouvez remarquer l'utilisation de la fonction `icon` (`icon = icon(...)`). L'argument de la fonction est un nom d'icône que nous pouvons trouver sur ces deux sites : https://fontawesome.com/ et https://getbootstrap.com/docs/4.3/components/alerts/ . En utilisant cette fonction, vous aurez une petite image (icônes) à gauche du nom de l'élément (par exemple un livre pour la lecture des données). Il est aussi possible de l'utiliser pour des boutons .

```
ui <- dashboardPage(
  dashboardHeader(title = "bioinfo-fr"),
  dashboardSidebar(
    sidebarMenu(
      menuItem("Lecture des données", tabName = "readData", icon = icon("readme")),
      menuItem("Visualisation des données", tabName = "visualization", icon = icon("poll"))
    )
  ),
  dashboardBody(
    tabItems(
      # Read data
      tabItem(tabName = "readData",
              h1("Lecture des données")
      ),

      # visualization
      tabItem(tabName = "visualization",
              h1("Visualisation des données")
      )
    )
  )
)
```

Visualisation de l'application

Création d'un lecteur de fichier

L'objectif est de proposer une interface simple pour lire un fichier dans l'application et qui permette à l'utilisateur de paramétrer la lecture et d'avoir une prévisualisation du fichier lu.

Importer un fichier

Pour importer un fichier, shiny propose la fonction `fileInput`. Il est possible de faire du "drag and drop" dans la zone de l'import ou de sélectionner un fichier dans l'explorateur de fichiers. Le type de fichier visible est paramétrable dans les arguments. Ici, nous utiliserons le paramétrage par défaut.

```
ui <- dashboardPage(
  dashboardHeader(title = "bioinfo-fr"),
  dashboardSidebar(
    sidebarMenu(
      menuItem("Lecture des données", tabName = "readData", icon = icon("readme")),
      menuItem("Visualisation des données", tabName = "visualization", icon = icon("poll"))
    )
  ),
  dashboardBody(
    tabItems(
      # Read data
      tabItem(tabName = "readData",
              h1("Lecture des données"),
              fileInput("dataFile",label = NULL,
                        buttonLabel = "Browse...",
                        placeholder = "No file selected")
      ),

      # visualization
      tabItem(tabName = "visualization",
              h1("Visualisation des données")
      )
    )
  )
)
```

Visualisation de l'application

Zone de paramétrage

Nous souhaitons maintenant paramétrer 3 points lors de la lecture du fichier : type de séparateur (virgule, tabulation, espace), type de quote (simple, double, aucune) et la présence/absence des noms de colonnes (header). Nous utilisons pour cela des radio boutons. La fonction utilisée est `radioButtons`. 5 arguments sont utilisés :

- **id** : identifiant du groupe de radio boutons (ici nous avons 3 groupes de radio boutons pour nos 3 paramètres),
- **label** : le titre présent au dessus du groupe de radio boutons,
- **choices** : les choix possibles dans le groupe de radio boutons. A noter, la zone située à gauche du "=" contient les informations qui seront affichées dans l'application alors que la partie droite indique ce que comprend R côté serveur. Pour le header par exemple, il sera affiché "Yes" côté UI et nous récupérerons côté serveur `TRUE` ("Yes" = TRUE) lorsque que nous récupérerons la valeur du radio bouton côté serveur.
- **Selected** : Nom du radio bouton sélectionné au lancement de l'application
- **inline = T** : pour avoir les radio boutons alignés

```
[...]
tabItem(tabName = "readData",
              h1("Lecture des données"),
              fileInput("dataFile",label = NULL,
                        buttonLabel = "Browse...",
                        placeholder = "No file selected"),

              h3("Parameters"),

              # Input: Checkbox if file has header
              radioButtons(id = "header",
                           label = "Header",
                           choices = c("Yes" = TRUE,
                                       "No" = FALSE),
                           selected = TRUE, inline=T),

              # Input: Select separator ----
              radioButtons(id = "sep",
                           label = "Separator",
                           choices = c(Comma = ",",
                                       Semicolon = ";",
                                       Tab = "\t"),
                           selected = "\t", inline=T),

              # Input: Select quotes ----
              radioButtons(id = "quote",
                           label= "Quote",
                           choices = c(None = "",
                                       "Double Quote" = '"',
                                       "Single Quote" = "'"),
                           selected = "", inline=T)

      ),

[...]
```

Visualisation de l'application

Zone de prévisualisation

Dans cette zone, nous allons visualiser les premières lignes du fichier que nous souhaitons lire. Il faut donc :

- Créer une zone d'affichage dans l'UI
- Lire les données côté serveur et envoyer les données dans la zone d'affichage

Côté UI

Pour afficher le tableau, nous utilisons la fonction `dataTableOutput`. Une zone va être créée pour afficher un tableau. Nous donnons à cette zone un identifiant en utilisant l'argument `outputId` . Cet identifiant est indispensable pour retrouver la zone côté serveur.

```
tabItems(
      # Read data
      tabItem(tabName = "readData",
              h1("Lecture des données"),
              fileInput("dataFile",label = NULL,
                        buttonLabel = "Browse...",
                        placeholder = "No file selected"),

              h3("Parameters"),

              # Input: Checkbox if file has header
              radioButtons(inputId = "header",
                           label = "Header",
                           choices = c("Yes" = TRUE,
                                       "No" = FALSE),
                           selected = TRUE, inline=T),

              # Input: Select separator ----
              radioButtons(inputId = "sep",
                           label = "Separator",
                           choices = c(Comma = ",",
                                       Semicolon = ";",
                                       Tab = "t"),
                           selected = "t", inline=T),

              # Input: Select quotes ----
              radioButtons(inputId = "quote",
                           label= "Quote",
                           choices = c(None = "",
                                       "Double Quote" = '"',
                                       "Single Quote" = "'"),
                           selected = "", inline=T),
              h3("File preview"),
              dataTableOutput(outputId = "preview")

      ),
```

Côté Server

Nous souhaitons à présent afficher de l'information. Du côté serveur, pour envoyer de l'information, la syntaxe commence quasiment toujours

par `output$` puis l'ID de la zone de sortie (ici notre tableau de prévisualisation avec l'id "preview"). Ce que nous souhaitons lui envoyer est un tableau. Nous utilisons donc la fonction `<- renderDataTable({ })`. Dans cette dernière fonction, nous allons lire le tableau qui va être renvoyé. Pour récupérer de l'information du côté UI, il faut utiliser la syntaxe suivante : `input$ID`. Par exemple, nous souhaitons récupérer le choix de l'utilisateur concernant le header : `input$header`.

```
output$preview <-  renderDataTable({

    req(input$dataFile)

    df <- read.csv(input$dataFile$datapath,
                   header = as.logical(input$header),
                   sep = input$sep,
                   quote = input$quote,
                   nrows=10
    )
  },  options = list(scrollX = TRUE , dom = 't'))
```

Si nous détaillons le code :

- **req(input$dataFile)** : bloque la suite du code si la zone d'import de fichier est vide
- **df <- read.csv()** : on stocke dans df la lecture du fichier
- **input$dataFile$datapath** : chemin d'accès au fichier importé
- **header = as.logical(input$header)** : récupération de la réponse de l'utilisateur pour savoir si présence ou absence d'un header. Le as.logical permet de convertir un TRUE ou FALSE en booléen.
- **sep = input$sep, quote = input$quote** : récupération du paramétrage de l'utilisateur pour le séparateur et les quotes. Ces informations sont données aux arguments de la fonction read.csv()
- **nrows=10** : Nous ne souhaitons pas lire tout le fichier. Seules les premières lignes sont nécessaires pour savoir si le tableau est lu correctement ou non. Nous lisons donc les 10 premières lignes.
- **options = list(scrollX = TRUE , dom = 't')** : Si le tableau a de nombreuses colonnes, cette option permet d'avoir un scroll horizontal

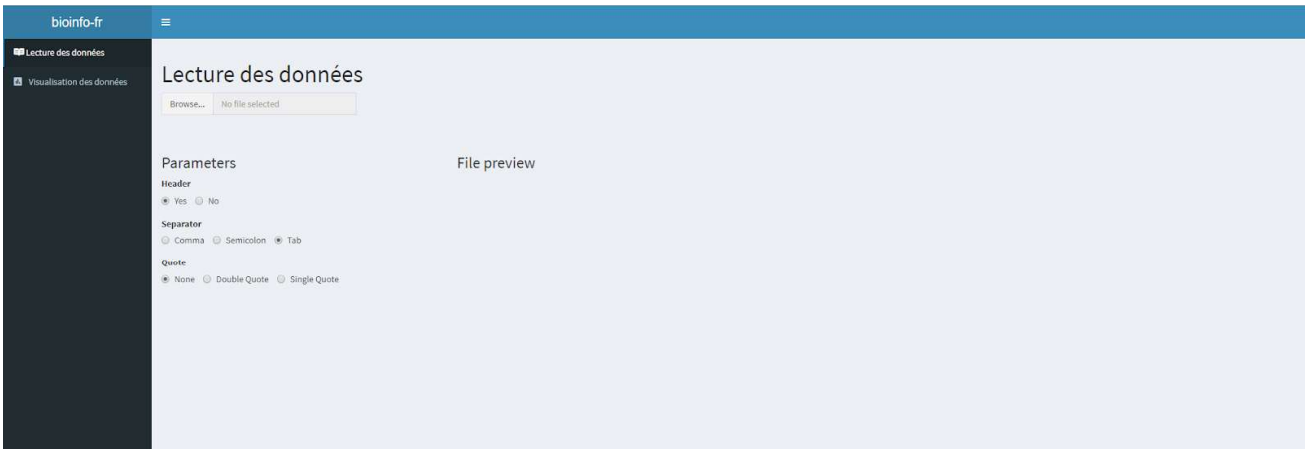
Vous pouvez maintenant tester sur un fichier texte contenant un tableau. Le changement de paramétrage a un effet direct sur la visualisation.

Visualisation de l'application

Organisation des éléments

Pour les connaisseurs de bootstrap, Shiny intègre son code. Pour les autres, il est possible d'organiser le contenu d'une page à l'aide d'une grille. La grille est composée de lignes (`fluidRow()`) elles-mêmes composées de 12 blocs. Nous allons placer les paramètres et la prévisualisation sur une même ligne. Nous souhaitons stocker les paramètres dans 3 blocs (`column(3,...)` : la colonne aura une taille de 3 blocs) et 9 blocs pour la prévisualisation (`column(9,...)`).

```
tabItem(tabName = "readData",
                h1("Lecture des données"),
                fileInput("dataFile",label = NULL,
                          buttonLabel = "Browse...",
                          placeholder = "No file selected"),

                fluidRow(
                  column(3,
                         h3("Parameters"),

                         # Input: Checkbox if file has header
                         radioButtons(inputId = "header",
                                      label = "Header",
                                      choices = c("Yes" = TRUE,
                                                  "No" = FALSE),
                                      selected = TRUE, inline=T),

                         # Input: Select separator ----
                         radioButtons(inputId = "sep",
                                      label = "Separator",
                                      choices = c(Comma = ",",
                                                  Semicolon = ";",
                                                  Tab = "t"),
                                      selected = "t", inline=T),

                         # Input: Select quotes ----
                         radioButtons(inputId = "quote",
                                      label= "Quote",
                                      choices = c(None = "",
                                                  "Double Quote" = '"',
                                                  "Single Quote" = "'"),
                                      selected = "", inline=T)
                  ),
                  column(9,
                         h3("File preview"),
                         dataTableOutput(outputId = "preview")
                  )
                )
```



Visualisation de l'application

Bouton de lecture

Pour finir avec cette page, nous allons créer un bouton pour valider le paramétrage de la lecture du tableau. En cliquant sur ce bouton, l'ensemble du fichier sera lu. Nous ne réalisons pas une lecture dynamique comme précédemment. En effet, à chaque changement de paramètre, l'ensemble du fichier est relu. Si le fichier est gros, le temps de lecture sera long.

Côté UI

Nous ajoutons un `actionButton`. L'identifiant de notre bouton est "actBtnVisualisation".

```
[...]
actionButton(inputId = "actBtnVisualisation", label = "Visualisation",icon = icon("play") )
[...]
```

Pour une question esthétique, nous ajoutons un saut de ligne avant le bouton et nous mettons le bouton dans une division pour pouvoir le centrer :

```
[...]
tags$br(),
div(actionButton(inputId = "actBtnVisualisation", label = "Visualisation",icon = icon("play") ), align = "center")
[...]
```

Côté serveur

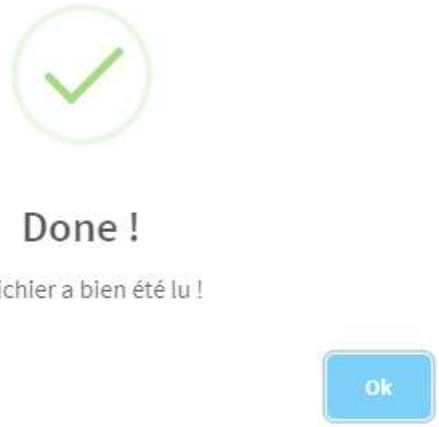
Lorsque que le bouton est cliqué, nous souhaitons à présent que le contenu du fichier soit stocké dans une variable. Il s'agit d'une variable particulière. Elle doit être visible par toutes les fonctions côté serveur et relancer toutes les fonctions qui l'utilisent si elle change. Il s'agit d'une variable réactive (`reactiveValues`). Si nous détaillons le code :

- Nous déclarons une reactiveValue avec comme nom `data`.
- Nous allons utiliser une fonction qui permet d'attendre une action particulière. Ici nous attendons que l'utilisateur clique sur le bouton. Une fois que le bouton a été cliqué, le code entre les { } sera exécuté. Ici, l'objectif sera de stocker le contenu du fichier importé dans la reactiveValue sous le nom `table` (data$table)

```
data = reactiveValues()

  observeEvent(input$actBtnVisualisation, {
    data$table = read.csv(input$dataFile$datapath,
                          header = as.logical(input$header),
                          sep = input$sep,
                          quote = input$quote,
                          nrows=10)
  })
```

Ainsi, à chaque clic du bouton, `data$table` sera mis à jour ainsi que toutes les fonctions qui l'utilise (ex : des graphiques).
Nous pouvons aussi ajouter un message pour confirmer la lecture du fichier. Nous utiliserons `sendSweetAlert` proposé dans le package shinyWidgets. La documentation est disponible ici : https://github.com/dreamRs/shinyWidgets .

```
observeEvent(input$actBtnVisualisation, {
    data$table = read.csv(input$dataFile$datapath,
                          header = as.logical(input$header),
                          sep = input$sep,
                          quote = input$quote,
                          nrows=10)
    sendSweetAlert(
      session = session,
      title = "Done !",
      text = "Le fichier a bien été lu !",
      type = "success"
    )
  })
```



Visualisation du message

Changement de page

Enfin, notre application étant composée de 2 pages, nous souhaitons changer de page une fois que le fichier est lu pour arriver sur la page de visualisation.

```
updateTabItems(session, "tabs", selected = "visualization")
```

Pour rappel, "tabs" est l'identifiant de notre sidebarMenu. Nous allons avec cette commande chercher dans la sidebarMenu la page qui a comme identifiant "visualization" et changer de page.

Visualisation

Exploration du tableau

Nous allons à présent afficher le tableau complet. Nous utilisons pour cela le package DT (https://rstudio.github.io/DT/). Il permet de rechercher, sélectionner ou trier les informations d'un tableau de données. Il faut pour cela créer une zone où sera affiché le tableau dans

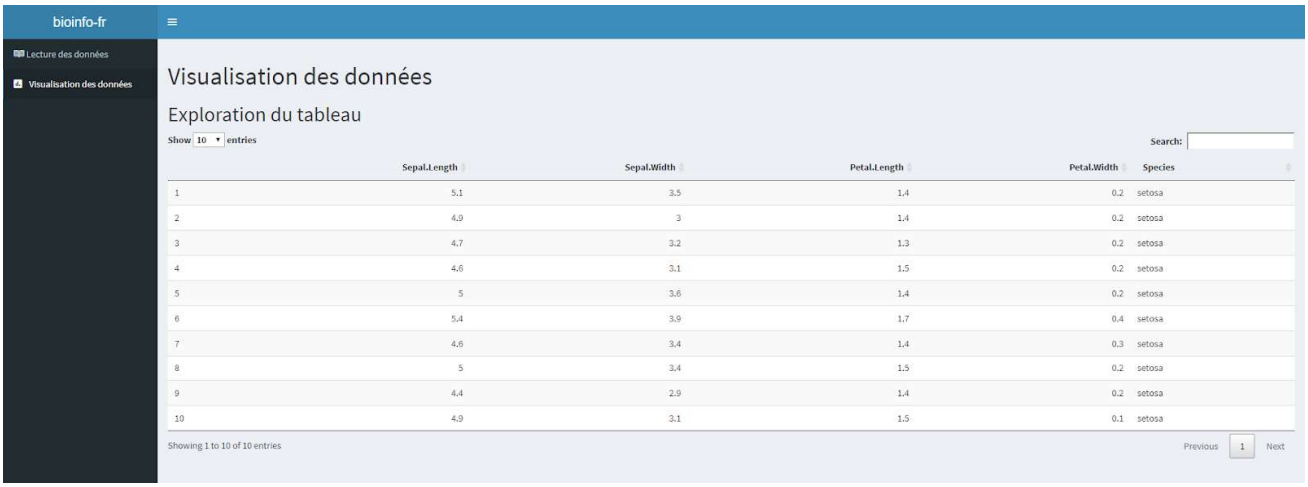
l'UI.

Côté UI

```
tabItem(tabName = "visualization",
              h1("Visualisation des données"),
              h2("Exploration du tableau"),
              dataTableOutput('dataTable')
      )
```

Puis du côté serveur, il ne reste plus qu'à envoyer le contenu de notre fichier dans ce tableau par le biais de la `reactiveValue`. Ainsi, le tableau sera automatiquement mis à jour si un nouveau fichier est lu.

```
output$dataTable = DT::renderDataTable(data$table)
```



Visualisation de l'application

Il est possible de faire de la mise en forme conditionnelle comme dans excel. Le code proposé par la suite est dépendant du tableau utilisé. En effet, nous allons cibler les colonnes d'intérêt par leur nom pour une question de lisibilité.

Voici une proposition de mise en forme conditionnelle de notre tableau (inspiré de l'exemple proposé dans la documentation du package DT).

- Histogramme des valeurs pour les colonnes Sepal.length et Petal.length
- Coloration par seuils multiples pour les colonnes Sepal.width et Petal.width (fond blanc écriture noire, fond rouge écriture blanche et fond rouge foncé écriture blanche)
- Coloration du fond en fonction de l'espèce pour la colonne espèce.

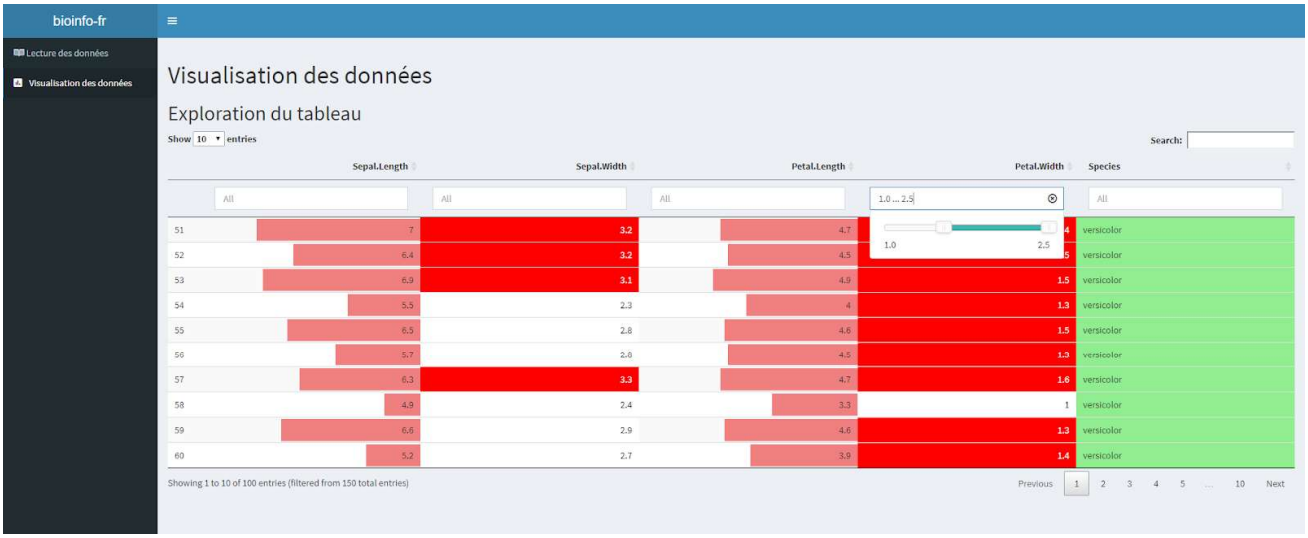
```
output$dataTable = DT::renderDataTable({
    datatable(data$table, filter = 'top') %>%
      formatStyle('Sepal.Length',
                  background = styleColorBar(data$table$Sepal.Length, 'lightcoral'),
                  backgroundSize = '100% 90%',
                  backgroundRepeat = 'no-repeat',
                  backgroundPosition = 'center'
      ) %>%
      formatStyle(
        'Sepal.Width',
        backgroundColor = styleInterval(c(3,4), c('white', 'red', "firebrick")),
        color = styleInterval(c(3,4), c('black', 'white', "white"))
      ) %>%
      formatStyle(
        'Petal.Length',
        background = styleColorBar(data$table$Petal.Length, 'lightcoral'),
        backgroundSize = '100% 90%',
        backgroundRepeat = 'no-repeat',
        backgroundPosition = 'center'
      ) %>%
      formatStyle(
        'Petal.Width',
        backgroundColor = styleInterval(c(1,2), c('white', 'red', "firebrick")),
        color = styleInterval(c(1,2), c('black', 'white', "white"))
      ) %>%
      formatStyle(
        'Species',
        backgroundColor = styleEqual(
          unique(data$table$Species), c('lightblue', 'lightgreen', 'lavender')
        )
      )
  })
```

Visualisation de l'application

Enfin, pour améliorer l'exploration, il est possible d'ajouter des filtres par colonnes. Pour les valeurs numériques, les données sont filtrées par un slider. Pour les colonnes contenant du texte, il y a deux possibilités :

- Peu de variabilité entre les éléments. Par exemple, la colonne Species ne contient que 3 éléments différents : setosa, versicolor et virginica. Dans ce cas, le filtre sera composé des éléments uniques de cette colonne qui seront cliquables. En les cliquant, toutes les lignes avec cet élément seront sélectionnées.
- Grande variabilité entre les éléments. Dans ce cas, une zone pour entrer du texte sera proposée. Le texte saisi sera recherché dans la colonne.

```
output$dataTable = DT::renderDataTable({
    datatable(data$table, filter = 'top') %>%
      [...]
```



Visualisation de l'application

Visualisation graphique

Nous allons créer de 4 façons différentes des graphiques et les afficher dans l'application shiny :

- des graphiques statiques
  - un plot de base avec R
  - un graphique avec ggplot2
- des graphiques dynamiques
  - Avec plotly
  - Avec google

Les graphiques seront représentés sur la même ligne avec une fluidRow et 4 colonnes (comme nous avons fait précédemment).

Graphique R

R propose une grande palette de graphiques de base. Cependant, il s'agit uniquement de graphiques statiques.

Côté UI

Il faut comme précédemment créer une zone pour indiquer où va être affiché le graphique. La fonction utilisée est plotOutput.

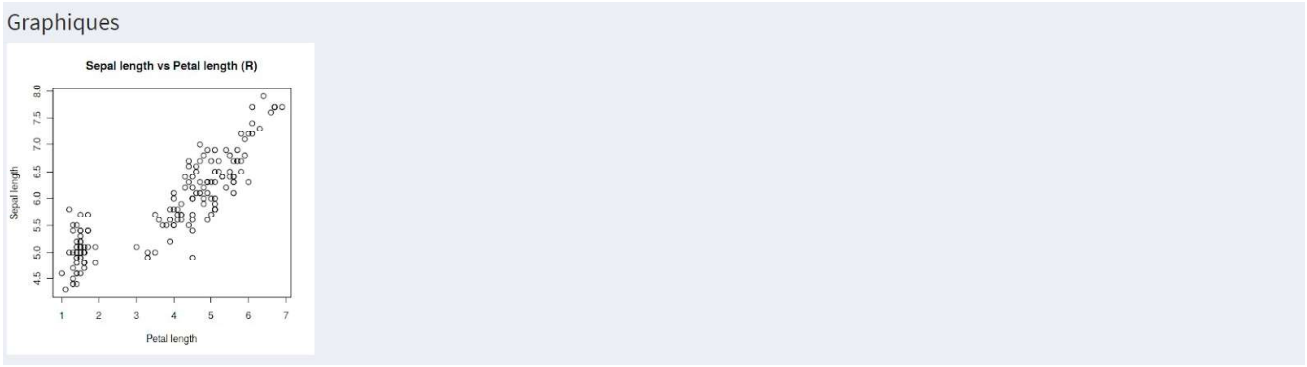
```
tabItem(tabName = "visualization",
              h1("Visualisation des données"),
              h2("Exploration du tableau"),
              dataTableOutput('dataTable'),
              h2("Graphiques"),
             fluidRow(
                column(3,plotOutput("plotAvecR") )
              )
      )
```

Côté serveur

Nous allons pour ce graphique comparer la corrélation entre la taille des sépales et des pétales. Comme pour le tableau, la syntaxe est la suivante pour envoyer de l'information du côté UI : `output$ID_de_la_zone`. Pour envoyer un plot, nous utilisons la fonction renderPlot. Dans cette fonction, vous pouvez mettre n'importe quel graphique de R. Afin de mettre à jour automatiquement les graphiques, nous utilisons notre reactiveValue : `data`. Chaque fois que `data` changera, le plot sera généré de nouveau. Pour accéder au contenu du fichier lu qui est stocké dans la reactiveValue `data` sous le nom de `table`, nous utilisons de nouveau la syntaxe suivante : `data$table`. Il s'agit d'un dataframe (la lecture par read.csv2 renvoie un dataframe). Les colonnes sont donc accessibles par un $ puis le nom. Au final, pour obtenir le vecteur contenant les valeurs de longueur des pétales, nous utiliserons la syntaxe suivante : `data$table$Petal.Length`.

```
output$plotAvecR <- renderPlot({
    plot(data$table$Petal.Length,data$table$Sepal.Length,
         main = "Sepal length vs Petal length (R)",
         ylab = "Sepal length",
         xlab = "Petal length")
  })
```

Le paramétrage du plot est libre et n'est pas contraint par shiny.



Visualisation de l'application

Graphique par ggplot2

Ggplot2 est une librairie graphique de plus en plus utilisée. Elle propose des graphiques plus évolués que ceux de base dans R. Vous trouverez une documentation très bien faite ici : https://ggplot2.tidyverse.org/. Nous allons comparer les largeurs et les longueurs des sépales. Une coloration en fonction de l'espèce est proposée.

Côté UI

Comme toujours, nous allons créer une zone où sera affiché le graphique. La fonction utilisée est encore `plotOutput`.

```
fluidRow(
                column(3,plotOutput("plotAvecR")),
                column(3, plotOutput("plotAvecGgplot2"))
              )
```

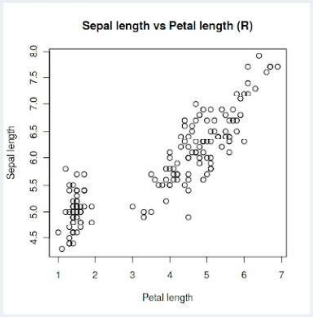
Côté serveur

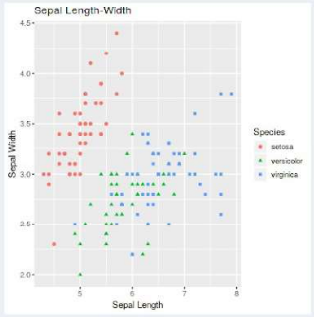
Nous allons procéder de la même façon que précédemment. La différence est liée au contenu de la fonction `renderPlot`. Nous allons cette fois-ci utiliser les fonctions de ggplot2.

```

```

```
output$plotAvecGgplot2 <- renderPlot({
    ggplot(data=data$table, aes(x = Sepal.Length, y = Sepal.Width)) +
      geom_point(aes(color=Species, shape=Species)) +
      xlab("Sepal Length") +  ylab("Sepal Width") +
      ggtitle("Sepal Length-Width (ggplot2")
  })
```

Graphiques





Visualisation de l'application

Graphique Plotly

Plotly est un package de j'affectionne particulièrement. Il propose énormément d'outils préprogrammés (enregistrement de l'image, zoom, informations supplémentaires). De plus, il n'est pas exclusivement réservé à R. Il est possible de l'utiliser aussi dans des projets en JS et en python (aussi simple d'utilisation).

Côté UI

De nouveau, nous allons créer une zone pour afficher le graphique. Attention, nous changeons de fonction. Nous utiliserons cette fois plotlyOutput.

```
fluidRow(
                column(3, plotOutput("plotAvecR")),
                column(3, plotOutput("plotAvecGgplot2")),
                column(3, plotlyOutput("plotAvecPlotly"))
              )
```

Côté serveur

Je n'expliquerai pas ici la syntaxe pour réaliser un graphique avec Plotly. La documentation sur le site est extrêmement bien faite avec de très nombreux exemples (https://plot.ly/r/). Vous pouvez mettre n'importe quel graphique plotly dans la fonction. Ici, nous comparons la largeur et la longueur des pétales.

```
plot_ly(data = data$table, x = ~ Petal.Length, y = ~ Petal.Width, color = ~ Species) %>%
        layout(title = 'Petal Length-Width (plotly)',
               yaxis = list(title = "Petal width"),
               xaxis = list(title = "Petal length"))
```

Je vous invite lorsque vous lancerez l'application à survoler ce graphique. Il y a énormément d'informations disponibles et d'outils d'exploration.

Graphique Google

Pour finir, les graphiques de Google sont de plus en plus populaires et offrent un plus large choix de représentations que Plotly (calendrier, etc.). Ici, nous allons réaliser un histogramme de la largeur des pétales.

Côté UI

Nous créons de nouveau une zone pour afficher le graphique. La fonction utilisée est htmlOutput. Cette fonction est capable d'interpréter du code HTML venant du serveur. Si vous souhaitez écrire du HTML directement dans la partie UI, il vous suffit d'utiliser la fonction HTML (ex : HTML("<h1>Titre 1</h1>")) .

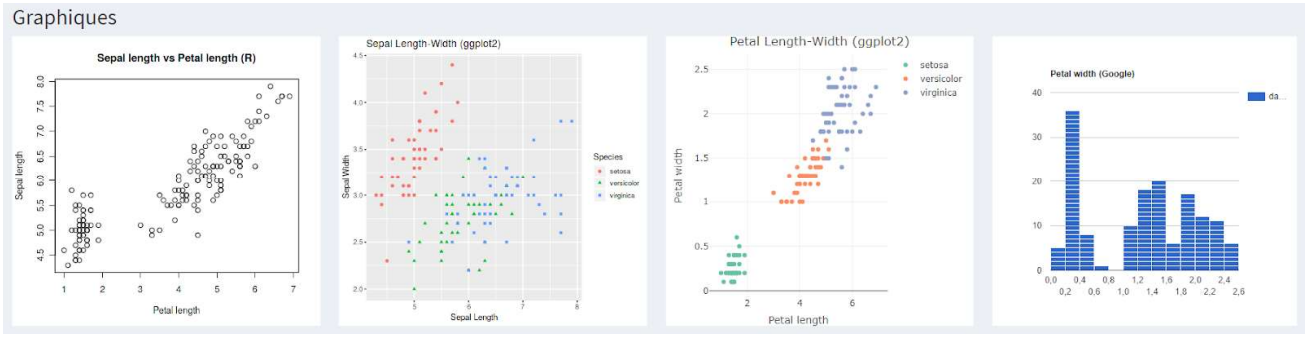
```
fluidRow(
                column(3, plotOutput("plotAvecR")),
                column(3, plotOutput("plotAvecGgplot2")),
                column(3, plotlyOutput("plotAvecPlotly")),
                column(3, htmlOutput("plotAvecGoogle"))
              )
```

Côté serveur

Pour les graphiques Google, nous utilisons les fonctions graphiques commençant par gvis et le rendu est fait avec la fonction `renderGvis`. Elles sont détaillées à la page suivante https://cran.r-project.org/web/packages/googleVis/vignettes/googleVis_examples.html .

```
output$plotAvecGoogle <- renderGvis({
      gvisHistogram(as.data.frame(data$table$Petal.Width),
                    options=list(title ="Petal width (Google)",
                                 height=400)
      )
  })
```

Visualisation de l'application



Visualisation du l'application

Gérer le tableau vide

En lançant l'application, si vous vous rendez sur la partie visualisation, vous trouverez plein d'erreurs. Ces erreurs sont la cause de l'utilisation d'une `reactiveValue`. En effet, lorsque rien n'a encore été lu, `data$table` est NULL (vide). Or toutes les fonctions que nous utilisons ne gèrent pas les NULL. Nous ajouterons pour le tableau et les graphiques un peu de code pour lui dire de renvoyer NULL si le tableau est vide.

```
if (!is.null(data$table)) {
    [représentation graphique ou le tableau]
} else {
    NULL
}
```

Interagir avec les graphique

Nous allons voir deux types d'interactions avec les graphiques pour illustrer la simplicité pour l'utilisateur d'interagir avec les données et les représentations :

- Sélectionner les données à afficher à l'aide du tableau
- Changer des paramètres graphiques sur le plot de base proposé par R (le premier graphique). Tous ces changements peuvent bien sûr être appliqués sur tous les graphiques.

Sélectionner les données à afficher à l'aide du tableau

Grâce à Shiny, il est possible de faire communiquer le tableau avec les graphiques. Nous profitons pour cela de la puissance du package DT qui génère le tableau. Les modifications que nous allons réaliser seront uniquement côté serveur. L'objectif est de récupérer les lignes qui sont affichées dans le tableau et de n'utiliser que ces lignes dans les graphiques. Comme précédemment, pour récupérer de l'information dans l'UI, il faut utiliser `input$ID_ZONE`. Nous souhaitons récupérer de l'information de notre tableau qui a comme identifiant `dataTable`. Ensuite, nous ajoutons `_rows_all` à la fin de l'ID pour obtenir les lignes. Ainsi, avec `input$dataTable_rows_all`, nous avons les lignes affichées dans le tableau. Il ne reste plus qu'à les sélectionner dans le vecteur de données. Le graphique est à présent dynamique.

```
output$plotAvecR <- renderPlot({
    if (!is.null(data$table)) {
      plot(data$table$Petal.Length[input$dataTable_rows_all],
           data$table$Sepal.Length[input$dataTable_rows_all],
           main = "Sepal length vs Petal length (R)",
           ylab = "Sepal length",
           xlab = "Petal length")
    } else {
      NULL
    }
  })
```

La même démarche est ensuite appliquée aux autres graphiques. Grâce aux filtres du tableau, nous avons ainsi la possibilité de sélectionner par les données numériques (longueur et largeur) et par l'espèce.

Changement de couleur pour le graphique de base R

L'objectif est de vous montrer une autre façon d'interagir avec les graphiques. En effet, il se peut que vous n'utilisiez pas de tableau dans votre application. De très nombreux exemples sont disponibles en ligne (ici par exemple : https://shiny.rstudio.com/gallery/). Nous allons

implémenter 4 changements sur ce graphique pour vous donner des exemples d'utilisation d'inputs :

- Changement de la couleur des points (avec l'utilisation d'un colour picker capable de gérer la transparence)
- Changement du type de point
- Changement de la taille des points
- Changement du titre

Côté UI

Pour plus de lisibilité lors de l'utilisation, nous avons changé la disposition des graphiques pour avoir sur une ligne le graphique R avec ses paramètres et sur une seconde les trois autres graphiques. Vous pouvez ainsi voir la simplicité de la réorganisation d'une page à l'aide du système de Grid.

```
tabItem(tabName = "visualization",
                h1("Visualisation des données"),
                h2("Exploration du tableau"),
                dataTableOutput('dataTable'),
                h2("Graphiques"),
                fluidRow(
                  column(4, plotOutput("plotAvecR")),
                  column(4, colourpicker::colourInput("colR", "Couleur graphique R", "black",allowTransparent = T),
                         sliderInput("cex", "Taille",
                                     min = 0.5, max = 3,
                                     value = 1,step = 0.2
                                    )),
                  column(4, selectInput(inputId = "pch", choices = 1:20, label = "Type de points",selected = 1),
                         textInput("title", "Titre", "Sepal length vs Petal length (R)") )
                ),
                tags$br(),
                fluidRow(
                  column(4, plotOutput("plotAvecGgplot2")),
                  column(4, plotlyOutput("plotAvecPlotly")),
                  column(4, htmlOutput("plotAvecGoogle"))
                )
        )
```

Pour faire entrer de l'information, nous avons besoin de 4 fonctions `input` : `colourInput` pour la couleur (du package colourpcicker), `sliderInput` pour la taille des points, `selectInput` pour le type de points et `textInput` pour le titre du graphique.

Côté serveur

Nous allons récupérer les entrées et les intégrer dans notre plot.

```
      plot(data$table$Petal.Length[input$dataTable_rows_all],
           data$table$Sepal.Length[input$dataTable_rows_all],
           main = input$title,
           ylab = "Sepal length",
           xlab = "Petal length",
           pch = as.numeric(input$pch),
           col = input$colR,
           cex = input$cex)
```

Visualisation dans l'application

Visualisation de l'application

Conclusion

Et voilà ! Vous avez réalisé une application complète capable de lire un fichier en fonction de paramètres et d'explorer ses données. Vous trouverez l'ensemble du code sur github ici : https://github.com/bioinfo-fr/bioinfo-fr_Shiny . A travers ce post, nous avons vu comment rendre interactive l'exploration d'un tableau de données à l'aide de Shiny. Vos utilisateurs n'auront plus à voir votre code. Ils auront simplement à appuyer sur Run App. Il existe de nombreuses solutions de partage (https://shiny.rstudio.com/tutorial/written-tutorial/lesson7/). De nombreuses autres possibilités sont disponibles et pourront être détaillées dans d'autres articles (concaténérisation et intégration continue d'une application Shiny, par exemple).

Merci à mes relecteurs Aurélien C. et Ismaël P. pour leur aide !

Versions des outils utilisés

```
R version 3.5.1 (2018-07-02)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Debian GNU/Linux 9 (stretch)

Matrix products: default
BLAS: /usr/lib/openblas-base/libblas.so.3
LAPACK: /usr/lib/libopenblasp-r0.2.19.so

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C               LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8     LC
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C                  LC_ADDRESS=C               LC_TELEPHONE=C             LC

attached base packages:
[1] parallel  stats4    stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] bindrcpp_0.2.2              shinycssloaders_0.2.0       shinyjs_1.0                 colourpicker_1.0
 [7] plotly_4.8.0                ggplot2_3.1.0               FactoMineR_1.41             DT_0.5
[13] DelayedArray_0.8.0          BiocParallel_1.16.5         matrixStats_0.54.0          Biobase_2.42.0
[19] IRanges_2.16.0              S4Vectors_0.20.1            BiocGenerics_0.28.0         shinydashboard_0.7.1

loaded via a namespace (and not attached):
 [1] bitops_1.0-6           bit64_0.9-7            RColorBrewer_1.1-2     httr_1.4.0             tools_3.5.1
 [8] rpart_4.1-13           Hmisc_4.1-1            DBI_1.0.0              lazyeval_0.2.1         colorspace_1.3-2
[15] tidyselect_0.2.5       gridExtra_2.3          bit_1.1-14             compiler_3.5.1         htmlTable_1.13.1
[22] checkmate_1.9.0        genefilter_1.64.0      stringr_1.3.1          digest_0.6.18          foreign_0.8-70
[29] pkgconfig_2.0.2        htmltools_0.3.6        htmlwidgets_1.3        rlang_0.3.0.1          rstudioapi_0.8
[36] jsonlite_1.6           acepack_1.4.1          dplyr_0.7.8            RCurl_1.95-4.11        magrittr_1.5
[43] leaps_3.0              Matrix_1.2-14          Rcpp_1.0.0             munsell_0.5.0          yaml_2.2.0
[50] MASS_7.3-50            zlibbioc_1.28.0        plyr_1.8.4             grid_3.5.1             blob_1.1.1
[57] miniUI_0.1.1.1         lattice_0.20-35        splines_3.5.1          annotate_1.60.0        locfit_1.5-9.1
[64] geneplotter_1.60.0     XML_3.98-1.16          glue_1.3.0             latticeExtra_0.6-28    data.table_1.11.8
[71] purrr_0.2.5            tidyr_0.8.2            assertthat_0.2.0       xfun_0.4               mime_0.6
[78] viridisLite_0.3.0      survival_2.42-3        tibble_1.4.2           AnnotationDbi_1.44.0   memoise_1.1.0
```

Bibliographie

[1] Winston Chang, Joe Cheng, JJ Allaire, Yihui Xie and Jonathan McPherson (2018). shiny: Web Application Framework for R. R package version 1.2.0.https://CRAN.R-project.org/package=shiny

[2] Winston Chang and Barbara Borges Ribeiro (2018). shinydashboard: Create Dashboards with 'Shiny'. R package version 0.7.1. https://CRAN.R-project.org/package=shinydashboard

[3] Victor Perrier, Fanny Meyer and David Granjon (2018). shinyWidgets: Custom Inputs Widgets for Shiny. R package version 0.4.4. https://CRAN.R-project.org/package=shinyWidgets

[4] Yihui Xie, Joe Cheng and Xianying Tan (2018). DT: A Wrapper of the JavaScript Library 'DataTables'. R package version 0.5. https://CRAN.R-project.org/package=DT

[5] Carson Sievert (2018) plotly for R. https://plotly-book.cpsievert.me

[6] H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

[7] Markus Gesmann and Diego de Castillo. Using the Google Visualisation API with R. The R Journal, 3(2):40-44, December 2011.

[8] Dean Attali (2017). colourpicker: A Colour Picker Tool for Shiny and for Selecting Colours in Plots. R package version 1.0. https://CRAN.R-project.org/package=colourpicker

[9] Aurelien Chateigner (2018). anyLib: Install and Load Any Package from CRAN, Bioconductor or Github. R package version 1.0.5. https://CRAN.R-project.org/package=anyLib

IV. L'application Web « MONet » pour l'intégration et la visualisation de réseaux multi-omiques

1. Le contexte du projet

Le développement du logiciel MONet est très récent. Il a été initié en mars 2020, à la suite d'un échange avec S. Terrier qui travaille sur la plateforme protéomique de l'Institut Jacques Monod[108]. Il souhaitait représenter un réseau d'interactions entre des protéines d'intérêt et ajouter des informations complémentaires, extraites de bases de données publiques. Dans le cadre du projet « Fer *C. glabrata* » (voir page 149), j'étais familiarisé avec différentes solutions techniques de représentation de réseaux et j'en avais éprouvé certains avantages et inconvénients.

2. Le logiciel MONet

MONet signifie Multi-Omics Networks. L'objectif du logiciel est de permettre la représentation de réseaux, à partir de données multi-omiques. Une présentation détaillée est rédigée ci-dessous. Comme bPeaks App et Pixel, le code source de MONet est disponible sur Github[109], un site internet de présentation est en cours de préparation, ainsi qu'une installation sur le serveur d'analyse de notre laboratoire.

3. La présentation détaillée (travail non publié)

a. Pourquoi un nouveau logiciel ?

Il existe de nombreux logiciels permettant d'intégrer des données multi-omiques et de les représenter en réseaux, par exemple Cytoscape (Shannon et al. 2003), VisANT (Hu et al. 2013), 3Omics (Kuo et al. 2013) ou Gephi (Bastian et al. 2009). Au laboratoire, nous avons l'habitude d'utiliser Cytoscape car il présente plusieurs avantages comme la possibilité d'automatiser l'analyse (Otasek et al. 2019) et la possibilité d'ajouter de nombreux *plugins*, tels que Bingo[110]

[108] https://www.ijm.fr/30/spectrometrie-de-masse.htm [Accessible le 01/06/2020]
[109] https://github.com/thomasdenecker/MONET [Accessible le 01/06/2020]
[110] http://apps.cytoscape.org/apps/bingo [Accessible le 05/05/2020]

(enrichissement fonctionnel) ou stringApp[111] (import et extension des réseaux à l'aide de données provenant de la base de données STRING).

Bien que Cytoscape soit très performant et simple d'utilisation, il présente certaines limites. Pour commencer, il est nécessaire d'avoir des compétences en programmation pour automatiser l'utilisation de Cytoscape. Si l'automatisation n'est pas mise en place, toutes les étapes nécessaires à la reproduction d'un même réseau sont très répétitives, notamment les étapes de personnalisations graphiques (formes, couleurs, positions des sommets). De plus, il est nécessaire de préparer les données pour pouvoir les importer dans le logiciel et générer le réseau. Une fois les réseaux générés, nous avons identifié plusieurs problématiques lors de leur exploration systématique :

1. Multiplicité des bases de données à parcourir pour obtenir diverses informations en relation avec les fonctions des gènes (ou des protéines) représentées dans le réseau ;
2. Annotations fonctionnelles limitées aux termes GO, alors que d'autres données sont disponibles dans la base de données STRING ;
3. Personnalisation du réseau (couleur, taille et positionnement des sommets, couleur des arêtes, etc.) qui n'est pas dynamique, ce qui impose de regénérer le réseau très fréquemment.

Toutes ces problématiques étaient partagées par nos collègues travaillant dans le domaine de la protéomique. Ils souhaitaient un outil automatique capable de récupérer toutes les connexions connues à partir d'une liste de protéines (STRING), d'avoir un rapport exportable contenant les informations principales pour chaque élément du réseau et un moyen de mettre en évidence des ensembles (*clusters*) d'éléments (enrichissement fonctionnel, annotations manuelles, etc). Le tout sans nécessiter de compétences techniques en informatique.

Dans ce contexte, nous avons décidé de créer l'outil MONet. Comme bPeaks App et Pixel, il s'agit d'une application WEB. MONet permet de visualiser des réseaux de gènes ou de protéines et de récupérer automatiquement des informations supplémentaires concernant les éléments qui composent ces réseaux. MONet s'utilise via un navigateur WEB (Chrome, Firefox, etc.) et permet d'effectuer toutes les étapes pour créer un réseau en seulement quelques

[111] http://apps.cytoscape.org/apps/stringapp [Accessible le 05/05/2020]

clics de souris. Chaque gène présent dans le réseau est accompagné d'un rapport de synthèse téléchargeable, constitué des informations principales présentes dans les bases de données de références (STRING et UniProt par exemple). L'ensemble des données collectées et structurées (enrichissement fonctionnel, lien entre les éléments du réseau, etc.) est regroupé dans un fichier Excel.

b. Informations techniques

Reproductibilité

Afin de garantir la reproductibilité de l'exploration des réseaux, nous avons décidé d'utiliser les technologies de développement *Open Source* les plus courantes : Github[112] pour le versionnage du code et Docker[113] pour garantir le même environnement de travail (même système d'exploitation, même version des outils, etc.). L'ensemble des *packages* et outils nécessaires au bon fonctionnement de l'application est installé dans cet environnement qui est téléchargeable sur une machine personnelle (travail local) ou sur un serveur *via* Docker Hub[114] (accès à distance).

Application dynamique

Pour fournir une interface dynamique, la partie *front-end*[115] a été développée avec le package R Shiny (Chang et al. 2019). Pour enrichir l'expérience utilisateur, des *packages*[116] complémentaires à Shiny ont été utilisés :

- `shinyjs` (Javascript) pour dynamiser l'application (Attali 2020);
- `shinyhelper` pour envoyer des messages d'aide (Attali et al. 2018) ;
- `shinydashboard` et `shinydashboardPlus` pour créer un *dashboard* (Chang et al. 2018; Granjon 2019),
- `shinyWidgets` pour avoir un plus grand choix de widgets (Perrier et al. 2020),

---

[112] https://github.com/ [Accessible le 04/05/2020]

[113] https://www.docker.com/ [Accessible le 04/05/2020]

[114] https://hub.docker.com/repository/docker/tdenecker/monet [Accessible le 04/05/2020]

[115] Eléments du site que nous voyons à l'écran lors de la navigation sur le site ou l'application et avec lesquels nous pouvons interagir (boutons, lien, …). Plus simplement, il s'agit de l'interface graphique.

[116] La différence entre un package et une librairie est expliquée dans la ressource numérique : https://thomasdenecker.github.io/thesisWebsite/annexes/packageLibrary/ [Accessible le 10/08/2020].

- `shinycssloaders` pour avoir une barre de progression lors des calculs et des imports de données (Sali et al. 2020)
- `colourpicker` un sélecteur de couleur pour la personnalisation du réseau (Attali 2017).

Sorties graphiques dynamiques

Pour obtenir des sorties graphiques dynamiques, les *packages* suivants ont été utilisés :

- `plotly` pour obtenir des histogrammes dynamiques (Sievert 2018).
- `visNetwork` pour générer le réseau de gènes / protéines (Almende B.V. et al. 2019) et `igraph` pour calculer le positionnement idéals des nœuds (Csardi et al. 2006) ;
- le visualiseur `ngl` pour afficher les structures 3D des protéines (Rose et al. 2018; 2015).
- `DT` pour organiser les données sous forme de tableaux dynamiques (tris, filtres et exports) (Xie et al. 2019).

Récupération des données via des APIs

Une *Application Programming Interface* ou API est un moyen permettant à des applications de communiquer entre elles et d'échanger mutuellement des services ou des données. Dans notre cas, les API utilisées permettent de récupérer des données directement dans les bases de données associées. Nous avons utilisé des APIs pour les bases de données STRING[117] et UniProt[118]. Un point important est à noter concernant l'API de STRING : il n'est pas possible de récupérer un réseau de plus de 500 nœuds. La solution proposée par l'équipe de STRING est d'importer la base de données complète et de l'interroger directement. Comme l'objectif premier de MONet est d'explorer des réseaux, cette limitation ne nous a pas semblé être un problème. En effet, explorer un réseau de plus de 500 nœuds est extrêmement complexe. De plus, l'import de la base de données aurait présenté 3 problèmes majeurs :

- Un espace de stockage important – Le fichier zippé contenant la base de données STRING occupe 70 Go ;

[117] http://version11.string-db.org/help/api/ [Accessible le 04/05/2020]
[118] https://www.uniprot.org/help/api [Accessible le 04/05/2020]

- Une nécessité de mise à jour – Si la base de données est stockée localement, les données ne sont plus constamment à jour comme avec l'API ;
- Une écriture des requêtes – Lorsque les API sont utilisées, les requêtes à la base de données sont automatiquement générées et optimisées par les créateurs de la base.

Des requêtes ont également été effectuées sur des bases de données externes à l'aide du *package* `httr` (Wickham 2019). Pour lire les données reçues, le *package* `xml2` a été utilisé pour celles au format XML (Wickham et al. 2018), le *package* `jsonlite` pour celles au format JSON (Ooms 2014) et `sequinr` pour celles au format fasta (Charif et al. 2007). Enfin, l'exploration et l'organisation des données ont été réalisées avec les *packages* `dplyr` et `reshape2` (Wickham et al. 2020; Wickham 2007).

Procédure d'installation

MONet peut être installé sur une machine locale (Linux, MacOS X et Windows 10) ainsi que sur un serveur. Lors d'un déploiement sur un serveur, l'application peut être utilisée simultanément par plusieurs utilisateurs. Toutes les informations nécessaires pour déployer l'application sont disponibles sur GitHub[119]. Les exigences minimales sont un OS 64 bits, Docker installé dans la version *community edition* (> v18) et un accès Internet. Pour rendre l'installation de MONet la plus simple possible, des scripts d'installation pour chaque OS sont disponibles. Finalement, aucune ligne de code n'est requise par l'utilisateur pour lancer tous les composants nécessaires au bon fonctionnement de l'application.

c. Exemples d'utilisation

Avec MONet, nos objectifs d'analyses étaient multiples. Nous souhaitions par exemple un outil capable de détecter des connexions entre les gènes / protéines à partir de données d'expression (transcriptomique) et de représenter ces connexions sous la forme d'un réseau. Ensuite, nous souhaitions savoir si les connexions découvertes étaient décrites dans la littérature. Enfin, nous souhaitions regrouper les principales informations disponibles pour un gène / une protéine dans une même interface pour faciliter l'exploration du graphe et éviter de parcourir de multiples bases de données. Afin de garantir la reproductibilité de l'exploration, l'ensemble des résultats

[119] https://github.com/thomasdenecker/MONET/blob/master/README.md [Accessible le 04/05/2020]

est téléchargeable et un rapport d'exploration est disponible pour tous les gènes / protéines représentés dans le réseau (accompagné de la version de tous les outils utilisés).

Génération de réseaux à partir d'une liste de gènes ou de protéines

Dans ce premier exemple, l'utilisateur renseigne une liste de gènes ou de protéines, que nous appellerons éléments pour simplifier, et MONet crée un réseau automatiquement (Figure 30). La liste d'éléments peut être copiée dans une zone de texte ou importée *via* un fichier texte. À partir de cette liste, MONet recherche des liens entre ces éléments et des éléments non présents dans la liste initiale sont ajoutés, grâce aux informations de la base de données STRING. L'utilisateur peut paramétrer le nombre de nouveaux éléments qui peuvent être ajoutés (par défaut, 5 nouveaux éléments par élément de la liste initiale). Pour les distinguer, les éléments de la liste initiale sont représentés sous forme de carré dans le réseau et sous forme de rond pour les nouveaux éléments.

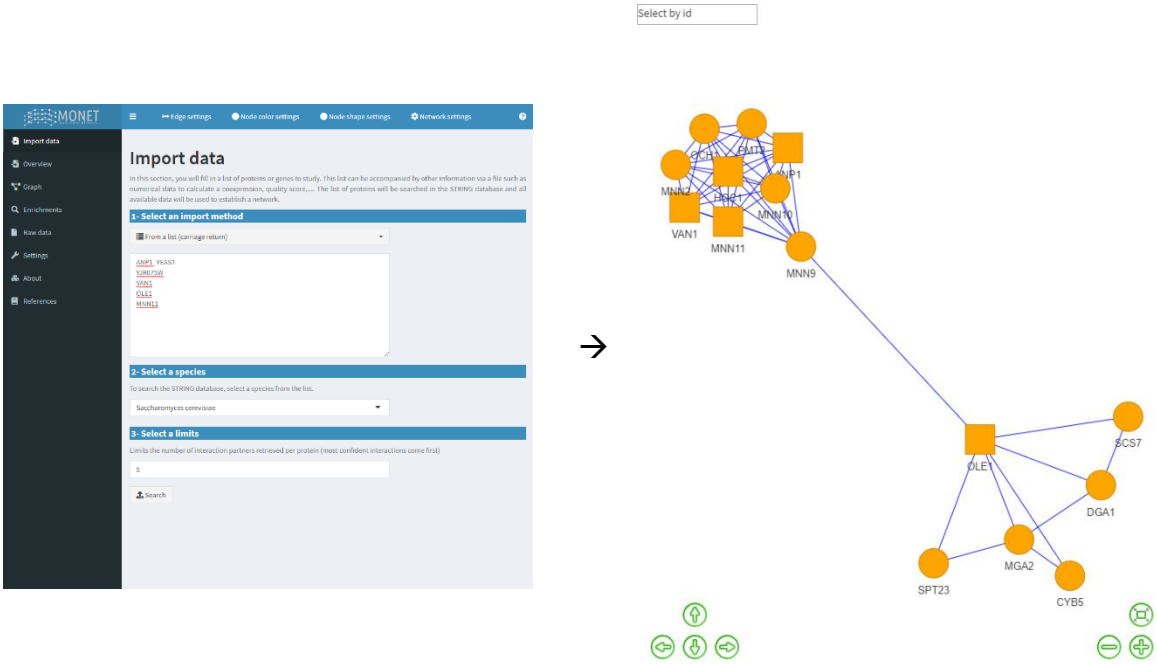


Figure 30 – Exemple de réseau obtenu lors de la recherche d'une liste de protéines de la levure Saccharomyces cerevisiae. À gauche une capture d'écran de la page WEB de MONet, dans laquelle la liste de gènes/protéines peut être saisie. À droite le réseau obtenu, après interrogation de la base de données STRING.

À chaque exploration, un *Dashboard* est mis à jour pour fournir 4 informations importantes : (1) le nombre d'éléments dans la liste initiale, (2) le nombre d'éléments qui n'ont pas été trouvés

dans la base de données STRING, (3) le nombre d'éléments dans le réseau final, et (4) le nombre de connexions dans le graphe final.

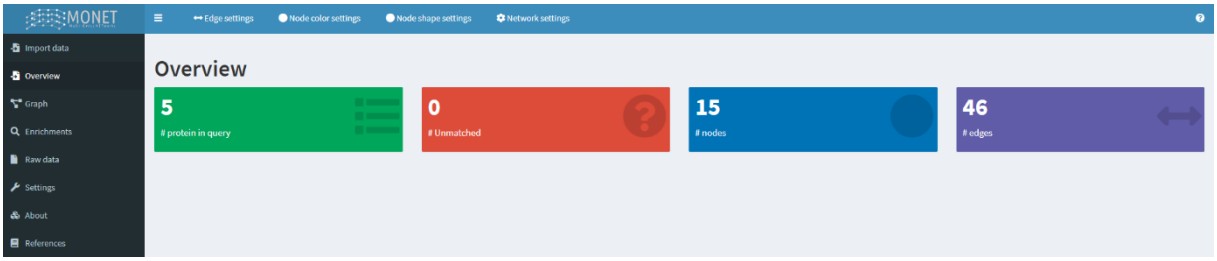


Figure 31 – Exemple de dashboard obtenu lors de la recherche d'une liste de protéines de la levure S. cerevisiae. Il s'agit de la même liste que celle présentée dans la figure précédente.

Dans le cas où l'utilisateur importe un fichier, celui-ci doit être au minimum composé d'une colonne contenant la liste des éléments à inclure dans le réseau. Une aide à l'import est proposée. Une prévisualisation permet aussi à l'utilisateur de vérifier la bonne lecture de ses données, de vérifier que le fichier est composé de plusieurs colonnes, etc. L'utilisateur peut également renseigner la colonne qui servira de liste d'éléments (par défaut, la première). Il est aussi possible d'importer d'autres colonnes du fichier pour utiliser des fonctionnalités avancées. Parmi ces fonctionnalités, il y a la possibilité de :

- Calculer un réseau de co-expression – Dans un tel réseau, les éléments sont représentés par des nœuds et les similitudes entre les profils d'expression de deux éléments sont représentées par des arêtes. Pour déduire un réseau, il est nécessaire de (1) calculer une mesure de la distance entre les profils d'expression des gènes (distance euclidienne par exemple), (2) définir une valeur seuil telle que si la distance entre deux profils d'expression des gènes est inférieure au seuil, les nœuds correspondants sur le réseau sont reliés par une arête et (3) appliquer des algorithmes dédiés pour calculer les positions des nœuds dans un espace bidimensionnel, de telle sorte que les gènes co-exprimés sont représentés par des nœuds positionnés côte à côte comme illustré ci-dessous avec les gènes A et B (Figure 32 – Réseau rouge à droite) et les gènes C, D et E (Figure 32 – Réseau gris à gauche). Actuellement, deux méthodes sont disponibles (calcul d'une distance euclidienne avec la fonction `dist()` et calcul d'une corrélation).

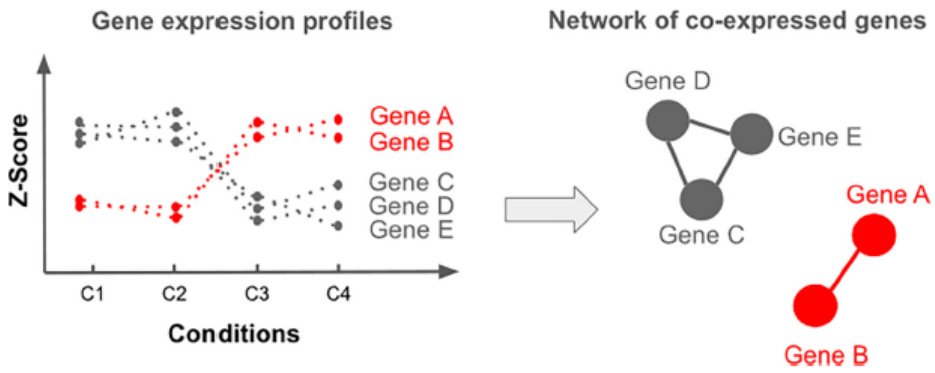


Figure 32 – Représentation schématique d'un réseau d'éléments co-exprimés. Cette figure est extraite de l'article présenté page 149.

- Définir la taille des éléments en fonction de l'abondance – Grâce à cette fonctionnalité, il est possible de distinguer les éléments très abondants (grande taille) et des éléments peu abondants (petite taille).
- Colorer les éléments en fonction d'un différentiel d'expression – Grâce à cette fonctionnalité, il est possible de mettre en évidence des *clusters* d'éléments qui ont une augmentation/diminution de leur niveau d'expression.
- Colorer les éléments en fonction d'un score de qualité – Grâce à cette fonctionnalité, il est possible de mettre en évidence des groupes de gènes plus importants que d'autres.
- Changer la forme des éléments en fonction d'une annotation – Dans notre étude sur l'homéostasie du fer chez *C. glabrata*, nous avons mis en évidence deux types de gènes : type I et type II (voir page 128). Grâce à cette fonctionnalité, il est possible de mettre en évidence ces types particuliers et découvrir rapidement avec quels éléments ils sont connectés.
- Ajouter de l'annotation dans le rapport d'exploration – Lors de la génération d'un rapport d'exploration pour un élément donné, toutes les informations disponibles sont collectées. Il est possible que l'annotation soit incomplète ou que nous ayons une annotation personnelle. Grâce à cette fonctionnalité, il est possible d'ajouter ces informations dans le rapport.

Une fois toutes les colonnes renseignées, toutes les zones de personnalisation sont automatiquement mises à jour pour permettre à l'utilisateur de les utiliser.

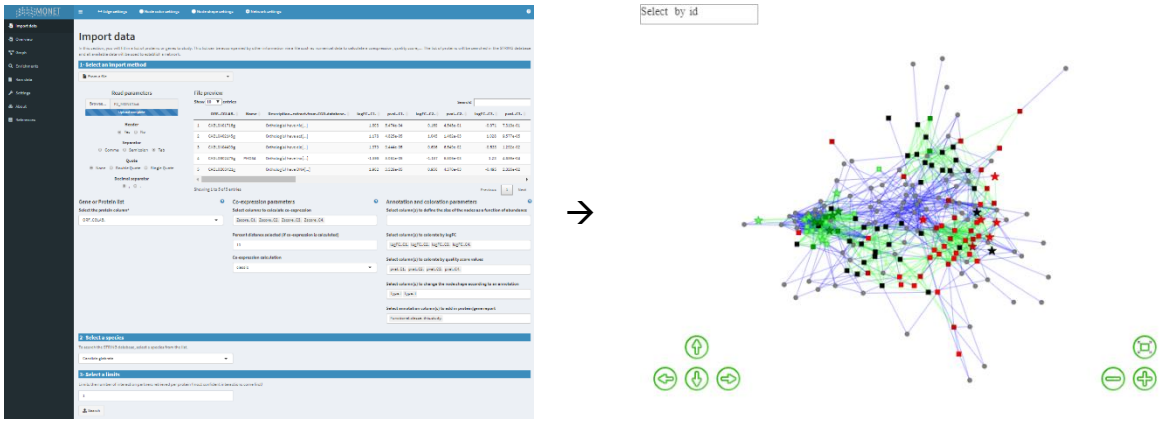


Figure 33 – Exemple de graphe obtenu à partir d'un fichier de données d'expression de gènes de la levure C. glabrata.

Lorsque des données d'expression sont renseignées, un histogramme de la distribution des distances et un histogramme de la distribution des données d'expression sont générés en plus du *dashboard* précédemment présenté.

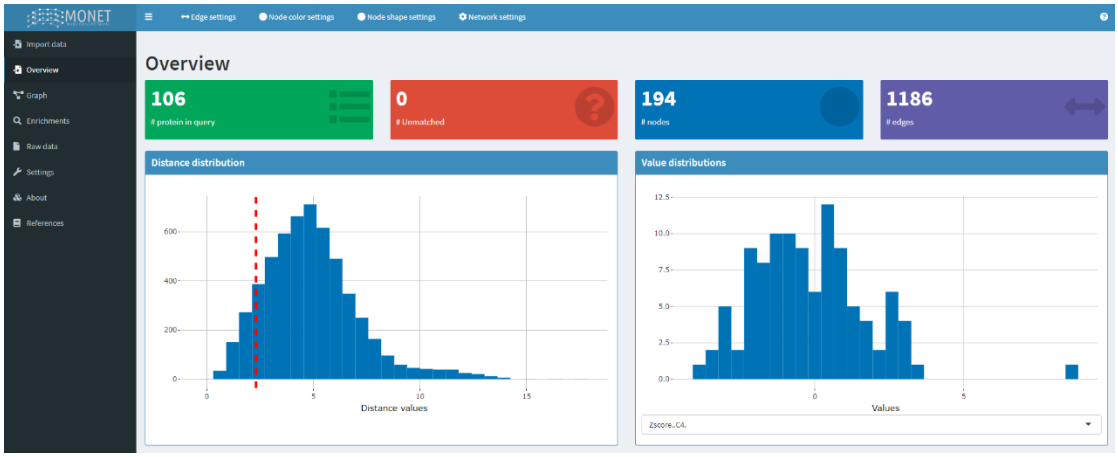


Figure 34 – Exemple de la page Overview après l'import d'un fichier.

Collecte automatique et regroupement des informations disponibles pour les éléments du réseau

Au final, MONet permet le regroupement d'informations provenant des bases de données UniProt (Consortium 2019), Refseq (O'Leary et al. 2016), KEGG (Kanehisa et al. 2000), GeneCards (Stelzer et al. 2016), ensembl (Yates et al. 2020), neXtProt (Zahn-Zabal et al. 2020), CGD (Skrzypek et al. 2017), SGD (Cherry et al. 2012), Pfam (El-Gebali et al. 2019), Smart

(Letunic et al. 2018), Reactome (Jassal et al. 2020), Interpro (Mitchell et al. 2019), PubMed[120], Quick GO et GO consortium (Carbon et al. 2019) et la PDB (Burley et al. 2019). Pour que l'utilisateur puisse sauvegarder son exploration ainsi que toutes les informations collectées, il peut générer une page HTML. Ce fichier contient aussi l'ensemble des versions des outils utilisés pour les obtenir. Un extrait d'un fichier généré est présenté Figure 35.

Figure 35 – Extrait du rapport exporté du gène TP53 de l'Humain.

[120] https://pubmed.ncbi.nlm.nih.gov/ [Accessible le 04/05/2020]

Enrichissement fonctionnel

Parmi les données disponibles via l'API de STRING, il est possible de récupérer des informations d'enrichissement fonctionnel, via les GO terms[121] (avec une distinction entre les types *cellular component*, *biological process* et *molecular function*), les termes Pfam, les termes InterPro, les termes Reactome, les termes UniProt et les termes KEGG. Chaque terme présentant un enrichissement important peut être sélectionné, ce qui change la coloration des éléments du réseau associé.

d. Conclusion et perspectives

Les objectifs de MONet sont d'aider à la découverte de nouvelles connexions entre des gènes / protéines par une co-expression, de récupérer les connexions déjà connues dans la base de données STRING et de générer un réseau simple à explorer (Figure 36). Grâce à Docker, l'installation des différents outils nécessaires n'est pas à réaliser. En effet, tous les *packages* et leurs dépendances sont installés dans l'image disponible sur Docker Hub. Grâce à l'utilisation de Shiny, l'utilisateur est guidé pour saisir les différents paramètres avec des bulles d'informations qui sont disponibles à chaque étape. Un rapport d'informations est disponible pour chaque gène / protéine du réseau. L'ensemble des données générées et importées est téléchargeable *via* un fichier Excel. L'utilisateur peut ainsi sauvegarder son analyse. À l'aide des différents *packages* associés à Shiny, l'utilisateur peut personnaliser le réseau (couleur, forme, taille des nœuds, etc.). Enfin, l'utilisateur peut mettre en évidence certaines caractéristiques dans le réseau comme un enrichissement fonctionnel. En conclusion, nous proposons une solution totalement *Open Source*, gratuite et conviviale pour découvrir et explorer de potentielles nouvelles interactions entre des gènes / protéines à partir de données multi-omiques.

[121] Pour en savoir sur les GO terms, nous proposons une ressource numérique : https://thomasdenecker.github.io/thesisWebsite/annexes/geneOntology/ [Accessible le 10/08/2020].

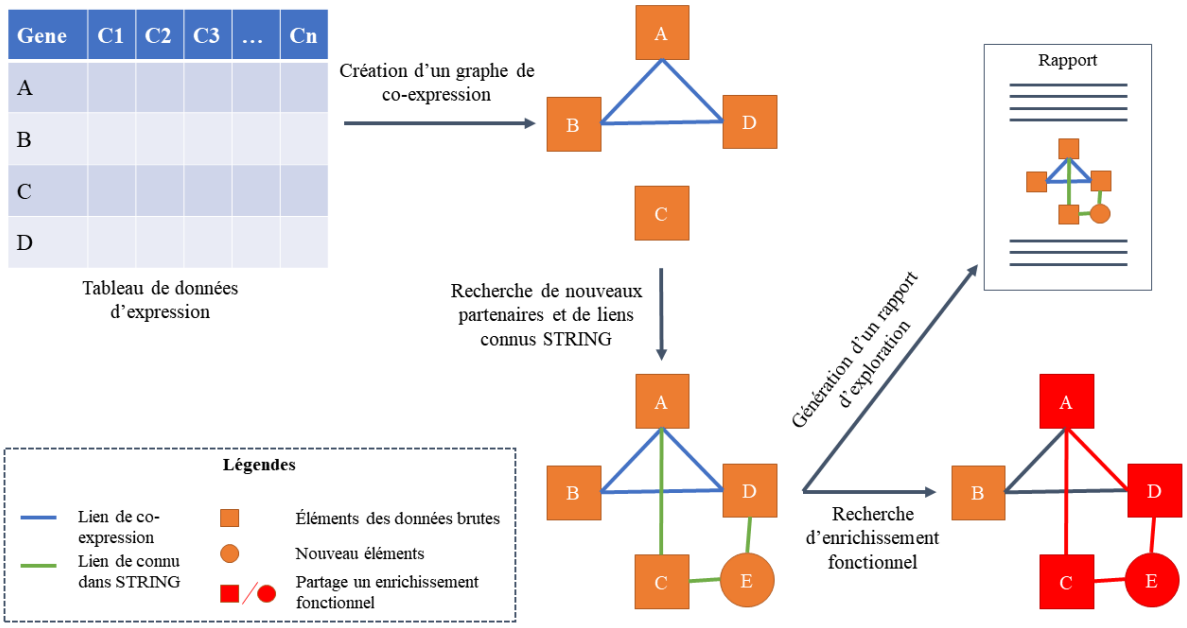


Figure 36 – Résumé schématique des différentes étapes de fonctionnement du logiciel MONet à partir d'un tableau de données d'expression de gènes.

MONet a été implémenté pendant le confinement lors de la pandémie de la COVID-19. Au moment de l'écriture de cette partie du manuscrit (juin 2020), nous n'avions pas encore eu les retours d'expériences des utilisateurs. En tant que développeur, je peux cependant envisager plusieurs voies d'amélioration telles que :

- L'implémentation de nouvelles approches de détection de connexions – Aujourd'hui, deux méthodes sont proposées : par un calcul de distance et par le calcul d'une corrélation. Nous pourrions ajouter par exemple les packages R WGCNA[122] ou coexnet[123] .
- L'implémentation de méthodes d'enrichissement fonctionnel – Aujourd'hui, l'enrichissement fonctionnel disponible est celui calculé par STRING. Cette nouveauté serait particulièrement intéressante dans le cas où STRING ne fournirait pas d'informations et que nous ayons par exemple accès à une liste de termes GO non disponibles en ligne.

[122] https://horvath.genetics.ucla.edu/html/CoexpressionNetwork/Rpackages/WGCNA/ [Accessible le 04/05/2020]

[123] https://www.bioconductor.org/packages/release/bioc/vignettes/coexnet/inst/doc/coexnet.pdf [Accessible le 04/05/2020]

V. La formation « FAIR_Bionfo » pour l'apprentissage des pratiques informatiques qui soutiennent la reproductibilité des résultats

1. Le contexte du projet

Pour rappel, l'objectif de la formation FAIR_Bioinfo était de sensibiliser les chercheurs de l'I2BC aux différentes techniques de reproductibilité, en appliquant des principes dérivés du « *FAIR data* ». Cette formation a été proposée en 2019 et s'est déroulée entre les mois de janvier et juillet, sous la forme de huit sessions de 2 h 00. Chacun était libre de participer, à la condition d'apporter son ordinateur de travail et de disposer d'un accès valide au WIFI de l'institut. En moyenne, une quinzaine de chercheurs étaient présents à chacune des sessions. J'ai souhaité créer cette formation pour plusieurs raisons. Tout d'abord, j'apprécie enseigner et partager les connaissances avec les autres. Dans le contexte des projets bPeaks App et Pixel, j'avais été confronté aux problématiques de reproductibilité en informatique. J'avais appris à utiliser des solutions techniques qui ne m'avaient pas été enseignées lors de ma formation universitaire. Également, j'avais rencontré C. Toffano-Nioche, responsable de l'animation du « Club Bioinformatique » du Département Biologie des Génomes de l'I2BC. Elle est également formatrice pour l'école de bioinformatique organisée par l'IFB. Ensemble, nous avons eu l'idée de transposer les principes « FAIR-data » (voir page 73) à un pipeline bioinformatique d'analyse de données. Nous avons ainsi défini un programme de formation, créé les ressources pédagogiques et animé les séances. À noter que la formation FAIR_Bioinfo sera renouvelée en 2020, avec le soutien cette fois de l'IFB[124].

2. La documentation pédagogique

a. Où trouver la formation ?

La formation est disponible dans deux formats. Sous forme de livre disponible sur Gitbook[125], où le contenu est rédigé en anglais, et sous la forme de cours accessibles sur GitHub[126]. Un

[124] https://www.france-bioinformatique.fr/en/evenements/principes-fair-appliques-a-la-bioinformatique [Accessible le 01/06/2020]

[125] https://fair-bioinfo.gitbook.io/fair-bioinfo/ [Accessible le 27/04/2020]

[126] https://github.com/thomasdenecker/FAIR_Bioinfo [Accessible le 27/04/2020]

WIKI[127] est aussi disponible pour préciser des notions essentielles au suivi de la formation (initiation au bash, git, github, Markdown). Le contenu est rédigé cette fois en français.

b. Programme

La formation est divisée en 8 sessions, chacune décrivant un niveau de reproductibilité supplémentaire à une analyse de données. Nous avons choisi comme exemple, de reproduire une analyse de données RNAseq (Lelandais et al. 2016). Il s'agit d'une étude transcriptomique classique, avec pour objectif d'identifier des gènes différentiellement exprimés entre des conditions comparées. Le fonctionnement des programmes bioinformatiques permettant de traiter les données n'est pas expliqué dans cette formation, car elle se concentre uniquement sur les outils permettant de réobtenir à tout moment les mêmes résultats. Le Tableau 4 présente les intitulés de séances, ainsi que des résumés de leurs contenus.

Session	Titre	Description	Outils
1	Ce n'est pas de la magie	Initiation à l'utilisation d'un terminal UNIX et récupération des données brutes (fichiers FASTQ) automatiquement à l'aide d'un script. Un ensemble de commandes bash ainsi que les notions de boucles et de variables sont présentées.	Shell Wget md5sum
2	La mémoire du code	Initiation au versionnage de code.	Git Github
3	Mise en place de l'analyse de données	Implémentation de la première partie du pipeline d'analyse (des fichiers FASTQ à la table de comptage).	Conda FastQC bowtie2 samtools htse-counts

[127] https://github.com/thomasdenecker/FAIR_Bioinfo/wiki [Accessible le 27/04/2020]

4	Une virée en mer	Contrôler l'environnement de travail en utilisant les *container* et lancement de l'analyse sur un *cloud*.	Docker SSH
5	I've got the power !	Parallélisation des analyses et utilisation d'un *cluster* de calculs.	Snakemake Slurm Singularity ascp
6	LoveR	Partager de façon dynamique des résultats.	Shiny et différents package R
7	Partager ses résultats et ses protocoles avec des notebooks	Créer un rapport d'analyse et le partager.	Jupyter Rmarkdown
8	Diffuser un projet reproductible.	Partager son projet Perspective d'améliorations	Github Zenodo

Tableau 4 – Programme proposé lors de la première formation FAIR_bioinfo. Cette formation a été proposée aux chercheurs de l'I2BC en 2019.

À la fin de chaque session, un exercice a été proposé pour remettre en application les concepts et compétences qui ont été discutés. Nous avons également ouvert un canal de communication (Slack) pour répondre aux questions ou aider à la résolution des problèmes rencontrés entre deux sessions. À la fin de la formation, les participants étaient capables de mettre en place les 7 grandes étapes nécessaires pour rendre une analyse de données reproductible (Figure 37).

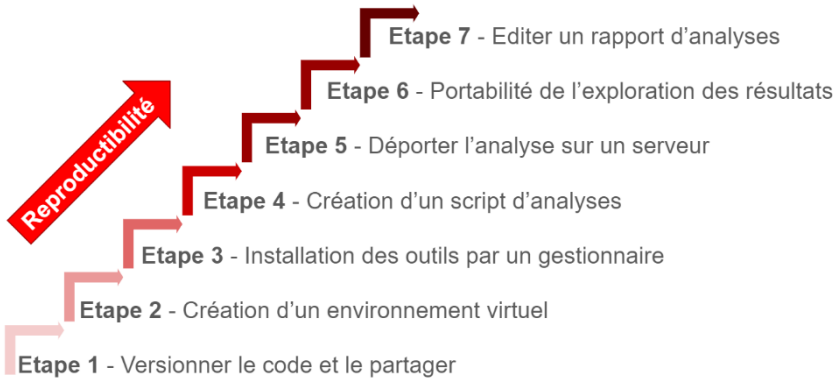


Figure 37 – Les 7 étapes proposées dans la formation FAIR_bioinfo pour rendre une analyse de données reproductible.

c. Diffusion du concept de la formation FAIR_Bioinfo

Une note de présentation de la formation FAIR_Bioinfo est disponible dans l'archive HAL[128]. J'ai également présenté le concept de FAIR_bioinfo lors de plusieurs communications orales :

- Aramis - La reproductibilité en pratique : méthodes et outils. La webcast de l'évènement est disponible en ligne[129].
- Au cours d'un séminaire interne au CNRGH[130] (Centre National de Recherche en Génomique Humaine) ;
- Aux Cafés LoOPS[131] ;
- Au séminaire interne de bioinformatique à l'Institut Jacques Monod, Diderot Mix.

3. Le futur de la formation FAIR_Bioinfo

a. Quelques questionnements

Sommes-nous reproductibles à 100% ?

Dans l'article *Practical Computational Reproducibility in the Life Sciences* (Grüning et al. 2018), nous pouvons constater que nous couvrons les recommandations proposées (Figure 38). Cependant, en échangeant avec la communauté des informaticiens, deux principales limites ont été mises en avant :

- La reproductibilité « bit à bit » - Même si un environnement contrôlé est créé pour lancer l'analyse, il utilise les ressources disponibles qui sont différentes d'un ordinateur à l'autre. Il existe des solutions pour résoudre ce problème mais elles sont complexes à mettre en place (Nix et Guix).
- La parallélisation – Lorsque les calculs sont parallélisés, il peut y avoir une modification de l'ordre dans lequel les calculs sont réalisés ou un changement de matériel. Comment

[128] https://hal.archives-ouvertes.fr/hal-02880655v1 [Accessible le 24/08/2020]

[129] https://webcast.in2p3.fr/video/la-reproductibilite-au-service-de-la-biologie-computationnelle [Accessible le 27/04/2020]

[130] http://jacob.cea.fr/drf/ifrancoisjacob/Pages/Departements/CNRGH.aspx [Accessible le 27/04/2020]

[131] Présentation du café LoOPS https://reseau-loops.github.io/cafes/ et lien vers l'abstract de la présentation https://reseau-loops.github.io/2019/06/03/cafe-loops [Accessible le 27/04/2020]

expliquer que le même calcul prenne 5 minutes la première fois et 25 minutes la seconde (en temps réel de travail) ? S'il s'agit d'un changement de matériel, comment garantir que les résultats sont strictement les mêmes ?

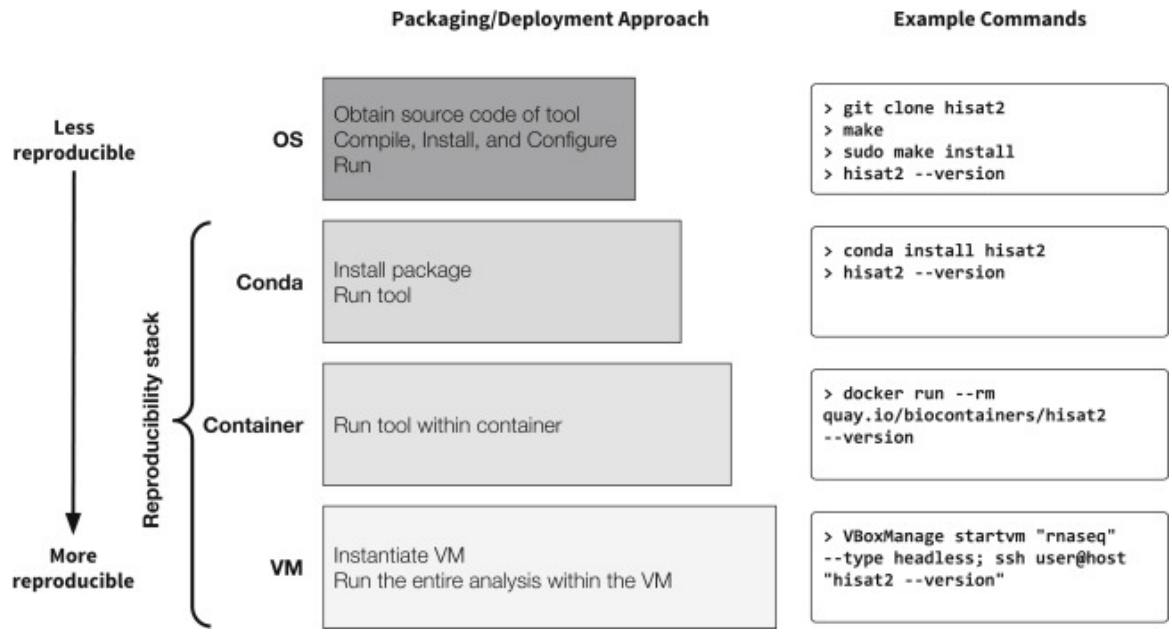


Figure 38 – Les différentes solutions proposées par B. Grüning pour augmenter la reproductibilité des analyses bioinformatiques (Grüning et al. 2018).

Est-il indispensable d'être reproductible à 100% ?

Deux stratégies sont possibles :

1. Être reproductible à tout moment, à tous les niveaux – Sans doute l'approche recommandée par les fervents de la reproductibilité.
    + Le projet peut être mené à plusieurs et n'est pas dépendant d'une personne ni d'un ordinateur. Il peut ainsi être mis en pause puis repris par d'autres.
    - La mise en route du projet est longue pour mettre en place tous les outils (Docker, installation, Github, etc.)
2. Rendre reproductible lorsque l'analyse de données est terminée – Sans doute l'approche la plus suivie actuellement.
    + L'obtention des résultats est plus rapide. En cas de réorientation du projet, il n'y aura pas eu de mise en place particulière.

- La mise en place de la reproductibilité à la fin peut être complexe. Nous pouvons par exemple nous rendre compte que les outils ne sont pas compatibles dans un même environnement. Et que faire si les résultats obtenus avant et après cette mise en place ne sont pas identiques ?

Je pense qu'une bonne solution se trouve entre les deux. La reproductibilité est sans aucun doute importante dans un projet d'analyse de données, mais elle ne doit pas devenir un frein. Personnellement, lorsque je démarre un nouveau projet, je crée un répertoire sur Github et un environnement Docker. La documentation (autres que les commentaires dans le code) et l'automatisation viennent dans un second temps.

b. Partenariat mis en place avec l'IFB

La prochaine édition est prévue à la fin de l'été 2020, avec le support de l'IFB : https://www.france-bioinformatique.fr/en/evenements/principes-fair-appliques-a-la-bioinformatique. Il s'agira de former les formateurs.

4. PDF de l'article déposé dans HAL (Denecker et al. 2020)

FAIR_Bioinfo: a turnkey training course and protocol for reproducible computational biology

Thomas Denecker, Claire Toffano-Nioche

FAIR_Bioinfo: a turnkey training course and protocol for reproducible computational biology

Thomas Denecker[1] (orcid: 0000-0003-1421-7641) & Claire Toffano-Nioche[1] (orcid: 0000-0003-4134-6844)

[1]CEA, CNRS, Univ. Paris-Sud, Institute for Integrative Biology of the Cell (I2BC), Gif-sur-Yvette, France.

Keywords: Reproducibility; Bioinformatics; Computational Biology; FAIR

Summary

Reproducibility plays an essential part in the success of a bioinformatics project. Indeed, Reproducibility makes it possible to guarantee the validity of scientific results and to simplify the dissemination of projects. To help disseminate Reproducibility principles among bioinformatics students, engineers and scientists, we created the FAIR_Bioinfo course, which presents a set of features we consider necessary to make a complete bioinformatics analysis reproducible. To illustrate the theoretical concepts of reproducibility, we use as an example a classic bioinformatics analysis (differential gene expression analysis from RNA-seq data). In short, we retrieve the data from public databases (ENA/SRA), we perform a reproducible analysis using a workflow management system (snakemake) in a virtual environment (Docker). The entire versioned (git) code is open source (Github [1] and dockerhub [2]). The visualization of the results is dynamic (Shiny app) and the PDF or HTML report (Rmarkdown) provides the results of the analysis and lists all user-selected parameters.

How to use

The whole example proposed in FAIR_Bioinfo can be simply reproduced with the following 4 command lines executed in a terminal (note that docker must be installed beforehand [3]):

```
$ git clone https://github.com/thomasdenecker/FAIR_Bioinfo
$ cd FAIR_Bioinfo
$ sudo docker run --rm -d -p 8888:8888 --name fair_bioinfo -v
${PWD}:/home/rstudio tdenecker/fair_bioinfo
$ sudo docker exec -it fair_bioinfo bash ./FAIR_script.sh
```

Once the analysis is complete, the web application can be launched with the following command:

```
$ sudo docker exec -it fair_bioinfo bash ./FAIR_app.sh
```

and used by copying http://localhost:8888/rstudio/p/4444/ into your browser.

Note: The use of the `sudo` command is not necessary for Mac OS and Windows. The demonstrations of the course have been made on a Linux environment (Ubuntu 18.04).

Statement of Need

Recent reanalyses of biology (Baker 2016) and computer science (Warren 2015) papers found a staggering ratio of non-reproducible results. Our daily experience as bioinformaticians indicate the same issues exist in our field. It is therefore essential that good practices are implemented to ensure data integrity and reproducibility of analytical results. This course was created originally at the request of colleagues who wanted to be trained in reproducibility practices in their own language (French). This course was born from the combination of our personal experiences in reproducibility acquired during our research work in Bioinformatics. We therefore created and delivered a set of training sessions in French, including a complete workflow and associated course materials available on Github [1]. Following positive initial feedback and at the request of our English-speaking colleagues, we decided to translate the workflow documentation to English and to convert it into a Gitbook document [4].

Target audience

No prior computational skills are required to complete the entire course. This training is therefore adapted to biologists willing to gain autonomy in their bioinformatics analyses. For example, we start by showing how to open a terminal (under any of the Windows, MacOSX or Ubuntu OS). At the end of the session, participants are able to reproduce an entire bioinformatics analysis, which can be launched either on their personal computer or on a distant server (or computational cloud or cluster services).

Learning Objectives and Content

In this FAIR_Bioinfo training, we present a stepwise protocol to ensure the reproducibility of Bioinformatics analysis and to guarantee identical results from the same data set and over time.

For this reason, we extended the FAIR principles popular in the field of data management (Wilkinson 2016) to propose FAIR principles for bioinformatics workflows:

- **F**indable: The tools used are references in their field (bowtie2, samtools, HTseq-counts, etc);
- **A**ccessible: Codes, slides, docker image are online (Github & Dockerhub)
- **I**nteroperable: The different tools will communicate with each other (conda & snakemake);
- **R**eusable / Reproducible: The workflow is saved in a file whose execution replays the entire analysis identically (Jupyter, Rmarkdown, etc).

Our protocol is composed of seven main steps which application gradually increases the level of reproducibility (Figure 1).

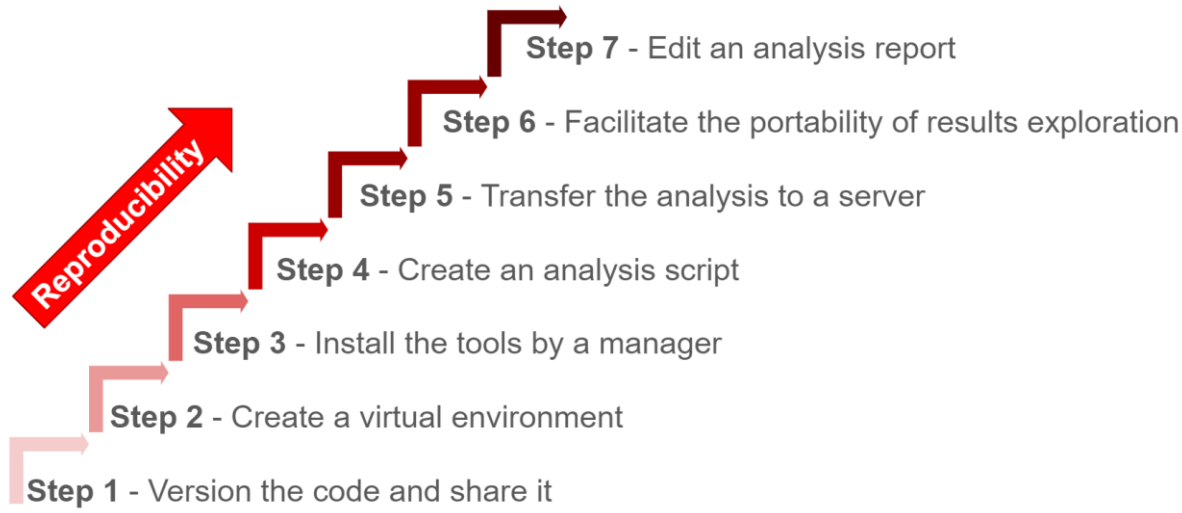


Figure 1 - The 7-step solution of FAIR_bioinfo

To illustrate FAIR principles for bioinformatics workflows, we used a classic RNA-seq data analysis workflow. Starting from raw data (FASTQ files), this analysis aims to identify genes that are differentially expressed between different conditions. The underlying biology is not detailed, as our focus is on the reproducibility aspects rather than the specific bioinformatics protocol or biological question. Eventually, the RNA-seq analysis protocol used as an example here could be replaced by any other type of bioinformatics analysis.

Instructor notes

The training course is divided into 8 sessions, each one bringing an additional level of reproducibility to the global workflow. Each session lasted an hour and a half; they were carried

out at the rate of one session per month. Table 1 shows the detailed program and learning objectives of each session.

Title	Session	Description	Tools
This is not magic	1	Open a terminal and retrieve data for the analysis pipeline (loop & variable concepts, bash command)	shell, wget, md5sum
The code memory	2	Initiation to code versioning	Git & Github
Play with analysis tools	3	Implementation of a first part of the analysis pipeline from the fastq file to the count table	conda, third-party tools used for bioinformatics analysis (FastQC, bowtie2, samtools, HTseq-counts)
A trip to the sea	4	Control the computing environment using a container and compute a cloud platform	Docker, ssh
I've got the power!	5	Parallel computing and use of a computer cluster	Snakemake, slurm, Singularity, ascp
LoveR	6	Upgrade to dynamic rendering using a web application and presentation of a few powerful R packages	Shiny, R packages
Sharing results and protocols using notebooks	7	Creating an analysis report and sharing it	Jupyter & Rmarkdown
Disseminate your project	8	Disseminate a reproducible project. Perspective for improvement	Githubpages, License, Release, Zenodo

Table 1 – Overview of sessions

The slides of each session are available on the FAIR_Bioinfo github in French. They have been translated into English and inserted as figures in the Gitbook. English speaking instructors can very easily create slides from the Gitbook and these figures.

Conclusion and Perspectives

We proposed a training course to help scientists improve their computational practice by ensuring reproducibility of bioinformatic analysis. The training course is divided into 8 sessions each introducing a core competence for reproducible analysis. We also address possible future developments (last session), including:

- Installing all tools in a Conda environment itself residing in a Docker image. The goal is to isolate the installation layer (Grüning et al. 2018);
- Setting up a Virtual machine to run the Docker container. The goal is to stand the test of time by also setting the Operating System;
- Implement continuous integration to ensure that code changes do not cause changes in results.

Through the FAIR_Bioinfo training, we offer users a solution to make Bioinformatics analyses entirely reproducible. All the tools used can be replaced by other equivalent solutions (docker by singularity, snakemake by nextflow, etc). Thus, with FAIR data and protocols according to the principles of FAIR_Bioinfo, we obtain FAIR-processed data. This same data can be used again and again to enter a virtuous circle of reproducible analyses.

References

Baker, Monya. 2016. "1,500 Scientists Lift the Lid on Reproducibility : Survey Sheds Light on the 'Crisis' Rocking Research." *Nature*. https://doi.org/10.1038/533452a.

Grüning, Björn, John Chilton, Johannes Köster, Ryan Dale, Nicola Soranzo, Marius van den Beek, Jeremy Goecks, Rolf Backofen, Anton Nekrutenko, and James Taylor. 2018. "Practical Computational Reproducibility in the Life Sciences." *Cell Systems*. Cell Press. https://doi.org/10.1016/j.cels.2018.03.014.

Warren, Alex. 2015. "Repeatability and Benefaction in Computer Systems Research — A Study and a Modest Proposal." http://reproducibility.cs.arizona.edu/v2/RepeatabilityTR.pdf.

Wilkinson, Mark D. 2016. “Comment: The FAIR Guiding Principles for Scientific Data Management and Stewardship.” *Nature Publishing Group*. https://doi.org/10.1038/sdata.2016.18.

Webography

[1] FAIR_Bioinfo GitHub https://github.com/thomasdenecker/FAIR_Bioinfo

[2] FAIR_Bioinfo docker https://hub.docker.com/r/tdenecker/fair_bioinfo

[3] Docker installation https://www.docker.com/products/docker-desktop

[4] FAIR_Bioinfo Gitbook https://fair-bioinfo.gitbook.io/fair-bioinfo/

Contributions aux projets d'analyse de données multi-omiques : génomique fonctionnelle des levures pathogènes *Candida glabrata* et *Candida albicans*

I. Les levures pathogènes *Candida glabrata* et *Candida albicans*

Les levures sont des champignons unicellulaires. Généralement non dangereuses, certaines levures sont pathogènes, c'est-à-dire qu'elles peuvent provoquer des infections fongiques chez l'humain. Ces infections sont un problème important dans les hôpitaux car elles sont à l'origine d'infections nosocomiales[132]. Au cours de mon travail de thèse, j'ai été impliqué dans des projets de génomique fonctionnelle menés avec deux espèces de levures pathogènes nommées *Candida glabrata* et *Candida albicans*. J'ai aussi travaillé avec des données provenant de la levure modèle (non pathogène) *Sacharomyces cerevisiae,* plus communément nommée « levure de boulanger » ou « levure de bière ».

1. Des risques importants pour la santé publique

Les levures *Candida* font partie de la composition normale de la flore commensale[133] des humains. Elles sont trouvées principalement à la surface des muqueuses de la cavité buccale ou des tractus gastrointestinal et urogénital (Underhill et al. 2014; Cho et al. 2012; Cui et al. 2013). Cependant, elles peuvent devenir opportunément pathogènes dans des conditions appropriées. Ainsi, elles sont qualifiées de « levures pathogènes opportunistes », c'est-à-dire qu'elles provoquent des troubles lorsque les défenses de l'hôtes sont affaiblies, comme lors d'une utilisation prolongée d'antibiotiques ou lorsque l'hôte est immunodéprimé (par exemple dans le cas d'atteinte par d'autres pathologies telles que le VIH, le cancer, etc.). Le nombre de sepsis[134] causés par les levures pathogènes est en augmentation depuis les années 90. L'espèce la plus souvent impliquée est *Candida albicans* suivie par *Candida glabrata* (Guinea 2014). À noter que seulement 5 espèces de *Candida* sont la cause de 92% des candidémies (*C. albicans, C. glabrata, C. tropicalis, C. parapsilosis,* et *C. krusei*). La Figure 39 montre l'évolution des fréquences d'infections en relation avec chacune de ces espèces. Il est intéressant d'observer que si le pourcentage des infections causées par *C. albicans* diminue depuis 20 ans (73% à

[132] « *Une infection nosocomiale ou infection associée aux soins est une infection contractée au cours d'un séjour dans un établissement de soins. Elle peut être directement liée aux soins ou survenir durant l'hospitalisation, en dehors de tout acte médical.* » (Assistance Publique – Hôpitaux de Paris).

[133] « *Organisme se nourrissant aux dépends d'un autre sans lui causer de dommage* » (Delamare 2009)

[134] Réponse inflammatoire généralisée associée à une infection grave.

65%), celui des infections causées par *C. glabrata* reste constant (de l'ordre de 11%). Ce phénomène est également décrit dans l'article de A. Doi (Doi et al. 2016).

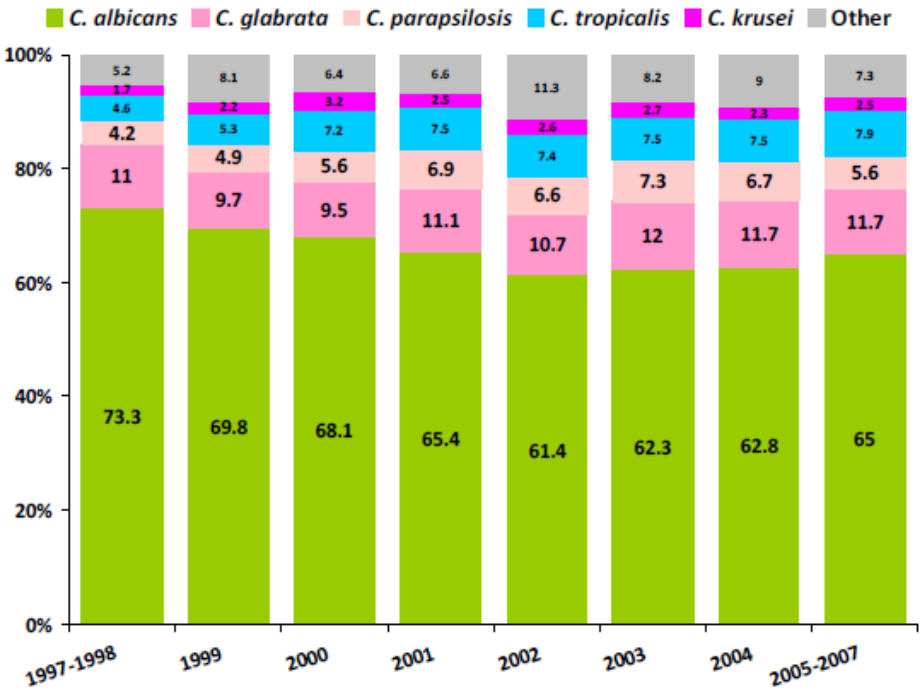


Figure 39 – Evolution au cours du temps (période 1997 à 2007) des pourcentages d'infections causées par les levures C. albicans (vert), C. glabrata (rose), C. parapsilopsis (rose claire), C. tropicalis (bleu clair) et C. krusei (magenta). Ce graphique est extrait de l'article de Guinea et al. et a été réalisé à partir des données ARTEMIS DISK (Guinea 2014).

Malgré la commercialisation de nouveaux antifongiques et les progrès réalisés dans le suivi médical des patients gravement malades, l'incidence[135] des candidémies reste importante (72.8 / million d'habitant / an). Le diagnostic des infections sanguines par une levure *Candida* (candidémies) est en effet difficile à réaliser du fait d'hémocultures peu sensibles et des résultats longs à obtenir (plusieurs jours). Plus de la moitié des cas ne sont ainsi pas détectés. Le taux de mortalité est très important, de l'ordre de 15-35% pour un adulte et 10-15% pour un nouveau-né (Guinea 2014). En cas de diagnostic tardif, le temps d'hospitalisation est prolongé et le risque de mortalité peut augmenter jusqu'à 50%. Ainsi, les coûts humains et financiers liés à ces infections sont très importants. Les candidémies sont la 7ème cause d'infection du sang nosocomiale au Brésil (5%) et une des plus fréquentes aux USA (Wisplinghoff et al. 2004; Magill et al. 2018). Aujourd'hui encore, le corps médical sous-estime souvent le nombre de

[135] Nombre de cas apparus pendant une année au sein d'une population.

décès liés aux levures *Candida* qui serait proche de 1.5 millions par an (G. D. Brown et al. 2012). L'étude de ces micro-organismes a donc un intérêt pour la santé publique.

2. Des spécificités importantes, malgré l'appellation partagée de levures « *Candida* »

a. Deux génomes entièrement séquencés, l'un diploïde et l'autre haploïde

Candida albicans est une espèce diploïde (les chromosomes sont présents par paires) qui est capable d'effectuer une conversion morphologique en hyphe[136]. Ce changement morphologique permet aux cellules de franchir activement la barrière épithéliale de l'hôte (sans effraction préalable par du matériel médical). La séquence complète du génome de *C. albicans* est disponible depuis 2004 (Jones et al. 2004). Elle est composée de 28 Mégabases (total pour le génome diploïde), sur lesquelles ont été positionnées 12 405 cadres ouverts de lecture ou ORF[137] (soit 6 198 en haploïdie) répartis sur 8 chromosomes (nommés de 1 à 7 ainsi que le chromosome R)[138]. La séquence complète du génome de *Candida glabrata* est également disponible depuis 2004 (Dujon et al. 2004). *C. glabrata* est cependant une espèce haploïde (un seul exemplaire de chaque chromosome). Son génome est composé de 12 Mégabases, sur lesquelles ont été positionnées 5 294 cadres ouverts de lecture[139] répartis sur 13 chromosomes (nommés de A à M).

b. Des infections menées avec des stratégies différentes

Les levures *Candida* sont capables de causer deux types infections. Les premières sont des infections superficielles des muqueuses et des phanères (peau et ongles) appelées « candidoses ». Ces infections superficielles se localisent principalement au niveau de la peau, de la cavité buccale et du tractus uro-génital et elles se soignent très bien. Parmi les plus fréquentes, il y a la candidose vaginale qui touche la plupart des femmes au cours de leur vie et les candidoses oropharyngées, telles que le muguet chez les jeunes enfants. À noter que les

---

[136] Longs filaments tubulaires avec des côtés complètement parallèles. La forme levure et la forme hyphe sont illustrées Figure 45 et nous conseillons la revue de P. Sudbery pour aller plus loin (Sudbery 2011).

[137] *Open Reading Frames.*

[138] http://www.candidagenome.org/cache/C_albicans_SC5314_genomeSnapshot.html [Accessible le 06/05/2020]

[139] http://www.candidagenome.org/cache/C_glabrata_CBS138_genomeSnapshot.html [Accessible le 06/05/2020]

candidoses oropharyngées sont les infections les plus courantes chez les patients atteints par le VIH (Fidel 2006).

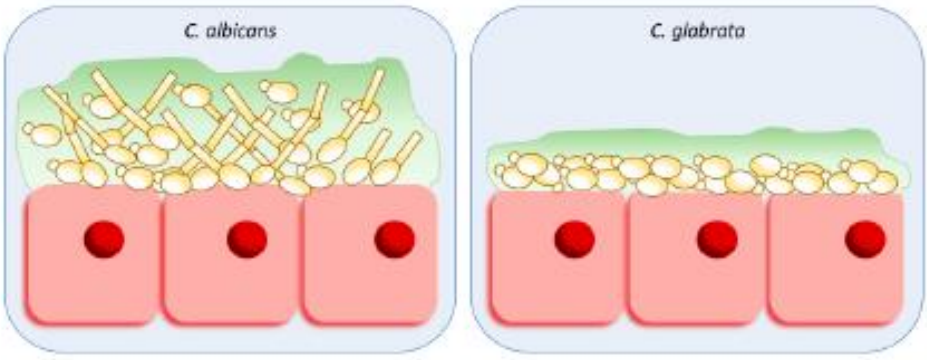


Figure 40 – Représentation schématique de la formation de biofilms par les levures C. albicans et C. glabrata. C. albicans forme des biofilms plus épais, avec beaucoup de biomasse à la fin de la formation du biofilm et produit plus de matrice extracellulaire que C. glabrata. Les biofilms matures de C. albicans sont composés d'un réseau dense de pseudohyphes, hyphes et cellules de levure, tandis que les biofilms de C. glabrata sont composés uniquement de cellules de levure compactes, formant un biofilm mince mais dense (Galocha et al. 2019).

Le second type d'infections causées par les levures *Candida* sont des infections systémiques des tissus. Elles sont beaucoup plus graves que les candidoses décrites précédemment. La stratégie de contamination de *C. glabrata* diffère de celle de *C. albicans* (Brunke et al. 2013). Contrairement à *C. albicans* (Sudbery 2011), *C. glabrata* n'est pas capable de changement morphologique en hyphe pour franchir la barrière épithéliale et disséminer dans le sang (Figure 41). Des pseudohyphes ont été mis en évidence en laboratoire (Csank et al. 2000) mais seule la forme levure a été observée et très rarement isolée en clinique (Kaur et al. 2005). Le risque le plus important d'infections par *C. glabrata* est donc lié aux techniques médicales invasives chez des patients âgés et/ou avec une pathologie sous-jacente telle qu'un diabète, une néoplasie ou une immunodépression comme celle induite lors des transplantations d'organe solide (Perlroth et al. 2007). Ces techniques médicales (l'utilisation d'un cathéter par exemple) provoquent des effractions de la barrière épithéliale et permettent la dissémination de *C. glabrata* dans le sang. Un fait intéressant est qu'il a été montré que *C. glabrata* peut profiter de l'infraction de la barrière épithéliale digestive causée par *C. albicans* pour proliférer (C. T. Alves et al. 2014; Tati et al. 2016). Ces infections systémiques sont appelées des « candidémies ». Elles sont très difficiles à diagnostiquer car elles ne provoquent souvent pas

d'état fébrile[140] (Leroy et al. 2008; 2009; 2016). Ces infections sont sévères avec un taux de mortalité proche de 50% (Jaillette et al. 2016).

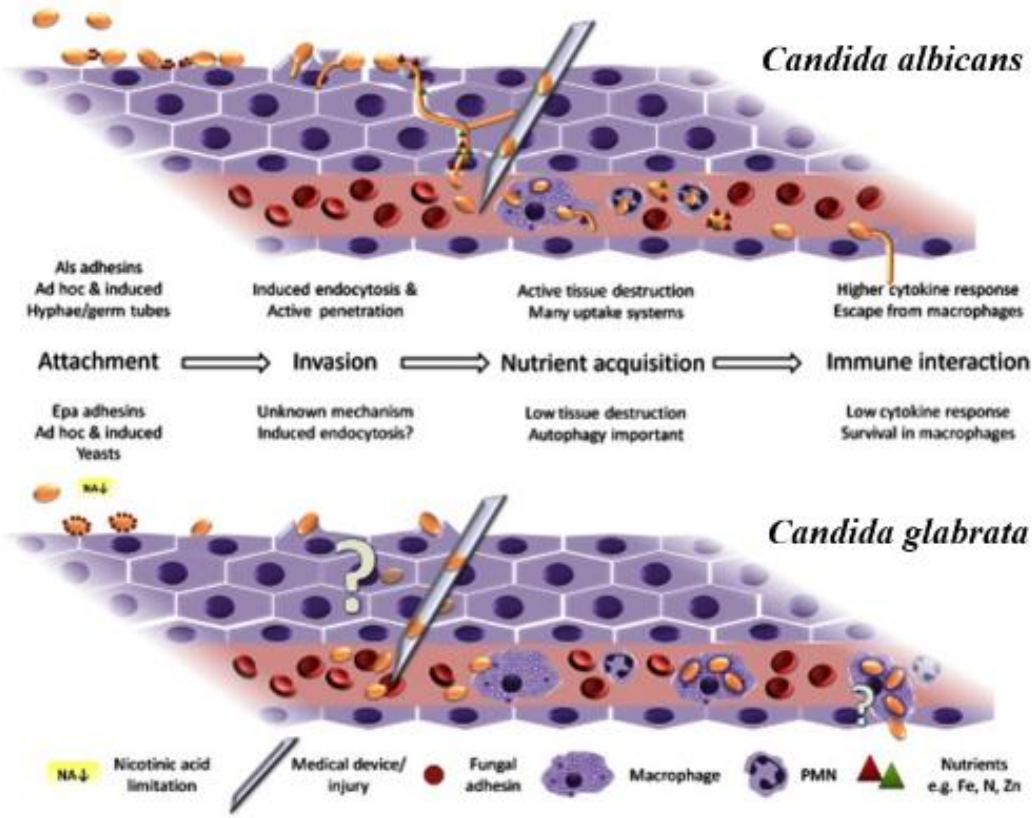


Figure 41 – Représentation schématique des stratégies d'infections causées par C. albicans et C. glabrata. C. albicans (en haut) est capable de changer de forme et devenir un hyphe capable de détruire les tissus, provoquant une forte réponse immunitaire. De nombreux aspects de la pathogénicité de C. glabrata sont encore inconnus, comme le mécanisme précis de l'invasion (en bas). Les lésions tissulaires actives de l'hôte sont faibles, tout comme la réponse immunitaire. Le franchissement de la barrière épithéliale peut se faire par le biais de matériels médicaux (Brunke et al. 2013).

c. Des annotations fonctionnelles des génomes de qualités différentes

Même si les séquences complètes des génomes de *C. glabrata* et *C. albicans* ont été publiées sur la même période, les quantités d'informations fonctionnelles pour chacune sont très différentes. En effet, dans la base de données *Candida Genome Database* (Skrzypek et al. 2017), moins de 5% des gènes de *C. glabrata* ont leur fonction vérifiée, alors que plus de 25%

[140] Élévation anormale de la température du corps, accélération des rythmes cardiaque et respiratoire, rigidité musculaire, etc.

des gènes de *C. albicans* ont leur fonction vérifiée. Ces niveaux d'informations fonctionnelles restent très inférieurs à celui de la levure modèle *S. cerevisiae*, pour laquelle près de 80% des gènes ont leur fonction vérifiée (*Saccharomyces Genome Database* (Cherry et al. 2012)). Ces différences sont la conséquence du nombre d'études très variables, publiées chez ces levures (Figure 43). Ainsi, les levures *C. albicans* et *S. cerevisiae* ont longtemps servi d'espèces modèles, respectivement pathogène et non pathogène.

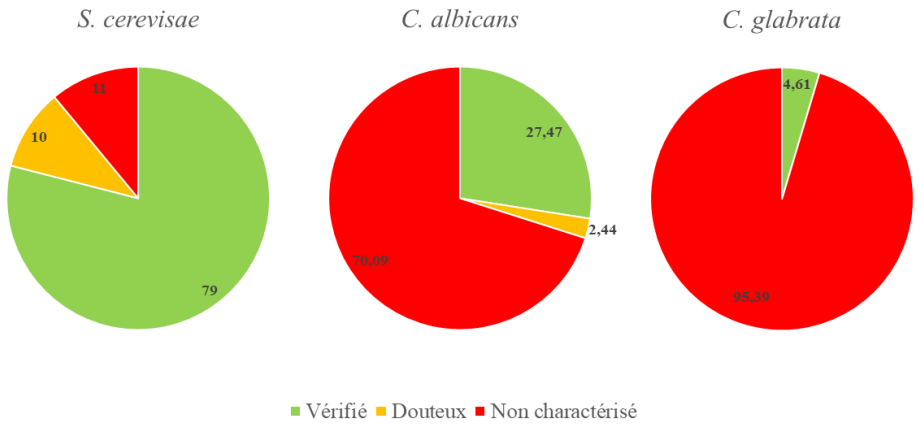


Figure 42 – Aperçu des informations fonctionnelles disponibles chez les levures S. cerevisiae, C. albicans et C. glabrata. La figure a été réalisée à partir de données mises à jour le 06/05/2020.[141] *sur les sites internet des bases de données SGD et CGD.*

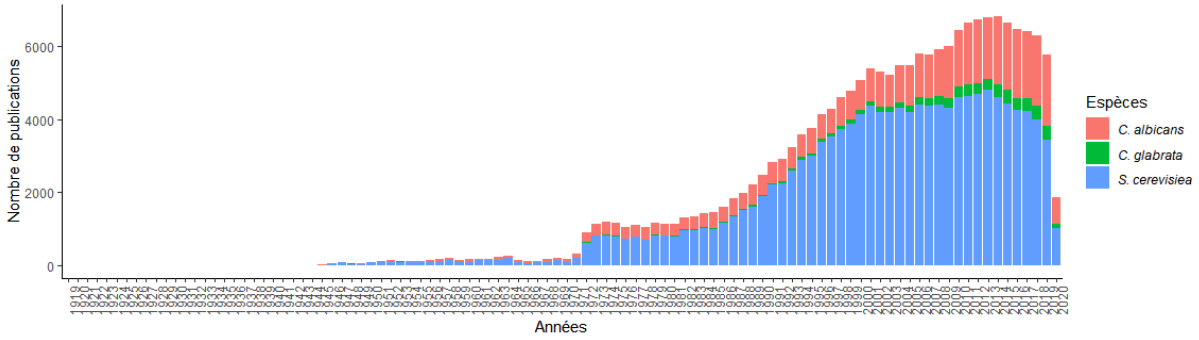


Figure 43 – Évolution du nombre de publications au cours du temps associées aux espèces de levures S. cerevisiae, C. albicans et C. glabrata. Les données ont été obtenues de la base de données PubMed après une recherche du nom de des espèces. La figure a été réalisée le 07/05/2020. Le code pour générer cette figure est disponible sur GitHub Gist.[142].

[141] Les données de *S. cerevisiae* sont disponibles ici : https://www.yeastgenome.org/genomesnapshot , celles de *C. albicans* ici : http://www.candidagenome.org/cache/C_albicans_SC5314_genomeSnapshot.html , et celles de *C. glabrata* ici : http://www.candidagenome.org/cache/C_glabrata_CBS138_genomeSnapshot.html . [Accessible le 07/05/2020]

[142] https://gist.github.com/thomasdenecker/64b44b7e37a1ed2176dd3c1c6cd5b785 [Accessible le 07/05/2020]

Dans ce contexte, l'étude de la levure *C. glabrata* est particulièrement intéressante. Elle a une distance phylogénétique plus faible avec la levure non pathogène *S. cerevisiae* qu'avec la levure pathogène *C. albicans* (Figure 44 et (Dujon 2010; Fitzpatrick et al. 2010; 2006)). L'ancêtre commun de *S. cerevisiae* et *C. glabrata* a subi une duplication entière du génome (WGD) qui a été suivie d'une perte massive de gènes et d'un recâblage important des familles multigéniques (Dujon et al. 2004; Roetzer et al. 2011). Phylogénétiquement, *C. glabrata* appartient au genre *Nakaseomyces* qui est issu de cet évènement WGD (Gabaldón et al. 2013). Dans cette situation, l'émergence de la virulence de *C. glabrata* est une problématique d'intérêt médical (Gabaldón et al. 2016; 2019).

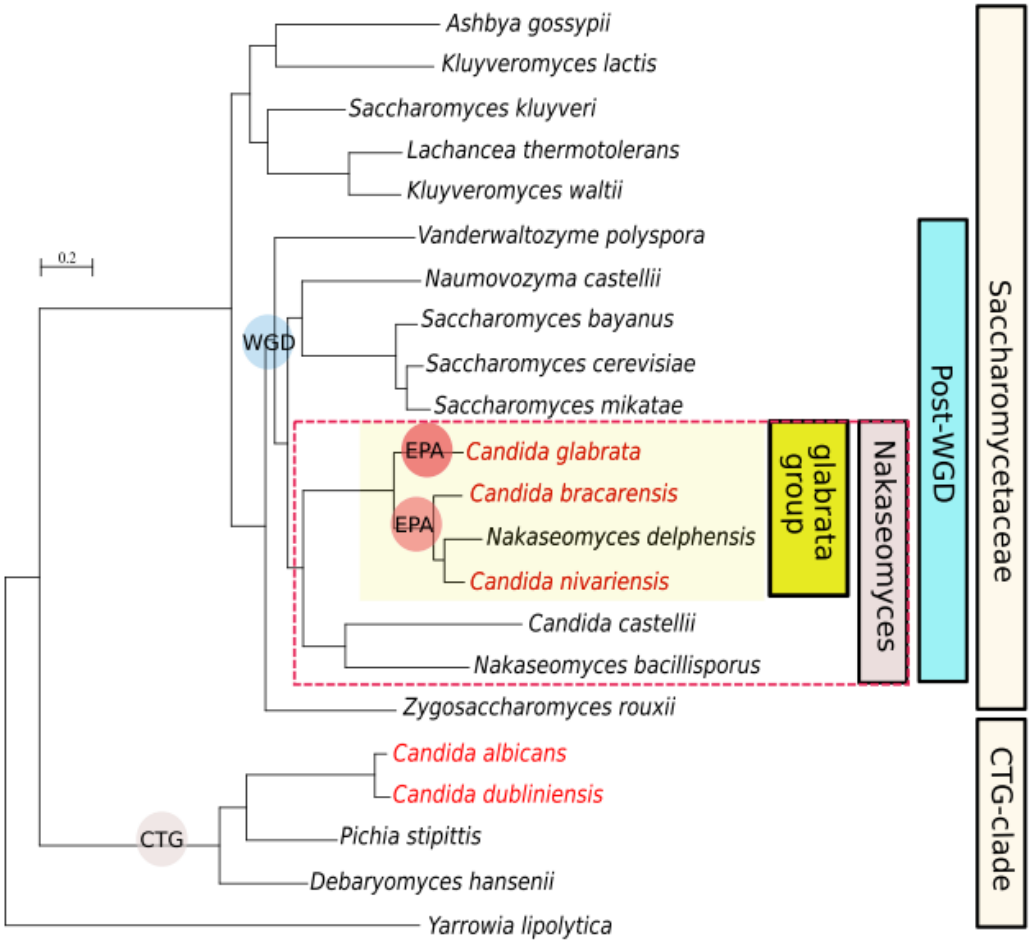


Figure 44 – Relations phylogénétiques entre C. glabrata, les espèces du clade des Nakaseomyces pour lesquelles le génome est entièrement séquencé et les autres espèces du clade des Saccharomycotina. Les espèces pathogènes sont indiquées en rouge. CTG indique un changement du code génétique ; WGD indique la duplication du génome entier ancestral ; et EPA indique la lignée où les deux expansions indépendantes des gènes EPA ont eu lieu. Ces gènes sont importants pour la pathogénicité de C. glabrata. Cette figure est extraite de l'article T. Gabaldón (Gabaldón et al. 2016).

3. Ce qu'il faut retenir

Les études sur le plan fonctionnel menées chez la levure *Candida glabrata* sont plus rares que celles chez *Candida albicans* (levure modèle des levures pathogènes). Néanmoins, elles se justifient pleinement par les nombreuses différences génomiques, morphologiques et fonctionnelles entre ces deux levures succinctement présentées dans ce chapitre. La Figure 45 présente en photo ces deux espèces avec lesquelles j'ai travaillé au cours de ma thèse. La taille de la levure *C. glabrata* est comprise entre 4 et 6 µm. *C. albicans* a une taille de 10 à 12µm sous sa forme levure et variable sous sa forme hyphe. Pour donner un ordre d'idée, sur la Figure 45, la barre grise en bas à gauche de la photographie de la forme hyphe a une taille de 1 mm.

C. glabrata *C. albicans*

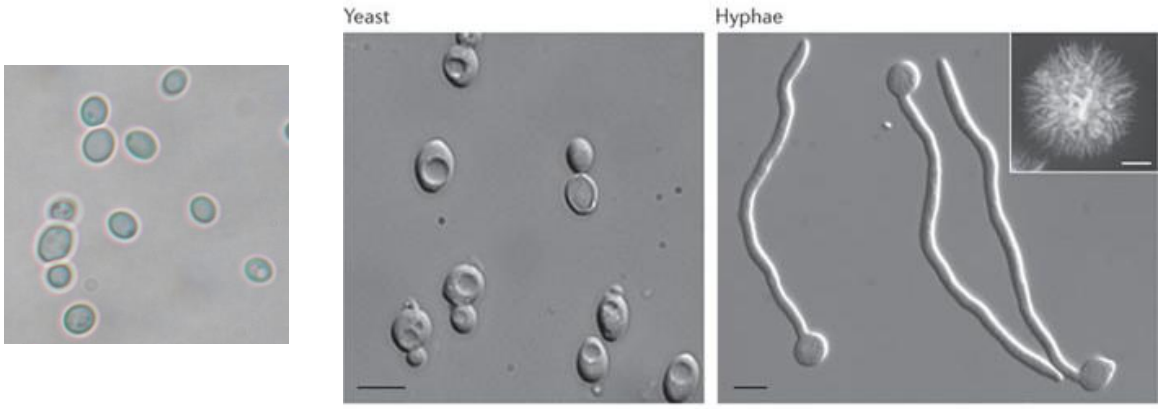


Figure 45 – Photographies des levures C. glabrata et C. albicans. À gauche des cellules de levure C. glabrata cultivées sur milieu Sabouraud obtenue par A. Angoulvant. À droite, des cellules de levure C. albicans avant et après la conversion morphologique (Sudbery 2011).

II. Étude transcriptomique pour l'exploration des mécanismes d'homéostasie du fer chez *Candida glabrata*

1. Le contexte de l'étude

a. Le fer, un élément essentiel pour le fonctionnement des cellules

Le fer est un micronutriment important pour tous les organismes vivants. Il s'agit d'un cofacteur de nombreuses enzymes impliquées dans des mécanismes biologiques cruciaux, par exemple la synthèse et la réparation de l'ADN des cellules, le transport de l'oxygène (le fer rentre dans la composition de l'hème) ou la production d'énergie (Dev et al. 2017; Ganz et al. 2015). Le fer est présent sous deux formes chez les organismes vivants : le fer ferreux (Fe2+) et le fer ferrique (Fe3+). La capacité du fer de passer d'une forme ferreuse réduite (Fe2+) à une forme ferrique oxydée (Fe3+) lui confère une polyvalence d'oxydoréduction qui est utilisée dans de nombreuses réactions où un transfert d'électrons est nécessaire. Cependant, une accumulation intracellulaire de Fe2+ peut être néfaste. En effet, elle entraîne la création de dérivés réactifs de l'oxygène (ROS) par la réaction de Fenton qui, en trop grande quantité, conduisent à la mort cellulaire (Fenton 1894). Par conséquent, si le fer est indispensable, il doit être présent à une concentration précise pour permettre le bon fonctionnement des cellules et un système de régulation est donc indispensable.

Le fer dans l'organisme humain

Le fer est le métal le plus abondant dans l'organisme humain, entre 3 et 5 g chez un adulte en moyenne, soit l'équivalent d'un clou. Il est rarement disponible librement (Ganz et al. 2015). Presque deux tiers des quantités de fer circulent dans le sang en tant que composant de l'hémoglobine dans les érythrocytes (globules rouges). Le fer restant est principalement couplé à la ferritine (30%) ou attaché à de plus petites protéines d'hèmes telles que la myoglobine (3-5%). Les dernières fractions de fer sont fixées à des enzymes héminiques et à la transferrine (voir le tableau ci-dessous).

	Adulte de 60 kg	**Pourcentage**
Hémoglobine	2000 mg	65%
Myoglobine	200 mg	3 à 5%
Enzymes héminiques	10 mg	0.3 %
Transferrine	3 à 4 mg	0.1 %
Réserves : Ferritine et hémosidérine	1000 mg	30 %
	Total = 3 à 5 g	

Tableau 5 – Répartition du fer dans un organisme humain adulte de 60 kg (Abbaspour et al. 2014).

Pour son bon fonctionnement, l'organisme humain doit maintenir cette répartition du fer constante. Un ensemble de mécanismes de régulation existent donc, pour maintenir l'équilibre entre les apports et les pertes en fer. L'absorption quotidienne de fer par l'alimentation *via* les entérocytes duodénaux est très faible (1 à 2 mg par jour) en comparaison des besoins quotidiens en fer, en particulier pour la synthèse d'hémoglobine, qui nécessite environ 20 à 25 mg de fer (Ganz et al. 2012). La différence est comblée par un recyclage important du fer principalement à partir d'érythrocytes endommagés ou sénescents qui ont une durée de vie de 120 jours.

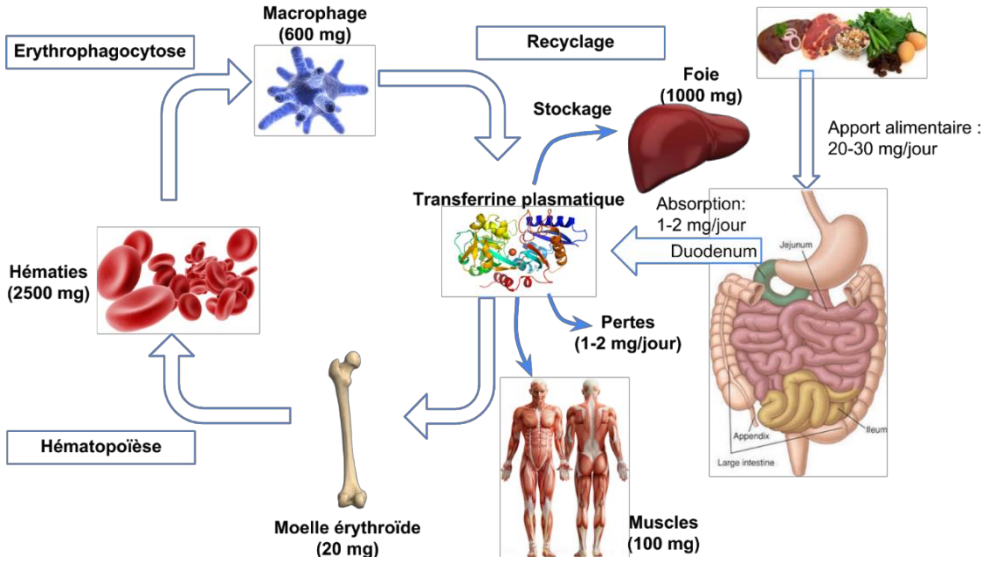


Figure 46 – Illustration schématique des apports et des pertes en fer dans l'organisme humain.

La perte de cet équilibre peut provoquer des pathologies. Un manque de fer peut entraîner une anémie[143] (ferriprive). Cette anémie peut entraîner des céphalées[144], des difficultés respiratoires, une insuffisance cardiaque, etc. À l'inverse, l'accumulation de fer lors d'une hémochromatose héréditaire[145] par exemple, entraîne une fatigue chronique, des douleurs articulaires, une insuffisance cardiaque, etc. Des mécanismes précis sont donc nécessaires pour réguler l'absorption, l'utilisation et le stockage du fer intracellulaire. Il s'agit de l'homéostasie du fer.

Le fer pour les micro-organismes qui vivent dans l'organisme humain

Si le fer est important pour l'organisme humain, il l'est également pour les micro-organismes qui y vivent comme ceux qui composent le microbiote intestinal. En temps normal, les micro-organismes prélèvent du fer dans la lumière du tractus gastrointestinal où il est largement disponible. En effet, la lumière du tube digestif est relativement riche en fer du fait de l'alimentation. Toutefois, en cas d'infection, les micro-organismes se localisent dans des compartiments où la concentration en fer est très variable et où le fer est généralement très peu accessible (Cassat et al. 2013). Pour renforcer cette notion, M. Nairz a titré un de ses articles « La lutte pour le fer - un métal à l'interface hôte-pathogène » (Nairz et al. 2010). Ainsi, le passage d'un milieu riche en fer à un milieu pauvre en fer est une étape importante pour les pathogènes opportunistes tels que la levure *C. glabrata.* Sa réussite dépend de mécanismes d'homéostasie du fer qui permettent de maintenir une concentration en fer intracellulaire compatible avec la survie des cellules de levures.

Défense de l'hôte spécifique au fer : le concept d'immunité nutritionnelle

Dans le corps humain, les ressources en fer sont essentiellement intracellulaires (hématies) ou fortement associées à des protéines de transport (transferrine) ou de stockage (ferritine). Cependant, une petite concentration de fer reste libre dans la circulation sanguine. Étant donnée la dépendance absolue de la plupart des micro-organismes (dont les levures) à l'égard du fer

---

[143] « Diminution de la quantité d'hémoglobine totale fonctionnelle circulante, en corrélation avec une diminution du nombre des hématies. » Dictionnaire médical de l'Académie de Médecine – version 2020.

[144] Maux de têtes.

[145] Pour en savoir plus, le site suivant est très bien fait : https://www.inserm.fr/information-en-sante/dossiers-information/hemochromatose-genetique [Accessible le 06/05/2020]

exogène pour leur survie, des chercheurs ont observé une hypoferrémie provoquée par l'infection et l'inflammation (Ganz et al. 2015) et ont supposé qu'elle avait une fonction de défense de l'hôte. Cette ligne de défense basée sur la séquestration du fer et la restriction de sa biodisponibilité est appelée l'immunité nutritionnelle (Weinberg 1975). L'immunité nutritionnelle liée au fer repose principalement sur 3 éléments :

- La lactoferrine - La lactoferrine est une glycoprotéine de la famille des transferrines. Elle possède une très grande affinité pour le fer ce qui lui confère des propriétés bactériostatiques, bactéricides et fongicides. Lors d'une inflammation, les granulocytes (globules blancs non spécifiques) délivrent des lactoferrines sur la zone infectée. Chez les nourrissons, l'apport en fer et une partie de la protection immunitaire se font principalement par le biais de la lactoferrine présente dans le lait maternel (50 fois plus concentré que le lait de vache) (Breakey et al. 2015).
- La sidérocaline – La sidérocaline est une protéine extracellulaire capable de lier les sidérophores[146] fabriqués par des micro-organismes (Ganz et al. 2015). La sidérocaline est produite principalement par les neutrophiles, les macrophages et les cellules épithéliales. En se liant aux sidérophores, elle rend inaccessible le fer aux micro-organismes.
- L'hepcidine – Lors d'une infection, il y a une induction de la production d'hepcidine ce qui conduit à une diminution de la concentration en fer dans le plasma (Nemeth et al. 2004; Ganz et al. 2012). L'hepcidine entraîne une diminution des concentrations plasmatiques de fer en inhibant l'absorption du fer et en favorisant la séquestration du fer dans les macrophages (Figure 47). Les macrophages vont ainsi stocker le fer susceptible d'être utilisé par l'agent infectieux en récupérant le fer dans les tissus à proximité (Potrykus et al. 2013) ou retirant de la circulation l'hémoglobine. La récupération et la séquestration du fer par les macrophages semblent avoir une double fonction : priver de fer les micro-organismes envahisseurs et protéger l'hôte des effets

[146] Chélateurs de fer synthétisés et sécrétés par les micro-organismes (dont les levures dans certains cas). Ils ont une très forte affinité pour les ions Fe3+ avec lesquels ils forment des complexes permettant l'internalisation du fer dans la levure. Pour un savoir plus, notamment sur leur fonctionnement, l'article de D. E. Crowley décrit les différents mécanismes d'acquisition du fer par les microorganismes dont les sidérophores (Crowley et al. 1991).

toxiques liés à l'augmentation du niveau de fer. En effet, lors d'une infection et d'une inflammation, l'hémoglobine peut être altérée et va libérer du fer.

Il n'est donc pas surprenant que de nombreux micro-organismes aient développé des mécanismes dont le but est d'échapper à l'immunité nutritionnelle. Ainsi, les micro-organismes pathogènes dont *C. glabrata* sont capables de modifier leurs stratégies d'acquisition du fer à mesure que la maladie progresse.

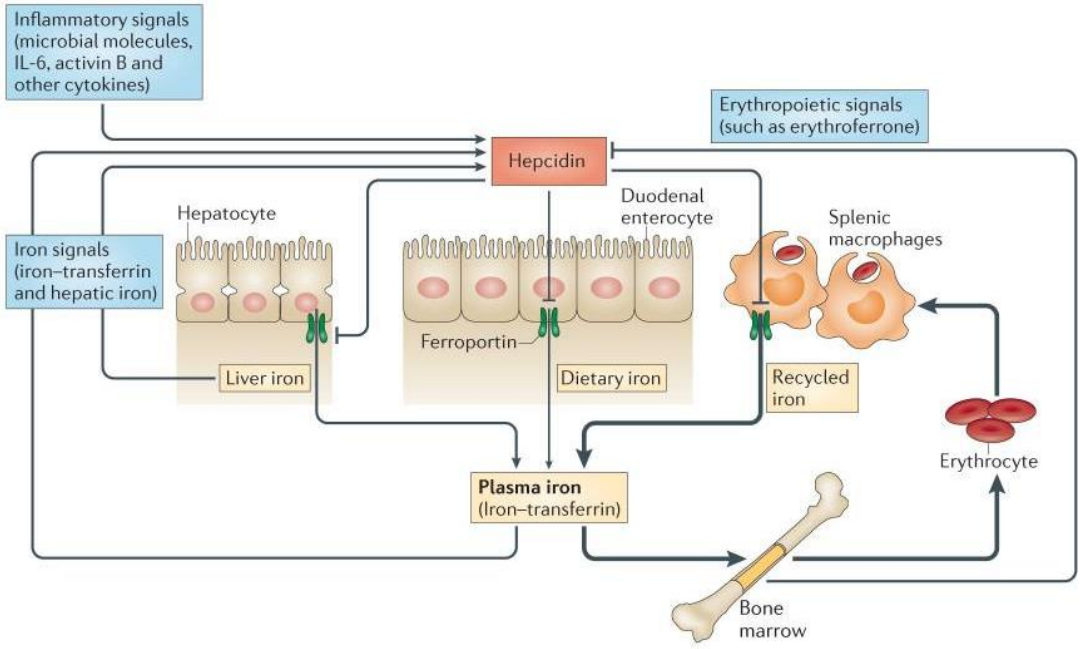


Figure 47 – L'homéostasie du fer et sa modulation par érythropoïèse et inflammation. L'hepcidine bloque les principaux flux du fer dans le plasma (provenant principalement des macrophages spléniques qui recyclent les érythrocytes mais aussi de l'absorption duodénale et des réserves dans les hépatocytes) en provoquant la dégradation de son récepteur, la ferroportine, exportateur de fer. La production d'hepcidine par le foie est régulée à la hausse par l'augmentation des niveaux de fer dans le plasma et des réserves de fer dans le foie. L'infection et l'inflammation stimulent également la transcription du gène codant pour l'hepcidine (Ganz et al. 2015).

b. Mécanismes d'homéostasie du fer chez les levures

Pourquoi l'homéostasie du fer est un processus important ?

Comme nous l'avons vu, lorsque les cellules de levure sont dans la flore digestive, le fer est largement disponible, conduisant la levure *C. glabrata* à réguler l'entrée et le stockage du fer pour éviter un stress oxydatif (réaction de Fenton) (Touati 2000; Higson et al. 1988). En effet, l'absorption du fer chez l'humain passe par son alimentation au niveau du jéjunum et du

duodénum (Figure 46). La lumière du tractus gastrointestinal offre donc à la levure un accès important au fer. À l'inverse, lors d'une candidémie, l'immunité nutritionnelle et la faible présence de fer libre circulant rendent le fer très difficilement accessible. De même, *C. glabrata* est capable de se développer dans les macrophages (Brunke et al. 2013; Kasper et al. 2015) où la concentration en fer est très variable en fonction du type (Nevitt et al. 2011; Seider et al. 2014). Une multiplication très importante de *C. glabrata* dans le macrophage aboutit à son éclatement et à une nouvelle dissémination dans le sang (Figure 48).

Figure 48 – Représentation schématique de la stratégie de C. glabrata pour s'échapper d'un macrophage. Cette levure est surtout connue pour une stratégie de persistance, survivant et prospérant à l'intérieur des macrophages, conduisant finalement à la lyse des cellules immunitaires en raison de la charge fongique (Galocha et al. 2019).

En résumé, *C. glabrata* peut se retrouver dans des environnements différenciés en fer. Elle doit ainsi être capable de mettre en place des stratégies pour survivre et s'adapter à la concentration en fer environnante. L'intensité de ces mécanismes est variable en fonction de sa localisation dans l'hôte et du stade de l'infection. En effet, aux prémices de l'infection, tous les mécanismes de défenses sont mis en place et rendent la disponibilité en fer très limitée. Les gènes impliqués dans l'homéostasie du fer chez *C. glabrata* sont donc différentiellement exprimés en fonction de la disponibilité du fer chez l'hôte.

Stratégies d'adaptation à un environnement différentié en fer

C. glabrata utilise principalement deux stratégies pour survivre dans un environnement différentié en fer :

- Adaptation métabolique – *C. glabrata* réduit sa consommation en fer au strict minimum en ralentissant des mécanismes utilisant le fer (comme en limitant la

création de protéines à centre fer-soufre par exemple), et en accélérant les mécanismes de libération du fer stocké dans la vacuole et les mécanismes de recyclage des protéines contenant du fer (comme les composés héminiques présents dans le cytoplasme) ;

- Adaptation de l'acquisition du fer circulant – En parallèle à l'adaptation métabolique fondée sur l'utilisation du fer intracellulaire, *C. glabrata* met en place des mécanismes pour capter un maximum de fer extracellulaire. Trois principales stratégies sont décrites dans la littérature : (1) l'absorption des sidérophores circulants ; (2) l'absorption d'hèmes et l'absorption réductrice de fer à haute affinité (HA) par la réduction du Fe3+ extracellulaire en Fe2+.

L'ensemble de ces mécanismes sont résumés dans la figure suivante (Figure 49), présentée dans un article de F. Gerwien (Gerwien et al. 2018).

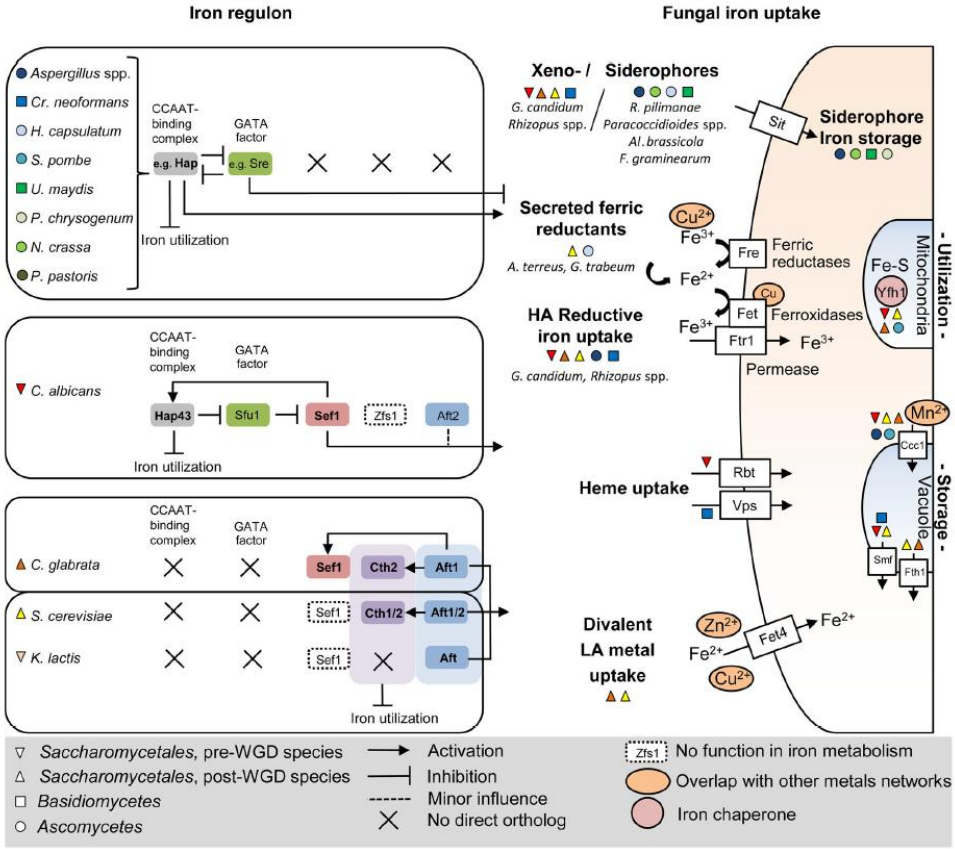


Figure 49 – Homéostasie du fer chez les champignons. La régulation de l'homéostasie du fer (côté gauche de la figure) est indiquée pour différentes espèces de champignons dont la levure C. glabrata (triangle pointant vers le haut orange). Les principaux facteurs de transcription régulés à la hausse pendant la privation de fer pour initier l'absorption du fer fongique (à droite) sont écrits en gras (Gerwien et al. 2018).

Comme nous l'avons vu précédemment, *C. glabrata* est largement moins étudiée que les levures modèles au regard du nombre de publications scientifiques (Figure 43). Cette observation est toujours valable pour les articles en relation avec le fer, avec 40 fois plus d'articles chez *S. cerevisiae* et 10 fois plus chez *C. albicans* que chez *C. glabrata*. Cependant, les articles disponibles nous permettent de définir une première liste de gènes liés au fer. Le tableau suivant regroupe les gènes les plus emblématiques trouvés dans la littérature et décrits comme impliqués dans l'homéostasie du fer chez *C. glabrata* (Tableau 6).

Fonctions	**Gènes**	**Publications**
Absorption du fer *via* les sidérophores	*SIT1*	(Srivastava et al. 2014; Nevitt et al. 2011; Gerwien et al. 2016)
Activité ferro-réductase	Absente	(Gerwien et al. 2017)
Activité hémolytique	Présente	(Luo et al. 2001; 2004; Rossoni et al. 2013; Seneviratne et al. 2016; Furlaneto et al. 2018; Nevitt et al. 2011)
Autophagie	*ATG1, ATG17, ATG19, ATG20, ATG21, ATG41*	(Nagi et al. 2016)
Cluster fer-soufre	*ISU1, ISU2, MGE1, SNQ1, GRX5*	(Devaux et al. 2019)
Hème oxygénase	*HMX1*	(Gerwien et al. 2016)
Métalloprotéines	*SDH2, CCP1, RIP1, CYT1, RLI1, ILV3, LIA1, CYC1, GLT1, MET5, YHB1, LEU1*	(Devaux et al. 2019)

Fonctions	Gènes	Publications
Mitochondrie	*ATG32, MGE1, SSQ1, SSC1, AAC3, POR1, HEM1, HEM14, HEM15, HEM25, YCM1 ISU1/2, ISA1/2, IBA57, NFU1, BOL1/3, JAC1, SSQ1, MEG1, MRS3/4, MNT1/2, ATM1*	(Devaux et al. 2019)
Mitophagie	*ATG3, ATG8, ATG11*	(Nagi et al. 2016)
Réducteur de la consommation en fer	*CTH2*	(Gerwien et al. 2016)
Régulation transcriptionnelle	*AFT1, AFT2, YAP5, GRX3, GRX4, HAP2, HAP3, HAP4, HAP5*	(Srivastava et al. 2014; Thiébaut et al. 2017; Devaux et al. 2019)
Respiration	*COX1, COR1*	(Thiébaut et al. 2017; Devaux et al. 2019)
Stérol	*ERG11, DAP1, AUS1, KES1, TIR3, CRS1*	(Thiébaut et al. 2017)
Stockage	*YFH1, FTH1, FET5, CCC1*	(Srivastava et al. 2014)
Spécifique à la surcharge	Voie PKC *KDX1*, *MKK2*, *ROM2*, *SLG1* et *RLM1* Voie HOG (*PBS2, SHO1, STE20* et *STE50* Sous-unité de régulation de PI3K *VPS15* Voie de signalisation de la calcineurine *BCY1, CCH1, CMP2, CNB1, CRZ1, ECM7*	(Sharma et al. 2016; Schwarzmüller et al. 2014)

Fonctions	Gènes	Publications
Stress ferrique	*GRX4, ISA1, RLI1, HEM3, TYW1, GLT1, CCC1, ACO1, SDH2*	(Thiébaut et al. 2017)
Transport et absorption	Faible affinité *FET4, SMF1, SMF2, CCC2, ATX1, CTR2* Haute affinité *FTR1, FET3, CCC2, FRE6, HOG1*	(Srivastava et al. 2014; Gerwien et al. 2016)
Utilisation de fer spécifique à l'hôte	*HMX1, CCW14, MAM3*	(Srivastava et al. 2014)
Vacuole	*FRE6, FRE8, FET5, FTH1, SMF3, CCC1, CTR2*	(Devaux et al. 2019)
Virulence	*VPS34, ATG32*	(Devaux et al. 2019)

Tableau 6 – Tableau récapitulatif d'une liste de gènes décrits dans la littérature comme impliqués dans l'homéostasie du fer de C. glabrata.

Cette liste est complétée par deux figures proposées par F. Devaux en 2019 qui regroupent les gènes décrits comme impliqués dans l'homéostasie du fer et pour lesquels des données expérimentales sont disponibles (Figure 50, une pour le manque fer (A) et une pour l'excès de fer (B)). Comprendre comment les levures pathogènes adaptent leur homéostasie du fer dans des conditions de manque et d'excès en fer pourrait déboucher sur des stratégies thérapeutiques innovantes.

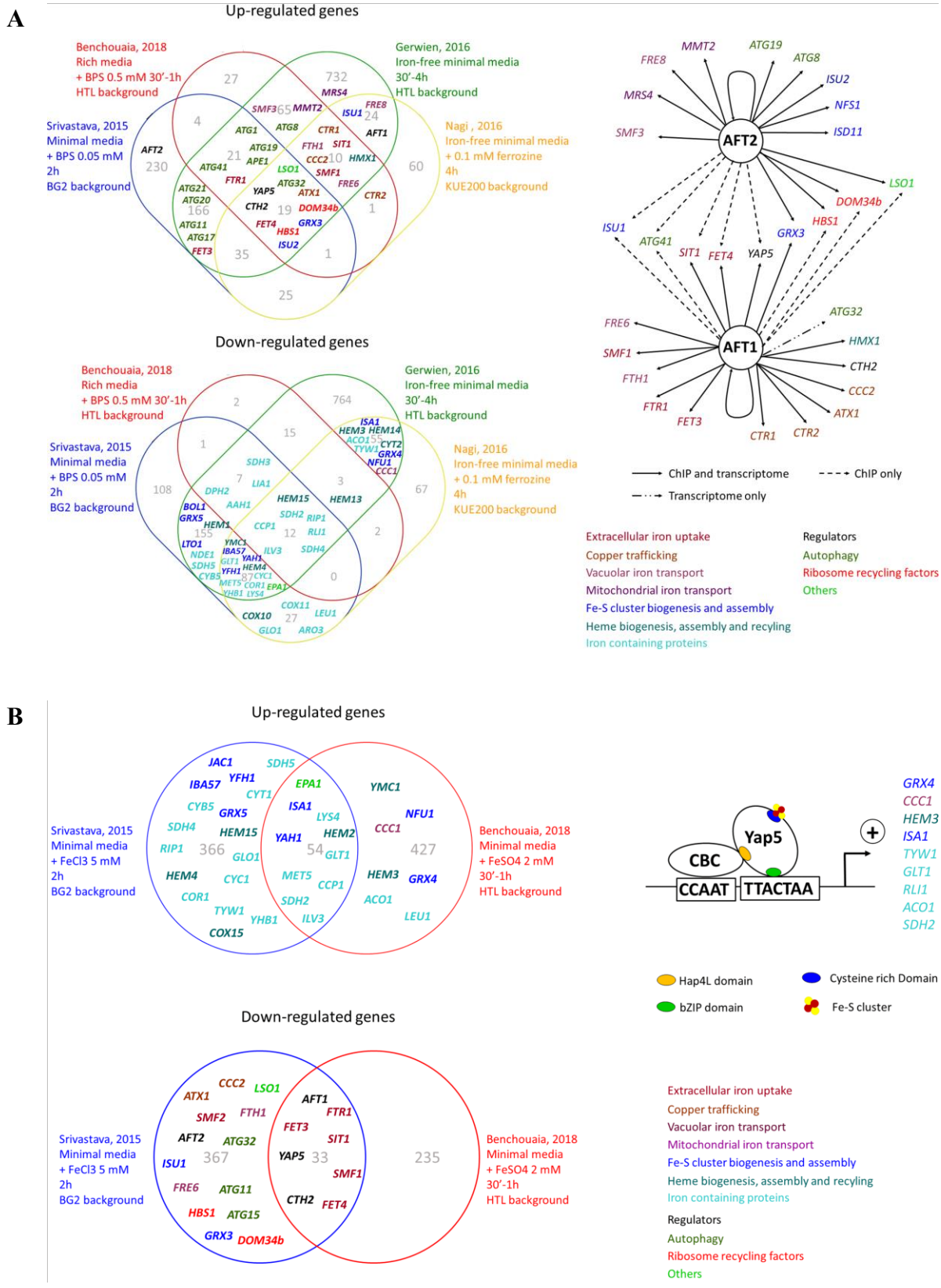


Figure 50 – Figures récapitulatives des gènes impliqués dans l'homéostasie du fer chez C. glabrata et décrits dans la littérature. La figure A est en condition de carence en fer et la figure B en condition de surcharge en fer. La partie gauche

correspond à des diagrammes de Venn. Chaque cercle représente un ensemble de gènes réagissant à la condition étudiée (carence ou surcharge) dans un jeu de données disponible dans la littérature. Le croisement entre deux cercles (deux expériences) correspond à un ensemble de gènes identifiés comme réagissant dans la condition étudiée dans deux expériences. La partie droite des figures correspond aux différents facteurs de transcription identifiés et les gènes avec lesquels ils interagissent. Ces deux figures sont extraites et expliquées en détails dans l'article de F. Devaux (Devaux et al. 2019).

2. La mise en application de la technologie des puces à ADN

a. Type de puces à ADN utilisées

La technologie des puces à ADN permet d'étudier la composition du transcriptome. Pour rappel, le transcriptome correspond à l'ensemble des transcrits présents dans une cellule à un moment donné et dans des conditions données. Ainsi les puces à ADN permettent d'obtenir des mesures de l'expression des gènes dans des conditions particulières.

Dans ce travail, nous avons utilisé la technologie « double marquage »[147]. Les puces à ADN utilisées pour cette expérience ont été fabriquées avec eArray[148] d'Agilent Technologies. Elles sont décrites dans la base de données Gene Expression Omnibus (GEO[149]) sous la rubrique "Platform GPL27653". Le Tableau 7 présente le nombre de caractéristiques chromosomiques décrites dans la base de données CGD (janvier 2020[150]), pour lesquelles il y a au moins une sonde utilisée sur les puces à ADN de notre étude.

Type de caractéristiques chromosomiques	Décrits dans la CGD	Avec des sondes dans sur la puce	Pourcentage
centromere	11	0	
long_terminal_repeat	4	0	
ncRNA|Uncharacterized	66	0	
ncRNA|Verified	2	0	

[147] Une présentation de toutes les étapes expérimentales est disponible sur la ressource numérique de la thèse : https://thomasdenecker.github.io/thesisWebsite/annexes/microarray/ ainsi que ses différents biais https://thomasdenecker.github.io/thesisWebsite/annexes/biaisPuces/ [Accessible le 08/08/2020]

[148] https://earray.chem.agilent.com/earray/ [Accessible le 11/05/2020]

[149] https://www.ncbi.nlm.nih.gov/geo/ [Accessible le 11/05/2020]

[150] http://www.candidagenome.org/download/chromosomal_feature_files/C_glabrata_CBS138/ [Accessible le 11/05/2020]

Type de caractéristiques chromosomiques	Décrits dans la CGD	Avec des sondes dans sur la puce	Pourcentage
ORF|Merged/Split|Uncharacterized	7	0	
ORF | Uncharacterized	5044	4955	**98.24**
ORF | Verified	242	239	**98.76**
pseudogene	18	0	
repeat_region	2	0	
rRNA | Uncharacterized	6	0	
tRNA | Uncharacterized	230	0	
Somme	**5632**	**5194**	

Tableau 7 – Nombre de caractéristiques chromosomiques pour lesquelles des sondes sont trouvées sur les puces à ADN utilisées dans cette étude. "Uncharacterized" signifie que les ORF ont été prédits sur la base de l'analyse des séquences mais qu'elles manquent de caractérisation expérimentale et "Verified" signifie qu'il existe des preuves expérimentales pour un produit fonctionnel.

Presque tous les cadres ouverts de lecture (ORF) signalés comme "non caractérisés" ou "vérifiés" sont étudiés (plus de 98%). Nous avons donc une très bonne couverture du transcriptome. À noter que les autres caractéristiques chromosomiques comme les centromères, les longues répétitions terminales, les ARNs non codant, les pseudogènes, les régions répétées, ARNrs et les ARNts ne sont pas étudiées. Par conséquent, elles ne sont pas présentes dans les fichiers de résultats.

b. Plan expérimental et traitement des données brutes

Avec les puces à ADN « double marquage », deux approches permettent de comparer les niveaux d'expression des gènes entre des conditions biologiques :

- L'approche directe – Les deux échantillons biologiques à comparer sont co-hybridés sur la même puce à ADN et le rapport obtenu fournit une mesure directe de l'expression relative ;
- L'approche indirecte – Chaque échantillon d'intérêt est co-hybridé avec un échantillon de référence commun, utilisé pour chaque puce à ADN de l'expérience (nommé plan de référence). Les rapports obtenus à partir de chaque puce à ADN

dans l'expérience peuvent ensuite être comparés pour identifier les gènes différentiellement exprimés.

L'un des avantages du plan de référence (approche indirecte) est qu'il permet d'élargir facilement une étude pour y inclure un nombre quelconque d'échantillons biologiques, à condition que le même échantillon de référence soit utilisé sur chaque nouvelle puce à ADN. Cependant, son utilisation augmente la variabilité dans les données par rapport au modèle direct. Dans le cadre de cette étude, le modèle expérimental utilisé était indirect (Figure 51). Ce choix avait été fait précédemment à mon arrivée en thèse.

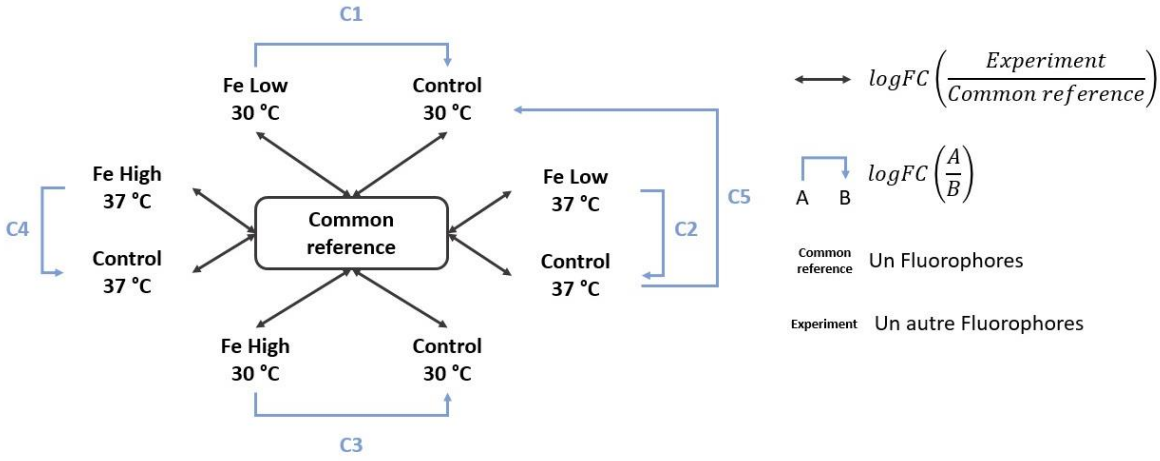


Figure 51 – Représentation du plan d'expérience utilisé dans l'étude de l'homéostasie du fer de la levure C. glabrata, à partir d'une référence commune. La condition C1 correspond à une étude dans un milieu pauvre en fer à 30 °C, C2 à un milieu pauvre en fer à 37 °C, C3 à un milieu riche en fer à 30 °C, C4 à un milieu riche en fer à 37 °C et C5 l'influence de la température (30 °C vs 37 °C). Ces conditions sont décrites en détail dans notre article inséré ci-dessous (page 149) (Denecker et al. 2020).

La souche sauvage de *C. glabrata* utilisée est la souche type[151] ATCC 2001 (CBS 138), décrite dans l'article de T. Gabaldón (Gabaldón et al. 2013). Toutes les conditions de culture ont été réalisées en YPD à 30°C ou 37°C pendant 4 h (phase logarithmique). Les cultures initiales ont été réalisées dans des conditions standards (indiquées comme « Control », Figure 53). L'ajout d'acide BathoPhénanthrolinediSulfonique (BPS, SIGMA-ALDRICH® France), un chélateur[152] du fer, a été utilisé pour engendrer une condition de carence en fer (BPS dans Figure 53). La dose a été choisie sur la base d'expériences préliminaires montrant qu'elle affecte mais n'arrête

[151] Première souche isolée et décrite de l'espèce.

[152] Ligand pouvant former un complexe avec un cation métallique (ici le fer).

pas la croissance des levures. L'ajout de $FeSO_4$ heptahydraté (Sigma-Aldrich®, France) a été utilisé pour entraîner un excès de fer ($FeSO_4$ dans Figure 53). Encore une fois, la dose a été déterminée dans des expériences préliminaires comme la concentration la plus élevée qui affecte mais n'arrête pas la croissance des levures.

Toutes les conditions de cultures cellulaires (carence ou excès en fer, à 30 ° C ou 37 ° C) ont été réalisées trois fois, à partir de précultures indépendantes. La normalisation inter-canaux des fluorochromes (Cy5 et Cy3) a été réalisée à l'aide du package R ManGO.[153]. Afin de contrôler la qualité des données, obtenues pour chacun des réplicats d'expériences, une analyse en composante principale.[154] (ACP) a été réalisée (Figure 52). Une bonne corrélation est observée entre les réplicats biologiques d'une même expérience. A noter que pour une expérience, un réplicat est manquant (BPS37-3), il avait été pointé comme « non conforme » par la personne en charge de l'analyse d'image des puces (travail effectué avant ma thèse).

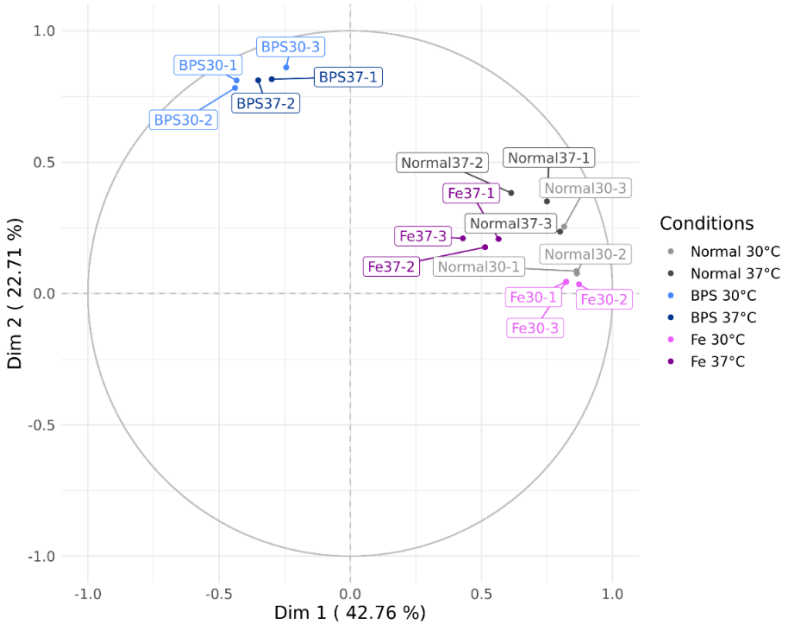


Figure 52 – Analyse en composante principale de l'ensemble des données de puces à ADN utilisées dans cette étude de l'homéostasie du Fer chez la levure pathogène C. glabrata. Le code pour réaliser cette figure est disponible sur *GitHub Gist*.[155].

[153] Méthode de normalisation par régression locale (loess). Cette méthode est recommandée pour ce type de données (Grant et al. 2007)

[154] Méthode d'analyse de données multivariées permettant de résumer et de visualiser un grand nombre de données quantitatives. Pour en savoir plus, François Husson propose une série de 3 vidéos (https://youtu.be/KrNbyM925wI [Accessible le 07/08/2020])

[155] https://gist.github.com/thomasdenecker/08b7f407f892179f0cae2443dd397392 [Accessible le 03/04/2020]

À partir de ces expériences, 5 grands axes ont été étudiés : l'impact d'une carence en fer à 30°C (C1) et à 37°C (C2), l'impact d'une surcharge en fer à 30°C (C3) et 37°C (C4) et l'influence du changement de température (30°C vs 37°C) pour la culture cellulaire en présence d'une concentration optimale en fer (C5). Un récapitulatif des différentes comparaisons expérimentales est proposé Figure 53.

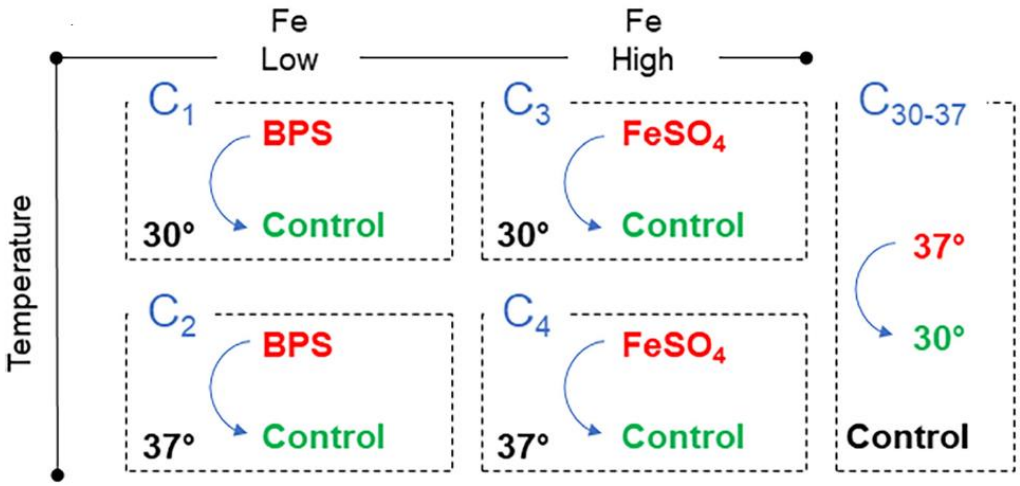


Figure 53 – Représentation schématique des différentes conditions expérimentales utilisées pour comparer l'abondance de l'ARNm avec la technologie des puces à ADN. Cinq conditions nommées C1, C2, C3, C4 et C30-37 ont été définies. Des valeurs de Z-Score ont été calculées pour chaque condition, en comparant les échantillons écrits en rouge (respectivement, "BPS", "FeSO4" et "37°") aux échantillons écrits en vert (respectivement, "Control" et "30°").

Pour identifier les gènes dont l'expression a été modifiée de manière significative en réponse à l'ajout de BPS (condition d'appauvrissement en fer) ou à l'ajout de FeSO4 (condition d'excès de fer), des modèles linéaires pour les données de puces à ADN (package R LIMMA (Ritchie et al. 2015)) ont été appliqués. Cet algorithme a été choisi parce qu'il s'agit de l'outil de référence pour l'analyse différentielle des puces à ADN. De plus, il permet de prendre en compte l'utilisation d'une référence commune[156] et de réaliser une analyse globale intégrant simultanément toutes les expériences. Les paramètres par défaut ont été utilisés lors de l'exécution de l'algorithme. Les gènes exprimés de manière différentielle ont été sélectionnés

[156] Ce cas correspond à un chapitre détaillé dans la documentation https://bioconductor.org/packages/release/bioc/vignettes/limma/inst/doc/usersguide.pdf [Accessible le 07/08/2020]

sur la base d'une valeur P ajustée inférieure à 1% et d'une valeur absolue de *Z-score* supérieure à 1,5.

c. Objectifs de l'analyse des données

Objectif 1 : identifier spécifiquement les gènes impliqués dans l'homéostasie du fer

À partir de ce plan expérimental, il nous était possible d'étudier l'influence de milieux respectivement pauvres (C1 et C2) et riches en fer (C3 et C4) sur la composition du transcriptome de *C. glabrata* (Figure 53). Notre hypothèse était que les gènes dont le niveau d'expression est affecté par le changement de la composition environnementale en fer étaient de bons candidats pour être impliqués dans l'homéostasie du fer de la levure. Dans notre étude, les gènes ont été considérés comme réagissant au fer[157] s'ils étaient observés comme différentiellement exprimés dans au moins une des quatre conditions expérimentales présentées Figure 53. Ainsi, nous avons obtenu une première liste de 637 gènes. Les gènes clés de l'homéostasie du fer (nommés iHKG[158]) constituaient des sous-ensembles de gènes réagissant au fer de façon un peu particulière puisqu'ils étaient dérégulés à la fois dans les conditions pauvres en fer (C1 ou C2) et riches en fer (C3 ou C4). Ces gènes avaient des fonctions importantes pour la cellule, avec pour rôle de contrebalancer les fluctuations externes de la disponibilité du fer, dans n'importe quel sens (diminution ou augmentation). À cet égard, nous avons défini la « dérégulation de Type I » qui regroupe des iHKG avec une dérégulation opposée dans des conditions de fer faibles et élevées et la « dérégulation de Type II » qui regroupe les iHKG avec une dérégulation constante (ou parallèle) dans des conditions de fer faibles et élevées (Figure 54).

[157] Appelés i*ron responsive genes* dans l'article publié.
[158] Cela signifie *iron Homeostasis Key Genes*.

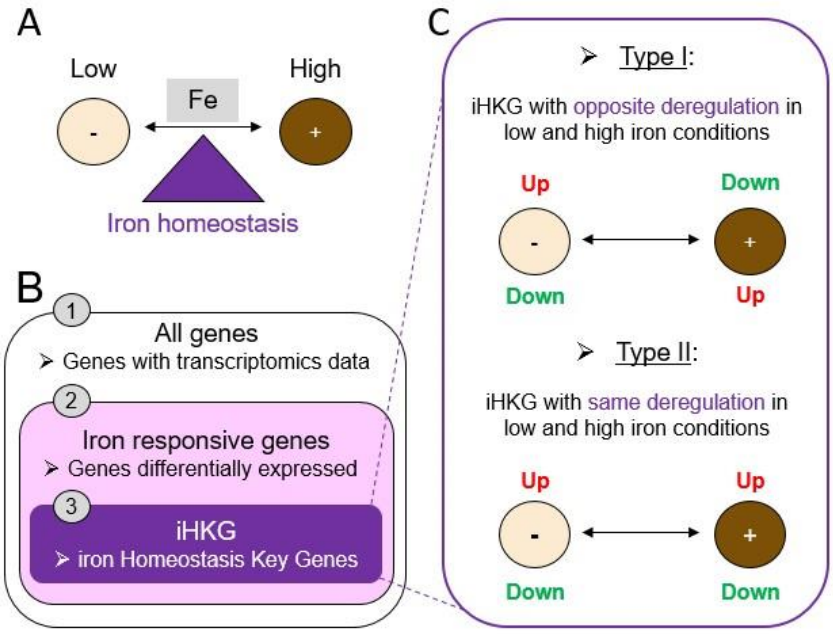


Figure 54 – Définitions et hypothèse de travail extraites de notre article (Denecker et al. 2020). *(A) Représentation schématique des changements extracellulaires de la disponibilité du fer, auxquels est confrontée la levure pathogène C. glabrata. « Low » signifie que la disponibilité en fer est faible, la cellule doit s'adapter à la carence en fer. « High » signifie que la disponibilité en fer est élevée, la cellule doit s'adapter à la surcharge en fer. L'homéostasie du fer est représentée ici comme le processus physiologique central qui permet de maintenir un environnement intracellulaire dans un état constant d'équilibre en fer, malgré les changements extérieurs. (B) Trois classes de gènes ont été étudiées dans cet article. La première classe "all genes" se réfère à tous les gènes pour lesquels nous avions des données. La deuxième classe, les "iron responsive genes", désigne les gènes pour lesquels des changements d'expression sont observés dans au moins une des expériences de transcriptomique. Enfin, la troisième classe "iHKG" fait référence à un nouvel ensemble de gènes ayant des fonctions particulières importantes pour la cellule afin de contrebalancer les fluctuations externes de la disponibilité du fer, dans n'importe quelle direction (faible ou forte). (C) Représentation schématique des deux types de iHKG sur la base des dérégulations observées dans notre ensemble de données, respectivement dans des conditions de fer "Low" (-) et "High" (+). Les "types I" sont des iHKG avec des dérégulations opposées en cas de fer faible et élevé alors que les "types II" sont des iHKG avec une dérégulation constante (ou parallèle) dans des conditions de fer faible et élevé.*

Objectif 2 : replacer les gènes iHKG de Type I et II au sein de grandes fonctions cellulaires

Pour comprendre l'intérêt biologique des gènes iHKG de Type I et II, la pertinence de cette liste de gènes a été évaluée. Pour cela, nous avons parcouru les descriptions fonctionnelles associées aux gènes (disponibles dans les bases de données CGD et GRYC) et recherché des termes GO enrichis. Cette approche a conforté l'intérêt de la liste par la mise en évidence de gènes et de fonctions conformes aux connaissances actuelles des mécanismes d'adaptation des cellules de levure à la carence ou à la surcharge en fer. Pour aller plus loin, nous avons souhaité

déconnecter notre exploration des gènes réagissant au fer des connaissances biologiques dont nous disposions déjà. Dans ce contexte, les méthodes de visualisation de données ont été très utiles. Elles nous ont aidé à changer notre perception des données et à imaginer de nouvelles analyses. Nous avons donc décidé de visualiser toute la liste des gènes réagissant au fer sous la forme d'un réseau, en recherchant des propriétés inattendues dans les relations entre ces gènes, ou dans la dynamique globale du transcriptome entre les conditions. Pour cela les relations de co-expression entre les profils d'expression des gènes ont été quantifiées et nous avons obtenu les résultats présentés à la Figure 55. Dans ce réseau, nous avons observé des gènes régulés à la hausse et à la baisse, respectivement dans des conditions de fer faibles ou élevées. Il s'agit du tableau global des modifications du transcriptome qui se produisent dans la cellule, mais pas seulement. En effet, il est important de souligner que dans ce réseau, les gènes co-exprimés sont représentés par des nœuds voisins. Cette propriété est intéressante car les gènes co-exprimés sont généralement considérés comme de bons candidats pour être impliqués dans des processus biologiques communs ou pour être régulés par les mêmes facteurs. Une façon pertinente de tirer avantage de ce réseau a donc consisté à rechercher des gènes non décrits (ou fonctionnellement non annotés), qui étaient dans le voisinage immédiat d'un gène connu comme sensible au fer.

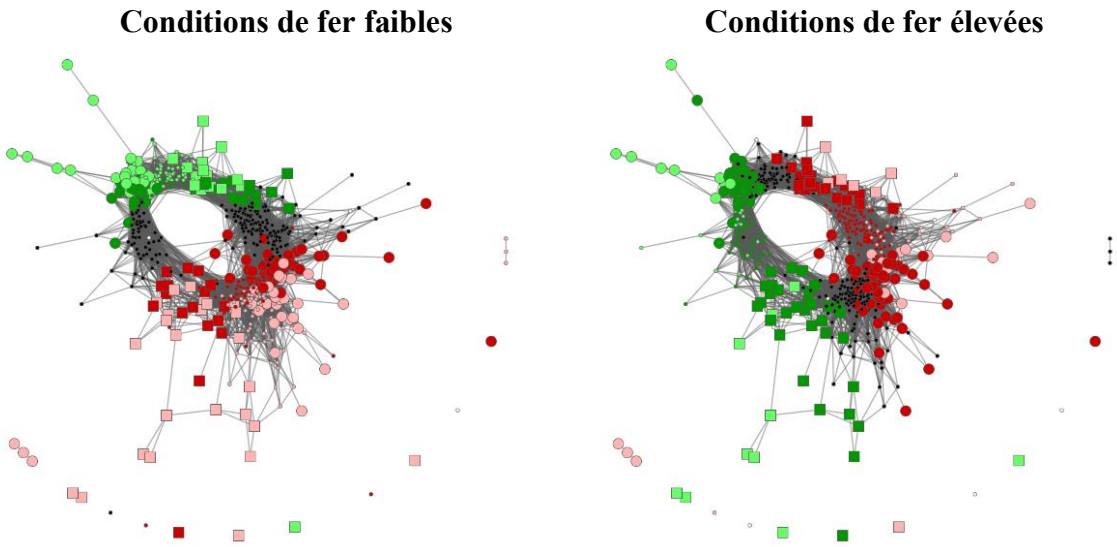


Figure 55 – Réseau de co-expression des 637 gènes réagissant au fer dans les conditions de fer faibles (gauche) et de fer élevées (droite). Chaque nœud correspond à un gène coloré en rouge s'il est surexprimé par rapport à une condition standard ou en vert s'il est sous exprimé. Les carrés sont les iHKG de Type I, les gros ronds de Types II et les petits ronds les autres gènes.

Dans cette optique, nous avons décidé de diviser le réseau de gènes co-exprimés[159] (qui comprenait les 637 gènes réagissant au fer). Nos objectifs étaient (1) de limiter le nombre de gènes dans chaque sous-réseau, facilitant ainsi la visualisation et l'exploration des données et (2) de fournir une image globale des fonctions cellulaires dans lesquelles un recâblage important de l'expression des gènes était observé. Nous avons donc dû définir un nombre limité de catégories fonctionnelles, regroupant l'ensemble des gènes réagissant au fer. Nous avons abouti à six sous-groupes fonctionnels nommés : "*Metabolism*", "*Regulation*", "*Redox Signaling*", "*Transport Trafficking*", "*Iron-Sulfur Cluster synthesis and assembly*" et "*Others*". Pour éviter des interprétations confuses des sous-réseaux fonctionnels associés, nous avons décidé d'attribuer à chaque gène une seule fonction. Le processus d'assignation est décrit en détail dans l'article et résumé dans la Figure 56.

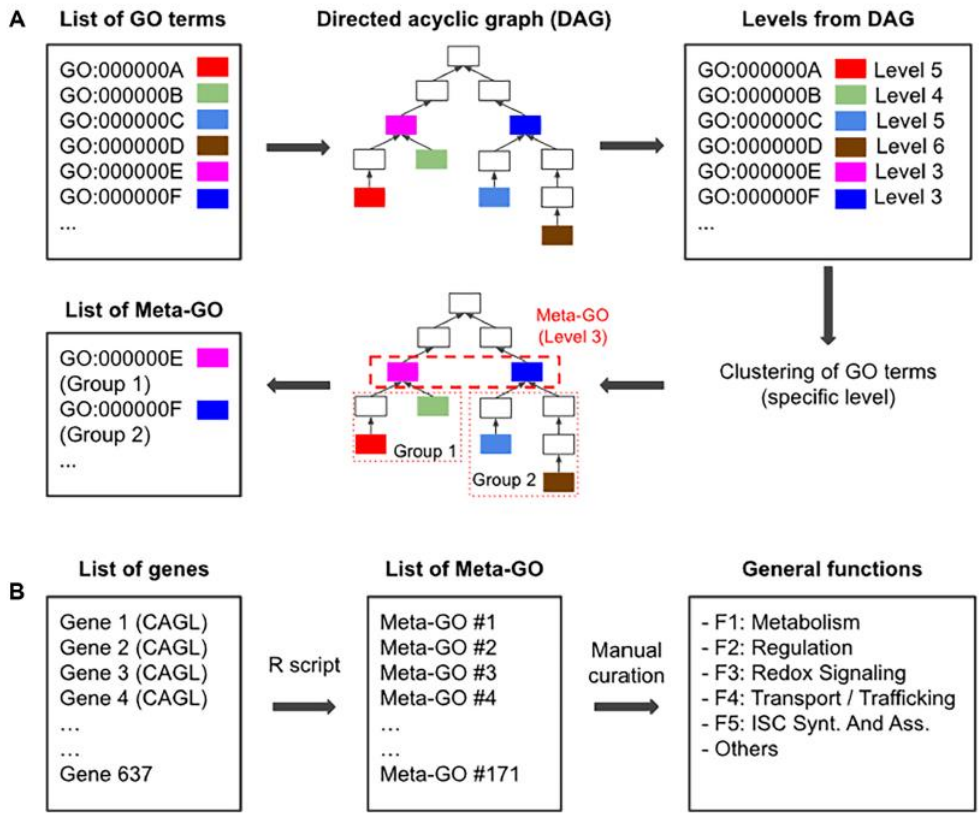


Figure 56 – Exploration des termes GO pour définir un nombre limité de fonctions générales auxquelles tous les gènes réagissant au fer peuvent être assignés. (A) Représentation schématique du processus appliqué pour traiter les termes GO

[159] Ensembles de gènes présentant un niveau d'expression proche dans toutes les conditions étudiées. Cette notion est illustrée à la Figure 4 de notre article (Denecker et al. 2020).

dans cette étude. À partir d'une liste de termes GO (dans notre cas, tous les termes dans "Processus biologique"), nous avons reconstruit le DAG fourni par la base de données GO. Les niveaux sont associés à chaque terme des GO en fonction de leur position dans le DAG (pour plus de détail, voir la section Matériels et Méthodes et la Figure supplémentaire S10 de l'article). Ceci est illustré ici par des cases de couleur. Les termes GO sont ensuite regroupés en fonction d'un niveau (niveau 3 dans cet exemple) dans la hiérarchie des GO. Nous obtenons ainsi des "Meta-GO", c'est-à-dire des groupes de termes GO qui partagent un ancêtre commun dans le DAG. Dans l'analyse présentée dans l'article et dans ce manuscrit, le niveau 4 a été utilisé pour créer les Meta-GO (voir Matériel et Méthodes et les données supplémentaires S3). (B) Fonctions générales définies dans cette étude pour mettre en évidence les rôles physiologiques des gènes réagissant au fer. Elles sont appelées "Metabolism". (F1), "Regulation" (F2), "Redox Signaling" (F3), "Transport / Trafficking" (F4), "Iron sulfur cluster synthesis and assembly" (F5) et "Others".

Il est important de souligner que d'autres choix auraient pu être faits pour la séparation fonctionnelle des gènes. Mais nous avons ici réussi à regrouper une grande majorité des 637 gènes réagissant au fer en seulement 5 fonctions. Notre objectif était avant tout de séparer les gènes sensibles au fer en sous-réseaux biologiquement cohérents pour simplifier leur exploration. Ces choix sont totalement déconnectés de la définition et de l'identification des gènes clés de l'homéostasie du fer (iHKG) présentées dans l'article.

Objectif 3 : proposer des nouvelles voies de recherche

Grâce à cette séparation des gènes en sous-réseaux fonctionnels, nous avons mis en évidence des cas intéressants. Par exemple, nous avons remarqué que le gène codant le facteur de transcription *Hap1* (CAGL0B03421g) était au voisinage immédiat du facteur de transcription *Aft1* (Figure 57). *Hap1* est un facteur de transcription activé par l'hème. Il n'était donc pas surprenant de trouver le gène *HAP1* dans notre liste des gènes clés de l'homéostasie du fer, car la biosynthèse de l'hème nécessite du fer. Par conséquent, nous nous attendions à ce que *Hap1* soit un capteur indirect de fer et nous savons que son activité est affectée par la carence en fer (Ihrig et al. 2010). Bien que le rôle principal de *Hap1* soit globalement conservé chez les différentes espèces de levures, un double rôle peut exister tel que celui décrit pour *Kluyveromyces lactis* (Bao et al. 2008). Nos résultats montrent pour la première fois chez *C. glabrata* que le gène *HAP1* est également dérégulé dans les conditions d'excès de fer. Ces données ne suffisent pas à elles seules pour aller plus loin. Mais cette exploration des réseaux nous a permis de formuler une nouvelle hypothèse de travail intéressante.

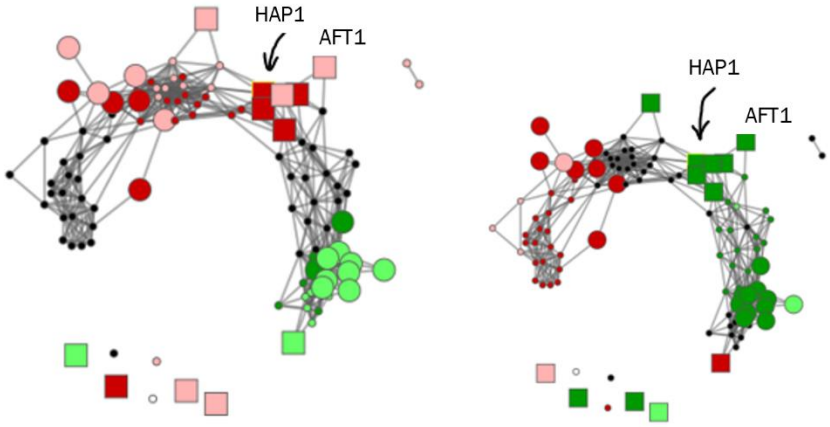


Figure 57 – Exploration des facteurs de transcription dans le sous réseau fonctionnel des gènes co-exprimés "Regulation" (Fer faible à gauche et fer élevé à droite). Le gène codant pour le facteur de transcription Hap1 est au voisinage immédiat du facteur de transcription Aft1.

En résumé, cette étude a permis de dresser une liste de gènes se présentant comme de bons candidats pour faire partie des mécanismes de l'homéostasie du fer chez *C. glabrata*. Elle se base sur l'intégration de plus de 100 000 données hétérogènes, qualitatives et quantitatives (termes GO, données de puces à ADN, annotations de gènes dans des bases de données publiques, etc.). Les gènes que nous avons identifiés permettent de proposer de nouvelles hypothèses de recherche qui, nous l'espérons, s'avéreront fructueuses pour la communauté des levuristes. Ces données sont accompagnées d'un outil de visualisation, iHKG viewer, pour simplifier leur exploration et leur réutilisation. L'ensemble des fonctionnalités de cet outil sont décrites en détail dans l'article inséré ci-dessous (Denecker et al. 2020).

3. Le PDF de l'article publié dans la revue « *NAR Genomics and Bioinformatics* »

Published online 20 April 2020

NAR Genomics and Bioinformatics, 2020, Vol. 2, No. 2 **1**
doi: 10.1093/nargab/lqaa027

Functional networks of co-expressed genes to explore iron homeostasis processes in the pathogenic yeast *Candida glabrata*

Thomas Denecker[1,†], Youfang Zhou Li[2,†], Cécile Fairhead[2], Karine Budin[1], Jean-Michel Camadro[3], Monique Bolotin-Fukuhara[2], Adela Angoulvant[2,4,†] and Gaëlle Lelandais[1,*,†]

[1]Université Paris-Saclay, CEA, CNRS, Institut de Biologie Intégrative de la Cellule (I2BC), 91198, Gif-sur-Yvette, France, [2]Université Paris-Saclay, INRAE, CNRS, Génétique Quantitative et Évolution Le Moulon, 91400, Orsay, France, [3]Université de Paris, CNRS, Institut Jacques Monod (IJM), 75013, Paris, France and [4]Parasitology and Mycology Department, Bicêtre University Hospital, Univ. Paris-Sud/Univ. Paris Saclay, Le Kremlin-Bicêtre, France

Received November 07, 2019; Revised February 27, 2020; Editorial Decision March 30, 2020; Accepted April 06, 2020

ABSTRACT

Candida glabrata **is a cause of life-threatening invasive infections especially in elderly and immunocompromised patients. Part of human digestive and urogenital microbiota, *C. glabrata* faces varying iron availability, low during infection or high in digestive and urogenital tracts. To maintain its homeostasis, *C. glabrata* must get enough iron for essential cellular processes and resist toxic iron excess. The response of this pathogen to both depletion and lethal excess of iron at 30°C have been described in the literature using different strains and iron sources. However, adaptation to iron variations at 37°C, the human body temperature and to gentle overload, is poorly known. In this study, we performed transcriptomic experiments at 30°C and 37°C with low and high but sub-lethal ferrous concentrations. We identified iron responsive genes and clarified the potential effect of temperature on iron homeostasis. Our exploration of the datasets was facilitated by the inference of functional networks of co-expressed genes, which can be accessed through a web interface. Relying on stringent selection and independently of existing knowledge, we characterized a list of 214 genes as key elements of *C. glabrata* iron homeostasis and interesting candidates for medical applications.**

INTRODUCTION

Infections due to *Candida* yeast species cause serious problems in aging populations and patients with compromised immunity, e.g. as a result of cancer treatment, organ transplantation or HIV infection (1, 2). A major cause of morbidity and mortality in healthcare structures (3), the frequency of candidemia and invasive candidiasis is increasing worldwide. While *Candida albicans* is known as the most common cause, the epidemiology varies according to geographical region, the populations involved and the survey period (1,2). In the United States and Northern Europe, *Candida glabrata* has been reported as the second cause of candidiasis (1). *Candida glabrata* is a pathogenic yeast species whose haploid genome was described in 2004 (4). It is composed of 13 chromosomes with ∼5200 genes. Despite its name, *C. glabrata* is phylogenetically more closely related to the model yeast *Saccharomyces cerevisiae* than to the pathogenic yeast *C. albicans* (4). Notably, *C. glabrata* infections remain challenging to treat owing to delayed diagnosis, natural low susceptibility to azole antifungals and acquired resistance to echinocandins (5–7).

During host infection, pathogens face abrupt physiological changes in their immediate environment (8). In this context, promising therapies are expected to emerge from a better understanding of the homeostatic processes used by pathogens to protect (or restore) an internal stability in their cellular functions. A major player is iron homeostasis, as iron bioavailability is a key factor involved in the 'nutritional immunity' host-defense mechanism (9). Remarkably, iron is a two-faced oligo-element for living organisms. On the one hand, iron is essential, as part of heme- and iron-sulfur cluster (ISC)-containing proteins involved in a variety of vital functions including oxygen transport, DNA synthesis, metabolic energy or cellular respiration (see (10) for review). On the other hand, iron is toxic. Its excess triggers oxidative stress, lipid peroxidation and DNA damage that ultimately compromise cell viability and can promote

*To whom correspondence should be addressed. Tel: +33 1 69 82 46 80; Email: gaelle.lelandais@universite-paris-saclay.fr
†Contributed equally.

programmed cell death (11). Iron homeostasis is therefore essential to allow pathogens to maintain a balance between iron utilization, storage, transport and uptake in the host environment.

Molecular mechanisms of iron acquisition and consumption in fungi are well described in the literature (see (12) or (13) for reviews). The yeast species *S. cerevisiae* and *C. albicans* were long considered paradigms for non-pathogenic and pathogenic species, respectively, but the situation is changing. Several articles describe the regulatory mechanisms involved in *C. glabrata* iron homeostasis ((14–21) and see (22) for a comprehensive review). Although *C. glabrata* has conserved the classical fungal iron regulon, it has also remodeled its own functional networks to maintain iron homeostasis. So far, experimental studies have essentially described the impact of iron deficiency on gene expression in *C. glabrata*. Iron deficiency is indeed a relevant system for mimicking infections of the human host, during which access to iron for the pathogen is severely limited by the host defense mechanisms (23,24). Nonetheless, iron deficiency is not representative of all situations to which *C. glabrata* is exposed during its life cycle in the human body. *Candida glabrata* is either a commensal or a colonizer, at least transiently, of the digestive and urogenital tracts. When digestive or urinary epithelium are damaged due to hypoperfusion, medications or invasive procedures, *C. glabrata* can translocate to blood and then disseminate. *Candida glabrata* in the digestive tract faces either high or low iron concentrations depending on dietary sources, gut motility and even microbiota (25). In the urinary tract, *C. glabrata* faces iron concentrations which can be high and increase with the host's age (26,27). In the blood phagocytes, iron concentrations are low (28). The notion of 'homeostasis' is thus particularly important for *C. glabrata* as colonizer since the pathogen must maintain an internal iron balance despite external fluctuations in iron bioavailability in the immediate environment.

The aim of the present work is to specifically study iron homeostasis. In particular, we want to highlight key genes, which are systematically deregulated when *C. glabrata* faces decreased or increased bioavailability of iron. We performed transcriptomic experiments (microarray technology) to monitor gene expression changes of *C. glabrata* to ferrous iron (Fe^{2+}) deficient and overload conditions at 30°C and 37°C. These temperatures are respectively the usual temperature at which *C. glabrata* is cultivated in laboratories and the temperature at which *C. glabrata* develops in the human body. The resulting dataset was analyzed to (i) clarify the potential effect of temperature on iron homeostasis, (ii) identify iron responsive genes, i.e. genes significantly up- or downregulated in at least one iron imbalanced situation and (iii) define a new set of genes, referred to hereafter as 'iron homeostasis key genes' (iHKG, Figure 1). These genes are good candidates to be chief components of iron homeostasis. Our exploration of the datasets was facilitated by the inference of functional networks of co-expressed genes, which can be accessed through a web interface (https://thomasdenecker.github.io/iHKG/). The philosophy of this work is to empower experimental researchers by providing access to all transcriptomics data and by generating easily interpretable graphical outputs. This should facilitate deep exploration of genome-wide functional data in the pathogenic yeast *C. glabrata* to advance our global understanding of iron homeostasis.

MATERIALS AND METHODS

Transcriptome analyses

Yeast strains and growth conditions. The wild-type *C. glabrata* strain used for gene expression analysis under iron deficiency and overload conditions (see below) is ATCC 2001 (CBS 138), as described in (29). Cells were frozen in 40% glycerol at −80°C and used thereafter. They were first cultured on yeast extract-peptone-glucose (YPD) agar plates and then sub-cultured in YPD liquid medium at 30°C, on a rotating shaker (150 rpm) for 24 h, by inoculating 10^7 yeast cells in 10 ml of medium.

Conditions of cell culture for RNA isolation. All culture conditions were performed in YPD, starting from a sample of 2×10^6 cells, inoculated in 10 ml of YPD medium, then cultured under gentle shaking (150 rpm) at 30°C or 37°C for 4 h (log phase, determined from growth curves). Initial cultures were performed in standard conditions (referred to as 'Control', Figure 2 and Supplementary Data S1). Addition of 100 μM bathophenanthrolinedisulfonic acid (4,7-diphenyl-1,10-phenanthrolinedisulfonic acid disodium salt hydrate or BPS, SIGMA-ALDRICH® France) was used as iron deficiency condition (referred to as 'BPS', Figure 2 and Supplementary Data S1). The dose was chosen based on preliminary experiments showing that it affects but does not stop growth of yeasts. Addition of 500 μM of $FeSO_4$ heptahydrate (SIGMA-ALDRICH®, France) was used as iron excess condition (referred to as 'FeSO4', Figure 2 and Supplementary Data S1). Again, this was determined in preliminary experiments as the highest concentration that affects but does not stop yeast growth. Note that the concentration of 500 μM of $FeSO_4$ heptahydrate is a modest excess, close to iron concentrations in the human gut and urine. All conditions of cell culture (iron deficiency or excess, at 30°C or 37°C) were performed three times, starting from independent pre-cultures, to cover biological variations. After cell cultures in standard, iron deficiency or excess conditions (each at 30°C or 37°C), an aliquot of 1 ml with 2×10^7 cells was frozen at −80°C for RNA purification.

RNA preparation, labeling and microarray hybridization. RNAs were isolated using RNeasy Mini Kit for purification of total RNA (QUIAGEN, GmbH, Germany) according to the manufacturer's instructions. Concentration was determined using a Nanodrop 2000 instrument. RNA quality was checked using the Agilent 2 bioanalyzer Nanochip system according to the manufacturer's instructions. Microarrays were manufactured with eArray (https://earray.chem.agilent.com/earray/) from Agilent Technologies. There are described in the Gene Expression Omnibus (GEO) database (30) under 'Platform GPL27653'. Technical information is provided in Supplementary Note S8. Probe preparation, labeling and hybridization were performed with Agilent Technologies according to the manufacturer's instructions. The reference sample is a mixture of RNA extracted from all different growth conditions.

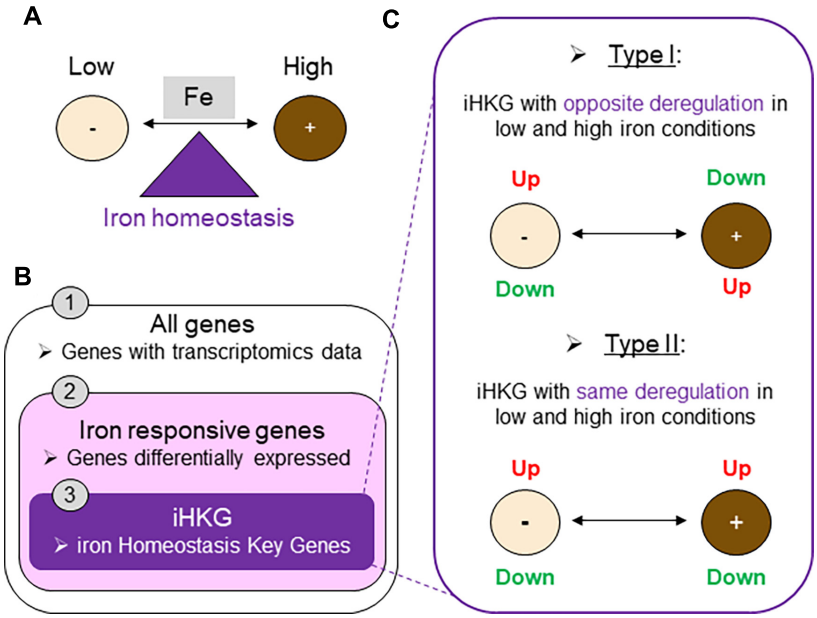


Figure 1. Definitions and working hypothesis. (**A**) Schematic representation of extracellular changes in iron availability, faced by the pathogenic yeast *C. glabrata*. 'Low' means that the iron availability is low, the cell must adapt to iron deficiency. 'High' means that the iron availability is high, the cell must adapt to iron overload. Iron homeostasis is represented here as the central physiological process to maintain an internal cellular environment in a constant state balance, despite the external changes. (**B**) Three classes of genes studied in this article. The first class 'All genes' refers to all the genes for which transcriptomics data is obtained, the second class 'Iron responsive genes' refers to the genes for which expression changes are observed in at least one transcriptomic experiment. At last, the third class 'iHKG' refers to a new set of genes with particular functions important for the cell to counterbalance external fluctuations in iron availability, in any direction (low or high). (**C**) Schematic representation of the two types of iHKG based on the de-regulations observed in our dataset, respectively in 'low' (−) and 'high' (+) iron conditions. 'Type I' are iHKG with opposite deregulations in low and high iron conditions, whereas 'Type II' are iHKG with constant (or parallel) deregulation in low and high iron conditions.

Microarray data acquisition, inter-channel normalization and public access. Microarrays were scanned using the Agilent Array Scanner according to manufacturer's instructions. Pre-processing and inter-channel (Cy5 and Cy3) fluorochrome normalization were performed with MAnGO software (31). Artefactual spots were eliminated from the analysis, duplicated spots were averaged. The data discussed in this publication have been deposited in NCBI's Gene Expression Omnibus (30) and are accessible through GEO Series accession number GSE139363 (https://www.ncbi.nlm.nih.gov/geo/).

Bioinformatics analyses

Source code availability, access to intermediate result files and reproduction of all analyses. All the source code written for this project is available in the Github repository https://github.com/thomasdenecker/iHKG/. The analysis starts with reading the MAnGO data file deposited in GEO (see previous section). This ensures continuity between the work on raw data (standard results of microarrays processed by a technical platform) and the work of data mining and analyses performed for this article (our contribution). To reproduce our data output tables, our selections of genes and our output graphs, a docker image is provided, with RStudio software (https://rstudio.com/) and all the necessary R packages installed: https://cloud.docker.com/repository/docker/tdenecker/ihkg/general. The website to explore the functional networks of co-expressed genes is available at https://thomasdenecker.github.io/iHKG/.

Statistics for differential expression. Our transcriptomic results are from two-color microarray experiments with a common reference, *i.e.* a mixture of RNA extracted from all the other growth conditions (see previous section). To compare gene expression levels between 'BPS' and 'Control' on the one hand, and between 'FeSO4' and 'Control' on the other hand, a design matrix was defined as explained in the LIMMA user guide available online https://bioconductor.org/packages/release/bioc/vignettes/limma/inst/doc/usersguide.pdf. This matrix allows identification in the entire dataset of the different conditions to be compared and the associated biological replicates. Statistics for differential expression can thus be calculated, applying the LIMMA linear model (32). Results include log2 fold changes (logFC), *t*-statistics and adjusted *P*-values (Benjamini–Hochberg correction). Note that statistics for differential expression were calculated independently using results of microarrays obtained either at 30°C or 37°C. This allowed us to obtain statistics for all the genes in four different conditions referred to as C1 (low Fe–30°C), C2 (low Fe–37°C), C3 (high Fe–30°C) and C4 (high Fe–37°C). Gene expression levels were also compared between 30°C and 37°C using 'Control' samples obtained at each temperature (referred to as C30–37, Figure 2). Variability observed in logFC values for C1, C2, C3, C4 and C30–37 conditions was standardized, calculating *Z*-Score

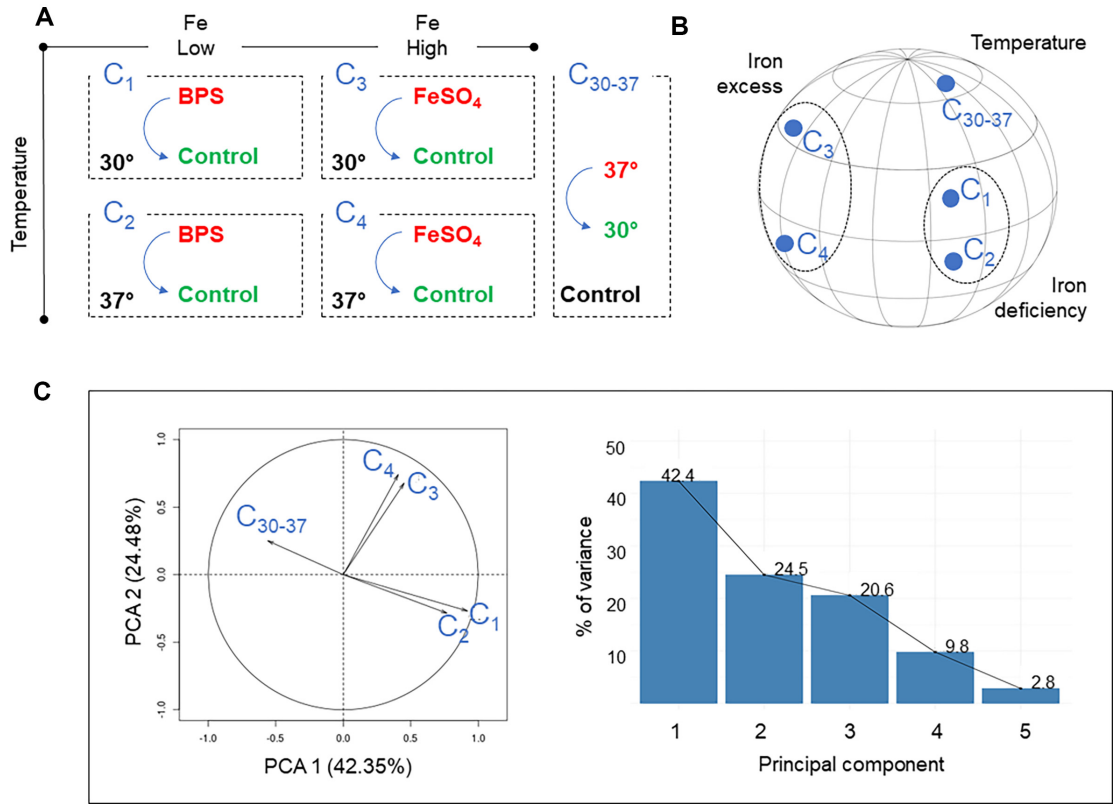


Figure 2. Global analysis of transcriptome changes in iron deficiency or iron overload conditions at 30°C and 37°C. (**A**) Schematic representation of the experimental design used for comparing mRNA abundance with microarray technology. Five conditions named C_1, C_2, C_3, C_4 and $C_{30–37}$ were defined. Z-Score values were derived for each condition, comparing samples written in red (respectively, 'BPS', 'FeSO4' and '37°C') to samples written in green (respectively, 'Control' and '30°C'). (**B**) Spherical representation of the correlation matrix between Z-Score values for the five conditions shown in (A). The smaller the distance between the points, the greater the correlation values. (**C**) Biplot representing the five conditions using the coordinate system defined by principal components 1 and 2 (left), and a histogram showing the percentage of variance represented by each principal component. Together, the principal components 1 and 2 account for more than 66%.

values, i.e. $Z-\text{Score} = \frac{\text{logFC} - \text{mean}}{\text{stand. dev.}}$ where 'mean' is the average value for logFC of all genes in a particular condition (this value is around 0 due to microarray raw data normalization and hence no significant difference is observed between conditions) and 'stand. dev.' is the standard deviation of logFC of all genes. This is the parameter that must be normalized (standard deviation equal 1 in *Z*-Score distribution) to ensure a constant stringency of gene selection in all conditions (see below).

Principal component analysis (PCA). Values of *Z*-Scores for all genes in all conditions were analyzed by principal component analysis (PCA) using the librairies FactoMineR (https://cran.r-project.org/package=FactoMineR) and psy (https://cran.r-project.org/package=psy) with default parameters. Detailed interpretations of the 3D sphere and the biplots can be found in (33,34).

Selection of iron responsive genes and iron homeostasis key genes (iHKG). Selection of iron responsive genes, as defined in Figure 1, was based on statistics for differential expression, i.e. *Z*-Score values and adjusted *P*-values. Upregulated (or induced) genes are those with a *Z*-Score value greater than two and an adjusted *P*-value lower than 5%, whereas downregulated (or repressed) genes are those with a *Z*-Score value lower than −2 and an adjusted *P*-value lower than 5%. Each gene that was observed as up- or downregulated in at least one of the conditions C1, C2, C3 or C4 was included in the final list of 'iron responsive genes'. From all the iron responsive genes, iHKG genes, as defined in Figure 1, were selected by applying the following successive rules: (i) adjusted *P*-value < 0.01, (ii) *Z*-Score value higher than 1.5 or lower than −1.5, (iii) validation of criteria (i) and (ii) in conditions 'low Fe' (C1 or C2) and in conditions 'high Fe' (C3 or C4). In iHKG, Type I genes are those which were found upregulated (respectively, downregulated) in 'low Fe' conditions (C1 or C2) and downregulated (respectively, upregulated) in 'high Fe' conditions (C3 or C4). Type II genes are those which were found upregulated (respectively, downregulated) in 'low Fe' conditions (C1 or C2) and also upregulated (respectively, downregulated) in 'high Fe' conditions (C3 or C4).

Data mining of GO terms, definition of Meta-GO and allocation of iron responsive genes to general functions. Gene Ontology (GO) terms were used to define the Meta-GO (Figure 3). They were extracted from the 'Biological Process' section of the GO database (35). The li-

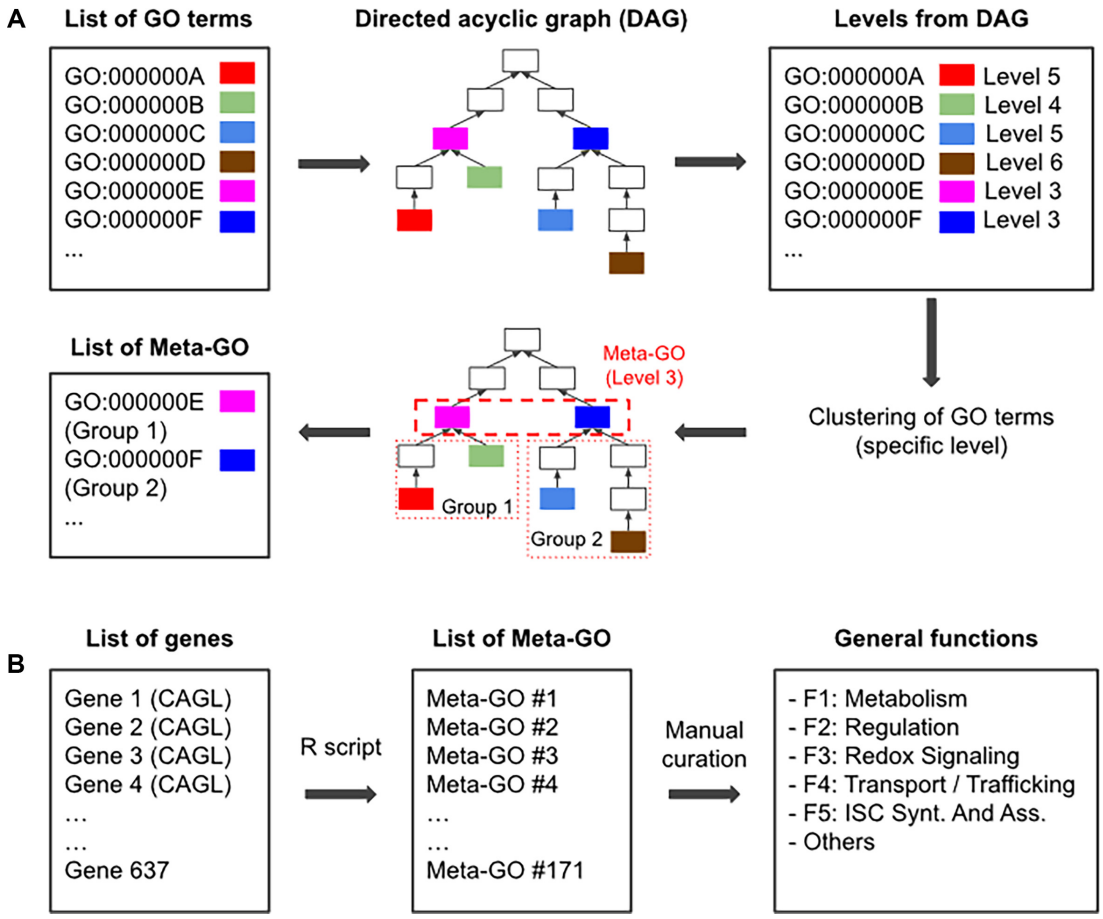


Figure 3. Data mining of GO terms to define a limited number of general functions to which all iron responsive genes can be assigned. (**A**) Schematic representation of the process applied to manage GO terms in this study. Starting from a list of GO terms (all terms in 'Biological Process' here), they are located in the DAG provided by the GO database. Levels are associated to each GO term according to their position in the DAG (see 'Materials and Methods' section and Supplementary Figures S10). This is illustrated here with colored boxes. GO terms are next clustered according to a specific level (Level 3 in this example) in the GO hierarchy. This produces 'Meta-GO', i.e. groups of GO terms which share a common ancestor in the DAG. Note that in the analysis of iron responsive genes presented in this article, Level 4 was used to create Meta-GO (see 'Materials and Methods' section and Supplementary Data S3). (**B**) General functions defined in this study to highlight physiological roles of iron responsive genes. They are named 'Metabolism' (F1), 'Regulation' (F2), 'Redox signaling' (F3), 'Transport/trafficking' (F4), 'Iron sulfur cluster synthesis and assembly' (F5) and 'Others'.

brary OntologyIndex (https://cran.r-project.org/package=ontologyIndex) was used to associate each term with a 'Level'. By definition, a level is the length of the longest path, which exist between a GO term (e.g. 'iron ion transmembrane transport'—GO:0034755) and the term 'biological process' (GO:0008150) that is located at the root of the GO hierarchy (see Supplementary Figures S10 for an illustration). Once the levels were assigned, only the terms related to at least one of the iron responsive genes (see previous section) were retained and used for Meta-GO assignments. Because we observed that GO terms in *C. glabrata* were not accurate enough, the GO terms associated with orthologous genes in *S. cerevisiae* were preferred at this step. Meta-GO assignment consisted to group the GO terms at a specific level in the GO hierarchy and assign all of them to the associated common ancestor in the DAG (see Figure 3A for an illustration). The Meta-GO defined in this study were obtained at Level 4. They were allocated (manually) to general functions named 'Metabolism' (F1), 'Regulation' (F2), 'Redox signaling' (F3), 'Transport/trafficking' (F4), 'Iron-sulfur cluster synthesis and assembly' (F5) and 'Others' (Figure 3B). This procedure allowed to further classify (through the Meta-GO) the iron responsive genes into the general functions. Note that in case of multiple functions for a gene (a gene can be associated with several GO terms, in several Meta-GO, in different general functions) the general function in which the gene had the highest number of associated GO terms was chosen.

Networks of co-expressed genes. Gene networks were inferred using a simple approach, which is based on distance calculations between gene expression measurements in conditions C1, C2, C3, C4. A schematic representation of the method can be found in Figure 4. In this work, Euclidean distances were calculated between all pairwise gene expression profiles (R function 'dist') and the threshold to connect nodes on the network was fixed such that only the 5% of pairs of genes with the smallest distances between

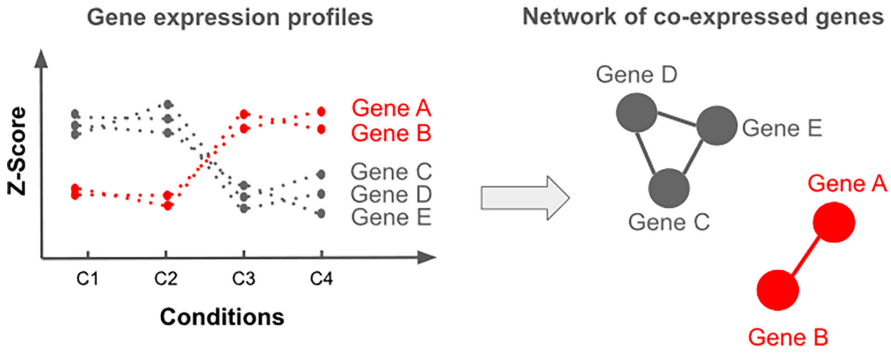


Figure 4. Schematic representation of a network of co-expressed genes. In such a network, genes are represented by nodes and similarities between gene expression profiles of two genes are represented by edges. To infer a network it is necessary to (i) calculate a measure of distance between gene expression profiles (Euclidean distance for instance), (ii) define a threshold value such that if the distance between two gene expression profiles is below the threshold, the corresponding nodes on the network are connected with an edge and (iii) apply dedicated algorithms to calculate node positions in a two dimensional space, such that co-expressed genes are represented by nodes positioned side by side as illustrated here with genes A and B (on the right) and genes C, D and E (on the left).

expression profiles would be connected. The network of co-expressed genes were represented using a combination of the library igraph (https://cran.r-project.org/package=igraph) for the calculation of the positions of nodes (R function 'layout_nicely') and the software Cytoscape (36) for dynamic network exploration in https://thomasdenecker.github.io/iHKG/.

Source of gene descriptions and annotations, sequence data and orthologous relationships between genes. Gene description and annotations were retrieved from the Candida Genome Database (CGD) (37) and the GRYC database (http://gryc.inra.fr/). Functional enrichment of GO terms was performed with the version of GOterm Finder (38) available in CGD (http://www.candidagenome.org/cgi-bin/GO/goTermFinder) and GO SLIM Mapper (http://www.candidagenome.org/cgi-bin/GO/goTermMapper). Orthologous relationships between *C. glabrata* and *S. cerevisiae* genes were downloaded from CGD (http://www.candidagenome.org/download/homology/orthologs/C_glabrata_CBS138_S_cerevisiae_by_CGOB/), as well as orthologous relationships between *C. glabrata* and *C. albicans* (http://www.candidagenome.org/download/homology/orthologs/C_glabrata_CBS138_C_albicans_SC5314_by_CGOB/).

RESULTS

Transcriptomic experiments in two unbalanced iron conditions (low and high) to reveal the 'key genes' involved in iron homeostasis

Definitions, working hypothesis and experimental design. In this study, our aim was to identify the 'key genes' required for iron homeostasis in the pathogenic yeast *C. glabrata*. Referred to as iHKG, these genes are expected to have an important activity for iron homeostasis that we define as the property of the cells to counterbalance intracellular physiological consequences of external iron changes (Figure 1A). In particular, we wanted to discriminate iHKG from the set of iron-responsive genes, which are all the genes exhibiting up- or downregulation in response to iron fluctuations. In this context, our working hypothesis was the following. Because iHKG are involved in iron homeostasis specifically, and that homeostasis is associated with a natural resistance to bidirectional changes (low or high iron availability), iHKG should exhibit de-regulation in both low and high iron source conditions. Other iron responsive genes should exhibit de-regulation in only one of the conditions, either low or high. The iHKG thus represent a subclass of the iron responsive genes (Figure 1B). Based on this, we defined two types of iHKG, 'Type I' and 'Type II' (Figure 1C). Type I genes are those for which the de-regulation is opposite in low and high iron conditions (respectively up- and downregulated), whereas Type II genes are those for which the de-regulation is identical in low and high iron conditions (constantly up- or downregulated). Note that we could anticipate, based on these definitions, that Type II genes would also include genes involved in the general stress response of cells (ESR genes for instance (39)) and hence would be less specific to iron physiological processes.

To reveal iHKG, we designed an experimental plan (Figure 2A) in which cells were cultured under two iron conditions: iron-deficiency (Fe 'low') and iron-overload (Fe 'high'), at two different temperatures: 30°C and 37°C. Thirty degrees is the usual temperature at which *C. glabrata* is cultivated in laboratories, while 37° is the temperature at which *C. glabrata* develops in the human body. Since iron homeostasis is an important physiological process involved in host-pathogen interactions, it was important to assess the potential effect of the temperature. Transcriptomics experiments were performed with microarray technology, to quantify gene expression changes with respect to appropriate controls (see 'Materials and Methods' section). As a result, experimental data were obtained in the five conditions referred to hereafter as C1, C2, C3, C4 and C30–37, for all *C. glabrata* genes that could be monitored with microarrays.

Temperature shift has a limited impact on gene expression changes associated with cell responses to unbalanced iron conditions. In the first step of the analysis, our aim was to evaluate the potential effect of temperature shift (30°C–37°C) on gene expression changes monitored in 'low' and

'high' iron conditions. For that, we applied PCA to study the correlation matrix between normalized expression measurements obtained for all genes in conditions C1, C2, C3, C4 and C30–37 (see 'Materials and Methods' section). Results are presented in Figure 2B and C. The five experimental conditions are represented by individual variables and they are on a unit hypersphere on which the distances between them are directly proportional to the initial correlation values stored in the matrix (34). Notably, we observed (i) that gene expression measurements are highly correlated between conditions C1 and C2 in the one hand, and conditions C3 and C4 on the other hand, and (ii) that the condition C30–37 is clearly separated from the others (Figure 2B and C). This observation indicates that if the temperature shift has an effect on the expression of *C. glabrata* genes, this effect is unrelated to any effects caused by extracellular iron imbalance. In other words, most of the 'iron responsive genes', which are genes de-regulated in response to iron deficiency or iron overload at 30°C, are also de-regulated at 37°C. This is an interesting result, which allows us to consider the generalization of experimental results obtained in a laboratory at 30°C to a more realistic living situation for *C. glabrata*, as a human pathogen. In the rest of this work, the transcriptomics data we obtained at 30°C and 37°C were combined in order to simplify iHKG identification. Note that another interesting property could be drawn from this PCA. We observed that projections of the initial condition vectors (C1-C2 and C3-C4) were orthogonal (90° angles, Figure 2C). This means that gene expression changes in response to 'low iron' conditions are essentially unrelated to gene expression changes in response to 'high iron'. This observation is important with regard to the respective roles of Type I and II genes (see next sections).

Type I and type II iHKG represent small sub-classes of the set of all iron responsive genes. Among the genes for which expression data were available, we identified those displaying significant changes in mRNA levels, respectively at 'low' or 'high' levels of extracellular iron availability. We used the LIMMA statistical procedure to identify differentially expressed genes based on replicated experiments in C1, C2, C3 and C4 (see 'Materials and Methods' section). As a result, we found that 637 genes were significantly up- or downregulated in at least one condition. Together, these genes represent our complete set of 'iron responsive genes' (Figure 1B). Among these genes, we found 214 iHKG of which 73 were Type I and 141 were of Type II (see 'Materials and Methods' section). Together iHKG represent 33% of iron responsive genes. They constitute, as anticipated from PCA analysis, a fairly small sub-class of genes, especially with regard to type I genes (11%). A detailed list of iron responsive genes and iHKG can be found in Supplementary Data S2.

Representation of iron responsive genes in functional networks of co-expressed genes

Definition and objectives. Starting from the list of 637 genes up- or downregulated in at least one condition (the iron responsive genes, see previous sections), we can infer functional networks of co-expressed genes. By 'functional networks', we mean graphs in which genes (represented as nodes) are (i) involved in a common cellular function and (ii) are connected by edges if they react similarly during iron homeostasis. On these graphs, we expected the iHKG to be easily highlighted and thus integrated into a more comprehensive functional context. Our computational strategy to infer and visualize the functional networks was divided into three steps: (i) placing all the iron responsive genes in a cellular function to which they contribute, (ii) representing them, in each function, by applying methods for gene co-expression network inference, (iii) developing an interactive web viewer for network exploration, rapid location of any gene and retrieval of its biological context. These three steps are detailed below.

Step 1: Computational strategy to place all the iron responsive genes in a small number of functional categories. Unlike the classical approach which consists in identifying functional categories (generally GO terms) significantly enriched in our list of genes (38), we wanted here to define a small number of cellular functions (fewer than 10) and assign all of the 637 iron responsive genes to only one of the pre-defined functions. This limited number of functions was required to allow subsequent gene network inference and visualization. To define our functions, we first classified all GO terms (more than 40 000) available in the 'Biological Process' section of the GO database into Meta-GO groups (see 'Materials and Methods' section). As illustrated Figure 3A, the 'Meta-GO' are GO terms, which share a specific level in the GO directed acyclic graph (DAG), the hierarchical organization of terms defined by the Gene Ontology Consortium (35). They serve as representatives for all other related terms in the DAG. Next, we selected all Meta-GO (level 4) in which at least one GO term was assigned to one iron responsive gene (one of the 637 genes). We found at this step 171 Meta-GO. They were inspected and manually classified into the general functions named 'Metabolism' (F1), 'Regulation' (F2), 'Redox Signaling' (F3), 'Transport/trafficking' (F4), 'Iron sulfur cluster synthesis and assembly' (F5) or 'Others' (Figure 3B). These general functions are emblematic of the key cellular processes by which yeast cells adapt their functioning to iron deficiency or iron overload. They are detailed in Table 1. All information regarding data mining of GO terms, Meta-GO assignments and clustering into general functions is available in Supplementary Data S3.

Step 2: Computational strategy to derive functional networks from gene expression measurements. We calculated the allocation of the 637 iron responsive genes into the functional categories described in the previous section. We observed 28% of genes in F1 (Metabolism), 17% in F2 (Regulation), 18% in F3 (Redox signaling), 13% in F4 (Transport/trafficking) and 3% in F5 (ISC synthesis and assembly). Altogether almost 80% of the iron responsive genes were classified in one of the general functions based on Meta-GO exploration. We also found that iHKG are well-represented in each function: 34% in Metabolism (F1), 29% in Regulation (F2), 42% in Redox Signaling (F3), 25% in Transport/trafficking (F4) and 36% in ISC synthesis and assembly (F5). To derive gene networks within each function, we calculated co-expression graphs (see 'Materials and

Table 1. Functional categories defined in this work to represent iron responsive genes using functional networks of co-expressed genes

Label	Name	Includes Meta-Go related to:	# of Meta-GO
F1	Metabolism	Nucleic acid, amino acid, fatty acid and lipid metabolism ; Carbon metabolism/energy production from respiratory and non respiratory origin ; Mitochondria functions (including biogenesis, mitophagy and functions other than redox signaling that are included in function F3, see below) ; Membrane and cell wall biogenesis ; Ribosomal biogenesis (biosynthesis, RNA processing and translation).	41
F2	Regulation	Transcriptional regulation (including general transcription) ; Post-translational modifications (including protein phosphorylation, glycosylation, structural modification and degradation) ; Ribosome activity (including translation) ; Signal transduction ; RNA and protein fate.	8
F3	Redox signaling	Proteins with functions directly or indirectly linked with redox mechanisms, i.e. oxygen dependant enzymes, flavo-hemoproteins including Cytochrome P dependent, membrane iron reductase, NADP/NADPH-dependent enzymes, metalloenzymes. These proteins are thus involved in thiol and redox signaling pathways, respiratory chain components, peroxisome activity, ROS detoxification, carrier proteins, folate biosynthesis, heme biosynthesis, etc.	7
F4	Transport/trafficking	Cell exchanges between intra and extracellular compartments (directed movement of substances such as macromolecules, small molecules, ions, etc.) ; Processes for internal cell trafficking.	3
F5	Iron sulfur cluster synthesis and assembly	Chemical reactions and pathways involving sulfur or compounds that contain sulfur.	1
OTHERS	Membrane/cell wall| Pathogenesis| Stress response| Unclassified	Remaining Meta-GO for which no classification was clear or not associated specifically to iron homeostasis: • 'Unclassified' (73 Meta-GO) • 'Stress Response' (26 Meta-GO) • 'Membrane/Cell Wall' (8 Meta-GO) • 'Pathogenesis' (5 Meta-GO).	112

These functions were meant to be representative of the cellular processes by which yeast cells adapt to iron deficiency or iron overload. *Candida glabrata* indeed tunes its response by adapting the biological systems devoted either to iron acquisition or to the mobilization of cellular iron storage (represented with the function 'Transport/trafficking'). Cellular processes that require iron for function are also modified (functions 'Metabolism' and 'Redox signaling'). The underlying regulatory pathways are well described, including transcriptional and post-transcriptional mechanisms (function 'Regulation'). At last, critical roles played by iron-sulfur clusters are known (function 'Iron sulfur cluster synthesis and assembly') and complex relationships between iron homeostasis and oxidative stress response is often emphasized (function 'Stress response' in 'Others' category). Methodology to explain the Meta-GO is presented Figure 3 and in 'Materials and Methods' section. Note that the category 'Others' comprise all remaining Meta-GO for which no classification was clear or not associated specifically to iron homeostasis.

Methods' section). The main idea is to measure the similarity between gene expression measurements of all genes in conditions C1–C4. If the similarity is high enough between two genes, these two genes are represented by connected nodes on the graph (see Figure 4). Graphs of co-expressed genes obtained for the functional categories F1–F5 are presented in Figure 5. Note that if a unique graph was obtained for a function then two complementary representations were derived by coloring the nodes according to gene expression measurements obtained in 'low' or 'high' iron conditions. This allowed us to observe the symmetrical de-regulations that characterize Type I and Type II iHKG (Figure 1).

Step 3: Computational strategy to interactively explore the functional networks of co-expressed genes. The main interest of Figure 5 is to give a global overview of the gene expression processes used by the pathogenic yeast *C. glabrata* to face low and high iron conditions. We integrated here more than 100 000 qualitative and quantitative heterogeneous kinds of information (GO terms, microarray data, gene annotations in public databases, etc.) to obtain a unified picture of multiple cellular processes. We added to Figure 5 the names of around fifty genes, which we found to agree with current knowledge of iron homeostasis in *C. glabrata* (see the next section). It is clear, however, that further exploration of these networks could be considered, examining for instance the disconnected (isolated) genes, or searching for the location of genes with new putative functions related to iron homeostasis. To ensure the diffusion of our data, as well as their subsequent exploitation by ourselves and by others, we developed an interactive web viewer (called 'iHKG viewer'). This website is publicly accessible (see 'Materials and Methods' section). Several features were implemented to (i) enable dynamic exploration of each functional network of co-expressed genes, (ii) enable rapid location of any genes in the network and (iii) obtain for any gene, all the available annotations available in the public databases GRYC and CGD (Figure 6).

Biological relevance of iron homeostasis key genes

Accordance with knowledge of iron homeostasis in Candida glabrata. The biological relevance of our results was manually verified based on a recent review of the regulation of iron homeostasis in *C. glabrata* (22). The genes cited in this article for their role in iron homeostasis were searched in our functional networks of co-expressed genes. As expected, we found orthologs of genes described in *S. cerevisiae* as involved in (i) iron-dependant cellular functions

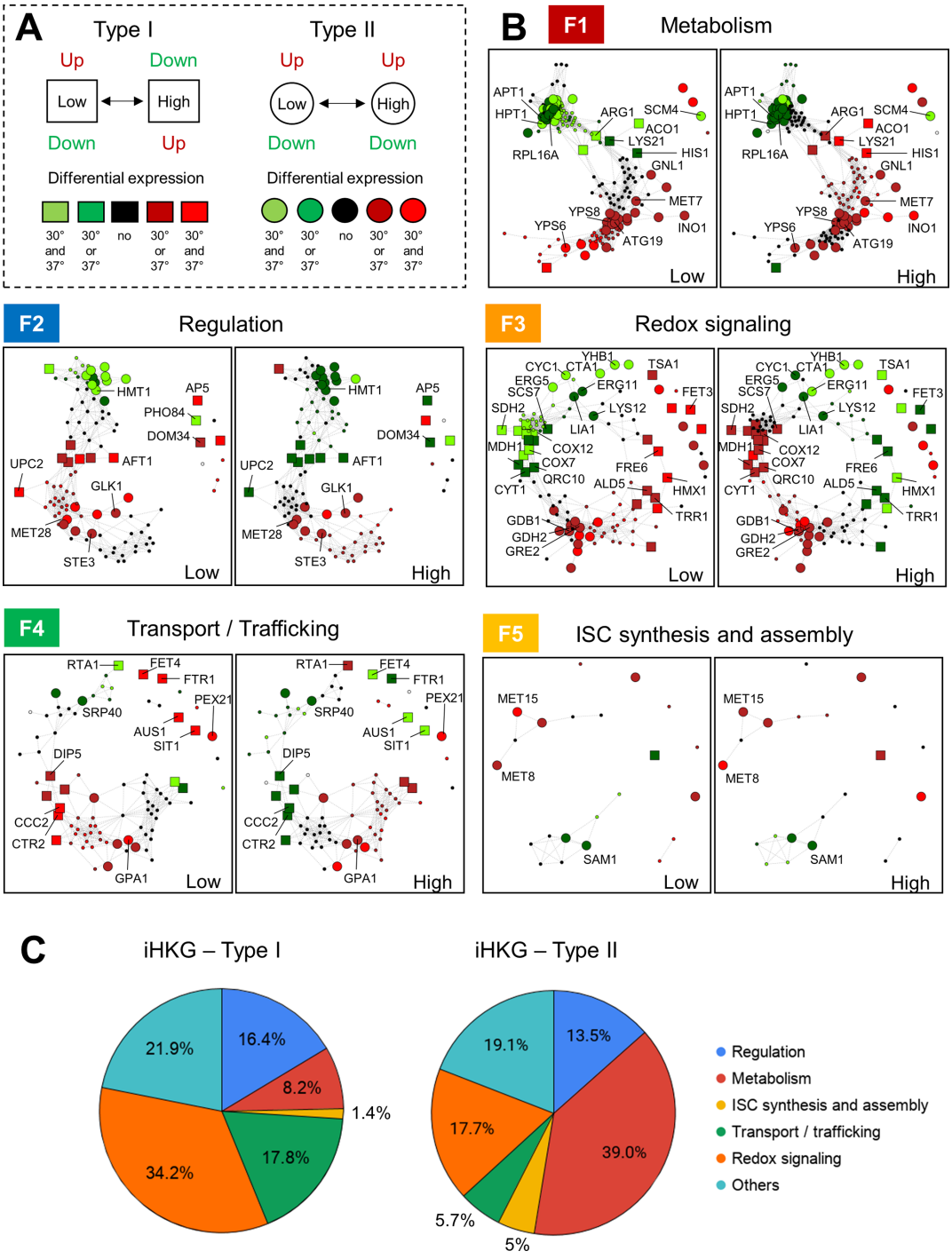


Figure 5. Functional networks of co-expressed genes derived from iron responsive genes. (**A**) Color coding and graphical style used to represent the vertices in the co-expression graphs shown in (**B**). Type I genes are represented by squares, while type II genes are represented by circles. Color is based on the deregulation status of the genes: upregulation in red and downregulation in green. (**B**) Co-expression graphs obtained based on distance calculations between gene expression profiles (C_1–C_4, Figure 4 and 'Materials and Methods' section). Genes were separated here according to their assignment into the general function 'Metabolism' (F1, 176 genes), 'Regulation' (F2, 106 genes), 'Redox signaling' (F3, 118 genes), 'Transport trafficking' (F4, 84 genes) and 'ISC synthesis and assembly' (F5, 22 genes). These two-faced functional networks highlight iHKG in the pathogenic yeast *C. glabrata.* An emblematic illustration of the typical homeostatic role played by iHKG is the co-expression graphs obtained for the function 'Transport/trafficking', in which most of the genes match the definition of iHKG of Type I. Gene names were placed on the graphs, when they were available, according to the CGD database. (**C**) Allocation of Type I and Type II genes in general functions defined in this work, with the percentage of genes assigned to each function, considering iHKG of Type I (left) or type II (right).

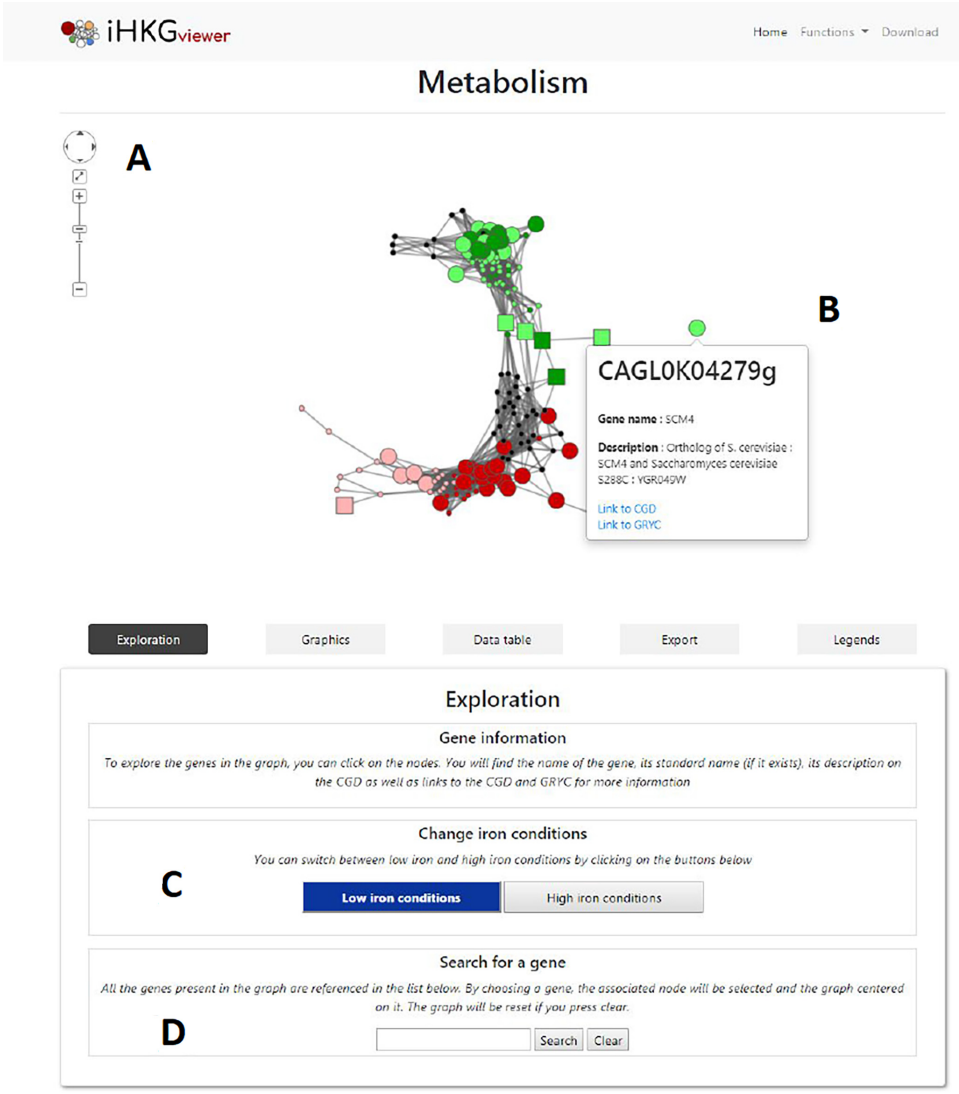


Figure 6. Screenshot of the iHKG viewer. Several functionalities were implemented to easily explore the functional networks of co-expressed genes described in this article: (**A**) Possibility to zoom in or out the graph, (**B**) Possibility to click a node with the mouse and obtain gene name, description and direct web links to CGD and GRYC databases. (**C**) Possibility to switch between low iron condition and high iron condition and (**D**) Possibility to search for a particular gene in the network. The functional network 'Metabolism' is shown here as an illustration.

such as respiration, e.g. *QCR2, QCR6, QCR7, QCR10, COX4, COX5B, COX6, COX7, COX9, COX12, COX15* (all in the functional network 'Redox Signaling'), *ACO1* (functional network 'Metabolism') or COX23 (functional network 'Unclassified'), (ii) genes that encode metalloproteins, e.g. *SDH2, CCP1, RIP1, CYT1, LIA1, CYC1, GLT1, YHB1* (all in the functional network 'Redox Signaling'), *RLI1* (functional network 'Regulation'), *ILV3* (functional network 'Metabolism') or (iii) genes involved in autophagy (general or mitochondria), e.g. *ATG19*, *ATG32*, *ATG41* (functional network 'Metabolism'). Genes involved in Fe-S clusters or heme biosynthesis and assembly were spread into the functional networks 'ISC synthesis and assembly' (e.g. *ISA1, CFD1*), 'Regulation' (e.g. *GRX4*), 'Metabolism' (e.g. *HEM4*, *HEM15*), 'Redox Signaling' (e.g. *HEM13*, *COX15*). This splitting between different networks arises from our decision to assign genes to only one functional category. Genes encoding proteins with functions directly or indirectly related to redox mechanisms were thus separated from the metabolic pathway in which they participate (as illustrated here with the heme biosynthetic pathway). The reader who is disturbed by this separation of functions can refer to the global network accessible here https://thomasdenecker.github.io/iHKG/all.html. As expected also, the genes *AFT1* (CAGL0H03487g) and *YAP5* (CAGL0K08756g), which encode transcriptional factors, were found in the functional network ('Regulation') along with CAGL0E01243g (the ortholog of *S. cerevisiae* genes *CTH1*/*CTH2*, which encode RNA binding proteins that promote degradation of mRNA of iron-associated genes). Also, we noticed (i) in the functional network 'Redox Signaling' the gene CAGL0A03905g (the ortholog of

S. cerevisiae gene *HMX1*), which encodes a heme oxygenase and thus allows iron recycling from heme and the gene *ERG11* (CAGL0E04334g), which encodes a heme-binding protein required for sterol biosynthesis, (ii) in the functional network 'Metabolism' the gene *DOM34* (CAGL0B04675g), which encodes a ribosome recycling factor, and (iii) in the functional network 'Others' the gene CAGL0G06798, the ortholog of *S. cerevisiae* gene *LSO1* for which the function in still undefined in *S. cerevisiae*. At last, we found genes involved in the three main routes for iron uptake in yeasts, i.e. the high-affinity system, e.g. *FTR1* (CAGL0I06743g), *FET3* (CAGL0F06413g), *CCC2* (CAGL0M08602g), the low-affinity system, e.g. *FET4* (CAGL0F00187g) and the uptake mediated by the capture of xeno-siderophores, e.g. CAGL0E04092g (the ortholog of the *S. cerevisiae* gene *ARN1*). These genes were all in the functional network 'Transport/trafficking'. The gene CAGL0C03333g (the ortholog of *S. cerevisiae* genes *FRE6*/*FRE4*), which is required for the appropriate functioning of the high-affinity uptake system, was found in the functional network 'Redox signaling'. It encodes a protein with a ferric reductase activity that allows an extracellular reduction of ferric iron (Fe3+) to ferrous iron (Fe2+). Genes involved in the copper transfer toward the protein Fet, e.g. the gene CAGL0J07980g (ortholog of *S. cerevisiae* gene *ATX1*), CAGL0M08602g (ortholog of the *S. cerevisiae* gene *CCC2*), CAGL0D04708g (ortholog of the *S. cerevisiae* gene *CTR1*) and CAGL0I02508g (ortholog of the *S. cerevisiae* gene *CTR2*) were also identified in the functional network 'Transport/trafficking', as well as genes coding for actors in the vacuolar or mitochondrial iron export system, e.g. CAGL0H08822g (ortholog of the *S. cerevisiae* gene *MMT1*), *FTH1* (CAGL0M05511g), *SMF3* (CAGL0A03476g).

Gene annotations in public databases. To further explore our list of iron responsive genes, independently of the literature, we used gene annotations available in public databases. We could thus identify a set of genes which (i) were classified as iHKG and (ii) were given a 'Standard Name' (for example, *TRR1* or *TSA1*). This means that the functional information available in CGD or GRYC for these genes was assigned on the basis of genetic, biochemical, or molecular characterization (standard gene names are optional, and genes that are completely uncharacterized generally only have systematic names, for example CAGL0H09592g or CAGL0I00286g). We thus found 11 genes with oxidoreductase activity (i.e. *ALD5*, *CTA1*, *ERG5*, *GDH2*, *GRE2*, *LYS12*, *MDH1*, *MET8*, *SCS7*, *TRR1* and *TSA1*), nine genes with transferase activity (i.e. *APT1*, *GDB1*, *GLK1*, *HIS1*, *HMT1*, *HPT1*, *LYS21*, *MET15* and *SAM1*) and four genes with transporter activity (i.e. *AUS1*, *DIP5*, *GAP1* and *PHO84*). Several genes are also involved in response to stress (i.e. *CTA1*, *ERG5*, *HSP12*, *PHO84*, *RSB1*, *RTA1*, *TRR1*, *TSA1* and *UPC2*), filamentous growth (i.e. *ERG5* and *PHO84*), cellular respiration (i.e. *MDH1*) or cellular homeostasis (i.e. *GLK1*, *TSA1*). They were all noted on the functional networks presented in Figure 5.

Potential specificities of Type I and Type II gene functions. To conclude our description of iHKG and associated functional networks, we examined the repartition of iHKG of Type I and Type II in each functional network. Results are presented in Figure 5C. We observed that the allocation of iHKG greatly differ between Type I and Type II. Differences were particularly important for networks 'Metabolism' (8% in Type I compared to 39% in Type II), 'Transport/trafficking' (18% in Type I compared to 6% in Type II) and 'ISC synthesis and assembly' (1.4% in Type I compared to 5% in Type II). To better explain this observation, we searched for significantly enriched GO terms in lists of Type I and Type II genes, respectively, using the GO database sections 'Cellular component' and 'Molecular function' ('Biological Process' was already used to define Meta-GO, Figure 3). Detailed results are available in Supplementary Data S4. In Type I genes, we found significantly enriched the terms 'membrane part' (33% of genes in the list, P-value = 0.0033), 'cell periphery' (30%, P-value = 0.00156) and 'transporter activity' (22%, P-value = 0.00011). This underlines the particular role of Type I genes in controlling the cell's interactions with its environment, in order to adjust the activity of its transporter proteins. This is relevant to the observation that Type I genes are more represented in the network 'Transport/trafficking' than Type II genes. For Type II genes, we rather found significantly enriched the terms 'ribonucleoprotein complex' (21%, P-value = 0.00277), 'cytosol' (17%, P-value = 0.01141), 'ribosome' (13%, P-value = 0.00016) or 'heme binding' (4%, P-value = 0.00076). It is the particular role of Type II genes to protect internal cellular functions which is highlighted, through the use of traditional mechanisms of stress response (ribosomal proteins) and stabilization of key processes that critically depend on iron (heme utilization or the pathway for iron-sulfur cluster synthesis and assembly). Again, this is relevant to the observation that Type II genes are more represented in the networks 'Metabolism' and 'ISC synthesis and assembly' than Type I genes.

To summarize this section, we observed in our functional networks of co-expressed genes, many genes already known for their role in iron homeostasis in *C. glabrata*. Such an observation gives credence to the biological interest of the transcriptomics dataset we produced in this study. We also described genes for which annotations were available in public databases and whose functions remain coherent with iron homeostasis processes in *C. glabrata*. At last, we observed interesting specificities for the functions supported by iHKG respectively of Type I and Type II. Of course, these observations merit further experiments. But the important point is that we provide with this work an original source of transcriptomics data, which can be interactively and comprehensively explored at any time by anyone, in any context related to iron homeostasis studies.

DISCUSSION

In this study, our goal was to investigate the mechanisms underlying iron homeostasis in the pathogenic yeast *C. glabrata*. In particular, we wanted to identify the 'key genes' whose role is to counterbalance the consequences for the cell not only of iron deficiency, but also of iron overload. In the literature the word 'homeostasis' is found in association with multiple cellular processes, such as 'energy homeosta-

sis' (40), 'pH homeostasis' (41), 'cellular redox homeostasis' (42) or 'cell size homeostasis' (43). In each case, homeostasis refers to mechanisms of return to equilibrium, necessary for the cell to maintain its biological functions in a state compatible with its survival. By definition, homeostasis is thus a dynamic process, which is reversible and requires multiple degrees of freedom to be effective in a panel of different situations. In this context, we believe that the iHKG identified in this study are of particular interest as fundamental actors of iron homeostasis *per se*.

In order to capture iHKG, we produced an original set of microarray experiments in which 'low' and 'high' iron conditions were faced, at two different temperatures (Figure 2). The microarrays we used allowed us to monitor more than 98% of 'verified' and 'uncharacterized' open reading frames in *C. glabrata* genome (see Supplementary Note S8). It is a high percentage that guarantees realistic snapshots of the transcriptome changes. In our experiments, we wanted to mimic the host environment of *C. glabrata* regarding iron sources. We thus used ferrous iron ($FeSO_4$) instead of ferric iron ($FeCl_3$) since in the human body, Fe^{3+} is very scarce or not available because it is rapidly reduced to Fe^{2+} or bound to proteins like transferrin, or ferritin. Fe^{2+} is therefore the major available form of iron for *C. glabrata* cells, both in extra (digestive or urogenital epithelia) and intracellular (macrophage) compartments. We paid special attention to subject the cells to limited environmental changes, in order not to induce cell death or irreversible changes in cell physiology. The concentrations of BPS and $FeSO_4$ used to generated respectively 'low' and 'high' iron conditions remain are equivalent or slighter than those applied in previous studies (see Supplementary Data S5). Also, the same yeast strain was systematically used, as well as the same protocols for RNA preparation, slide hybridization, raw data normalization, etc. This is an important added value for subsequent data analyses, greatly limiting potential sources of experimental noise. At last, we performed the experiments at two different temperatures, i.e. 30°C and 37°C. It was important to verify that at 37° (the human body temperature), the transcriptomic responses remained consistent with those described at 30°, which is the classical temperature used for *C. glabrata* cell cultures in laboratories. To our knowledge, this represents the first transcriptomic dataset published for *C. glabrata* that allows gene expression comparison between 30° and 37°. Even if we observed that the temperature did not have a critical effect on the transcriptomic response which arises either in low or in high iron conditions, we still observed several hundred genes with differential expression between the two conditions (Supplementary Data S1). These results will be a valuable resource to control in any transcriptomics project that genes identified based on laboratory experiments performed at 30°C, are not completely de-regulated at 37°C.

In this work, our mining of the data relies greatly on the representation of the *C. glabrata* genes in functional networks of co-expressed genes (Figures 4 and 5). To infer these networks, we combined a large amount of information, both numerical (measurements of gene expression) and descriptive (GO terms, Types I or II classifications, gene names, etc.). We divided the global network of co-expressed genes (which initially comprised the 636 iron responsive genes) into six functional sub-networks, referred to as 'Metabolism', 'Regulation', 'Redox Signaling', 'Transport trafficking', 'ISC synthesis and assembly' and 'Others' (https://thomasdenecker.github.io/iHKG/). Our aims were (i) to limit the number of genes in each network, thus improving the visualization and exploration of graphs and (ii) to provide a global picture of the cellular functions in which substantial gene expression rewiring was observed. Of course, other choices for gene assignment in subnetworks could have been made (see Supplementary Note S9), but they are, in any case, totally disconnected from the definition and the identification of the iHKG.

We identified with this work a list of 214 genes as good candidates for being key elements of iron homeostasis (iHKG of Type I and II, Supplementary Data S2). Because they rely on stringent selection, iHKG can be considered trustworthy information, which is, notably, independent of existing knowledge obtained in the model species *S. cerevisiae* and *C. albicans*. We present in Supplementary Note S7 examples of interesting biological insights that arise from our observations: (i) We propose a shortlist of 27 genes that did not appear in previous emblematic publications in the field (15,22). For 10 of them, we observed interesting variations in expression levels reported in recent multi-omics datasets (RNAseq and mass spectrometry technologies). (ii) We detected differentiated regulatory mechanisms underlying transcription of the iHKG of Type I and Type II. In Type II genes, we found as enriched DNA sequence in promoters, the motif AGGG. It is the stress response element (STRE), i.e. the DNA binding site of transcription factors Msn2 and Msn4 (44). Such observation is relevant with the manner Type II genes were defined in this study (Figure 1) and our anticipation they could include genes involved in the general stress response of cells (ESR genes for instance (39)). In Type I genes, we found as enriched motif, the sequence TGCACCC. It corresponds to the DNA binding site of the transcription factor Aft1 (45), one of the main regulators of iron homeostasis (15). Type I genes are very strongly linked to the Aft1 regulon in *C. glabrata* and certainly includes new Aft1 targets so far not described in the literature (e.g. CAGL0A01199g or CAGL0K06259g, Supplementary Note S7). (*iii*) We noticed in our network of co-expressed genes (function 'Regulation') that the gene coding the transcription factor Hap1 (CAGL0B03421g) was in the immediate neighbourhood of the transcription factor Aft1 (Supplementary Figures S10). Hap1 is the heme activated transcription factor. It is thus not surprising to find the gene *HAP1* in our list of iHKG, because heme biosynthesis requires iron. Hap1 is thus expected to be an indirect iron sensor and its activity is known to be affected by iron deficiency (46). Although the main role of Hap1 is globally conserved in yeast species, a dual role may exist as described for *Kluyveromyces lactis* (47). Our results show for the first time that, in *C. glabrata*, *HAP1* is also deregulated in excess iron conditions. At last, even if we observed the global responses at 30°C and 37°C are similar, we could see several iKHG that were more de-regulated at 37°C, which is the human host temperature. Among them, we found *AWP2* (CAGL0K00110g) and *ERG5* (CAGL0M07656g). We provide here interesting transcriptomic data showing that they may be essential for the yeast homeostasis in iron varying

conditions, and hence for the *C. glabrata* adaptation in the human body compartments.

To conclude, the data we provide in this work is full of new research perspectives. It can help to improve the functional annotation of *C. glabrata* genes (only 5% of genes are 'verified' in the CGD database) and increase our understanding of the specificities of this human pathogen. In that respect, we found in our data several genes for which no clear orthologous genes could be identified in *S. cerevisiae* and *C. albicans* (Supplementary Data S6). A natural further direction would be to search for genomic mutations in these genes, using *C. glabrata* strains isolated from patients (48). The rapid increase in genomic sequence availability together with the decreasing cost for deep sequencing represent unprecedented opportunities for population genomics studies. We finally hope our web server will be valuable resources for such exciting analyses, particularly in perspective of medical applications.

DATA AVAILABILITY

Microarray raw data files are accessible through GEO Series accession number GSE139363 (https://www.ncbi.nlm.nih.gov/geo/). Source code and result data files can be found here: https://github.com/thomasdenecker/iHKG/. The website to explore the functional networks of co-expressed genes is available here: https://thomasdenecker.github.io/iHKG/.

GEO Series accession number is GSE139363.

SUPPLEMENTARY DATA

Supplementary Data are available at NARGAB Online.

ACKNOWLEDGEMENTS

G.L. would like to thank Linda Sperling for careful reading of the manuscript and comments, the research team 'Mitochondria, Metals and Oxidative Stress' (Jacques Monod Institute, Paris), the research team 'Fungal Epigenomics and Development' (I2BC, Orsay) and the partners from ANR CANDIHUB for helpful discussions.

FUNDING

Ecole doctorale de l'Université Paris-Saclay (SDSV http://www.ed-sdsv.u-psud.fr/); Agence Nationale pour la Recherche (CANDIHUB project, Grant Number ANR–14–CE14–0018–02).

Conflict of interest statement. None declared.

REFERENCES

1. Pfaller,M.A. and Diekema,D.J. (2007) Epidemiology of invasive candidiasis: a persistent public health problem. *Clin. Microbiol. Rev.*, **20**, 133–163.
2. Goemaere,B., Lagrou,K., Spriet,I., Hendrickx,M. and Becker,P. (2018) Clonal spread of candida glabrata bloodstream isolates and fluconazole resistance affected by prolonged exposure: a 12-year single-center study in Belgium. *Antimicrob. Agents Chemother.*, **62**, e00591-18.
3. Epelbaum,O. and Chasan,R. (2017) Candidemia in the intensive care unit. *Clin. Chest Med.*, **38**, 493–509.
4. Dujon,B., Sherman,D., Fischer,G., Durrens,P., Casaregola,S., Lafontaine,I., Montigny,J. de, Marck,C., Neuvéglise,C., Talla,E. *et al.* (2004) Genome evolution in yeasts. *Nature*, **430**, 35–44.
5. Pfaller,M.A., Castanheira,M., Lockhart,S.R., Ahlquist,A.M., Messer,S.A. and Jones,R.N. (2012) Frequency of decreased susceptibility and resistance to echinocandins among fluconazole-resistant bloodstream isolates of candida glabrata. *J. Clin. Microbiol.*, **50**, 1199–1203.
6. Pham,C.D., Iqbal,N., Bolden,C.B., Kuykendall,R.J., Harrison,L.H., Farley,M.M., Schaffner,W., Beldavs,Z.G., Chiller,T.M., Park,B.J. *et al.* (2014) Role of FKS Mutations in Candida glabrata: MIC Values, Echinocandin Resistance, and Multidrug Resistance. *Antimicrob. Agents Chemother.*, **58**, 4690–4696.
7. Vallabhaneni,S., Cleveland,A.A., Farley,M.M., Harrison,L.H., Schaffner,W., Beldavs,Z.G., Derado,G., Pham,C.D., Lockhart,S.R. and Smith,R.M. (2015) Epidemiology and Risk Factors for Echinocandin Nonsusceptible Candida glabrata Bloodstream Infections: Data From a Large Multisite Population-Based Candidemia Surveillance Program, 2008–2014. *Open Forum Infect. Dis.*, **2**, ofv163.
8. Brunke,S. and Hube,B. (2013) Two unlike cousins: Candida albicans and C. glabrata infection strategies. *Cell. Microbiol.*, **15**, 701–708.
9. Sutak,R., Lesuisse,E., Tachezy,J. and Richardson,D.R. (2008) Crusade for iron: iron uptake in unicellular eukaryotes and its significance for virulence. *Trends Microbiol.*, **16**, 261–268.
10. Wang,J. and Pantopoulos,K. (2011) Regulation of cellular iron metabolism. *Biochem. J.*, **434**, 365–381.
11. Nakamura,T., Naguro,I. and Ichijo,H. (2019) Iron homeostasis and iron-regulated ROS in cell death, senescence and human diseases. *Biochim. Biophys. Acta Gen. Subj.*, **1863**, 1398–1409.
12. Bairwa,G., Jung,W.H. and Kronstad,J.W. (2017) Iron acquisition in fungal pathogens of humans. *Metallomics*, **9**, 215–227.
13. Gerwien,F., Skrahina,V., Kasper,L., Hube,B. and Brunke,S. (2018) Metals in fungal virulence. *FEMS Microbiol. Rev.*,**42**, doi:10.1093/femsre/fux050.
14. Thiébaut,A., Delaveau,T., Benchouaia,M., Boeri,J., Garcia,M., Lelandais,G. and Devaux,F. (2017) The CCAAT-binding complex controls respiratory gene expression and iron homeostasis in Candida Glabrata. *Sci. Rep.*, **7**, 1–10.
15. Gerwien,F., Safyan,A., Wisgott,S., Hille,F., Kaemmer,P., Linde,J., Brunke,S., Kasper,L. and Hube,B. (2016) A novel hybrid iron regulation network combines features from pathogenic and nonpathogenic yeasts. *Mbio*,**7**, e01782-16.
16. Gerwien,F., Safyan,A., Wisgott,S., Brunke,S., Kasper,L. and Hube,B. (2017) The fungal pathogen Candida glabrata does not depend on surface ferric reductases for iron acquisition. *Front. Microbiol.*, **8**, 1055.
17. Sharma,V., Purushotham,R. and Kaur,R. (2016) The Phosphoinositide 3-Kinase Regulates Retrograde Trafficking of the Iron Permease CgFtr1 and Iron Homeostasis in Candida glabrata. *J. Biol. Chem.*, **291**, 24715–24734.
18. Srivastava,V.K., Suneetha,K.J. and Kaur,R. (2015) The mitogen-activated protein kinase CgHog1 is required for iron homeostasis, adherence and virulence in Candida glabrata. *FEBS J.*, **282**, 2142–2166.
19. Srivastava,V.K., Suneetha,K.J. and Kaur,R. (2014) A systematic analysis reveals an essential role for high-affinity iron uptake system, haemolysin and CFEM domain-containing protein in iron homoeostasis and virulence in Candida glabrata. *Biochem. J.*, **463**, 103–114.
20. Seider,K., Gerwien,F., Kasper,L., Allert,S., Brunke,S., Jablonowski,N., Schwarzmüller,T., Barz,D., Rupp,S., Kuchler,K. *et al.* (2014) Immune evasion, stress resistance, and efficient nutrient acquisition are crucial for intracellular survival of Candida glabrata within macrophages. *Eukaryot. Cell*, **13**, 170–183.
21. Hosogaya,N., Miyazaki,T., Nagi,M., Tanabe,K., Minematsu,A., Nagayoshi,Y., Yamauchi,S., Nakamura,S., Imamura,Y., Izumikawa,K. *et al.* (2013) The heme-binding protein Dap1 links iron homeostasis to azole resistance via the P450 protein Erg11 in Candida glabrata. *FEMS Yeast Res.*, **13**, 411–421.
22. Devaux,F. and Thiébaut,A. (2019) The regulation of iron homeostasis in the fungal human pathogen Candida glabrata. *Microbiol. Read. Engl.*, **165**, 1041–1060.

23. Cassat,J.E. and Skaar,E.P. (2013) Iron in Infection and Immunity. *Cell Host Microbe*, **13**, 509–519.
24. Nairz,M., Schroll,A., Sonnweber,T. and Weiss,G. (2010) The struggle for iron—a metal at the host-pathogen interface: Iron at the host-pathogen interface. *Cell. Microbiol.*, **12**, 1691–1702.
25. Yilmaz,B. and Li,H. (2018) Gut microbiota and iron: the crucial actors in health and disease. *Pharmaceuticals*, **11**, E98.
26. Pfrimer,K., Micheletto,R.F., Marchini,J.S., Padovan,G.J., Moriguti,J.C. and Ferriolli,E. (2014) Impact of aging on urinary excretion of iron and zinc. *Nutr. Metab. Insights*, **7**, 47–50.
27. van Raaij,S.E.G., Srai,S.K.S., Swinkels,D.W. and van Swelm,R.P.L. (2019) Iron uptake by ZIP8 and ZIP14 in human proximal tubular epithelial cells. *Biometals*, **32**, 211–226.
28. Abreu,R., Quinn,F. and Giri,P.K. (2018) Role of the hepcidin-ferroportin axis in pathogen-mediated intracellular iron sequestration in human phagocytic cells. *Blood Adv.*, **2**, 1089–1100.
29. Gabaldón,T., Martin,T., Marcet-Houben,M., Durrens,P., Bolotin-Fukuhara,M., Lespinet,O., Arnaise,S., Boisnard,S., Aguileta,G., Atanasova,R. *et al.* (2013) Comparative genomics of emerging pathogens in the Candida glabrata clade. *BMC Genomics*, **14**, 623.
30. Edgar,R., Domrachev,M. and Lash,A.E. (2002) Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Res.*, **30**, 207–210.
31. Marisa,L., Ichanté,J.-L., Reymond,N., Aggerbeck,L., Delacroix,H. and Mucchielli-Giorgi,M.-H. (2007) MAnGO: an interactive R-based tool for two-colour microarray analysis. *Bioinform. Oxf. Engl.*, **23**, 2339–2341.
32. Ritchie,M.E., Phipson,B., Wu,D., Hu,Y., Law,C.W., Shi,W. and Smyth,G.K. (2015) limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res.*, **43**, e47.
33. Ringnér,M. (2008) What is principal component analysis? *Nat. Biotechnol.*, **26**, 303–304.
34. Falissard,B. (1996) A spherical representation of a correlation matrix. *J. Classif.*, **13**, 267–280.
35. Ashburner,M., Ball,C.A., Blake,J.A., Botstein,D., Butler,H., Cherry,J.M., Davis,A.P., Dolinski,K., Dwight,S.S., Eppig,J.T. *et al.* (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, **25**, 25–29.
36. Shannon,P., Markiel,A., Ozier,O., Baliga,N.S., Wang,J.T., Ramage,D., Amin,N., Schwikowski,B. and Ideker,T. (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, **13**, 2498–2504.
37. Skrzypek,M.S., Binkley,J., Binkley,G., Miyasato,S.R., Simison,M. and Sherlock,G. (2017) The Candida Genome Database (CGD): incorporation of Assembly 22, systematic identifiers and visualization of high throughput sequencing data. *Nucleic Acids Res.*, **45**, D592–D596.
38. Boyle,E.I., Weng,S., Gollub,J., Jin,H., Botstein,D., Cherry,J.M. and Sherlock,G. (2004) GO::TermFinder–open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinform. Oxf. Engl.*, **20**, 3710–3715.
39. Gasch,A.P., Spellman,P.T., Kao,C.M., Carmel-Harel,O., Eisen,M.B., Storz,G., Botstein,D. and Brown,P.O. (2000) Genomic expression programs in the response of yeast cells to environmental changes. *Mol. Biol. Cell*, **11**, 4241–4257.
40. Zhang,J., Vemuri,G. and Nielsen,J. (2010) Systems biology of energy homeostasis in yeast. *Curr. Opin. Microbiol.*, **13**, 382–388.
41. Eskes,E., Deprez,M.-A., Wilms,T. and Winderickx,J. (2018) pH homeostasis in yeast; the phosphate perspective. *Curr. Genet.*, **64**, 155–161.
42. Ayer,A., Gourlay,C.W. and Dawes,I.W. (2014) Cellular redox homeostasis, reactive oxygen species and replicative ageing in Saccharomyces cerevisiae. *FEMS Yeast Res.*, **14**, 60–72.
43. Millar,J.B.A. (2002) A genomic approach to studying cell-size homeostasis in yeast. *Genome Biol.*, **3**, REVIEWS1028.
44. Martínez-Pastor,MT., Marchler,G., Schüller,C., Marchler-Bauer,A., Ruis,H. and Estruch,F. (1996) The Saccharomyces cerevisiae zinc finger proteins Msn2p and Msn4p are required for transcriptional induction through the stress response element (STRE). *EMBO J.*, **15**, 2227–2235
45. Conde e Silva,N., Gonçalves,I.R., Lemaire,M., Lesuisse,E., Camadro,J.M. and Blaiseau,P.L. (2009) KlAft, the Kluyveromyces lactis ortholog of Aft1 and Aft2, mediates activation of iron-responsive transcription through the PuCACCC Aft-type sequence. *Genetics.*, **183**, 93–106.
46. Ihrig,J., Hausmann,A., Hain,A., Richter,N., Hamza,I., Lill,R. and Mühlenhoff,U. (2010) Iron regulation through the back door: iron-dependent metabolite levels contribute to transcriptional adaptation to iron deprivation in Saccharomyces cerevisiae. *Eukaryot. Cell*, **9**, 460–471.
47. Bao,W.G., Guiard,B., Fang,Z.A., Donnini,C., Gervais,M., Passos,F.M., Ferrero,I. and Fukuhara,H. (2008) Oxygen-dependent transcriptional regulator Hap1p limits glucose uptake by repressing the expression of the major glucose transporter gene RAG1 in Kluyveromyces lactis. *Eukaryot. Cell.*, **7**, 1895–1905.
48. Carreté,L., Ksiezopolska,E., Gómez-Molero,E., Angoulvant,A., Bader,O., Fairhead,C. and Gabaldón,T. (2019) Genome comparisons of Candida glabrata serial clinical isolates reveal patterns of genetic variation in infecting clonal populations. *Front. Microbiol.*, **10**, 112.

4. Pour aller plus loin : ouverture sur les levures du clade des *Nakaseomyces*

a. Contexte du projet

Dans la littérature, l'homéostasie du fer est un processus globalement conservé chez les levures, tout en gardant certaines spécificités (Devaux et al. 2019). Un point particulièrement intéressant serait donc d'étudier cette conservation chez des levures très proches sur le plan phylogénétique. Une hypothèse serait que plus les espèces sont proches phylogénétiquement, plus la conservation des gènes et de leur niveau d'expression dans une condition donné est importante. Comme nous l'avons vu, *C. glabrata* fait partie du clade des *Nakaseomyces*. Or, nous disposions au laboratoire de données de transcriptomique obtenues en condition de faible accès au fer pour 3 autres levures de ce clade : *Candida bracarensis*, *Nakaseomyces delphensis* et *Candida nivariensis*. Notre hypothèse scientifique était que les gènes réagissant au fer chez *C. glabrata* devaient, au moins en partie, réagir au fer au sein du clade, du fait de la faible distance phylogénétique entre les membres. Si ces gènes étaient importants pour la survie de *C. glabrata*, il devait y avoir de bonnes chances que ces mêmes gènes aient des fonctions conservées chez les autres levures du clade. L'intérêt de cette étude était renforcée par la présence de deux espèces de levures pathogènes au sein du clade : *C. bracarensis* et *C. nivariensis* (Angoulvant et al. 2016; Małek et al. 2018). L'étude n'étant pas terminée, les résultats que je vous présente dans ce chapitre sont encore préliminaires et nécessiteront des approfondissements et des confirmations.

b. Aperçu des levures *Candida bracarensis*, *Nakaseomyces delphensis* et *Candida nivariensis*.

Un point important est à garder à l'esprit : si la littérature est considérée comme peu abondante pour l'homéostasie du fer chez *C. glabrata*, elle est inexistante pour ces 3 levures (aucun article n'est retourné sur PubMed lorsque le nom de l'espèce est associé à *iron*[160]). Le Tableau 8 ci-

[160] Cette recherche a été faite le 12/05/2020.

dessous regroupe les différentes informations que j'ai pu collecter dans la littérature, de la base de données GRYC[161] et du livre *The Yeasts: A Taxonomic Study* .

	C. bracarensis	*C. nivariensis*	*N. delphensis*
Taille du génome	12 Mb	12 Mb	11 Mb
Statut de l'annotation	Partielle	Partielle	Partielle
Nombre de scaffolds	40	29	30
Nombre de chromosomes	12 (Gabaldón et al. 2013)	10–13 (Ahmad et al. 2014)	10 (Gabaldón et al. 2013)
Nombre de séquences codantes (CDS)	5265	5206	5112
Reproduction	Asexuée (Gabaldón et al. 2013)	Asexuée (Gabaldón et al. 2013)	Sexuée (Gabaldón et al. 2013)
Forme	Haploïde	Haploïde	Haploïde
Pathogène	Oui	Oui	Non
Décrit dans le flux sanguin	Oui (Warren et al. 2010; Campos-Garcia et al. 2019)	Oui (Campos-Garcia et al. 2019)	Non

Tableau 8 – Informations génomiques disponibles sur le site Internet de la base de données GRYC concernant C. bracarensis, C. nivariensis et N. delphensis.

[161] Ces données sont accessibles via la page de téléchargement : http://gryc.inra.fr/index.php?page=download [Accessible le 12/05/2020]

L'annotation partielle de ces 3 levures rend leur étude particulièrement délicate. Elles ont également été renommées depuis leur séquençage. Par exemple, la levure *Kluyveromyces delphensis* est devenue la levure *Nakaseomyces delphensis*. Un article propose un aperçu de l'évolution au sein du clade de *C. glabrata* et en particulier de ces 3 levures (Gabaldón et al. 2013).

c. Plan expérimental

Les données transcriptomiques ont été obtenues par la même approche que celle présentée pour *C. glabrata* (culture, condition de carence en fer, analyse différentielle, etc.). Nous disposions de 3 jeux de données, un par levure, obtenus à 30 °C en condition de carence en fer (BPS).

d. Résultats préliminaires

Pour déterminer si les gènes que nous avons observés comme réagissant au fer chez *C. glabrata*, réagissaient aussi au fer chez les 3 autres levures, nous avons commencé par rechercher les orthologues[162] entre les espèces. Nous avons pour cela construit une table d'orthologie pour chaque espèce à partir de données disponibles sur phylomeDB[163]. Il s'agit d'une base de données phylogénétique développée par T. Gabaldón (Huerta-Cepas et al. 2014). Lors de l'exploration des différents phylomes, nous pouvons trouver un extrait de l'arbre phylogénétique regroupant les 4 espèces d'intérêt (Figure 58). À noter que *N. delphensis* est sur une autre branche que les 3 autres *Candida*.

[162] La notion d'orthologue est traitée dans la ressource numérique de la thèse : https://thomasdenecker.github.io/thesisWebsite/annexes/orthologue/ [Accessible le 10/08/202020]

[163] http://www.phylomedb.org/ [Accessible le 12/05/2020]

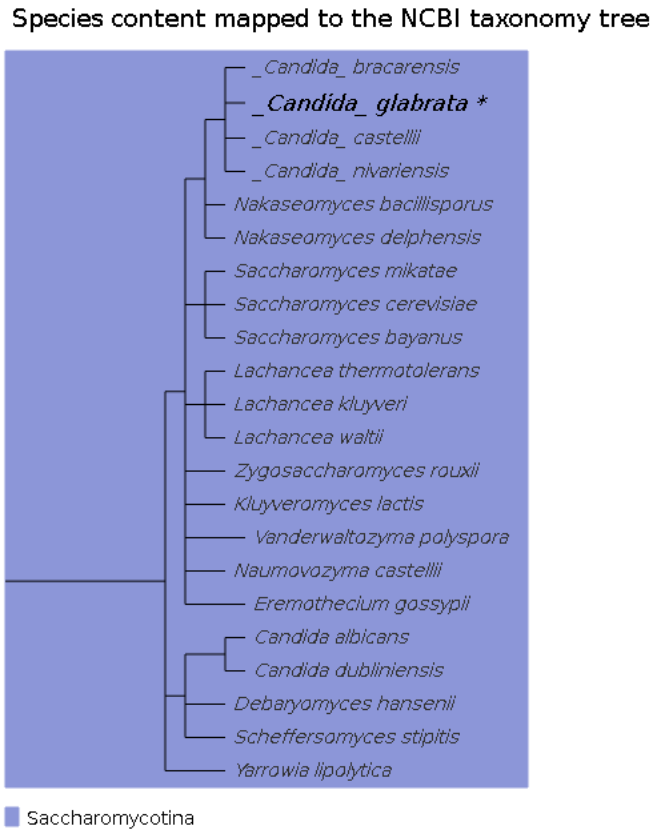


Figure 58 – Arbre phylogénétique composé des 4 levures d'intérêt : C. glabrata, C. bracarensis, C. nivariensis et N. delphensis extrait du site phylomeDB.

À partir des tables d'orthologie construites, nous avons trouvé un orthologue pour la plupart des gènes réagissant au fer chez *C. glabrata* dans chacune des 3 espèces *C. bracarensis*, *C. nivariensis* et *N. delphensis* (Figure 59). Pour les gènes n'ayant pas d'orthologues, nous avons regardé leur type (Type I, Type II ou aucun des deux). Le premier constat est qu'un orthologue a été trouvé pour quasiment tous les gènes réagissant au fer. Nous pouvons déjà imaginer une bonne conservation de ces gènes au sein des différentes espèces. Le second constat est que la répartition des types des gènes ne possédant pas d'orthologues est plutôt constante. Nous pouvons donc nous demander si ces gènes sont toujours les mêmes.

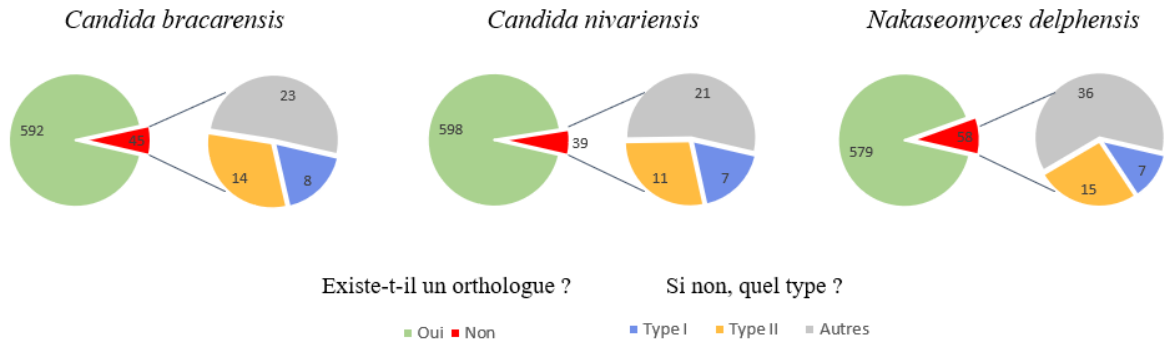


Figure 59 – Nombre d'orthologues trouvés entre les gènes réagissant au fer chez C. glabrata et une des 3 espèces : C. bracarensis, C. nivariensis et N. delphensis (diagramme en secteur de gauche). Si aucun orthologue, nous avons regardé le type (diagramme en secteur de droite).

Pour répondre à cette question, la Figure 60 a été réalisée à l'aide du *package* UpsetR. Il permet de croiser des listes de gènes et de représenter les intersections de façon plus lisible qu'un diagramme de Venn. Nous avons pu constater que sur les 70 gènes qui réagissaient au fer chez *C. glabrata* qui n'avaient pas d'orthologues dans au moins une des 3 espèces, 30 gènes n'étaient jamais retrouvés chez les 3 espèces étudiées. Un point intéressant était que plus de 50% de ces gènes avaient été classés dans la fonction générale 'Others' du sous-réseau de co-expression. En considérant les 70 gènes sans orthologue pour au moins une des 3 espèces, nous étions à plus de 60%. Une hypothèse est qu'il s'agit de gènes impliqués dans la réponse au stress. Un dernier point, nous avons pu constater que 19 gènes n'avaient pas d'orthologue spécifiquement chez *N. delphensis*. Ces gènes font peut-être partie des gènes expliquant le changement de branche entre les 3 *Candida* et cette levure.

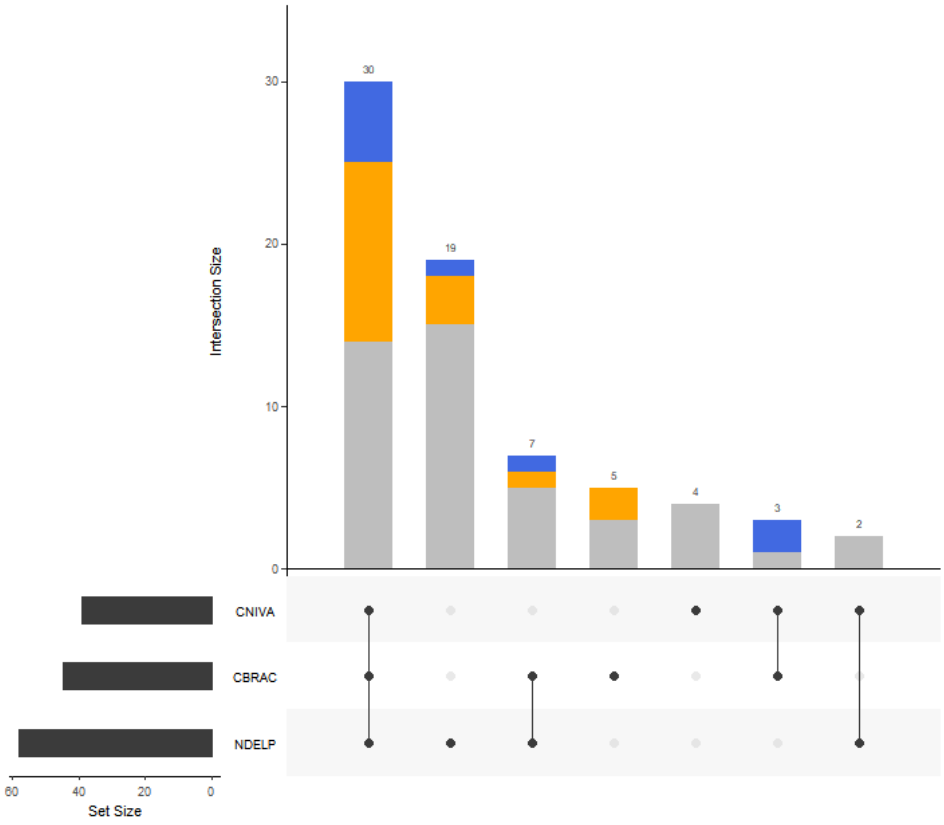
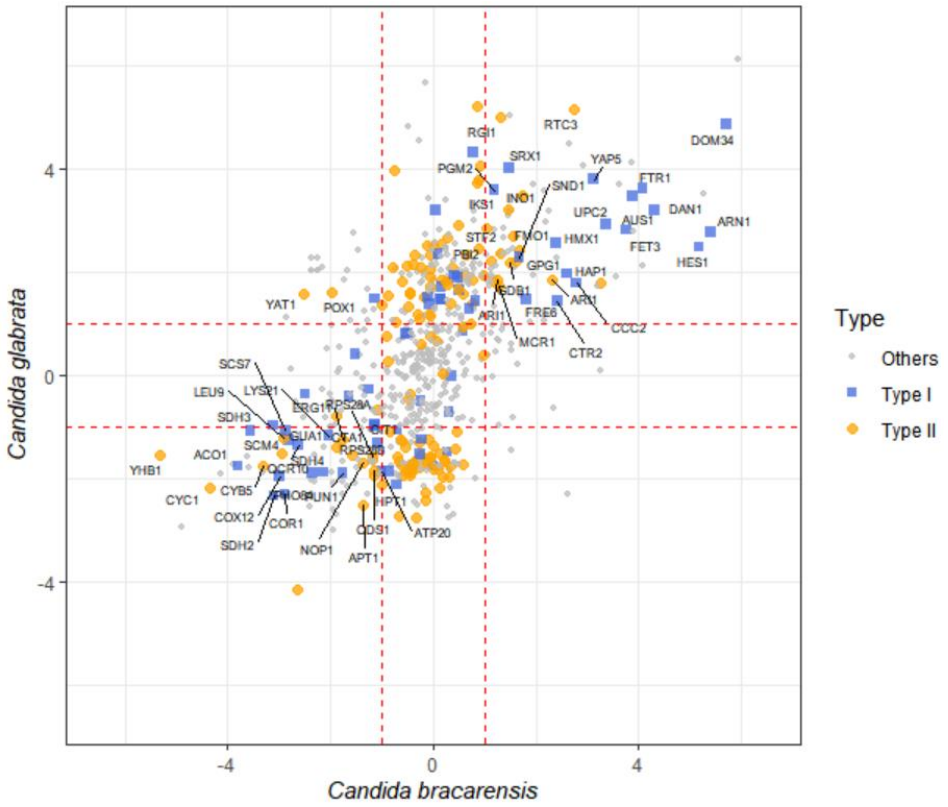


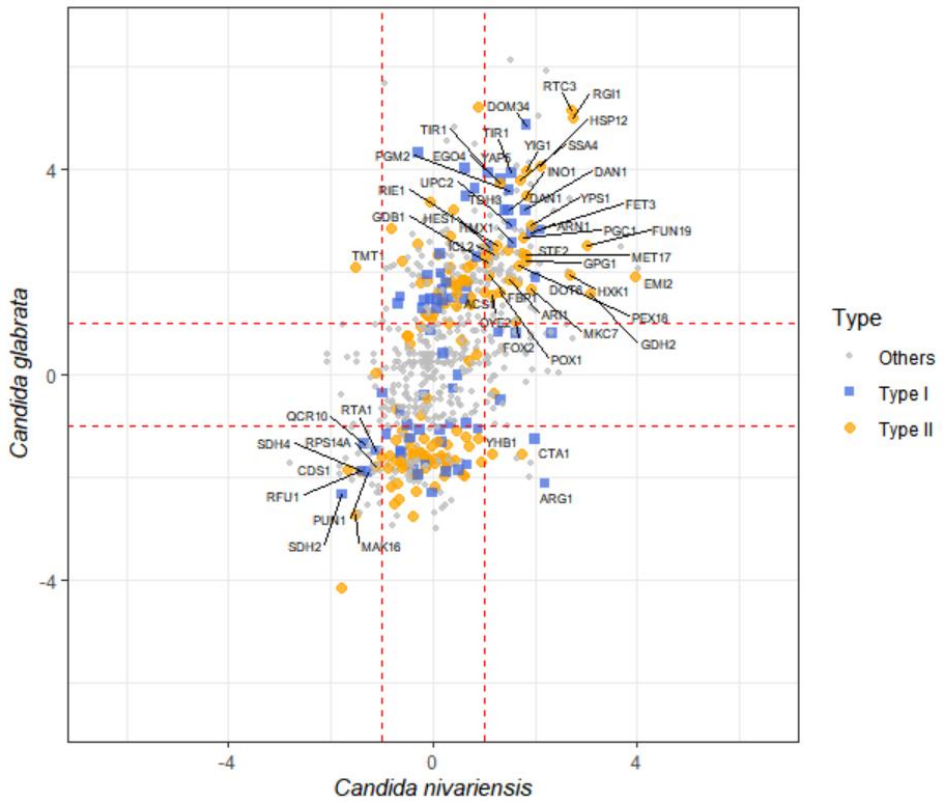
Figure 60 – Résultats du croisement des listes de gènes ne possédant pas d'orthologues chez C. glabrata et faisant partie de la liste des gènes réagissant au fer chez C. glabrata. Les abréviations CNIVA, CBRAC et NDELP correspondent respectivement aux espèces C. nivariensis, C. bracarensis et N. delphensis. Chaque barre correspond au nombre de gènes communs aux listes sélectionnées (identifiées en dessous par des points noirs). Par exemple, la barre la plus à gauche correspond au croisement des listes des 3 espèces qui partagent 30 gènes. Les couleurs présentes sur les barres correspondent aux types des gènes réagissant au fer chez C. glabrata (bleu pour les gènes de types I, orange pour les gènes de types II et gris pour les autres).

Une fois que nous avions des orthologues pour presque tous les gènes réagissant au fer chez *C. glabrata*, nous avons souhaité savoir si les gènes se comportaient de la même façon chez les autres membres du clade des *Nakaseomyces*. Nous avons donc comparé les logFCs de *C. glabrata* des gènes réagissant au fer (en condition C1 : 30 °C + BPS) avec les logFCs des orthologues dans chacune des 3 espèces. La Figure 61 a ainsi été obtenue. Plusieurs constats peuvent être faits à partir de ces graphiques. Le premier est que si le gène n'est pas différentiellement exprimé chez *C. glabrata*, l'orthologue ne l'est pas non plus chez les autres levures. Pour ceux qui le sont chez *C. glabrata* et chez les autres levures, nous pouvons remarquer que ce sont des gènes clés de l'homéostasie du fer que nous avons déjà évoqués et qui sont de nombreuses fois décrits chez *S. cerevisiae*, *C. albicans* ou *C. glabrata*. Nous pouvons citer par exemple : *FTR1, FET3, CCC2, HMX1, ACO1, HAP1, FRE6, YAP5*, etc.

C. bracarensis



C. nivariensis



N. delphensis

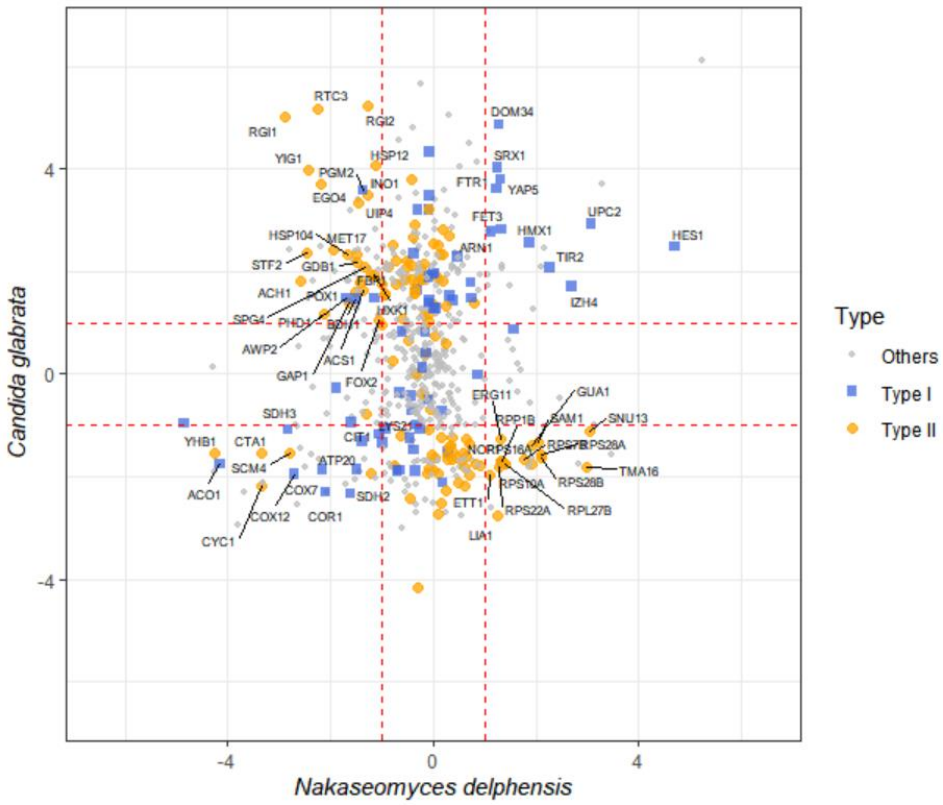


Figure 61 – Nuages de points avec sur l'axe des ordonnées les logFC de gènes réagissant au fer chez C. glabrata et en abscisse les logFC des orthologues dans l'une des 3 espèces étudiées : C. bracarensis, C. nivariensis et N. delphensis. Le point bleu à la forme carrée sont les gènes de Type I dans notre étude, les oranges à la forme ronde les type II et gris à la forme ronde les gènes qui ne sont ni Type I, ni Type II. Les gènes différentiellement exprimés chez C. glabrata et dans la levure étudiée (abs(logFC) > 1) sont nommés sur les nuages de points.

À noter que l'expérience *N. delephensis* n'a pas très bien fonctionné (l'intensité lumineuse des puces était globalement faible) justifiant la différence visuelle générale par rapport aux deux autres levures. Néanmoins, nous mettons en évidence *FTR1, FET3* et *ACO1* qui sont des gènes clés de l'homéostasie du fer chez *C. glabrata*. Si nous regardons les gènes qui sont dérégulés de la même façon entre *C. glabrata* et *C. bracarensis* et *C. nivariensis*, nous trouvons une liste de 73 gènes dont les fonctions générales sont dominées par des fonctions clés dans l'homéostasie du fer (Figure 62).

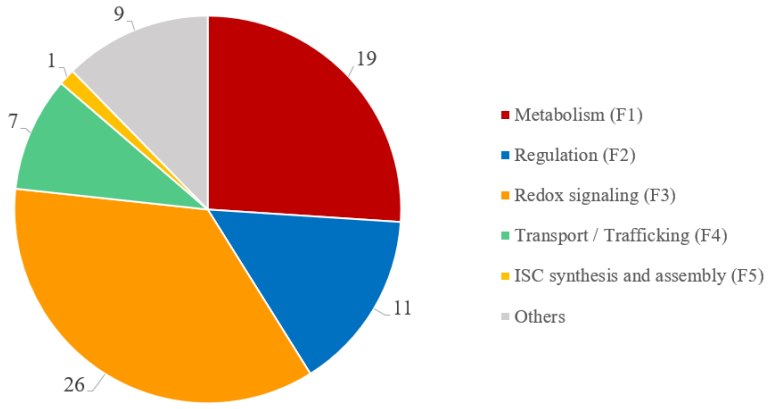


Figure 62 – Répartition des gènes réagissant au fer chez C. glabrata, C. nivariensis et C. bracarensis dans les grandes fonctions créées chez C. glabrata.

En conclusion, cette étude préliminaire nous permet d'envisager une conservation de gènes clés dans l'homéostasie du fer entre *C. glabrata* et d'autres levures du clade des *Nakaseomyces* : *C. nivariensis*, *C. bracarensis* et *N. delphensis*. Une valorisation de ce travail par l'équipe de C. Fairhead est actuellement discutée.

5. Pour conclure, de nouvelles stratégies thérapeutiques impliquant le fer ?

La disponibilité du fer chez l'hôte a une influence directe sur la gravité de l'infection. Une dérégulation de l'homéostasie du fer (de l'hémochromatose à la carence en fer) peut augmenter la sensibilité de l'hôte à l'infection (Kumar et al. 2010). En cas de carence en fer chez l'hôte, il a été montré qu'il y avait une prévalence plus importante de candidose buccale (Lu 2016). Lors des infections systémiques à *Candida*, une réaction inflammatoire est générée et celle-ci induit une carence en fer. À l'inverse, une surcharge en fer peut quant à elle favoriser une croissance microbienne (Fourie et al. 2018; Cassat et al. 2013). En fonction du type de micro-organismes pathogènes, une piste thérapeutique serait de réguler la concentration en fer environnementale. Dans le cadre des infections à *Candida*, les chercheurs explorent la mise en place d'un environnement pauvre en fer induisant une carence en fer intracellulaire chez la levure et donc sa mort. Pour générer un environnement avec une concentration faible en fer, plusieurs pistes sont explorées principalement chez *C. albicans*. La première est l'utilisation de chélateurs du fer comme a pu le faire l'équipe de Meyer lors d'une co-infection VIH / *Mycobacterium tuberculosis* (Meyer 2006) et qui a été testée récemment chez *C. albicans* (Puri et al. 2019). Ces

chélateurs peuvent ralentir le développement de maladies inflammatoires, avoir des propriétés antivirales, peuvent être utilisés dans le traitement des tumeurs malignes et comme traitement alternatif lors de résistances aux antibiotiques (Meyer 2006; Thompson et al. 2012; Lehmann et al. 2015). Même si les chélateurs du fer semblent être une piste prometteuse dans le traitement de certaines infections, ils peuvent également avoir l'effet inverse sur d'autres. Les chélateurs chargés en fer peuvent être capturés par les micro-organismes qui utilisent ensuite certains mécanismes pour libérer le fer. De plus, en fonction des chélateurs utilisés, ils peuvent avoir un impact sur l'hôte comme une carence en fer importante. De nombreux travaux ont été menés pour contourner ces problèmes et sont motivés par une absence constatée d'une éventuelle résistance aux chélateurs du fer. En effet, des résistances aux antifongiques ont commencé à apparaître (Santos et al. 2018) et de façon plus étonnante certains antifongiques ont tendance à favoriser certaines levures. Une étude menée en France montre par exemple que l'utilisation de fluconazole ou de capsofungine a tendance à promouvoir *C. glabrata* (Lortholary et al. 2011). D'autres pistes ont aussi été explorées notamment chez *C. albicans* comme :

- L'utilisation d'un anticorps monoclonal bloquant une voie d'absorption réductrice du fer et induisant une carence en fer intracellulaire chez la levure (Brena et al. 2011) ;
- L'administration de lactoferrine qui conduirait à une séquestration importante du fer et donc réduirait la disponibilité en fer à la levure (Kuipers et al. 2002; Bai et al. 2006; Velliyagounder et al. 2015) ;
- L'administration de transferrine a montré dans plusieurs études la diminution de la croissance microbienne et aucune émergence d'une forme de résistance (Han 2014; Lin et al. 2014) ;
- L'induction d'une carence en fer en association à un traitement antifongique. Il a été montré que l'utilisation d'un chélateur générait un stress chez la levure et une meilleure sensibilité aux traitements antifongiques (Prasad et al. 2006; Fiori et al. 2012; Savage et al. 2018).

Les recherches portent essentiellement sur *C. albicans* qui est la première levure pathogène. Cependant, il est à noter qu'une étude a été menée chez *C. glabrata* et avance de bons résultats pour un traitement basé sur le fer (Hirai et al. 2016).

III. Étude systématique des modifications post-traductionnelles des protéines chez *Candida albicans*

1. L'introduction générale

a. Etude du protéome par spectrométrie de masse

Le protéome correspond à l'ensemble des protéines synthétisées à partir de l'expression du génome, à un moment donné, pour une condition donnée[164]. La composition du protéome est extrêmement dynamique. Ainsi, il a été montré chez *S. cerevisiae* que le nombre de protéines individuelles par cellule peut varier de quelques milliers à quelques millions (Ho et al. 2018). Pour mieux comprendre ces variations, il est indispensable d'avoir à l'esprit que l'état du protéome reflète les répercussions des événements cellulaires tant au niveau traductionnel que post-traductionnel (modifications post-traductionnelles des protéines).

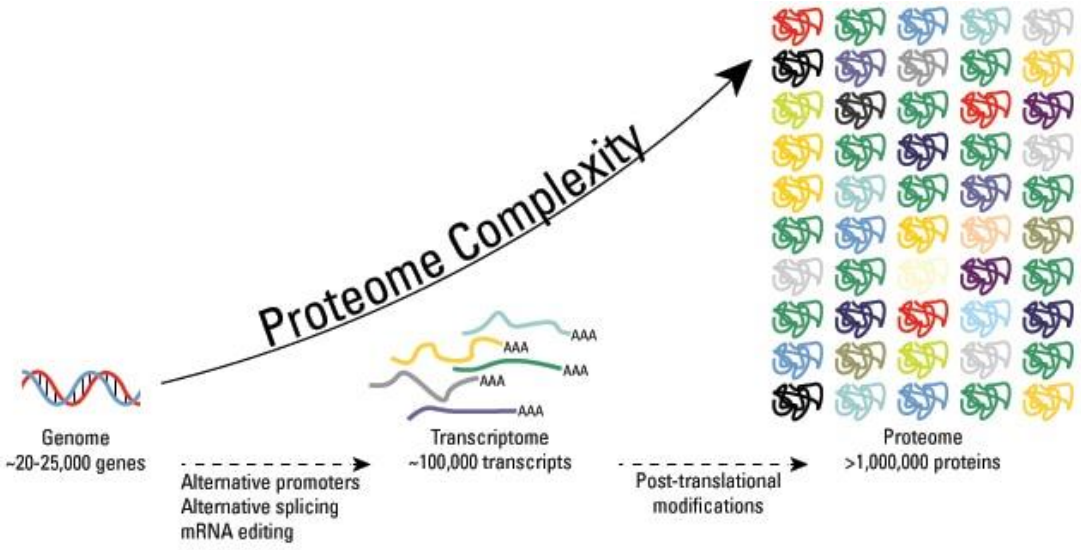


Figure 63 – De nombreuses modifications se produisent entre le génome et le protéome. L'accumulation de toutes ces modifications augmente considérablement la complexité du protéome. Cette illustration est extraite du site de Thermo Fishier scientific[165] et concerne l'Homme.

[164] Une ressource numérique a été mise en place pour tracer l'évolution de l'acide aminé jusqu'au protéome : https://thomasdenecker.github.io/thesisWebsite/annexes/aaProteome/ [Accessible le 14/08/2020]

[165] https://www.thermofisher.com/fr/fr/home/life-science/protein-biology/protein-biology-learning-center/protein-biology-resource-library/pierce-protein-methods/overview-post-translational-modification.html [Accessible le 18/05/2020]

L'étude du protéome (protéomique) est donc complexe[166]. Elle a pour objectif d'identifier les protéines présentes dans un échantillon obtenu dans des conditions particulières et éventuellement les quantifier. Pour cela, la spectrométrie de masse est devenue une technologie incontournable ces dernières années en biologie. Cette approche permet d'obtenir une bonne couverture du protéome dans un délai relativement court (80% des protéines de *Saccharomyces cerevisiae* peuvent être identifiées et quantifiées en moins de 24h (Richards et al. 2015)).

À partir d'un échantillon dont nous souhaitons connaître la composition, la manière la plus évidente serait de séparer les protéines présentes dans l'échantillon puis de les analyser dans un spectromètre de masse une par une pour les identifier. Bien que très simple conceptuellement, les nombreuses difficultés techniques de cette approche dite *Top down* ont poussé les chercheurs à se tourner vers une approche dite *Bottom up* (ou encore *Shotgun*). Cette approche repose sur une stratégie ascendante : les protéines sont digérées (souvent par de la trypsine) pour obtenir des peptides (beaucoup plus courts que la protéine entière). La majorité des peptides produits par la trypsine ont une longueur comprise entre 4 et 26 acides aminés. Leur masse est comprise dans une gamme de masses d'environ 450-3000 Da, ce qui est idéal pour un spectromètre de masse[167]. Ces peptides sont ensuite séparés, identifiés et utilisés pour déduire quelles protéines se trouvaient initialement dans l'échantillon. La technique la plus courante aujourd'hui pour mettre en place cette approche est de réaliser une spectrométrie de masse en tandem LC-MS/MS.

b. Spectrométrie de masse en tandem LC-MS/MS : principes et composants

Les principes et les composants d'un spectromètre de masse en en tandem LC-MS/MS sont détaillés dans la ressource numérique de la thèse[168]. En résumé, lors d'une analyse par spectrométrie de masse en tandem LC-MS/MS, un échantillon biologique est mis en présence d'une enzyme de clivage (le plus souvent la trypsine) permettant d'obtenir des peptides. Ces peptides sont ensuite séparés par une chromatographie liquide (le LC de LC-MS/MS). Une

[166] Au cours de mes études, j'ai souvent entendu « *étudier les protéines, c'est forcément plus complexe que les gènes, on ne se limite pas à 4 composants !* ».

[167] Pour information, la gamme de masse explorée sur la plateforme de l'Institut Jacques Monod est comprise entre 400 et 2000 Da.

[168] https://thomasdenecker.github.io/thesisWebsite/annexes/spectroMass/ [Accessible le 10/08/2020]

première étape de spectrométrie de masse (MS1) permet de déterminer le poids moléculaire de chaque peptide. Les peptides qui présentent la plus grande intensité et abondance sont alors sélectionnés puis fragmentés dans une cellule de collision. Pour chaque peptide fragmenté, une nouvelle étape de spectrométrie de masse est réalisée (MS2) pour obtenir un spectre de masse des fragments. À partir de ces spectres, il sera possible de déduire la séquence du peptide et ainsi remonter jusqu'à la protéine d'origine. L'ensemble des étapes sont résumées à la Figure 64.

Figure 64 – Résumé des différentes étapes pour identifier des protéines par l'approche Bottom up.

c. Contexte du projet

Difficultés d'identification des protéines

Théoriquement, la spectrométrie de masse devrait être capable d'identifier et de quantifier les protéines dans des mélanges / échantillons biologiques complexes. Dans la réalité, elle n'est pas en mesure d'y arriver en raison de nombreuses difficultés rencontrées à toutes les étapes en commençant par la conception expérimentale jusqu'à l'analyse de données (Prakash et al. 2007). Plusieurs causes peuvent expliquer l'absence d'identification pour une protéine présente dans les cellules étudiées :

- La protéine n'a pas été extraite – Certaines protéines sont plus complexes que d'autres à extraire comme notamment les protéines transmembranaires. À moins de mettre en place un protocole particulier, ces protéines sont souvent éliminées lors de l'extraction ;
- La protéine n'est pas suffisamment présente pour être détectée – Dans ce cas la protéine a bien été extraite mais en faible quantité. Certaines protéines sont tellement abondantes qu'elles « masquent » les protéines moins abondantes. Les protéines faiblement abondantes deviennent alors « invisibles » car en dessous du seuil de détectabilité/ sensibilité du spectromètre de masse dans une analyse non ciblée LC-MS/MS ;
- Aucun spectre de masse n'a permis d'identifier la protéine – À l'heure actuelle, il a été constaté sur la plateforme de protéomique de l'Institut Jacques Monod qu'un spectre de

masse sur deux ne conduit pas à une identification (pas d'association peptide protéine). Une de nos intuitions porte sur la mauvaise prise en compte des modifications post-traductionnelles qui pourraient être présentes dans les spectres de masses, ne conduisant pas à une identification.

Le travail que j'ai mené dans l'équipe de JM. Camadro portait sur cette hypothèse. **Pouvons-nous améliorer le taux d'identification des protéines si nous prenons en compte de façon systématiques les modifications post-traductionnelles ?**

Modifications post-traductionnelles

Une modification post-traductionnelle est « *une modification biochimique qui se produit sur un ou plusieurs acides aminés d'une protéine après que la protéine ait été traduite par un ribosome.* »[169] (Carter et al. 2015). Ces modifications sont très variées et jouent un rôle fondamental dans la régulation des processus cellulaires. Le plus fréquemment, ces modifications post-traductionnelles sont la conséquence d'une activité enzymatique. Il a été estimé que 5% du protéome comprend des enzymes qui effectuent plus de 200 types de modifications post-traductionnelles (Thermo Fisher Scientific 2015). En fonction de l'enzyme, les modifications post-traductionnelles peuvent être séparées en deux grandes catégories (Walsh et al. 2005) :

- Addition ou retrait d'un groupement chimique par des enzymes comme des kinases, des phosphatases, des transférases et des ligases ;
- Clivage de liaisons peptidiques pour enlever des séquences spécifiques ou des sous-unités de régulation par l'action de protéases ou plus rarement par clivage autocatalytique ;
- Certaines modifications post-traductionnelles sont réversibles (comme la phosphorylation) permettant un contrôle rapide et économique (d'un point de vue bioénergétique) de la fonction des protéines (Bürkle 2001). Elles permettent au protéome d'être dynamique et se modifient en fonction des conditions. Après la traduction, elles permettent par exemple un repliement correct de la protéine ou sa

[169] *"a biochemical modification that occurs to one or more amino acids on a protein after the protein has been translated by a ribosome.*"

redirection vers le bon compartiment cellulaire. Une fois la protéine à la bonne localisation, les modifications post-traductionnelles permettront par exemple une activation / inactivation de l'activité catalytique de la protéine. Plus de 300 modifications post-traductionnelles chez les eucaryotes sont aujourd'hui décrites (Csizmok et al. 2018).[170].

Conséquences des modifications post-traductionnelles pour la spectrométrie de masse

La conséquence des modifications post-traductionnelles est une modification du poids de l'acide aminé modifié. Le Tableau 9 regroupe par exemple les modifications post-traductionnelles présentes dans le programme d'identification RAId pour le résidu glycine dont la masse est 57.02 Da.

Nom de la modification post-traductionnelle	Masse après la modification post-traductionnelle	Différence de masse (Da)	Localisation
Cholesterol glycine ester	425.37	+ 368.35	C-terminal
N-formylglycine	85.02	+ 28	N-terminal
1-thioglycine	73	+ 15.98	Dans la protéine
Phosphatidylethanolamine amidated glycine	756.54	+ 699.52	C-terminal
N-acetylglycine	99.03	+ 42.01	N-terminal
Glycine amide	56.04	- 0.98	C-terminal
N-palmitoyl glycine	295.25	+ 238.23	N-terminal
N-myristoyl glycine	267.22	+ 210.2	N-terminal

Tableau 9 – Liste des modifications post-traductionnelles renseignées dans le programme d'identification RAId. En fonction de la modification post-traductionnelle, la différence de masse est plus ou moins importante. Ces différences sont détectables par le spectromètre de masse. Les masses sont exprimées en Da.

[170] Les plus fréquentes sont décrites dans la ressource numérique de la thèse https://thomasdenecker.github.io/thesisWebsite/annexes/ptms/ [Accessible le 10/08/2020]

Les spectromètres de masse de la plateforme sont capables de détecter ces changements[171]. Par conséquent, comme nous pouvons le voir sur la Figure 65, pour deux peptides identiques, les spectres sont différents lorsqu'il y a une modification post-traductionnelle (S1A présente une modification post-traductionnelle sur une méthionine en rouge).

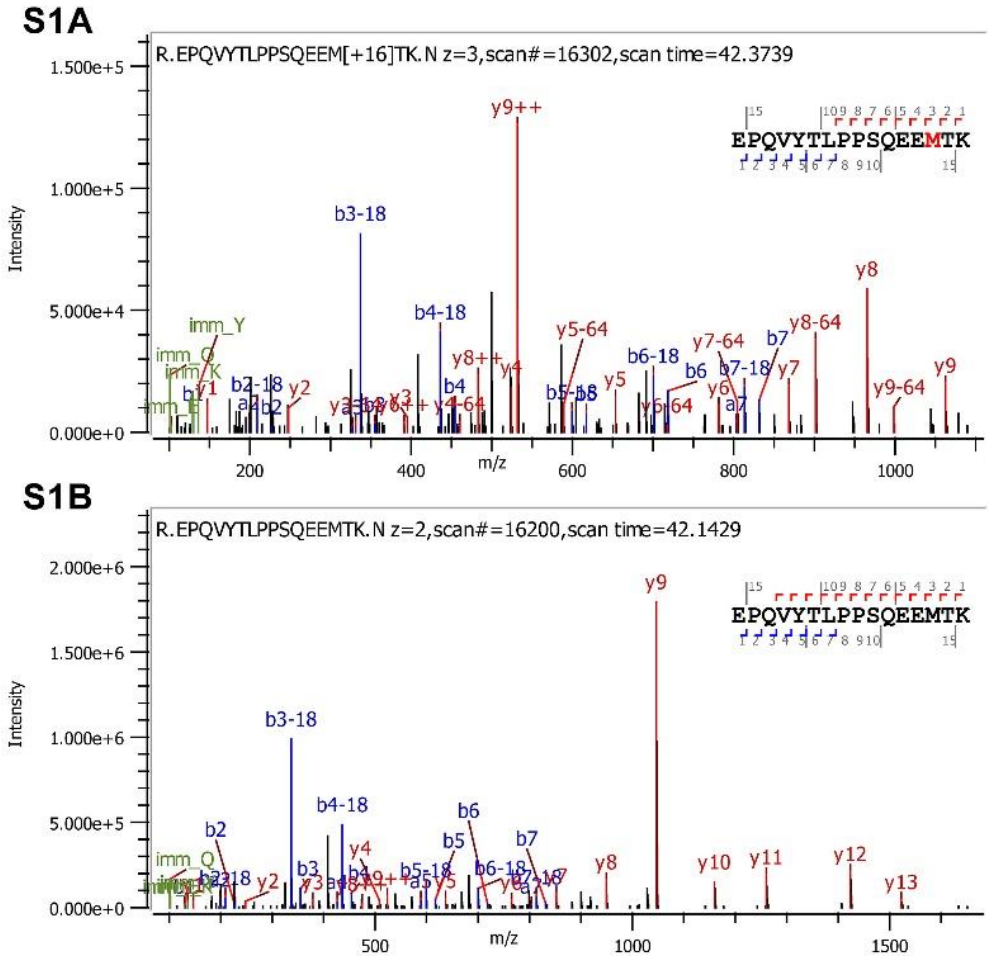


Figure 65 – Spectres de masses pour un peptide avec une modification post-traductionnelle (oxydation de la méthionine - en haut) et le même peptide sans modification post-traductionnelle (en bas). Ces spectres sont issus de l'article de Xu et al [c]. De nombreux autres exemples sont disponibles en figures supplémentaires de cet article.

Comme cela est décrit dans la ressource numérique de la thèse[172], l'identification des peptides passe par une comparaison avec une banque de spectres théoriques. Si lors de la construction

[171] Il est capable de détecter une différence de 10^{-5} Da

[172] https://thomasdenecker.github.io/thesisWebsite/annexes/spectroMass/#identification-des-protéines-à-partir-des-peptides [Accessible le 10/08/2020]

de cette banque, il n'a pas été précisé qu'il pouvait y avoir des modifications post-traductionnelles, ses spectres théoriques n'en contiendront pas. Par conséquent, il n'y aura pas de concordance possible entre les spectres théoriques de la banque et les spectres expérimentaux contenant des modifications post-traductionnelles et donc une absence d'identification. Pour éviter ce type de problème, une idée simple serait de chercher systématiquement toutes les modifications post-traductionnelles. Dans la pratique, cela n'a pas effectué tout simplement parce que les temps de calculs seraient trop importants. Aujourd'hui, seules les 3 modifications post-traductionnelles les plus fréquentes sont testées et nécessitent 1 à 2 heures pour obtenir les résultats avec Proteome discoverer 2.4 (Thermo Scientific) pour une expérience.

Pleinement conscients de la sous-exploitation de plusieurs téraoctets de données, nous souhaitions explorer une nouvelle approche permettant de systématiser la recherche des modifications post-traductionnelles et ainsi augmenter le taux d'identification des protéines. Pour cela, nous avons utilisé l'outil d'une équipe collaboratrice, RAId, réputé pour être rapide, performant et gérant l'identification à partir de peptides contenant des modifications post-traductionnelles. Dans ce chapitre, nous verrons comment nous avons traité les fichiers de sortie RAId pour les rendre exploitables (génération de données) et je présenterais quelques résultats préliminaires obtenus lors de l'analyse exploratoire des données (création d'informations).

2. Le matériel et les méthodes

a. Données collectées

Pour réaliser cette étude, nous avons utilisé des fichiers bruts .RAW[173] obtenus par spectrométrie de masse sur la plateforme de protéomique de l'institut Jacques-Monod. Les spectromètres de masse utilisés sont le spectromètre de masse hybride quadripôle Orbitrap Q Exactive™ plus et le spectromètre de masse Tribrid™ Orbitrap Fusion™ de Thermo Scientific™. L'organisme étudié est *Candida albicans*. La Table 1 regroupe les différentes conditions expérimentales étudiées.

[173] Le format RAW est un format propriétaire de Thermo Scientific. Il est présenté en annexe.

Nom du fichier	Forme	Informations complémentaires
1703006-F1	Levure	Forme levure cultivée sur carbone normal
1703006-F1_170124045953	Levure	Forme levure cultivée sur carbone normal
1703006-F1_170124224955	Levure	Forme levure cultivée sur carbone normal
1703006-F2	Hyphe	Forme hyphe cultivée sur carbone normal
1703006-F2_170124092722	Hyphe	Forme hyphe cultivée sur carbone normal
1703006-F2_170125031736	Hyphe	Forme hyphe cultivée sur carbone normal
1703006-F3	Levure	Forme levure cultivée sur carbone C12.[174]
1703006-F3_170124135447	Levure	Forme levure cultivée sur carbone C12
1703006-F3_170125074531	Levure	Forme levure cultivée sur carbone C12
1703006-F4	Hyphe	Forme hyphe cultivée sur carbone C12
1703006-F4_170124182218	Hyphe	Forme hyphe cultivée sur carbone C12
1703006-F4_170125121326	Hyphe	Forme hyphe cultivée sur carbone C12
1913001-Q1	Levure	Réplicats biologiques. Il s'agit du même échantillon que 1847003
1913001-Q2	Hyphe	Réplicats biologiques. Il s'agit du même échantillon que 1847003
1913001-Q3	Levure	Réplicats biologiques. Il s'agit du même échantillon que 1847003
1913001-Q4	Hyphe	Réplicats biologiques. Il s'agit du même échantillon que 1847003
1913001-Q5	Levure	Réplicats biologiques. Il s'agit du même échantillon que 1847003
1913001-Q6	Hyphe	Réplicats biologiques. Il s'agit du même échantillon que 1847003
1847003-Q1	Levure	Levure
1847003-Q2	Levure	Levure + SAHA (inhibiteur de la protéine déacetylase)
1847003-Q3	Hyphe	Hyphe
1847003-Q4	Hyphe	Hyphe + SAHA (inhibiteur de la protéine déacetylase)
1847003-Q5	Levure	Levure
1847003-Q6	Levure	Levure + SAHA (inhibiteur de la protéine déacetylase)
1847003-Q7	Hyphe	Hyphe
1847003-Q8	Hyphe	Hyphe + SAHA (inhibiteur de la protéine déacetylase)

[174] L'utilisation d'un milieu riche en C12 permet d'augmenter l'intensité du pic monoisotopique et de limiter la présence d'isotopologues pour chaque peptide étudié.

1847003-Q9	Levure	Levure
1847003-Q10	Levure	Levure + SAHA (inhibiteur de la protéine déacetylase)
1847003-Q11	Hyphe	Hyphe
1847003-Q12	Hyphe	Hyphe + SAHA (inhibiteur de la protéine déacetylase)

Tableau 10 – Liste des fichiers RAW étudiés avec différentes caractéristiques expérimentales. Les lignes rouges correspondent à la forme hyphe et les lignes bleues à la forme levure.

b. Logiciel RAId

RAId pour *Robust Accurate Identification,* est un ensemble de programmes *Open source*[175] permettant l'analyse de données de spectrométrie de masse en tandem avec des statistiques précises (G. Alves et al. 2005; 2010; 2007). Il a été développé par l'équipe *Quantitative Molecular Biological Physics* (NCBI) dirigée par Y. Yu. La grande qualité de l'outil d'identification de protéines proposé est qu'il repose sur des statistiques solides[176]. Il est aussi capable d'intégrer 236 modifications post-traductionnelles dans la recherche. RAId fait partie des outils d'identification hybrides[177]. Il détermine la séquence des peptides à partir des spectres de masses, filtre la base de données de peptides théoriques créée et compare les spectres. Chaque identification est ensuite associée à un score de qualité. RAId peut être utilisé sur Internet[178] ainsi qu'en local à l'aide d'une interface graphique (G. Alves et al. 2008; Joyce et al. 2018; Ogurtsov et al. 2019).

RAId présente aussi de nombreux avantages très utiles pour notre projet. Il est capable de traiter des fichiers RAW. Il n'est donc pas nécessaire de les convertir au format mzml comme la plupart des algorithmes d'identification actuels et ainsi risquer de perdre de l'information. Ensuite, il est possible de paralléliser[179] l'identification et ainsi diminuer considérablement le

[175] Le prix de licence de Proteome Discoverer, l'outil utilisé actuellement, est très élevé.

[176] Pour les utilisateurs, l'équipe de Y.K. Yu a amélioré les statistiques de BLAST (Gerts et al. 2006).

[177] Les différentes approches d'identification sont détaillées dans la ressource numérique https://thomasdenecker.github.io/thesisWebsite/annexes/spectroMass/ [Accessible le 10/08/2020]. En résumé, le principe d'une identification hybride est de déterminer la séquence peptidique à partir d'un spectre de masse pour (1) réaliser une sous-sélection de spectres de masse dans une banque de données de spectres de masse théoriques puis (2) de comparer le spectre expérimental avec ces spectres sélectionnés.

[178] https://www.ncbi.nlm.nih.gov/CBBresearch/Yu/raid/index.html [Accessible le 25/05/2020]

[179] Technique informatique permettant de traiter des données simultanément.

temps d'analyse. Enfin comme évoqué précédemment, RAId dispose d'une liste de 236 modifications post-traductionnelles préenregistrées. Cette liste peut être complétée très facilement notamment à partir des autres modifications post-traductionnelles disponibles sur le site UNIMOD[180] (plus de 1500 modifications post-traductionnelles y sont recensées). La Figure 66 est un exemple d'une modification post-traductionnelle enregistrée dans RAId.

```
ID    Glutathionylation
AC    C28
TG    Cysteine
RW    103.009186
MW    408.07727
PA    Amino acid side chain.
PP    Anywhere.
CF    None
MM    305.068081
KY    None
LT    None
```

Figure 66 – Exemple des informations renseignées pour la modification post-traductionnelle Glutathionylation dans RAId.

RAId présente cependant deux limites. La première est qu'il n'est pas simple à installer en local. Pour remédier à ce problème et pour des questions de reproductibilité, nous avons créé une image Docker contenant RAId (prochainement disponible sur DockerHub). La seconde limite réside dans les fichiers de sorties générés à chaque identification (recherche d'une modification post-traductionnelle dans un fichier RAW). Ils ne sont pas aux formats standards et toutes les données générées sont dispersées dans 4 fichiers différents. Le Tableau 11 regroupe certaines des données les plus importantes que nous avons extraites des fichiers de sortie.

[180] http://www.unimod.org/modifications_list.php [Accessible le 25/05/2020]

Description	Exemple
Numéro de passage dans le scan	9745
Masse du peptide par rapport à sa charge	1791.912
Etat de charge du peptide	2
Temps de rétention	1492.2
Score RAId de qualité de l'identification	4.447 e-06
Acide aminé présent dans la protéine avant le peptide	K
Acide aminé présent dans la protéine après le peptide	N
Position de début du peptide dans la protéine	12
Position de fin du peptide dans la protéine	29
Protéine identifiée	A0A1D8PDD1
Taille de la protéine	308
Description de la protéine	Lipid-binding protein
Séquence de la protéine	MHRTYSLRSSKAPTASQLQSPPPPPSS ….
Séquence du peptide avec un PTM (si présent)	APTASQ(MOD:00685)LQSPPPPPSSTK
Séquence du peptide sans PTM	APTASQLQSPPPPPSSTK
Position de la modification dans le peptide	6

Tableau 11 – Liste des informations principales disponibles après une identification avec RAId

Une grande partie du temps consacré à ce projet a donc eu pour objectif de rendre exploitable ces données en mettant en place un protocole de retraitement. Il avait deux objectifs :

1. Permettre aux outils d'analyses de données classiques en spectrométrie de masse (comme ceux disponibles dans OpenMS[181]) d'exploiter les résultats de RAId. Le script de

[181] https://www.openms.de/ [Accessible le 26/05/2020]

retraitement a été implémenté pour générer 3 nouveaux fichiers au format standard : mzID, idXML et pepXML. Ces différents formats sont détaillés en Annexe (page 229) ;

2. Permettre de combiner simplement les résultats d'une recherche de multiples modifications post-traductionnelles pour de multiples fichiers RAW. Le script génère une table pour chaque identification contenant toutes les informations nécessaires à une future analyse de données.

Les différents scripts de retraitement des données ont été implémentés en R[182]. Ce langage a été choisi pour plusieurs raisons. Pour commencer, il traite de façon optimal les tableaux de données (les fichiers de sortie RAId sont sous forme de tableaux). Ensuite, il est possible de faire de la programmation orientée objet avec R accélérant considérablement l'accès aux données. Enfin, R est un langage parfaitement adapté à la visualisation et l'analyse de données. D'autres langages potentiellement plus rapides comme C++ (langage de programmation de RAId) auraient pu être utilisés mais R offre la possibilité de réaliser toutes les étapes (et ainsi éviter de compiler plusieurs langages). De plus, nous avons comparé les temps de calculs avec un premier script de retraitement implémenté en C++ au laboratoire par N. Senecaut. Pour une expérience, le script C++ est certes plus rapide mais que de quelques secondes seulement.

À partir des différentes tables générées (7140 dans ce projet et représentées en orange sur la Figure 67), nous avons créé un objet R pour explorer rapidement les données et alléger l'ensemble des résultats. À titre d'exemple, pour ce projet, l'objet R utilisé lors de l'exploration des données présentés dans la partie Résultats ne pèse que 102 Mo contre 70 Go uniquement pour les fichiers au format mzid. Les différents attributs de cet objet sont détaillés à la Figure 67.

[182] Dans le Rstudio server disponible sur le cluster de l'IFB https://rstudio.cluster.france-bioinformatique.fr/ [Accessible le 26/05/2020]

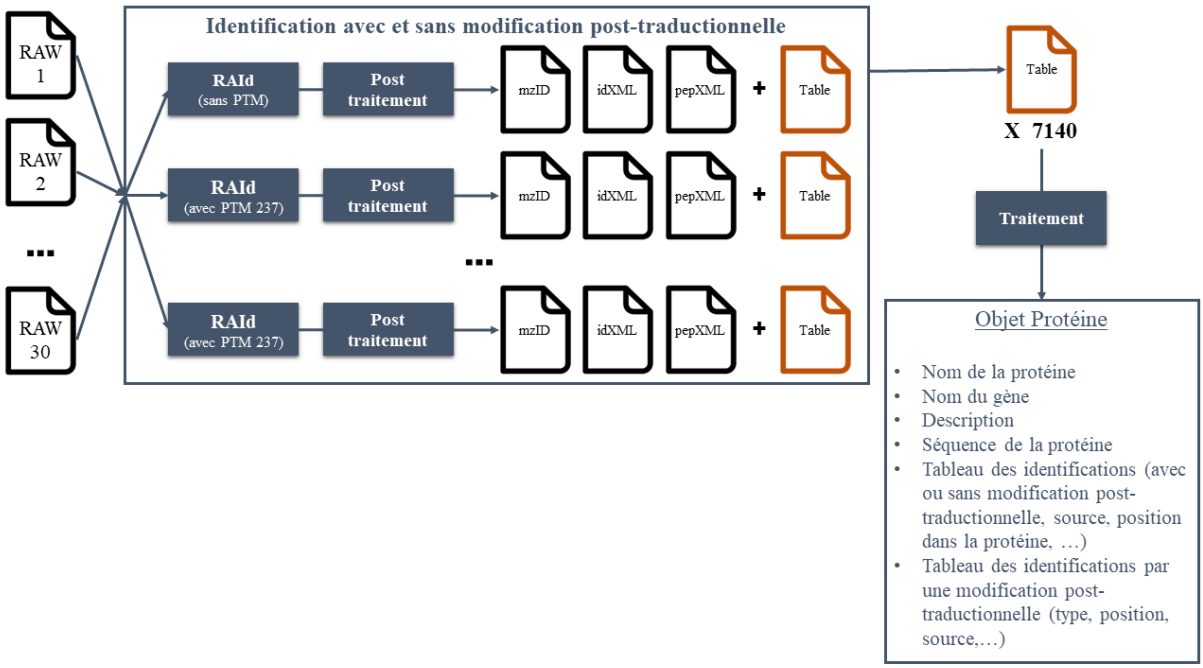


Figure 67 – Vue d'ensemble des différents fichiers générés lors de notre analyse.

c. Automatisation des calculs en utilisant le *cluster* informatique de l'IFB

Comme nous venons de le voir, nous avons un très grand volume de données expérimentales et de très nombreuses modifications post-traductionnelles à tester. Sur mon ordinateur de thèse (en utilisant l'interface graphique), il fallait 30 minutes pour réaliser une identification pour un fichier RAW pour une modification post-traductionnelle. En faisant une estimation très large, il faudrait donc 3570h (soit 149 jours) pour réaliser les calculs en mode graphique sur mon ordinateur. Une seconde estimation concernait la taille totale de l'analyse évaluée à plus de 250 Go. Ne disposant pas d'autant de temps pour faire tout le projet et d'espace sur mon ordinateur, je me suis tourné vers les solutions de *clusters*[183] proposées par l'Institut Français de Bioinformatique (IFB). En utilisant le mode ligne de commande de RAId et sa capacité à paralléliser les identifications, nous avons pu lancer 20 tâches[184] en parallèle avec 30 GB de

[183] Pour en savoir plus sur le *cluster* : https://www.france-bioinformatique.fr/le-cluster-ifb/ [Accessible le 04/08/2020]).

[184] 1 tâche = 1 identification pour 1 fichier pour une modification post-traductionnelle

RAM en multicœur hyperthreadés (parallélisation de chaque tâche). Ainsi, l'ensemble de l'analyse suivie du retraitement des fichiers de sortie RAId ne nécessite plus qu'une nuit.

3. Les résultats

Une fois l'ensemble des données générées (recherche de 237 modifications post-traductionnelles sur 30 fichiers expérimentaux de *C. albicans*), nous avons réalisé une analyse exploratoire préliminaire qui sera présentée dans cette partie. Ces données nous permettent de nous interroger sur deux niveaux. Le premier niveau repose sur la partie computationnelle de l'identification des protéines. Existe-t-il une plus-value à rechercher systématiquement des modifications post-traductionnelles chez *C. albicans* ? Existe-t-il une liste de modifications post-traductionnelles à rechercher en priorité chez cette levure ? Le second niveau est au niveau biologique. Existe-il une dynamique des modifications post-traductionnelles entre la forme hyphe et la forme levure chez *C. albicans* ? Pouvons-nous mettre en évidence des protéines spécifiques d'une forme ? Pouvons-nous mettre en évidence des modifications post-traductionnelles spécifiques d'une forme ?

a. Identification plus performante

Lors de l'utilisation de RAId, nous fournissons un fichier FASTA contenant l'ensemble des séquences protéiques de l'organisme à étudier. Ce fichier sera utilisé pour générer la base de données de spectres théoriques. Celui que nous avons utilisé pour étudier *C. albicans* est composé de 7146 protéines différentes. Cette approche a permis d'identifier 6687 protéines avec un seuil de qualité très bas ($< 5x10^{-4}$) dans au moins une de nos expériences (Figure 68). Nous avons donc une très bonne couverture du protéome lorsque nous utilisons RAId en prenant en compte des modifications post-traductionnelles. Parmi les protéines qui n'ont pas été identifiées, 53% sont non caractérisés sur Uniprot.

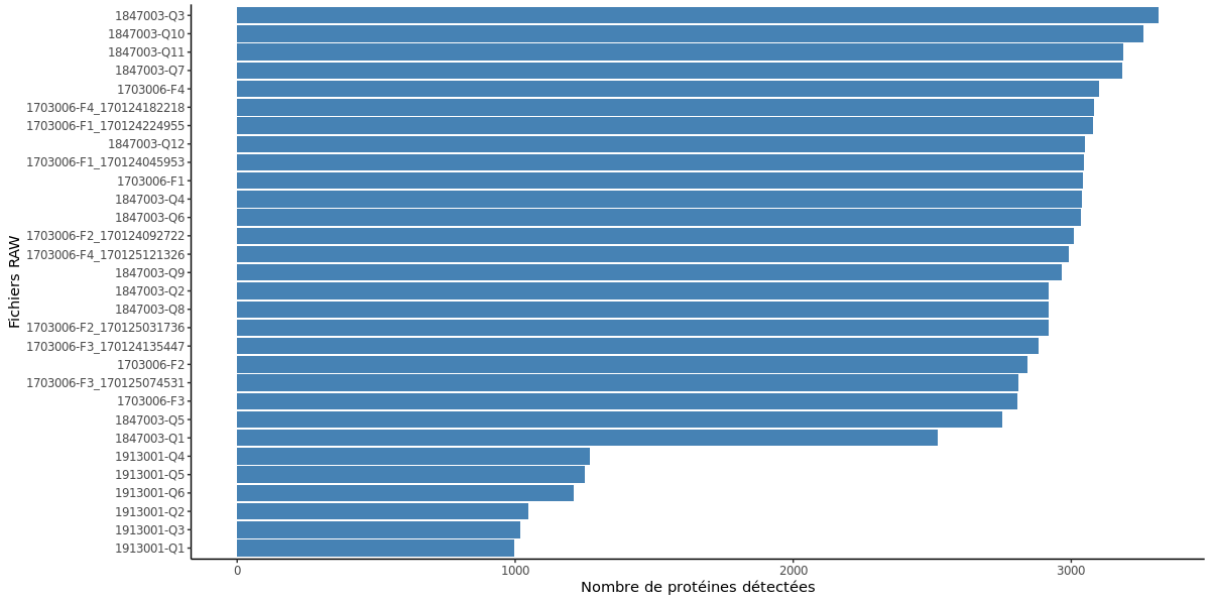


Figure 68 – Nombre de protéines identifiées par fichier RAW. Comparé aux autres expériences, les expériences 1913001 conduisent à moins d'identification que les autres (les 6 lignes du bas). Il s'agit pourtant des réplicats biologiques de l'expérience 1847003. La seule différence entre les deux est un passage dans un congélateur à -80°C pendant un an. Après discussion avec l'équipe, il est connu que la congélation peut entraîner des pertes de protéines.

En moyenne, une protéine est identifiée à partir de 12 peptides différents pouvant être associés plusieurs fois à une même protéine (Figure 69) et 786 protéines ont été identifiées avec un seul peptide (une confiance moindre par rapport aux autres identifications). La protéine identifiée à partir du plus grand nombre de peptides est Q5A4W7, une protéine essentielle impliquée dans la synthèse des acides gras.

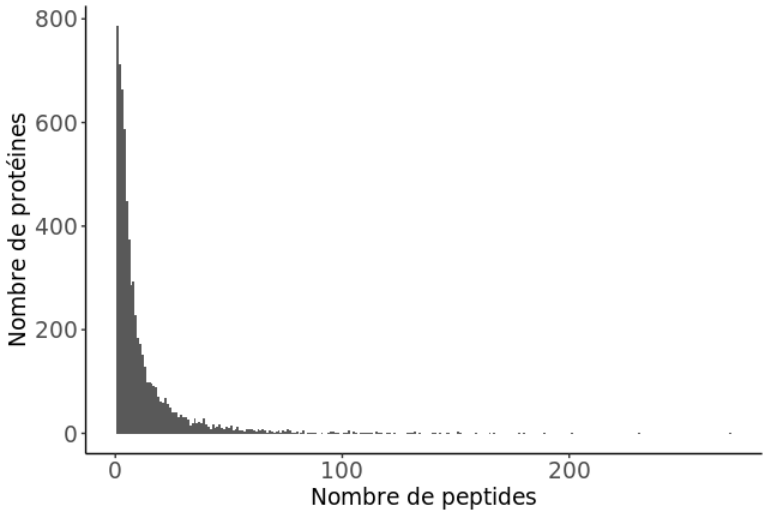


Figure 69 – Distribution du nombre de peptides différents conduisant à l'identification d'une protéine.

b. Modifications post-traductionnelles chez *Candida albicans*

Notre première interrogation était de savoir si les protéines étaient identifiées dans la forme levure et/ou la forme hyphe. La grande majorité des protéines sont identifiées dans les deux formes (5504 protéines) et certaines n'ont été identifiées que dans la forme levure (571 protéines) et d'autres que dans la forme hyphe (612 protéines) (Figure 70 – droite). Nous avons étudié cette répartition dans 3 cas : (1) en n'utilisant que des peptides ne contenant aucune modification post-traductionnelle (Figure 70 – gauche), en n'utilisant que des peptides contenant des modifications post-traductionnelles (Figure 70 – centre) ou tous les peptides avec ou sans modifications post-traductionnelles (Figure 70 – droite). Cette proportion de répartition reste stable dans les 3 cas. Environ 80% des protéines sont retrouvés sous les deux formes et 10% des protéines sont spécifiques à la forme levure et 10% à la forme hyphe. En résumé, nous observons une conservation de la répartition des protéines entre les deux formes avec des spécificités sans doute liées à l'état morphologique.

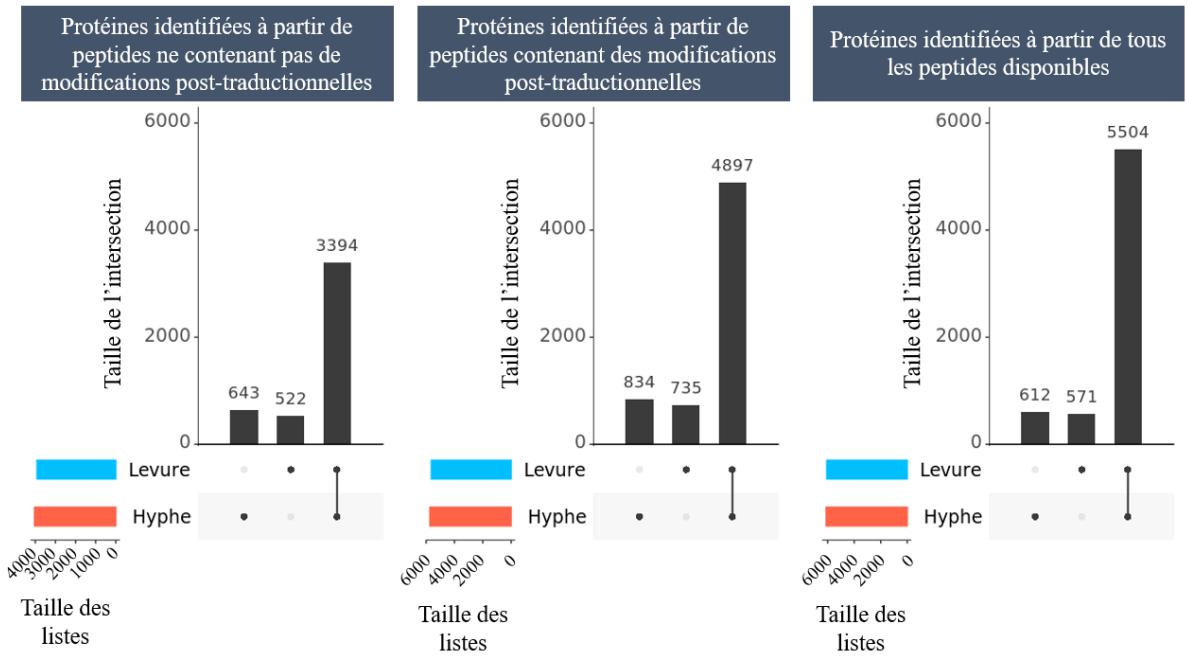


Figure 70 – Répartition des protéines identifiées entre les formes hyphe et levure dans différents cas : à gauche, en étudiant uniquement les protéines identifiées à partir de peptides ne contenant pas de modification post-traductionnelle, au centre en n'étudiant que les protéines identifiées à partir de peptides contenant des modifications post-traductionnelles et à droite en étudiant toutes les protéines sans distinction des peptides.

Ensuite, nous souhaitions savoir si les protéines pouvaient être identifiées uniquement sans modification post-traductionnelle ou uniquement avec des modifications post-traductionnelles en prenant en compte la forme de *C. albicans*. Sur la Figure 71, nous pouvons constater que très peu de protéines sont identifiées à partir de peptides ne contenant aucune modification post-traductionnelle (environ 5% des protéines identifiées). À l'inverse, de nombreuses protéines sont identifiées avec des peptides contenant des modifications post-traductionnelles (environ 30 %). Ce résultat conforte l'intérêt de la prise en compte des modifications post-traductionnelles lors de l'identification chez *C. albicans*. De plus, les changements morphologiques ne semblent pas influencer l'identification en prenant en compte les modifications post-traductionnelles (les proportions sont stables entre les deux formes).

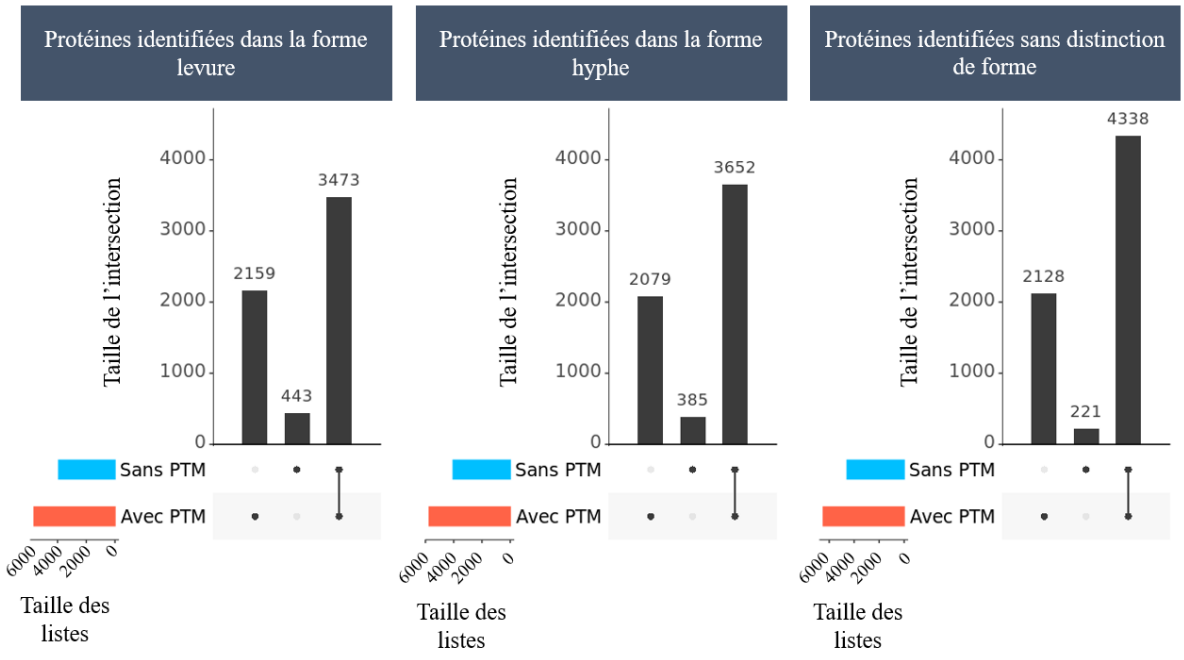


Figure 71 – Répartition des protéines identifiées avec ou sans modifications post-traductionnelles dans différents cas : à gauche, en étudiant les protéines identifiées dans la forme levure, au centre en étudiant que les protéines identifiées dans la forme hyphe et à droite en étudiant toutes les protéines sans distinction de forme.

En moyenne pondérée, environ 7 modifications post-traductionnelles sont retrouvées par protéine (Figure 72). Ce résultat conforte un peu plus l'intérêt de la recherche systématique des modifications post-traductionnelles. La protéine identifiée avec le plus de modifications post-traductionnelles est Q5APD2 issue du gène *MLS1*. Cette protéine est une malate synthase impliquée dans le cycle du glyoxylate essentiel pour la vie de la levure.

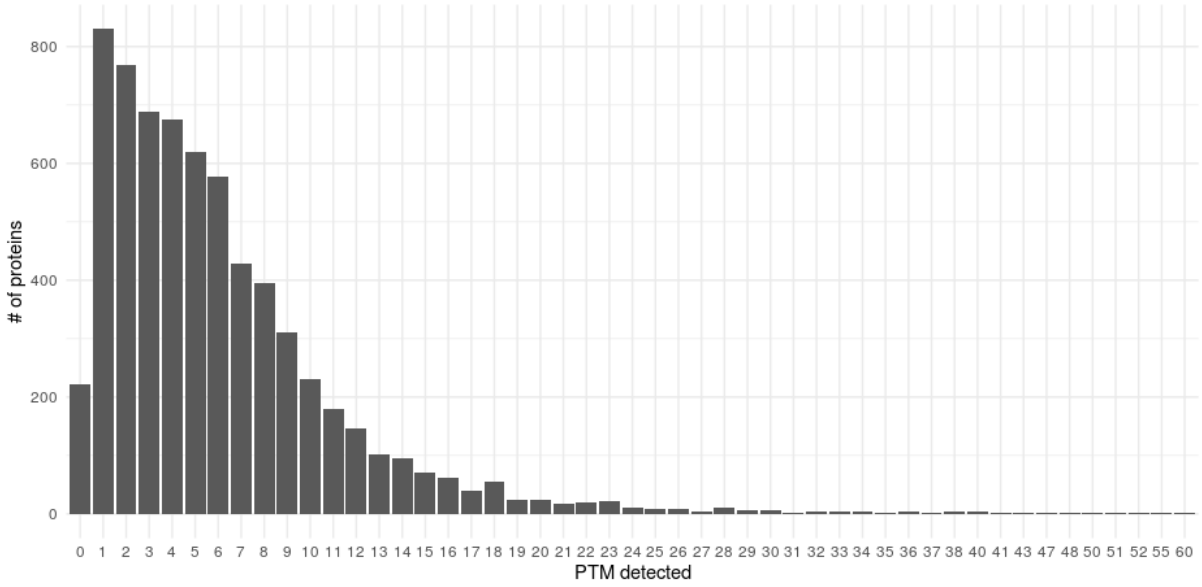


Figure 72 – Nombre de protéines en fonction du nombre de modifications post-traductionnelles détectées.

Pour aller plus loin, nous souhaitions savoir si des modifications post-traductionnelles étaient plus présentes que d'autres. Conscients du besoin de rapidité lors des analyses de routine, l'idée est de proposer une liste de modifications post-traductionnelles plus courte à rechercher systématiquement. Une première observation est que les modifications post-traductionnelles actuellement recherchées en routine ne sont pas les modifications post-traductionnelles les plus présentes dans les protéines identifiées (flèches rouges sur la Figure 73). Parmi les modifications post-traductionnelles les plus présentes, nous retrouvons la Glutathionylation (C28) qui est très étudiée dans le laboratoire de J.M. Camadro (Gergondey et al. 2016) et représentée par la flèche verte sur la Figure 73. Enfin, nous avons pointé par des flèches bleues des modifications post-traductionnelles permettant d'identifier des protéines uniquement grâce à elles. Si nous devions proposer des modifications post-traductionnelles à explorer en association à celles explorées actuellement, il faudrait rechercher : Aspartyl isopeptide, Deamidated asparagine, Deamidated glutamine, Citrulline, Glutathionylation et l'acide 3-phenyllactique (les 6 premières barres en partant de la droite sur la Figure 73)

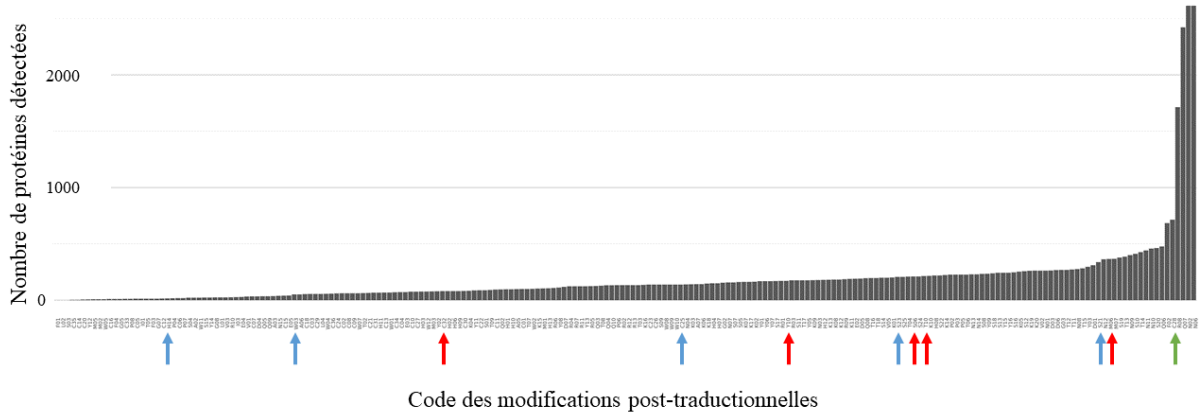


Figure 73 – Nombre de protéines identifiées pour une modification post-traductionnelle donnée. Les flèches rouges pointent les modifications post-traductionnelles qui sont actuellement recherchées en routine (Oxidation (Met), phosphorylation (Ser, Thr, Tyr), acetylation (N-term of protein) et carbamidomethylation (Cys)). Les flèches bleues pointent des modifications post-traductionnelles qui ont permis de mettre en évidence de nouvelles protéines uniquement grâce à elles. La flèche verte pointe la modification post-traductionnelle Glutathionylation très étudiée dans le laboratoire de Jean-Michel Camadro.

Cette liste reste inchangée si nous ne considérons que la forme levure d'une part et la forme hyphe d'autre part. Certaines modifications post-traductionnelles semblent cependant plus fréquentes dans la forme hyphe comme N-acetylproline (P08) et N-methylmethionine (M02) et d'autres plus fréquentes dans la forme levure comme N-methylphenylalanine (F03), Cysteine N-terminal Carbamidomethylation (C33) et S-archaeol cysteine (C14). Une étude plus approfondie de l'existence de modifications post-traductionnelles spécifiques à une forme ou une autre serait à réaliser.

En conclusion, la recherche systématique des modifications post-traductionnelles chez *C. albicans* permet d'augmenter considérablement le nombre d'identifications. Si les délais d'analyse sont trop courts pour rechercher toutes les modifications post-traductionnelles, nous proposons une liste de modifications post-traductionnelles qui sont intéressantes à rechercher chez *C. albicans*. Cette liste est une piste à explorer et à compléter en étudiant d'autres organismes. Ce travail a été présenté à JOBIM 2020 et initie de futurs travaux dans l'équipe de JM. Camadro.

c. Une levure, deux formes

Dans cette partie, nous nous sommes concentrés sur la morphologie de *C. albicans*. Nous avons commencé par étudier le nombre d'expériences où une même protéine a été identifiée. En

moyenne pondérée, une protéine est retrouvée dans un tiers des expériences (Figure 74). Concernant les cas les plus extrêmes, 600 protéines sont identifiées uniquement à partir d'une seule expérience et quasiment 400 protéines sont identifiées dans toutes les expériences. Nous avons donc un faible niveau de confiance pour le premier groupe car nous nous attendions à retrouver ces protéines au moins dans les réplicats (3 expériences). Une analyse fonctionnelle a été réalisée sur les protéines du second groupe (haut niveau de confiance car identifiées dans toutes les expériences) à l'aide de l'outil Gene Ontology Term Finder[185] disponible sur la *Candida Genome Database* (Skrzypek et al. 2017). Parmi ces protéines, 70% ont une activité catalytique et environ 20% ont une activité d'oxydoréduction dont la moitié en tant qu'accepteur dans les groupes de donneurs CH-OH, NADP ou NAD. Nous retrouvons aussi 40% de ces protéines impliquées dans les processus de biosynthèse d'un composé organonitrogéné. Plus largement, 90% de ces protéines sont étiquetées comme étant impliquées dans les processus cellulaires. Nous pouvons supposer que ces 400 protéines sont impliquées dans des processus indispensables à la survie cellulaire.

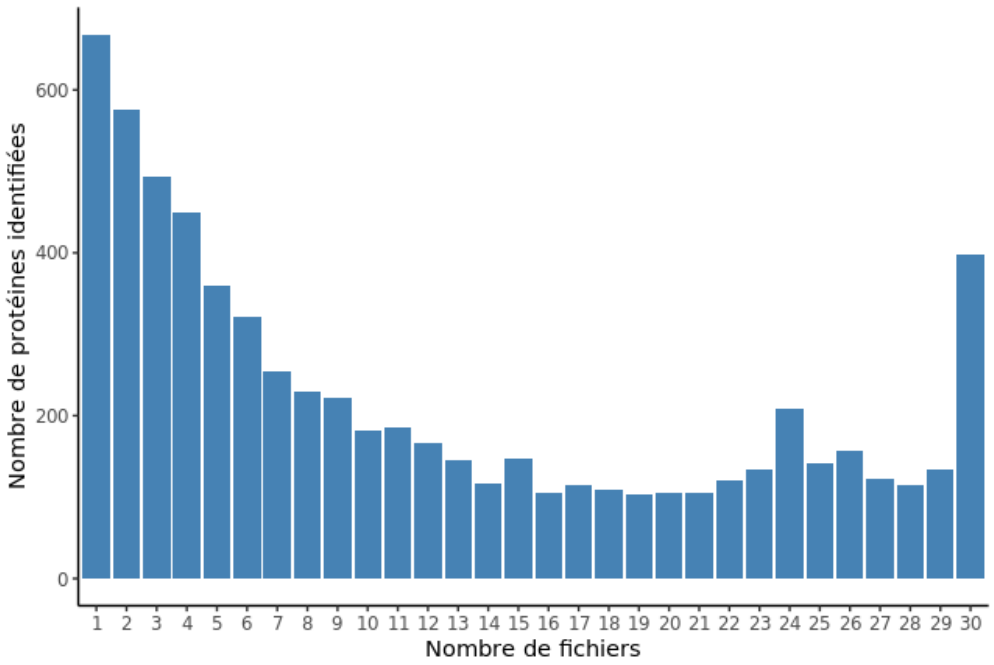


Figure 74 – Nombre de protéines identifiées en fonction du nombre de fichiers utilisés pour les identifier.

[185] http://www.candidagenome.org/cgi-bin/GO/goTermFinder [Accessible le 25/05/2020]

L'étape suivante a consisté à mettre en évidence des protéines spécifiques à la forme levure et à la forme hyphe. Nous avons pour cela calculé un log FoldChange dont la distribution est représentée Figure 75 :

$$logFC = log_2(\frac{Nombre\ d'identification\ dans\ la\ forme\ hyphe + 1}{Nombre\ d'identification\ dans\ la\ forme\ levure + 1})$$

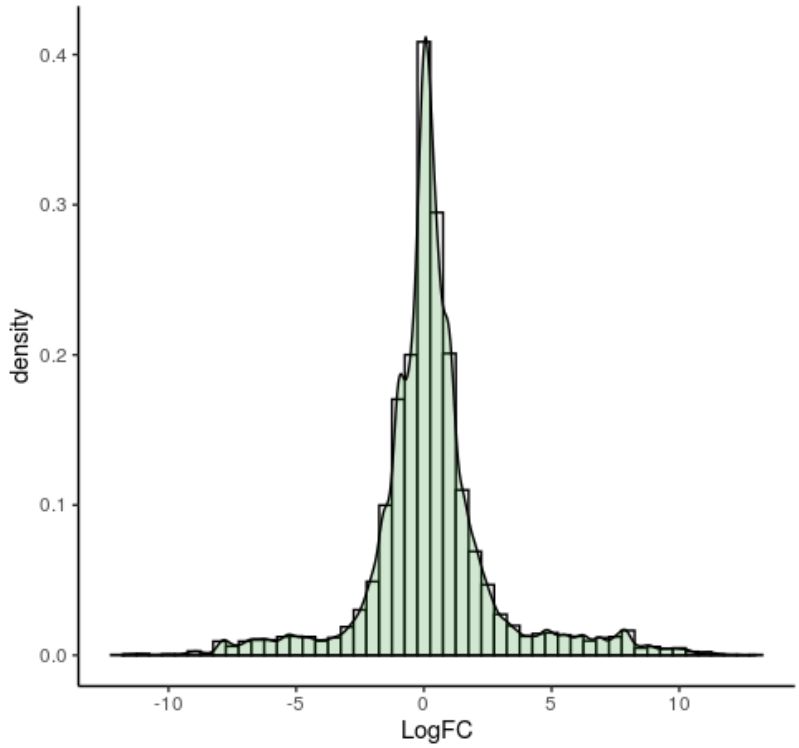


Figure 75 – Distribution des logFCs permettant de mettre en évidence des protéines principalement identifiées dans la forme hyphe (à droite) ou dans la forme levure (à gauche).

Les protéines présentant les logFCs les plus extrêmes sont de bons candidats pour être spécifiques d'une forme : 931 protéines ont été identifiées majoritairement dans la forme hyphe (logFC > 2) et 642 protéines ont été identifiées majoritairement dans la forme levure (logFC < -2). Le Tableau 12 regroupe les protéines les plus représentées dans la forme hyphe et le Tableau 13 dans la forme levure. Parmi les 10 protéines les plus spécifiques de la forme hyphe, elles sont majoritairement impliquées dans l'induction de biofilm. Le lien entre le biofilm et la forme hyphe a été confirmé plusieurs fois dans la littérature (Lee et al. 2018; McCall et al. 2019).

Nom de la protéine	Nom du gène	Description (CGD)	Nombre d'identifications dans la forme hyphe	Nombre d'identifications dans la forme levure
A0A1D8PME3	CFL2	Oxidoreductase; iron utilization; Sfu1/Sef1/Hap43/Nrg1/Tup1/Rim101 regulated; alkaline/low iron/fluphenazine/ciclopirox olamine, flucytosine, fluconazole, **Spider/flow model/rat catheter biofilm induced**; caspofungin/amphotericin B repressed	8151	0
A0A1D8PN12	GIT2	Putative glycerophosphoinositol permease; fungal-specific; repressed by alpha pheromone in SpiderM medium; Hap43-repressed; **Spider biofilm induced**	4937	0
Q59TP1	RBT1	Cell wall protein with similarity to Hwp1; **required for virulence**; predicted glycosylation; fluconazole, Tup1 repressed; farnesol, alpha factor, serum, hyphal and alkaline induced; Rfg1, Rim101-regulated	3661	0
Q5AIA1	EXG2	GPI-anchored cell wall protein, similar to S. cerevisiae exo-1,3-beta-glucosidase Exg2p; predicted Kex2p substrate; induced during cell wall regeneration; possibly an essential gene, disruptants not obtained by UAU1 method; Hap43p-repressed	3031	0
Q59VX6	CWH8	Putative dolichyl pyrophosphate (Dol-P-P) phosphatase; ketoconazole-induced; expression is increased in a fluconazole-resistant isolate; clade-associated gene expression; Hap43p-induced gene	2817	0

Q5A362	CYS3	Cystathionine gamma-lyase; induced by alkaline, amphotericin B, cadmium stress, oxidative stress via Cap1; possibly adherence-induced; Hog1 regulated; **reduced levels in stationary phase yeast cells; Spider and flow model biofilm induced**	11893	4
Q5AD07	SOD5	Cu-containing superoxide dismutase; protects against oxidative stress; induced by neutrophils, hyphal growth, caspofungin, osmotic/oxidative stress; oralpharyngeal candidiasis induced; **rat catheter and Spider biofilm induced**	2359	0
A0A1D8PNL1	MIG1	C2H2 transcription factor; repressor; regulates genes for carbon source utilization; Tup1-dependent and independent functions; hyphal, Hap43 and caspofungin repressed; **Spider and flow model biofilm induced**	2353	0
A0A1D8PI22	orf19.2175	Mitochondrial apoptosis-inducing factor; induced by nitric oxide; **Spider biofilm induced**; rat catheter biofilm repressed	4674	1
Q5A446	CFL1	Protein similar to ferric reductase Fre10p; possible functional homolog of S. cerevisiae Fre1p (reports differ); transcription is negatively regulated by Sfu1p, copper, amphotericin B, caspofungin; induced by ciclopirox olamine	6334	2

Tableau 12 – Protéines les plus spécifiques de la forme hyphe (logFC les plus élevés). Elles sont beaucoup plus identifiées dans la forme hyphe que dans la forme levure.

Parmi les 10 protéines les plus spécifiques de la forme levure, elles sont majoritairement impliquées dans le *switch white-opaque* spécifique à la forme levure. Il s'agit d'un état métastable de la forme levure (Sasse et al. 2013).

Nom de la protéine	Nom du gène	Description (CGD)	Nombre d'identifications dans la forme hyphe	Nombre d'identifications dans la forme levure
A0A1D8PKD9	FAA2	Putative acyl CoA synthetase; expression regulated upon **white-opaque switch**; rat catheter biofilm induced; Spider biofilm induced	2	10905
Q5A3Z5	CDG1	Putative cysteine dioxygenases; role in conversion of cysteine to sulfite; transcript regulated upon **white-opaque switch**; rat catheter, Spider and flow model biofilm induced	0	2981
A0A1D8PFU1	orf19.7209	Putative Rho GDP dissociation inhibitor; induced by nitric oxide independent of Yhb1p	1	5259
A0A1D8PE61	orf19.5069	Ortholog of S. cerevisiae Sae3; meiosis specific protein involved in DMC1-dependent meiotic recombination in S. cerevisiae; Spider biofilm induced	0	2364
A0A454J8I6	L150_04834	Pseudogène	0	2067
Q5AIB2	SCW4	Putative cell wall protein; substrate for Kex2p processing in vitro; expression regulated by	1	3623

		white-opaque switch; alkaline repressed; possibly essential (UAU1 method); flow model biofilm induced; Spider biofilm induced		
C4YLW0	CAWG_01830	Methyltransferase involved in sphingolipid homeostasis, methylates a drug cantharidin; **decreased expression in hyphae compared to yeast**; expression regulated during planktonic growth; flow model biofilm induced; Hap43-repressed gene	0	1226
Q5A1L6	GIT3	Glycerophosphocholine permease; **white cell specific transcript**; fungal-specific; alkaline repressed; caspofungin, macrophage/pseudohyphal-repressed; flow model biofilm induced; Spider biofilm induced	0	1025
A0A1D8PCH9	POX18	Similar to Pox18, a peroxisomal protein; induced during chlamydospore formation in C. albicans and C. dubliniensis; A21 sequence updated based on new sequence and analysis, the allelic orf19.10841 was reinstated; Spider biofilm induced	0	905
Q5AAV4	orf19.6205	Uncharacterized protein	1	1322

Tableau 13 – Protéines les plus spécifiques de la forme levure (logFC les plus élevés). Elles sont beaucoup plus identifiées dans la forme levure que dans la forme hyphe.

Pour conclure cette analyse exploratoire et valider l'approche, nous avons collecté des gènes décrits dans la littérature comme spécifiques d'une forme ou l'autre et nous les avons comparés à nos données. Nous avons ensuite calculé le ratio suivant (comme dans les articles étudiés) :

$$R = \frac{\text{Nombre de peptides permettant l'identification d'une protéine dans la forme } \mathbf{levure}}{\text{Nombre de peptides permettant l'identification d'une protéine dans la forme } \mathbf{hyphe}}$$

Les données ne provenant pas de spectrométrie de masse quantitative, notre hypothèse est que si R est largement supérieur à 1, la protéine est certainement plus présente dans la forme levure et si R est largement inférieur à 1, la protéine est plus présente dans la forme hyphe. Nous pouvons voir que de nombreuses protéines sont en cohérence avec la littérature. Ces résultats pourront être confirmés par la réalisation d'une analyse par spectrométrie de masse quantitative.

Protéine	Gène	Ratio R dans nos données (Levure/Hyphe)	Forme dans la littérature
A0A1D8PRR7	ACC1	0.776	Hyphe (Ebanks et al. 2006)
A0A1D8PFR4	ACT1	0.94	Hyphe (Ebanks et al. 2006)
Q59L12	ALS3	0.07	Hyphe (Ebanks et al. 2006)
Q5A768	BUL1	0.011	Hyphe (Singh et al. 2005)
Q5ALV5	COX4	0.933	Hyphe (Singh et al. 2005)
Q5APK5	COX5	0.844	Hyphe (Singh et al. 2005)
P43079	CPH1	0.6	Hyphe (Singh et al. 2005)
Q59NP1	CTR1	0.511	Hyphe (Singh et al. 2005)
A0A1D8PM35	EFB1	0.816	Hyphe (Singh et al. 2005)
Q59X67	EFG1	0.475	Hyphe (Ebanks et al. 2006)
Q5A0M4	EFT2	0.903	Hyphe (Ebanks et al. 2006)
P30575	ENO1	0.919	Hyphe (Ebanks et al. 2006)
Q5A4W7	FAS1	0.729	Hyphe (Singh et al. 2005)
A0A1D8PK65	FAS2	0.765	Hyphe (Ebanks et al. 2006)
Q5AG77	GAP1	0.004	Hyphe (Singh et al. 2005)
Q59P43	GSP1	0.923	Hyphe (Singh et al. 2005)
A0A1D8PF56	GAR1	0.911	Hyphe (Singh et al. 2005)
Q59NC4	GLY1	0.133	Hyphe (Singh et al. 2005)
Q59RQ6	LPD1	0.898	Hyphe (Ebanks et al. 2006)
Q59S06	NOP58	0.787	Hyphe (Singh et al. 2005)
Q5A747	OLE1	0.001	Hyphe (Singh et al. 2005)
Q5A786	PFY1	0.878	Hyphe (Singh et al. 2005)

Q5ALK3	PRS1	0.691	Hyphe (Singh et al. 2005)
A0A1D8PGA2	REG1	0.695	Hyphe (Singh et al. 2005)
A0A1D8PLC9	RPL20B	0.721	Hyphe (Ebanks et al. 2006)
Q5AG43	RPS5	0.848	Hyphe (Singh et al. 2005)
A0A1D8PK22	RPS15	0.903	Hyphe (Singh et al. 2005)
Q59KY8	SIT4	0.672	Hyphe (Kavanagh 2007)
P46587	SSA2	0.94	Hyphe (Ebanks et al. 2006)
A0A1D8PSB9	SRP40	0.858	Hyphe (Singh et al. 2005)
A0A1D8PRR7	ACC1	0.776	Hyphe (Ebanks et al. 2006)
Q59RR0	ACE2	Forme levure uniquement	Levure (Singh et al. 2005)
A0A1D8PCM2	GAL7	1.419	Levure (Singh et al. 2005)
A0A1D8PHQ6	GRE2	1.238	Levure (Singh et al. 2005)
Q5A3M2	MSO1	1.187	Levure (Singh et al. 2005)
O74711	PEX5	2.217	Levure (Singh et al. 2005)
A0A1D8PJ01	PMA1	1.112	Levure (Ebanks et al. 2006)
Q59SE2	STB3	1.16	Levure (Singh et al. 2005)
Q5ANI6	STP3	2.5	Levure (Singh et al. 2005)
Q59QC7	UPC2	195	Levure (Singh et al. 2005)

Tableau 14 – Listes de quelques protéines collectées dans la littérature connues comme spécifiques d'une forme (hyphe ou levure). Cette spécificité est comparée au ratio R (Levure/Hyphe).

En conclusion, comme le montrent de multiples publications sur le sujet référencées dans le Tableau 14, nous avons pu mettre en évidence des protéines spécifiques d'une forme de *C. albicans*. Nous avons ainsi montré une cohérence entre les protéines identifiées à partir de nos données et la littérature et validé notre protocole d'identification. De plus, ces résultats confirment la pertinence d'une analyse de données plus approfondie.

4. Les conclusions et perspectives

a. Nouvelle stratégie d'identification des protéines

Aujourd'hui, la prise en compte des modifications post-traductionnelles lors de l'identification de protéine en spectrométrie de masse n'est pas systématique et seulement 3 sont testées en routine. Les temps de calculs pour les prendre en compte sur les outils d'identification classiques restent encore trop élevés pour une analyse de routine. Nous avons donc mis en place un protocole de création de données à partir des fichiers bruts (.raw) issus de spectrométrie de

masse en Tandem LC-MS/MS par approche *Bottom Up*. La nouveauté de ce protocole consiste à prendre en compte de 237 modifications post-traductionnelles et proposer des fichiers d'identification au format standard (mzID, idXML et pepXML) ainsi qu'un fichier optimisé pour une analyse de données réalisée avec R. Grâce à l'utilisation du *cluster* de calcul de l'Institut Français de Binformatique et l'optimisation des scripts, une identification à partir d'un fichier RAW pour une modification post-traductionnelle prend moins de 5 minutes. Plus largement, il ne faut qu'une nuit pour traiter 30 fichiers RAW en recherchant systématiquement les 237 modifications post-traductionnelles disponibles actuellement dans RAId.

Nous avions à notre disposition des données chez *C. albicans* dans ces deux formes morphologiques : la forme levure et la forme hyphe. Quinze expériences ont été réalisées pour chacune de ces deux formes. À partir de toutes les données d'identification générées, nous avons réalisé une analyse exploratoire visant à répondre à plusieurs questions :

- Faut-il rechercher systématiquement les modifications post-traductionnelles lors d'une identification de protéines par spectromètre de masse en Tandem LC-MS/MS avec une approche *Bottom Up ?* Existe-t-il des modifications post-traductionnelles à rechercher en priorité ?
- Existe-t-il une dynamique des modifications post-traductionnelles chez *C. albicans* entre la forme levure et la forme hyphe ? Ce protocole peut-il mettre en évidence des protéines spécifiques de ces deux formes morphologiques de *C. albicans* ?

Pour commencer, nous avons montré que la prise en compte des modifications post-traductionnelles permettait d'identifier plus de 2000 nouvelles protéines chez *C. albicans* (quasiment un tiers du protéome). De plus, moins de 20% des spectres sont à présent sans identification. L'apport de cette recherche systématique est important. Ensuite, nous avons montré que certaines modifications post-traductionnelles sont spécifiques à une des deux formes mais qu'une étude plus approfondie devra être menée pour confirmer ce résultat. Certaines modifications post-traductionnelles semblent particulièrement présentes chez *C. albicans* dans les deux formes étudiées. Elles semblent intéressantes à explorer systématiquement puisqu'elles sont présentes dans une très grande majorité de protéines identifiées chez *C. albicans*. Une fois confirmée, cette liste pourrait être utile par la suite pour limiter les temps de calcul. Enfin, pour valider le protocole, nous avons listé à partir de la littérature des protéines connues comme spécifiques d'une des deux formes de *C. albicans* et

nous les avons confrontées à nos données. Pour un grand nombre des protéines testées (> 80%), la spécificité morphologique décrite dans la littérature a été retrouvée dans les données. Nous avons donc une bonne confiance dans le protocole mis en place.

b. Pour aller plus loin

Les résultats présentés dans ce chapitre sont très préliminaires. Le temps consacré à la génération de données a limité celui consacré à l'exploration. Il reste encore de nombreuses pistes à approfondir ou explorer comme :

Étudier l'influence de l'utilisation de SAHA

Dans les expériences 1847003 (Tableau 10), certains milieux de culture ont été complémentés avec du SAHA, un inhibiteur de l'histone désacétylase. Cette histone est une enzyme qui catalyse la perte du groupement acétyl sur l'extrémité N-terminale d'une histone. Ces histones modifiées ont un rôle essentiel dans la régulation de l'expression génétique. L'utilisation de SAHA a pour objectif de limiter la perte de groupement acétyl et ainsi obtenir plus de peptides à identifier. Nous nous attendions donc à une augmentation du taux d'identification de protéines pour des modifications post-traductionnelles d'acétylation. Lors d'une première exploration, nous avons observé que certaines modifications post-traductionnelles conduisent à plus d'identification en présence de SAHA. Ce constat est particulièrement vrai sous la forme levure mais plus discret dans la forme hyphe. Il a été montré que la forme hyphe est plus résistante aux drogues (Desai et al. 2014) et notamment lors de l'utilisation de SAHA comme traitement (Fedier et al. 2007). Il est donc cohérent d'avoir un effet limité de SAHA dans la forme hyphe. Comme le nombre de peptides identifiés dans la forme levure en présence de SAHA est proche de celui de la forme hyphe, nous pouvons supposer que la forme hyphe présente plus d'acétylations que la forme levure. Une seconde hypothèse est que le SAHA ne serait pas vraiment actif dans la forme hyphe parce que le protéome acétylé serait possiblement au maximum dans cette forme. Craignant d'avoir utilisé une dose trop faible de SAHA, les expérimentateurs ont réalisé une expérience de confirmation en utilisant un anticorps anti-acétyle lysine. La lysine est la modification post-traductionnelle la plus recherchée sous sa forme acétylée. Cette expérience a confirmé que la dose de SAHA n'était pas assez forte. Notre protocole d'identification à l'aide des modifications post-traductionnelles a permis de confirmer cette tendance puisque le taux d'identification pour la modification post-

traductionnelle acétyle lysine reste stable dans les différentes formes en présence ou non de SAHA. Cette analyse renforce la volonté de prendre en compte systématiquement les modifications post-traductionnelles et l'utilisation de RAId comme outil d'identification.

Étudier l'influence d'une culture dans un environnement riche en Carbone 12

Un axe de recherche très important de l'équipe de J.M. Camadro est d'augmenter l'intensité du pic monoisotopique afin d'améliorer le taux d'identification de protéines par spectrométrie de masse en approche *Top down*. Pour y arriver, le milieu de culture contient du glucose composé à 100% de carbone C12 limitant ainsi la formation d'isotopes moins stables. Même si l'utilisation d'un tel environnement n'est pas obligatoire pour une approche *Bottom up* (par expérience le pic le plus intense est généralement le pic monoisotopique), il pourrait néanmoins aboutir à plus de peptides identifiés. Nous avons exploré pour cela les expériences 1703006-F1 à F4 (avec et sans C12). Pour commencer, la grande majorité des protéines sont identifiées dans les deux conditions (quasiment 5000 protéines). Cultiver les levures dans un milieu riche en C12 conduit-il à une identification de meilleure qualité (plus de peptides permettent l'identification de la protéine) ? Pour répondre à cette question, nous avons comparé la moyenne du nombre de peptides ayant conduit à une identification à partir de levures cultivées dans un milieu traditionnel (les expériences 1703006-F1 et 1703006-F2) avec la moyenne du nombre de peptides ayant conduit à une identification à partir de levures cultivées dans un milieu riche en C12 (les expériences 1703006-F3 et 1703006-F4). Nos hypothèses statistiques sont :

- H0 : Il n'y a pas de différence entre les moyennes (l'enrichissement en C12 n'a pas d'influence sur le nombre de peptides conduisant à une identification)
- H1 : il y a une différence significative entre les moyennes (l'enrichissement en C12 a une influence sur le nombre de peptides conduisant à une identification)

En moyenne, 166 peptides sont associés à une protéine à partir d'un milieu de culture traditionnel contre 273 dans un milieu enrichi en C12. Après avoir testé la normalité de la distribution de nos valeurs (test de Shapiro), le test de comparaison de moyenne de Student met en évidence une différence significative à un risque alpha 0.05 (valeur P (*P-value*) $< 2*10^{-16}$). H0 est rejetée, l'utilisation d'un milieu riche en C12 a donc une influence sur le nombre de

peptides conduisant à une identification. Il serait à présent intéressant de valider ce résultat sur d'autres organismes.

Étudier l'enrichissement fonctionnel des modifications post-traductionnelles

L'objectif de cette étude serait de mettre en évidence des modifications post-traductionnelles présentant un enrichissement fonctionnel. L'idée serait pour chaque modification post-traductionnelle d'isoler les protéines la possédant et de calculer l'enrichissement de ces protéines. Les premières explorations laissent penser que certaines modifications post-traductionnelles présentent un enrichissement fonctionnel et de nouvelles analyses doivent être réalisées prochainement pour valider ces résultats. Pour aller plus loin, il serait aussi intéressant de séparer les formes levures et les formes hyphes et rechercher si les enrichissements fonctionnels mis en évidence sont conservés d'une forme à l'autre ou sont spécifiques d'une forme.

c. Intérêt pour le projet sur l'homéostasie du fer mené chez *Candida glabrata*

Lors de l'étude des gènes réagissant au fer chez *C. glabrata*, nous avons évoqué un jeu de données que nous avons publiées en protéomique (Lelandais et al. 2019b) (présenté en détail page 193). Très brièvement, il s'agit de données quantitatives obtenues par spectrométrie de masse *label free*[186]. Un des objectifs était d'étudier la réponse de *C. glabrata* à un stress pH alcalin. À ce type de pH, le fer précipite et ce qui mime les effets d'un milieu pauvre en fer. À partir des données brutes disponibles sur PRIDE[187], nous avons appliqué le même protocole que celui présenté dans ce chapitre pour *C. albicans*. Comme pour *C. albicans*, le taux d'identification avec notre approche est beaucoup plus élevé avec 1642 nouvelles protéines identifiées (Figure 76). Quasiment toutes les protéines identifiées par la méthode classique le sont aussi par notre approche (1773 protéines). Les 106 protéines identifiées uniquement par la méthode classique sont identifiées par notre approche mais pas sélectionnées. La raison est l'utilisation d'un seuil de sélection par les valeurs P (*P-values*) très stringent ($< 5x10^{-4}$). Ces

[186] Pour en savoir plus, je vous conseille la très bonne ressource francophone : http://mass-spectro.com/proteomique [Accessible le 27/05/2020]

[187] https://www.ebi.ac.uk/pride/archive/projects/PXD014125 [Accessible le 27/05/2020]

106 protéines identifiées mais non sélectionnées ont en effet des valeurs P comprises entre 10^{-2} et 10^{-3}.

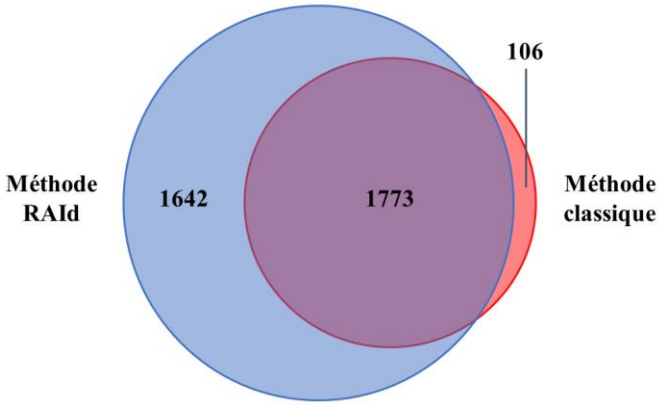


Figure 76 – Comparaison des protéines identifiées par la méthode classique (Proteome discoverer) et notre protocole basé sur RAId.

En se focalisant sur les 637 gènes que nous avons identifiés comme réagissant au fer (voir Chapitre II, page 128), un des premiers constats est que plus de la moitié des gènes de notre liste n'avait pas de valeurs associées par l'approche classique[188]. La Figure 77 compare la liste des 637 gènes réagissant au fer avec les protéines identifiées par une méthode classique (Proteome discoverer) et notre protocole basé sur RAId. Presque la moitié des gènes réagissant au fer est retrouvé dans les deux méthodes d'identification et 176 gènes ne sont toujours pas identifiés. Après exploration de la localisation cellulaire de ces gènes, 20 % aboutissent à des protéines membranaires or il est connu que ces protéines très hydrophobes sont complexes à extraire par des méthodes classiques en LC-MS/MS. Pour ces protéines, le défaut d'identification semble lié à la partie expérimentale. Pour les autres protéines, il s'agit encore du seuil de sélection puisque les identifications ont des valeurs P (*P-values*) comprises entre 10^{-2} et 10^{-3}. Ensuite, 155 nouveaux gènes ont été identifiés grâce à notre approche. Seulement 8 n'ont été trouvés que par l'approche classique. Encore une fois, ces 8 protéines sont identifiées mais ne sont pas sélectionnées à cause leur valeur P comprise entre 10^{-2} et 10^{-3}.

[188] Les données sont disponibles dans iHKG viewer https://thomasdenecker.github.io/iHKG/OtherDatasets.html [Accessible le 27/05/2020]

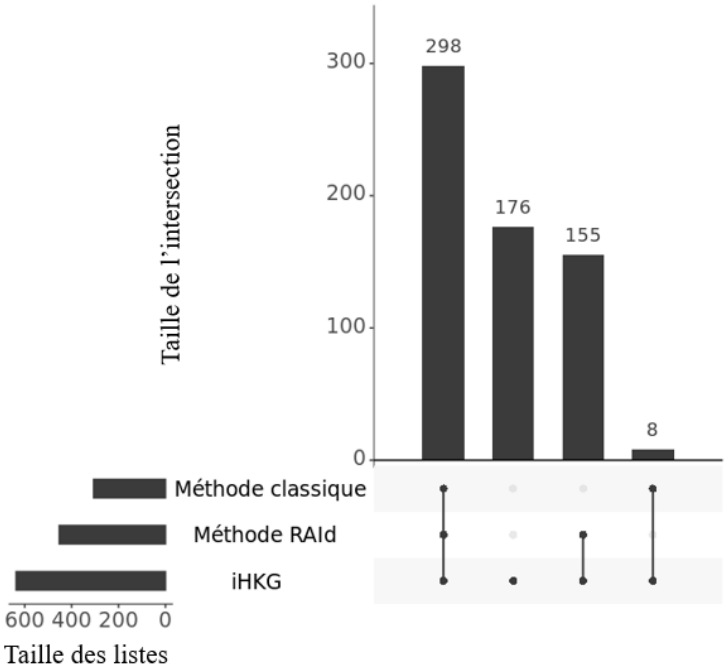


Figure 77 – Comparaison de la liste des 637 gènes réagissant au fer avec les protéines identifiées par une méthode classique (Proteome discoverer) et notre protocole basé sur RAId.

Nous avions volontairement choisi un seuil très stringent dans l'analyse précédente pour avoir une grande confiance dans les résultats. Nous pourrions envisager d'augmenter ce seuil au même niveau que celui utilisé dans Proteome discoverer. En conclusion, chez *C. glabrata* aussi, la prise en compte systématique et le choix de RAId permettent une meilleure identification des protéines que l'approche classique. Cette étude pourrait être approfondie pour mettre en évidence des modifications post-traductionnelles d'intérêt chez les protéines issues des gènes réagissant au fer.

Pour conclure, nous avons pu voir dans ce chapitre que la prise en compte systématique des modifications post-traductionnelles dans l'identification des protéines par spectrométrie de masse apporte une amélioration du taux d'identification chez les levures pathogènes *C. albicans* et *C. glabrata*. De plus en plus étudiées, les modifications post-traductionnelles ont été identifiées comme des marqueurs importants du vieillissement et des maladies liées à l'âge telles que les maladies d'Alzheimer et de Parkinson, le cancer ou le diabète (A. L. Santos et al. 2017). Avoir un protocole robuste pour les détecter deviendra donc indispensable pour les analyses de routine d'autant plus qu'il ouvre de nouveaux horizons de recherche.

IV. Contributions à d'autres projets

1. Étude de la reproductibilité des résultats obtenus par spectrométrie de masse

a. Contexte du projet

Dans une démarche de contrôle qualité des données de spectrométrie de masse, G. Lelandais avait mis en place un projet annexe visant à évaluer la reproductibilité des analyses réalisées par des approches de type *bottom up*. En particulier, elle souhaitait aborder deux aspects associés aux études protéomiques haut débit : l'aspect qualitatif (quelles protéines sont identifiées) et l'aspect quantitatif (mesures des abondances des protéines identifiées). Ainsi, elle a produit en 2014 chez les levures *Candida glabrata* et *Candida albicans*, deux jeux de données incluant à la fois des situations de réplicats techniques et de réplicats biologiques, pour deux points de temps d'analyse différents. J'ai travaillé sur l'analyse de ces données lors de mon stage de Master 1 (2016) et nous les avons rendues publiques et publiées en 2019. La version PDF de l'article est disponible ci-après.

b. PDF de l'article publié dans la revue « BMC Research Notes » (Lelandais et al. 2019)

Lelandais *et al. BMC Res Notes* (2019) 12:470
https://doi.org/10.1186/s13104-019-4505-8

BMC Research Notes

DATA NOTE

Open Access

Label-free quantitative proteomics in *Candida* yeast species: technical and biological replicates to assess data reproducibility

Gaëlle Lelandais[1*], Thomas Denecker[1], Camille Garcia[2], Nicolas Danila[3], Thibaut Léger[2] and Jean-Michel Camadro[2,3]

Abstract

Objective: Label-free quantitative proteomics has emerged as a powerful strategy to obtain high quality quantitative measures of the proteome with only a very small quantity of total protein extract. Because our research projects were requiring the application of bottom-up shotgun mass spectrometry proteomics in the pathogenic yeasts *Candida glabrata* and *Candida albicans*, we performed preliminary experiments to (i) obtain a precise list of all the proteins for which measures of abundance could be obtained and (ii) assess the reproducibility of the results arising respectively from biological and technical replicates.

Data description: Three time-courses were performed in each *Candida* species, and an alkaline pH stress was induced for two of them. Cells were collected 10 and 60 min after stress induction and proteins were extracted. Samples were analysed two times by mass spectrometry. Our final dataset thus comprises label-free quantitative proteomics results for 24 samples (two species, three time-courses, two time points and two runs of mass spectrometry). Statistical procedures were applied to identify proteins with differential abundances between stressed and unstressed situations. Considering that *C. glabrata* and *C. albicans* are human pathogens, which face important pH fluctuations during a human host infection, this dataset has a potential value to other researchers in the field.

Keywords: Mass spectrometry, Label-free quantitative proteomics, *Candida glabrata*, *Candida albicans*, Alkaline pH

Objective

Studying proteome dynamics is a key step in systems biology projects. In this context, label-free bottom-up shotgun MS-based proteomics produces quantitative analyses of proteomes. This technique has emerged from significant improvements achieved by mass spectrometry (MS) instrumentation, chromatographic separation systems and a stronger correlation between the relative measured ion intensity and the original molecule abundance in the electrospray ionization process [1–3]. Members of our research team were involved in functional genomics studies in pathogenic yeasts *Candida glabrata* and *Candida albicans* [4–8]. We observed how the experimental design is a critical step to empower the statistics used to assess the robustness of the results.

"How many replicates is enough?" is certainly one of the most frequently asked questions in wet laboratories. This question is especially critical in situations where the experiments are expensive, and/or the preparation of the biological samples is challenging. Here, our objective was to assess the robustness of the results arising from label-free bottom-up shotgun MS-based proteomics performed in *C. glabrata* and *C. albicans*, in case of technical and biological replicates. If the importance of biological replicates was indisputable when we started

*Correspondence: gaelle.lelandais@u-psud.fr
[1] CEA, CNRS, Institute for Integrative Biology of the Cell (I2BC), Univ. Paris-Sud, Gif-Sur-Yvette, France
Full list of author information is available at the end of the article

this project, the interest for technical replicates was more questionable.

We induced proteome modifications applying an alkaline pH stress to *Candida* cells grown in minimal liquid medium. Our final dataset comprises quantitative proteomics for 24 samples (two species, three time-courses, two time points and two runs of mass spectrometry, see below) [9, 10]. We believe it could be useful for other researchers, either interested in a statistical exploitation of the results (to model for instance the variability of protein quantifications associated with biological or technical replicates respectively) or interested in a better understanding of the cellular mechanisms which underly adaptation of pathogenic yeasts to pH changes, a key process during a human host infection [11].

Data description

In this analysis, we performed in *Candida glabrata* (CGLAB) and *Candida albicans* (CALB) yeast species, three cultures referred as CTRL, ALK1 and ALK2. CGLAB and CALB strains are respectively the ones used in [4] and [7], and they were cultured in the same standard conditions as described in [4, 7]. Here, CTRL means "Control", i.e. the cells were grown in minimal liquid medium. ALK means "alkaline pH stress", i.e. the cells were subjected to an alkaline stress by adding 1 M of Tris base. This dose was appropriate to slightly affect cell growth without killing the cells. ALK1 and ALK2 referred to two biological replicates, i.e. independent cell growth cultures. T10 and T60 means respectively "time point 10 min" and "time point 60 min", i.e. the time after stress induction at which the cells were collected for mass spectrometry experiments. These time points were chosen because the cells were then in the exponential phase. Finally, REP1 and REP2 referred to two technical replicates, i.e. independent MS acquisition from the same protein extract and trypsin digestion.

Overall, two datasets were associated to this paper note (Table 1). Data set 1 comprises 24 raw data files, obtained from a Q-Exactive Plus mass spectrometer coupled to a Nano_LC Proseon 1000 equipped with an easy spray ion source (all from Thermo Fisher Scientific); 48 search files, obtained with the Proteome Discovered software (Thermo Scientific, version 2.1) and the Mascot search engine (Matrix Science, version 2.5.1); 2 quantification files obtained with the Progenesis QI for Proteomics software (version 4.1, Waters) and 2 FASTA files obtained for the CGD website and used for the MS/MS identification step. Note that detailed descriptions of (i) sample processing protocol and (ii) data processing protocol can be found in [9]. Data file 2 explains the relationship between MS files and associated experimental conditions (CTRL, ALK1, ALK2, T10, T60, REP1 and REP2).

Limitations

We produced this dataset to assess our ability to properly quantify protein abundances in yeasts *Candida glabrata* and *Candida albicans*. An open question for us was the impact of technical replicates compared to biological replicates. We thus performed cell cultures under two different conditions (control and induced stress), collected cells at two separate time points (10 and 60 min) after stress induction, extracted the proteins, performed trypsin digestions and analysed the composition of samples by mass spectrometry. As a result, we were first able to observe a good coverage of proteome in yeasts *C. glabrata* and *C. albicans*, respectively. Between 1500 and 2000 proteins were identified in a reproducible way, representing ~ 30% of the total protein repositories in these species. It should be noted that a problem in two sample preparations occurred in *Candida glabrata*. Less than 250 proteins were found in technical replicates 1445007-Q3 and 1445007-Q9, which are CGLAB, ALK2, T10, REP1 and REP2 [10]. This is the main limitation for our data. Second, we observed that technical replicates were critical to increase the number of identified proteins, as ~ 25% of them were found in only one technical replicate. In this context, having a third technical replicate would have been of interest to see if better proteome coverage can still be obtained. Finally, we were able to

Table 1 Overview of the data files related to the study of label-free quantitative proteomics in *Candida* yeasts species, assessing data reproducibility in technical and biological replicates

Label	Name of data file	File types	Data repository and identifier (accession number)
Data set 1 [9]	Proteomics project in *Candida* yeast species	RAW files (.raw), Search files from Proteome Discoverer software (.pdResults and.xlsx), Quantification files from Progenesis QI for Proteomics software (.csv), FASTA files (.fasta)	http://identifiers.org/pride.project:PXD014125
Data file 2 [10]	Detailed correspondence between MS files and experimental conditions	PDF (.pdf)	https://doi.org/10.5281/zenodo.3334949

observe very high positive correlation values (higher than 0.9) between abundances of proteins obtained from biological replicates. If this result is very encouraging, it may also reflect that our cell cultures were not totally "independent". Indeed, they were performed simultaneously, starting from the same over-night pre-culture. We believe it could be interesting to replicate these experiments paying more attention to this last point, in the design of experiments.

Abbreviations

C. glabrata and CGLAB: *Candida glabrata*; *C. albicans* and CALB: *Candida albicans*; MS: mass spectrometry; CTLR: control; ALK1 and ALK2: alkaline stress 1 and 2; T10 and T60: time point 10 min and time point 60 min; REP1 and REP2: replicate 1 and replicate 2.

Acknowledgements

We would like to thank the research team "Mitochondria, Metals and Oxidative Stress" at the Jacques Monod institute for helpful discussions.

Authors' contributions

GL and JMC designed the experiments, GL and ND performed yeast cell cultures, GL and TD performed statistical and bioinformatics analyses, CG performed protein extractions, TL and JMC performed mass spectrometry experiments, TL uploaded the dataset on PRIDE, GL wrote the manuscript. All authors read and approved the final manuscript.

Funding

This work was funded by the Agence Nationale pour la Recherche (CANDIHUB project, Grant Number ANR-14-CE14-0018-02). The funders had no role in study design, data collection and analysis, decision to publish or preparation of the manuscript.

Availability of data and materials

The mass spectrometry proteomics data have been deposited to the ProteomeXchange Consortium via the PRIDE partner repository [10] with the dataset identifier PXD014125 [9]. Please see Table 1 for details and links to the data.

Ethics approval and consent to participate

Not applicable.

Consent to publish

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1] CEA, CNRS, Institute for Integrative Biology of the Cell (I2BC), Univ. Paris-Sud, Gif-Sur-Yvette, France. [2] Mass Spectrometry Laboratory, CNRS, Institut Jacques Monod, UMR 7592, Université de Paris, 75205 Paris, France. [3] CNRS, Institut Jacques Monod (IJM), Univ. Paris Diderot, Paris, France.

Received: 14 June 2019 Accepted: 20 July 2019
Published online: 01 August 2019

References

1. Léger T, Garcia C, Videlier M, Camadro J-M. Label-free quantitative proteomics in yeast. In: Methods in molecular biology (Clifton, NJ). 2016. p. 289–307.
2. Bantscheff M, Schirle M, Sweetman G, Rick J, Kuster B. Quantitative mass spectrometry in proteomics: a critical review. Anal Bioanal Chem. 2007;389(4):1017–31.
3. Neilson KA, Ali NA, Muralidharan S, Mirzaei M, Mariani M, Assadourian G, et al. Less label, more free: approaches in label-free quantitative mass spectrometry. Proteomics. 2011;11(4):535–53.
4. Merhej J, Thiebaut A, Blugeon C, Pouch J, Ali Chaouche MEA, Camadro J-M, et al. A network of paralogous stress response transcription factors in the human pathogen *Candida glabrata*. Front Microbiol. 2016;7:645.
5. Merhej J, Delaveau T, Guitard J, Palancade B, Hennequin C, Garcia M, et al. Yap7 is a transcriptional repressor of nitric oxide oxidase in yeasts, which arose from neofunctionalization after whole genome duplication. Mol Microbiol. 2015;96(5):951–72.
6. Benchouaia M, Ripoche H, Sissoko M, Thiébaut A, Merhej J, Delaveau T, et al. Comparative transcriptomics highlights new features of the iron starvation response in the human pathogen *Candida glabrata*. Front Microbiol. 2018;9:2689.
7. Léger T, Garcia C, Ounissi M, Lelandais G, Camadro J-M. The Metacaspase (Mca1p) has a dual role in farnesol-induced apoptosis in *Candida albicans*. Mol Cell Proteomics. 2015;14(1):93–108.
8. Léger T, Garcia C, Collomb L, Camadro J-M. A simple light isotope metabolic labeling (SLIM-labeling) strategy: a powerful tool to address the dynamics of proteome variations in vivo. Mol Cell Proteomics. 2017;16(11):2017–31.
9. Lelandais G, Denecker T, Garcia C, Danila N, Léger T, Camadro J-M. Label-free quantitative proteomics in Candida yeast species: technical and biological replicates to assess data reproducibility. PRIDE. 2019. http://identifiers.org/pride.project:PXD014125.
10. Lelandais G. Label-free quantitative proteomics in Candida yeast species: technical and biological replicates to assess data reproducibility. ZENODO. 2019. https://doi.org/10.5281/zenodo.3334949.
11. Fernandes TR, Segorbe D, Prusky D, Di Pietro A. How alkalinization drives fungal pathogenicity. PLOS Pathog. 2017;13(11):e1006621.

Publisher's Note

2. Collaboration avec l'équipe de Jean-Charles Cadoret : le logiciel START-R

a. Contexte du projet

La réplication de l'ADN doit être fidèle et suivre un programme spatio-temporel bien défini, étroitement lié à l'activité transcriptionnelle, aux marques épigénomiques, aux structures intra-nucléaires, aux taux de mutation et à la détermination du devenir des cellules. Les analyses du programme temporel de la réplication nécessitent des procédures expérimentales complexes, précises et longues, ainsi que l'étude de fichiers informatiques de grande taille. Pour faciliter son étude, nous avons d'une part amélioré le protocole expérimental pour l'accélérer et augmenter sa qualité et sa reproductibilité notamment par l'automatisation de certaines étapes et d'autre part proposé un ensemble d'outils bioinformatiques pour réaliser simplement l'analyse de données.

b. Logiciel START-R

Pour simplifier l'analyse des données générées lors d'une étude du programme temporal de la replication de l'ADN, nous avons développé la suite START-R (*Simple Tool for the Analysis of the Replication Timing based on R*). La suite START-R est un ensemble de deux applications web *Open Source* : START-R Analyzer, pour analyser les données et START-R Viewer pour visualiser les résultats. Elles ont été implémentées à l'aide du package R Shiny et permettent d'analyser des données provenant de puces à ADN ou de séquençage à haut débit. Ces outils peuvent être utilisés sans connaissances spécifiques en bioinformatique. Ils réduisent également le temps nécessaire pour générer et analyser simultanément les données de plusieurs échantillons. La suite START-R détecte les régions de temps constant (CTR) mais aussi, et c'est une nouveauté, des régions de transition temporelle (TTR) et détecte des différences significatives entre deux conditions expérimentales. L'analyse informatique globale nécessite moins de 10 minutes. La suite START-R est présentée en détails dans la publication disponible en ligne[189] et insérée à la section suivante. Le code source est disponible sur Github[190].

[189] https://pubmed.ncbi.nlm.nih.gov/30286789 [Accessible le 02/06/2020]
[190] https://github.com/thomasdenecker/START-R [Accessible le 02/06/2020]

c. PDF de l'article en cours de publication pour la revue « NAR Genomics and Bioinformatics » (Hadjadj et al., 2020)

Published online 19 June 2020

NAR Genomics and Bioinformatics, 2020, Vol. 2, No. 2 **1**
doi: 10.1093/nargab/lqaa045

Efficient, quick and easy-to-use DNA replication timing analysis with START-R suite

Djihad Hadjadj[1,†], Thomas Denecker[2,†], Eva Guérin[1], Su-Jung Kim[1], Fabien Fauchereau[1], Giuseppe Baldacci[1], Chrystelle Maric[1,‡] and Jean-Charles Cadoret[1,*,‡]

[1]Pathologies de la Réplication de l'ADN, Université de Paris; Institut Jacques-Monod, UMR7592, CNRS, F-75006 Paris, France and [2]Institut de Biologie Intégrative de la Cellule, UMR9198, CNRS, Université Paris-Saclay, Université Paris-Sud, F-91405 Orsay, France

Received February 20, 2020; Revised May 19, 2020; Editorial Decision May 28, 2020; Accepted June 15, 2020

ABSTRACT

DNA replication must be faithful and follow a well-defined spatiotemporal program closely linked to transcriptional activity, epigenomic marks, intranuclear structures, mutation rate and cell fate determination. Among the readouts of the spatiotemporal program of DNA replication, replication timing analyses require not only complex and time-consuming experimental procedures, but also skills in bioinformatics. We developed a dedicated Shiny interactive web application, the START-R (Simple Tool for the Analysis of the Replication Timing based on R) suite, which analyzes DNA replication timing in a given organism with high-throughput data. It reduces the time required for generating and analyzing simultaneously data from several samples. It automatically detects different types of timing regions and identifies significant differences between two experimental conditions in ∼15 min. In conclusion, START-R suite allows quick, efficient and easier analyses of DNA replication timing for all organisms. This novel approach can be used by every biologist. It is now simpler to use this method in order to understand, for example, whether 'a favorite gene or protein' has an impact on replication process or, indirectly, on genomic organization (as Hi-C experiments), by comparing the replication timing profiles between wild-type and mutant cell lines.

INTRODUCTION

DNA replication is a highly regulated process involved in the maintenance of genome stability (1–3). Its accuracy relies partly on a spatio-temporal program that regulates timing and location of origin firing (4,5). Based on this program, replication is organized into large-scale domains that replicate at different times in S phase (6–8). Protocols developed to study the replication timing (RT) in specific cell lines have been established in different laboratories (9–13). A script dedicated to RT analysis was previously developed by David Gilbert's laboratory (10,12), but it required skills in bioinformatics and R language. In order to make the analysis of experimental results easier for biologists, we implemented an interactive suite, START-R (Simple Tool for the Analysis of the Replication Timing based on R) Analyzer and START-R Viewer, showing user-friendly interfaces. This START-R suite is dealing with RT experiments made with microarrays or with Repli-seq data (either Early/Late or S/G1 ratios) from different organisms. These web applications would make easier differential analyses of RT, by comparing conditions such as treated/untreated cells or mutated/normal cells. They are free and may be improved by developers, according to specific needs. In addition, RT profiles correlate with A/B compartment profiles predicted by chromosome conformation methods. Regions of the genome defined by Hi-C profiles as A compartments are also identified as Early replicated domains, whereas regions defined as B compartments are Late replicating domains (14). Furthermore, some replicating domains coincide with a subset of topologically associating domains and more closely with the ones located at compartment boundaries (15). Studies of RT programs with START-R suite take shorter time to perform and therefore open new research perspectives for many laboratories working in DNA replication, in chromosome conformation and in other closely related molecular processes.

MATERIALS AND METHODS

START-R suite

START-R Analyzer and START-R Viewer (doi:10.5281/zenodo.3251905) were developed using the Shiny R package (W. Chang, J. Cheng, J. Allaire,

*To whom correspondence should be addressed. Tel: +33 1 57 27 80 74; Email: jean-charles.cadoret@ijm.fr
†The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.
‡The authors wish it to be known that, in their opinion, the last two authors should be regarded as Joint Last Authors.

Y. Xie and J. McPherson, Shiny: web application framework for R, R package version 1.2.0, 2018, https://CRANR-project.org/package=shiny). The source code and the installation procedure are available on GitHub (https://github.com/thomasdenecker/START-R). All R packages used in the START-R suite are listed in the Readme file on GitHub. Users should install Docker and then follow installation procedures described for each operating system (Windows, Linux and Mac OS) in the Readme file (Figure 1). Although the installation of these web-based applications has been simplified as much as possible, it may even so require some computer knowledge (especially on Linux). Once installed, the START-R suite is an easy-to-use tool requiring no computer skills (see Supplementary Figure S1 and the Wiki section on GitHub for more details).

Validation of START-R suite using microarray data from other laboratories

We analyzed with the START-R suite the microarray data obtained by Hiratani *et al.* (16) of D3esc and D3npc9 cell lines during mouse cells' differentiation (GEO accession numbers: GSM450273 and GSM450285, respectively). As data extracted from the Nimblegen platform are in PAIR format, we used a script to convert data into a valid format for START-R Analyzer (convertPair.R, available on GitHub in 'supplement script' file, https://github.com/thomasdenecker/START-R).

Early/Late-seq data and conversion

In order to validate our software, we used data corresponding to Repli-seq 46C mouse cells—Early S fraction or Late S fraction, respectively (GSM2496038 and GSM2496039). Read mapping was obtained using Bowtie2 (2.3.4.2 version) with the very sensitive end-to-end option. Then, PCR duplicates were removed by RmDUP from SAMTools (2.0.1 version). BamCoverage parameters were fixed (3.1.2.0.0 version with default parameters) to a bin size of 10 kb (corresponding to the genomic distance between microarray probes) and reads per kilobase million (RPKM) normalization to generate a bedGraph file. A headline was added to the file to name the four columns (chr, start, end and gProcessedSignal for Early file or rProcessedSignal for Late file, respectively). Then, a script to convert and merge the bedGraph files from Early and Late samples to a format compatible with START-R Analyzer was developed. This script is available on GitHub in 'supplement script' file as convert_bamcoverage_file.R.

Validation of START-R suite using S/G1 data from multiple species

Different laboratories analyze variations of DNA copy number between G1 and S phase cells (S/G1 ratio) to study the RT program with Repli-seq. We used data obtained from different organisms such as *Drosophila*, zebrafish and humans (17–19), to validate the START-R suite (GSM3154888, GSM3154890, GSM2282090, SRX3413939-40). As previously, BAM coverage files from G1 and S fractions are converted to a format compatible with START-R Analyzer with the aforementioned 'convert_bamcoverage_file.R.' script.

RESULTS

RT analysis with START-R suite allows robust statistical analysis

We developed a software allowing an automatic detection of RT regions and a differential analysis between two conditions. The START-R suite is implemented into an HTML interface for more efficient use and sharing by biologists. START-R is built-in and packaged into a virtual environment with Docker (Figure 1). Thus, START-R can be easily deployed on a personal computer or on a server, and can run independently of any library updating. START-R provides as many parameters as possible for a comprehensive analysis of the RT program (Supplementary Figure S1A–K). Furthermore, we added new scaling, normalization and smoothing methods (Supplementary Figure S2) and also novel statistical approaches to detect differences between two samples. A classical differential analysis performed with START-R takes ∼15 min with a standard laptop computer. START-R Analyzer can run data from all organisms with different genome assemblies. This flexibility is one of the new aspects of the START-R suite that allows to analyze RT program in every organism (Supplementary Figure S1B). In addition, START-R generates RT profiles in PDF and all the files necessary for a better and customized visualization with START-R Viewer. START-R Analyzer also produces specific files that could be visualized with START-R Viewer, a specific genome browser tool dedicated to START-R suite.

A large panel of new settings and tools for RT analysis

We based our method on four major steps: *normalization* (between Early and Late fractions, between two replicates and between two independent experiments; Supplementary Figure S2A), *smoothing* (different options are available; Supplementary Figure S2B), *identification* of transition timing regions (TTRs; Supplementary Figure S3) and *segmentation*. The originality of our approach is to first detect TTRs in order to better identify constant timing regions (CTRs; Supplementary Figure S3). The identification of TTRs is based on their intrinsic properties: regions that include more than three consecutive probes with significantly different Early/Late intensity log ratios are considered as TTRs (Supplementary Figure S3). The statistical significance of differences between intensity log ratios is calculated by the outlier box plot method (Supplementary Figure S4) (20). Following TTR detection, START-R Analyzer localizes CTRs: TTRs are subtracted from the genome (Supplementary Figure S3) and the remaining regions are considered as CTRs. At the end of these steps, START-R Analyzer automatically generates a BED file for CTRs and TTRs making easier further bioinformatic analyses and the display of the RT domains via a genome browser (Supplementary Figure S1F). A codebook is also generated to ensure the traceability of options chosen for each analysis.

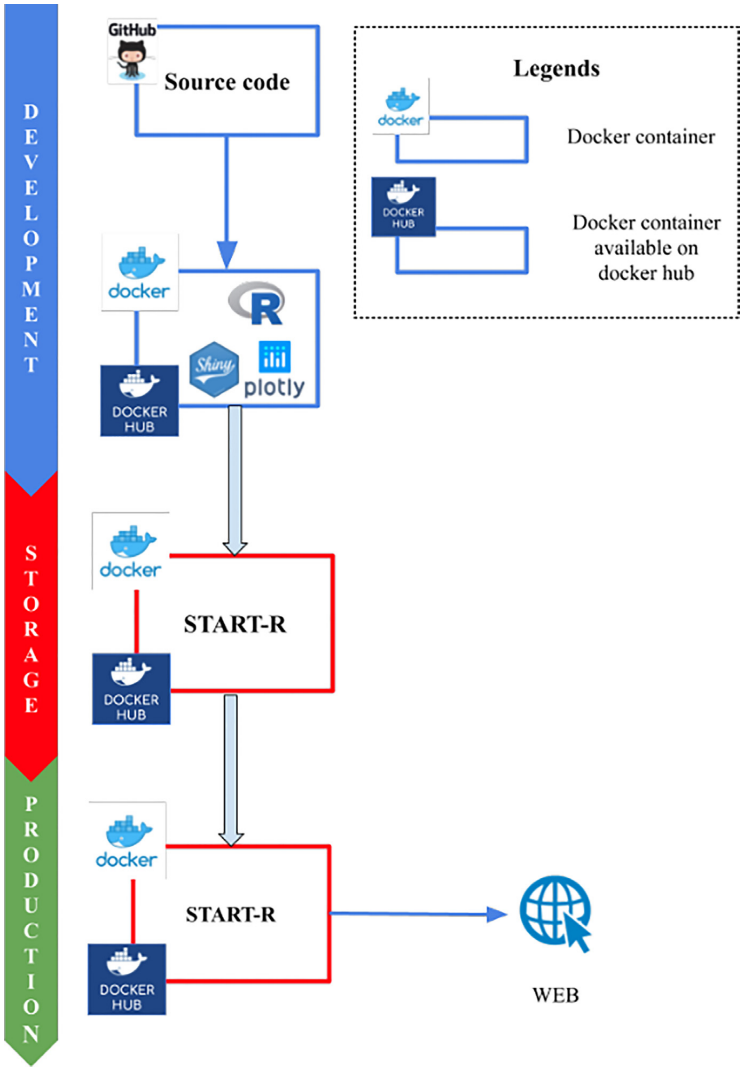


Figure 1. Stack overview of the START-R suite. The START-R suite is able to analyze all types of genome-wide RT data formats like microarray data, Repli-seq data with Early/Late and S/G1 ratios, and RT data from multiple organisms. The START-R suite was developed using the Shiny R package (W. Chang, J. Cheng, J. Allaire, Y. Xie and J. McPherson, Shiny: web application framework for R, R package version 1.2.0, 2018, https://CRANR-project.org/package=shiny) and Plotly visualization tools (C. Sievert, C. Parmer, T. Hocking, S. Chamberlain, K. Ram, M. Corvellec and P. Despouy, plotly: create interactive web graphics via 'plotly.js', R package version 4.7.1, 2017, https://CRANR-project.org/package=plotly). The START-R software (START-R Analyzer and START-R Viewer) are open-source web-based applications (doi:10.5281/zenodo.3251905). For the storage and production steps, the START-R suite was concatenated using Docker in order to install, use and share it easily. These software can be used with different operating systems: Windows, Mac OS and Linux. The source code and the installation procedure are available on GitHub (https://github.com/thomasdenecker/START-R). To install the START-R suite, users should install Docker and follow the Readme file containing the installation procedure (https://github.com/thomasdenecker/START-R/blob/master/README.md). Finally, in order to run the START-R suite, users should double-click on the START-R file (Windows) or launch the command line (Mac OS X, Linux), followed by opening an internet browser at the following URLs: http://localhost:3838/ for START-R Analyzer and http://localhost:3839/ for START-R Viewer.

We added a step allowing the differential analysis of RT programs from two experiments. Thus, we can now compare RT profiles obtained in different conditions and/or with different cell lines to identify genomic elements that can modify the RT program. Our differential analysis includes three different methods of comparison: the Mean method, the Euclidean method and the Segment comparison method. When the goal is to identify most regions with strong RT changes, we recommend using the most stringent Mean method. The less stringent Segment and Euclidean methods allow the detection of a larger set of RT changes, while increasing the risk of obtaining false positives.

The last major implementation is START-R Viewer (Supplementary Figure S1K). This web-based interface allows the visualization of the RT profile generated by START-R Analyzer in dynamic charts obtained with the Plotly library (C. Sievert, C. Parmer, T. Hocking, S. Chamberlain, K. Ram, M. Corvellec and P. Despouy, plotly: create interactive web graphics via 'plotly.js', R package version 4.7.1, 2017, https://CRANR-project.org/package=plotly).

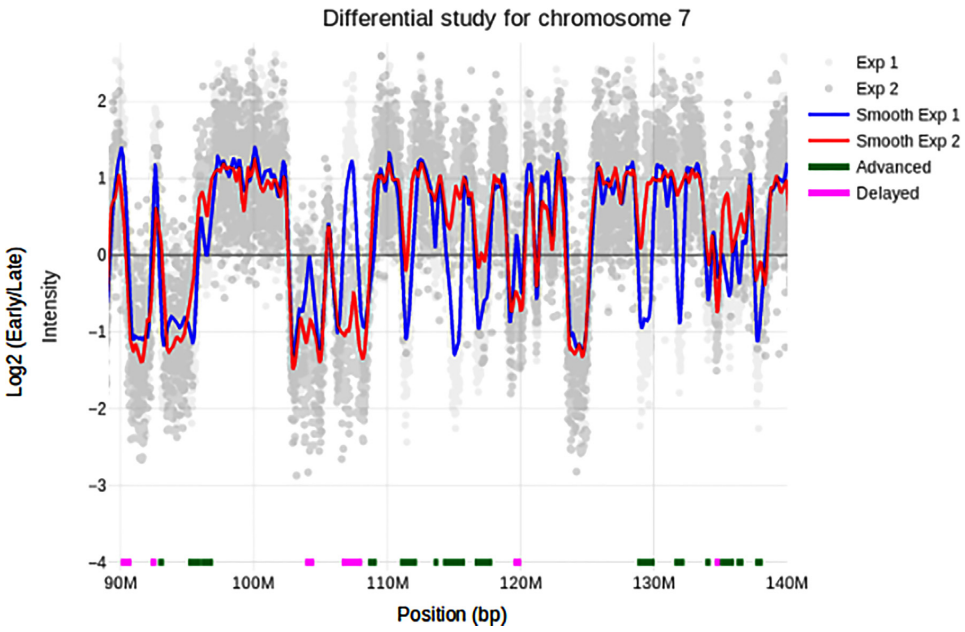


Figure 2. Genomic characteristics of regions harboring different RT programs. START-R Viewer allows the visualization of RT data with many features. The START-R differential analysis of RT profiles is shown here for a portion of chromosome 7 in mouse D3esc (blue) and D3npc9 (red) cells. We used START-R Analyzer with the standard options: Loess Early/Late normalization, scale inter-replica normalization, inter-experiment standardization, Loess method for smoothing (span = 300 kb), 2.5 for SD difference between two segments and mean comparison analysis with a Holm's method *P*-value of 0.05 for the differential analysis. The advanced (green) and delayed (pink) regions identified with START-R Analyzer are indicated underneath the RT profiles. Light gray and gray spots indicate data from both RT experiments. With these parameters, 2066 CTRs were detected in the genome and 910 regions showed different RT between D3esc and D3npc9 cells. Box plots illustrating genomic characteristics (GC content, LINE-1 content and gene coverage) of regions harboring different RT programs are shown in Supplementary Figure S6.

One can easily identify CTRs, TTRs (Supplementary Figure S5A) and significantly advanced or delayed regions (Supplementary Figure S5B). We therefore developed a genome browser to optimally display the maximum of data generated by START-R Analyzer. The START-R suite also automatically generates files with different output formats essential for further molecular characterizations and compatible for classical bioinformatic tools and/or for GALAXY genomic tools (21).

START-R analysis of RT programs during differentiation in mouse: a new analysis of previous data

To validate our START-R based-approach without *a priori* consideration, we decided to re-analyze the data generated by the Gilbert's group concerning the changes of RT program during cell differentiation in mouse D3esc and D3npc9 cell lines (16). We converted these raw data with the convertPair.R script available in our GitHub project into the correct format for START-R Analyzer. RT profiles generated with START-R Analyzer (Figure 2) and molecular signatures (Supplementary Figure S6) are identical to the ones previously described by Gilbert's group. Each modified timing region had a particular molecular signature: Late-to-Early or advanced regions show a GC/LINE-1 density and gene coverage similar to constant Early regions, while Early-to-Late or delayed regions showed GC/LINE-1 density and gene coverage similar to constant Late regions.

Validation of START-R with Early–Late Repli-seq data from mouse

Nowadays, many data of RT program are generated with Repli-seq experiments, but their analysis is time-consuming and often requires bioinformatics skills. We analyzed the Early/Late Repli-seq data from Marchal *et al.* (12). We specifically developed a supplemental script to convert the BAM coverage file to a log Early/Late file (convert_bamcoverage_file.R) to be sure that the integration into the START-R pipeline was correct. Then, we compared the RT smooth profiles generated from Early/Late Repli-seq data with those generated by the same group using microarrays (Figure 3A). Profiles are almost identical to the ones described by Marchal *et al.* (12). Thus, START-R Analyzer and START-R Viewer can be easily used to analyze Early/LateIDEX Repli-seq data, showing their versatility and their simplicity of use.

Validation of START-R with S–G1 Repli-seq data from *Drosophila*, zebrafish and humans

Other laboratories use the ratio of DNA content between G1 and S phases to analyze the RT program. We wanted to know whether START-R suite can run the correct analyses with this type of data and also with other organisms than mice and humans. We performed exactly the same pipeline used for Early/Late Repli-seq data described above with *Drosophila*, zebrafish and human S/G1 data (17–19). Then and as expected, START-R can be run with S/G1 log ratio

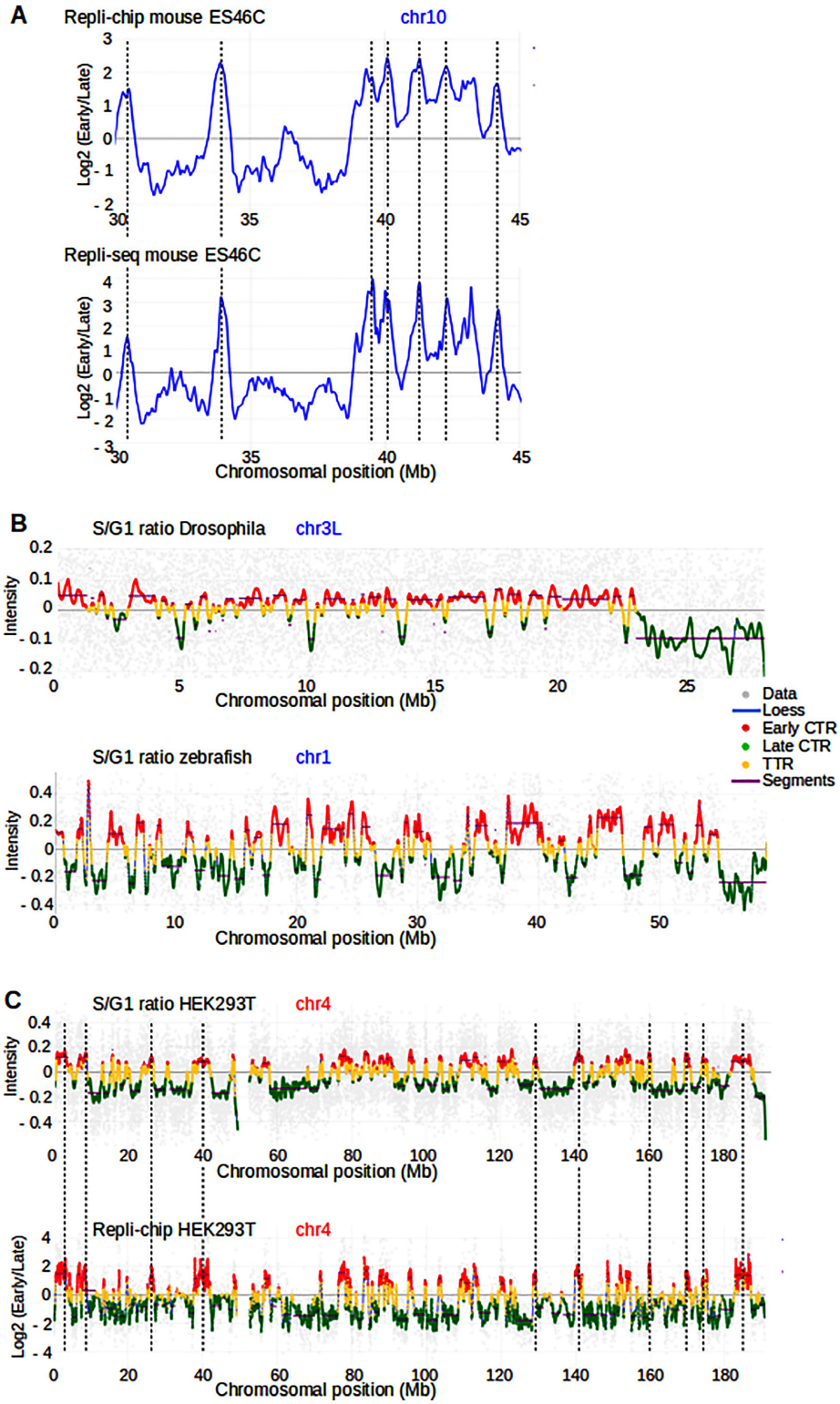


Figure 3. The START-R suite allows analysis and visualization of both Repli-chip and Repli-seq data from different model systems. (**A**) RT profiles of a portion of mouse chromosome 10 from ES46C cell line are generated using Repli-chip (top panel) and Repli-seq data (bottom panel) with START-R web applications. Dashed vertical lines show common RT regions between both profiles. (**B**) RT profiles obtained by S/G1 ratios are shown for the left part of *Drosophila* chromosome 3 (3L) and for zebrafish chromosome 1 (blue lines). The profiles display distribution of Early and Late CTRs, in red and green, respectively, and of TTRs, in yellow. Segments corresponding to regions of constant timing are shown in purple. Gray spots indicate data from RT experiments. (**C**) RT profiles of human HEK293T chromosome 4 are generated using S/G1 ratio and Repli-chip data. The empty space inside the RT profiles represents the centromere region.

data (Figure 3B and C). We observed similar profiles as the ones already observed for these different organisms.

DISCUSSION

In this study, we show a new automated protocol for analyzing RT profiles (Figure 1), obtained with different methods, in all organisms. As a proof of concept, we succeed in generating RT analyses for human, mouse, *Drosophila* and zebrafish genomes (Figures 2 and 3). START-R suite's user-friendly interface allows choices between different parameters at all steps used to generate RT profiles (Supplementary Figure S1). Compared to the previous methods (10,12), START-R Analyzer first detects TTRs and thus better refines and improves the CTR detection (Supplementary Figures S3 and S5A). In addition, START-R Analyzer contains new calculation methods for differential analyses between two conditions or cell lines (Supplementary Figure S5B).

This flexibility gives the users the opportunity to choose the differential analysis method and different parameters according to their questions. It also automatically generates files with different output formats essential for further molecular characterizations and compatible for classical bioinformatic tools and/or for GALAXY genomic tools (21). Then, START-R Viewer produces a nice interface to visualize all the data generated by START-R Analyzer. Furthermore, START-R suite freeware are available on GitHub and their source codes are open to anyone who wants to improve them, as, for example, for studies of allelic changes of RT.

In conclusion, it is now possible for any biologist or laboratory to readily explore new or previous RT data simply and quickly. Thus, a large number of laboratories can today use our software to find out whether their experimental conditions are affecting the RT process or are correlated with other molecular mechanisms. START-R also allows to determine what parts of the genome are impacted and in which proportion and to characterize further those loci. Thanks to the accessibility of our approaches and software, their speed and efficiency, new research perspectives can be efficiently envisaged.

SUPPLEMENTARY DATA

Supplementary Data are available at NARGAB Online.

ACKNOWLEDGEMENTS

We thank Gaëlle Lelandais for helpful discussions. We also acknowledge the ImagoSeine core facility of the Institut Jacques-Monod, member of the France BioImaging (ANR-10-INBS-04) supported by the Region Île-de-France (E539). This project was supported by the generous legacy from Ms Suzanne Larzat to our group.

FUNDING

La Ligue Nationale Contre le Cancer RS16/75-108, RS17/75-135; GEFLUC; Institut National du Cancer INCa-10493; IDEX Université de Paris ANR-18-IDEX-0001.

Conflict of interest statement. None declared.

REFERENCES

1. Hanahan,D. and Weinberg,R.A. (2011) Hallmarks of cancer: the next generation. *Cell*, **144**, 646–674.
2. Macheret,M. and Halazonetis,T.D. (2015) DNA replication stress as a hallmark of cancer. *Annu. Rev. Pathol. Mech. Dis.*, **10**, 425–448.
3. Técher,H., Koundrioukoff,S., Nicolas,A. and Debatisse,M. (2017) The impact of replication stress on replication dynamics and DNA damage in vertebrate cells. *Nat. Rev. Genet.*, **18**, 535–550.
4. Dileep,V., Rivera-Mulia,J.C., Sima,J. and Gilbert,D.M. (2015) Large-scale chromatin structure–function relationships during the cell cycle and development: insights from replication timing. *Cold Spring Harb. Symp. Quant. Biol.*, **80**, 53–63.
5. Rivera-Mulia,J.C. and Gilbert,D.M. (2016) Replicating large genomes: divide and conquer. *Mol. Cell*, **62**, 756–765.
6. Ryba,T., Hiratani,I., Lu,J., Itoh,M., Kulik,M., Zhang,J., Schulz,T.C., Robins,A.J., Dalton,S. and Gilbert,D.M. (2010) Evolutionarily conserved replication timing profiles predict long-range chromatin interactions and distinguish closely related cell types. *Genome Res.*, **20**, 761–770.
7. Cornacchia,D., Dileep,V., Quivy,J.P., Foti,R., Tili,F., Santarella-Mellwig,R., Antony,C., Almouzni,G., Gilbert,D.M. and Buonomo,S.B.C. (2012) Mouse Rif1 is a key regulator of the replication-timing programme in mammalian cells. *EMBO J.*, **31**, 3678–3690.
8. Desprat,R., Thierry-Mieg,D., Lailler,N., Lajugie,J., Schildkraut,C., Thierry-Mieg,J. and Bouhassira,E.E. (2009) Predictable dynamic program of timing of DNA replication in human cells. *Genome Res.*, **19**, 2288–2299.
9. Hansen,R.S., Thomas,S., Sandstrom,R., Canfield,T.K., Thurman,R.E., Weaver,M., Dorschner,M.O., Gartler,S.M. and Stamatoyannopoulos,J.A. (2010) Sequencing newly replicated DNA reveals widespread plasticity in human replication timing. *Proc. Natl Acad. Sci. U.S.A.*, **107**, 139–144.
10. Ryba,T., Battaglia,D., Pope,B.D., Hiratani,I. and Gilbert,D.M. (2011) Genome-scale analysis of replication timing: from bench to bioinformatics. *Nat. Protoc.*, **6**, 870–895.
11. Dileep,V., Didier,R. and Gilbert,D.M. (2012) Genome-wide analysis of replication timing in mammalian cells: troubleshooting problems encountered when comparing different cell types. *Methods*, **57**, 165–169.
12. Marchal,C., Sasaki,T., Vera,D., Wilson,K., Sima,J., Rivera-Mulia,J.C., Trevilla-García,C., Nogues,C., Nafie,E. and Gilbert,D.M. (2018) Genome-wide analysis of replication timing by next-generation sequencing with E/L Repli-seq. *Nat. Protoc.*, **13**, 819–839.
13. Petryk,N., Kahli,M., d'Aubenton-Carafa,Y., Jaszczyszyn,Y., Shen,Y., Silvain,M., Thermes,C., Chen,C.L. and Hyrien,O. (2016) Replication landscape of the human genome. *Nat. Commun.*, **7**, 10208–10220.
14. Miura,H., Takahashi,S., Poonperm,R., Tanigawa,A., Takebayashi,S.I. and Hiratani,I. (2019) Single-cell DNA replication profiling identifies spatiotemporal developmental dynamics of chromosome organization. *Nat. Genet.*, **51**, 1356–1368.
15. Marchal,C., Sima,J. and Gilbert,D.M. (2019) Control of DNA replication timing in the 3D genome. *Nat. Rev. Mol. Cell Biol.*, **20**, 721–737.
16. Hiratani,I., Ryba,T., Itoh,M., Yokochi,T., Schwaiger,M., Chang,C.W., Lyou,Y., Townes,T.M., Schübeler,D. and Gilbert,D.M. (2008) Global reorganization of replication domains during embryonic stem cell differentiation. *PLoS Biol.*, **6**, 2220–2236.
17. Armstrong,R.L., Penke,T.J.R., Strahl,B.D., Matera,A.G., McKay,D.J., MacAlpine,D.M. and Duronio,R.J. (2018) Chromatin conformation and transcriptional activity are permissive regulators of DNA replication initiation in *Drosophila*. *Genome Res.*, **11**, 1688–1700.
18. Siefert,J.C., Georgescu,C., Wren,J.D., Koren,A. and Sansam,C.L. (2017) DNA replication timing during development anticipates

transcriptional programs and parallels enhancer activation. *Genome Res.*, **8**, 1406–1416.
19. Massey,D.J., Kim,D., Brooks,K.E., Smolka,M.B. and Koren,A. (2019) Next-generation sequencing enables spatiotemporal resolution of human centromere replication timing. *Genes (Basel)*, **10**, E269.
20. Krzywinski,M. and Altman,N. (2013) Error bars. *Nat Methods*, **10**, 921–922.
21. Afgan,E., Baker,D., Batut,B., Van Den Beek,M., Bouvier,D., Ech,M., Chilton,J., Clements,D., Coraor,N., Grüning,B.A. *et al.* (2018) The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. *Nucleic Acids Res.*, **46**, W537–W544.

3. Collaboration avec l'équipe de Cécile Fairhead : caractérisation des orthologues au sein du clade des *Nakaseomyces*

a. Contexte du projet

La collaboration avec l'équipe de C. Fairhead avait pour objectif d'étudier les données de transcriptomique (puce à ADN) en conditions de carence en fer (BPS) à 30 °C disponibles au laboratoire pour 4 levures du clade des *Nakaseomyces*. Cette collaboration a déjà été évoquée lors de la discussion du chapitre sur l'homéostasie du fer chez *C. glabrata* (page 150). Pour rappel, nous avons à notre disposition 4 jeux de données en conditions de carence en fer à 30 °C : un provenant de l'étude de *C. glabrata* (condition C1), un pour *C. bracarensis*, un pour *C. nivariensis* et un pour *N. delphensis*. Plus de détails sur les jeux de données et ces levures sont disponibles à la page 150. Nous y proposons aussi quelques résultats préliminaires. En partant de ces résultats, nous avons pu voir qu'un certain nombre de gènes n'avaient pas d'orthologue (Figure 60). À ce stade, deux hypothèses étaient possibles : (1) il n'y a pas de liens d'orthologie parce qu'ils n'existent pas ou (2) parce qu'ils n'ont pas été détectés par les algorithmes utilisés lors de la création des tables d'orthologie de phylomeDB. En suivant la philosophie d'une approche globale proposée par l'équipe d'O. Lespinet (Pereira et al. 2014), nous avons combiné plusieurs méthodes pour proposer des tables d'orthologie plus complètes ainsi que de nouvelles tables (pas d'orthologues décrits entre *C. bracarensis*, *C. nivariensis* et *N. delphensis*). Nous commencerons cette partie par rappeler ce qu'est un orthologue, nous présenterons ensuite les différents outils utilisés pour construire les nouvelles tables d'orthologie que nous explorerons pour finir.

b. Quelques généralités

Orthologues

Pendant cette collaboration, nous nous sommes concentrés sur les orthologues. Ce choix est basé sur la « conjecture de fonctions orthologiques» (Altenhoff et al. 2012; Rogozin et al. 2014). Les orthologues conservent la même fonction et peuvent donc être utilisés pour le transfert d'annotations fonctionnelles de gènes caractérisés expérimentalement vers des gènes non caractérisés (Gabaldón and Koonin 2013). Il est à noter que la recherche d'orthologues n'échappe pas aux défis imposés par le *Big data* (Sonnhammer et al. 2014). Pour approfondir

ces notions, une page est disponible sur la ressource numérique de la thèse[191] et nous conseillons la lecture de l'article de R. Fernández (Fernández et al. 2019).

Outils utilisés

Lorsque nous recherchons des orthologues, 3 approches principales peuvent être envisagées :

- Par similarité – Les approches fondées sur la similarité reposent sur la comparaison des séquences génomes et le regroupement des gènes les plus ressemblants ;
- Par phylogénie – Les approches fondées sur la phylogénie utilisent des familles de gènes candidats déterminées par similarité, puis s'appuient sur la phylogénie des espèces pour confirmer l'orthologie ;
- Par synténie – Deux gènes sont confirmés orthologues si les gènes qui les entourent sont aussi orthologues entre eux et qu'il y a une conservation de l'ordre et de l'orientation de ces gènes.

Pour réaliser les tables d'orthologie, nous avons combiné ces 3 approches. Un aperçu des outils actuellement disponibles est présenté proposé par B. Nichio (Nichio et al. 2017). Pour générer chaque table d'orthologie, nous avons utilisé :

- La table d'orthologie issue de phylomeDB (approches par phylogénie) – Nous avons utilisé la même table que celle présentée page 152 ;
- L'outil OrthoFinder (approche par similarité et par phylogénie) – OrthoFinder est un outil de référence rapide, précis et complet pour rechercher des orthologues (Emms et al. 2019). Il fait partie des outils plébiscités dans le domaine de la génomique comparative (Nichio et al. 2017).
- La méthode BHR (approche par similarité) – BHR est l'acronyme de *Best Reciprocal Hits*. Deux gènes sont orthologues si lorsque nous les alignons, le meilleur résultat pour un des deux gènes est l'autre gène. Nous devons retrouver le même résultat si nous inversons les gènes. La Figure 78 résume la méthode.

[191] https://thomasdenecker.github.io/thesisWebsite/annexes/orthologue/ [Accessible le 10/08/202020]

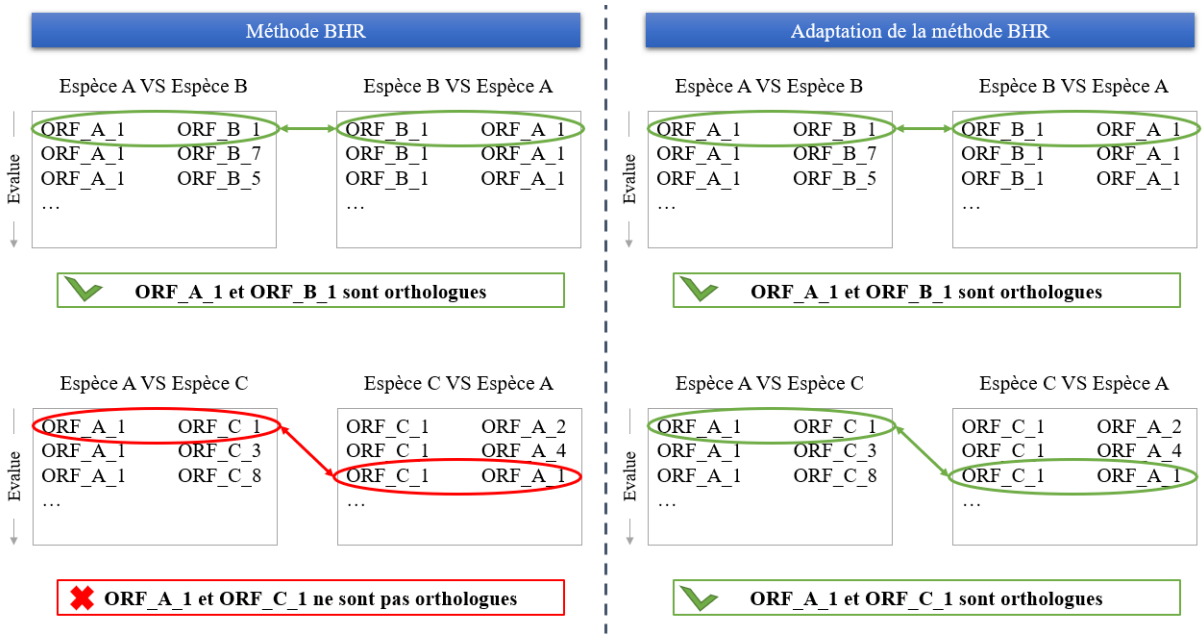


Figure 78 – Illustration schématique du principe de la méthode BHR et BHR adaptée.

- Une adaptation de la méthode BHR (approche par similarité) – Cette approche est basée sur la méthode BHR. Deux gènes sont orthologues si lorsque nous les alignons, le meilleur résultat pour un des deux gènes est l'autre gène. Si nous inversons les gènes, le premier gène doit être dans la liste des meilleurs gènes du gène testé. Plusieurs niveaux peuvent être explorés :
    - Nucléotidique – Les séquences alignées sont des séquences nucléotidiques. Ces séquences sont les plus simples d'accès ;
    - Protéique – Les séquences alignées sont des séquences protéiques. La conservation des séquences protéiques est meilleure et donc donne théoriquement de résultats plus complets lors d'une recherche d'orthologues (Chapitre 4 du livre de Eugene Koonin (Koonin et al. 2003)).
- Une recherche de synténie (approche par synténie) – Pour confirmer (ou proposer) un lien d'orthologie par synténie, les gènes qui encadrent un gène d'intérêt doivent être orthologues entre les deux espèces et dans le même ordre (Figure 79). Nous avons pour cela regardé 2 gènes avant et 2 gènes après. Nous avons dû cependant rester prudents avec les résultats de cette approche. En effet, nous avons rencontré 2 problématiques :
    - L'ordre des ORFs n'était pas le même entre les fichiers de la CGD et ceux de GRYC pour *C. glabrata*. Par conséquent chaque fichier a été contrôlé pour les mettre dans le bon ordre.

- Comme nous l'avons vu dans la présentation des 3 levures *C. bracarensis*, *C. nivariensis* et *N. delphensis*, l'annotation est encore partielle et les positions chromosomiques ne sont pas encore fixées. Dans ce cas, comment générer les gènes aux extrémités ? Comment ne pas écarter une synténie simplement parce que nous n'avons pas l'ordre définitif ? Actuellement, nous perdons de l'information notamment pour les gènes aux extrémités.

Cette méthode est pertinente et nous a permis de découvrir de nouveaux orthologues mais elle ne se suffit pas à elle-même (il faut que des orthologues soient identifiés initialement). Pour résumer, il s'agit d'une excellente méthode complémentaire.

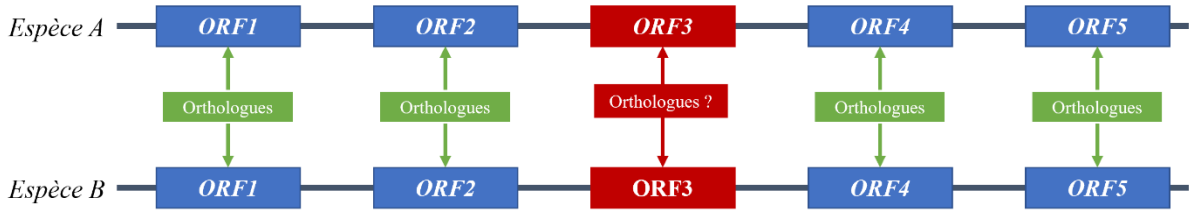


Figure 79 – Confirmation d'un lien d'orthologie entre deux éléments. Ici, nous savons que les ORF1, ORF2, ORF4 et ORF5 sont orthologues entre l'espèce A et B. Nous souhaitons confirmer l'orthologie entre les ORF3. Comme ORF1, ORF2, ORF4 et ORF5 sont conservés et dans le même ordre, nous pouvons conclure que les ORF3 sont orthologues entre les espèces A et B

Une fois toutes ces données générées, nous avons recherché un orthologue consensus. Chaque gène a une annotation pour évaluer l'association : Parfaite (tous les outils sont en accords), Bonne (la majorité des outils ont donné une réponse identique), Prudence (tous les outils n'ont pas donné de réponse), Douteux (les outils ont donné des réponses mais pas les mêmes), Multiples (les outils ont donné des réponses multiples) et Pas d'information (aucun orthologue n'a été trouvé). Par la suite, pour simplifier la lecture des résultats obtenus, nous avons regroupé sous « Validé » les annotations Parfaite et Bonne, « Absent » lorsque nous n'avions pas d'information disponible et les autres catégories dans « Prudence ».

Les résultats ont été collectés, organisés et visualisés à l'aide de R. Plusieurs packages ont été utilisés dont `dplyr` pour la gestion des données (Wickham et al. 2020) et `ggplot2`, `ggrepel` et `UpSetR` pour la visualisation des données (Wickham 2016; Slowikowski 2020; Gehlenborg 2019)

L'environnement de travail (DockerHub), l'ensemble des scripts et des données (GitHub) sont pour le moment en accès restreint aux collaborateurs en attendant la publication des résultats.

c. Résultats

Nouvelle table d'orthologie

En utilisant les différents outils présentés dans le paragraphe précédent, nous avons abouti à une table d'orthologie entre *C. glabrata* et chacune des levures étudiées. Les annotations des orthologues trouvés sont regroupées dans la Figure 80.

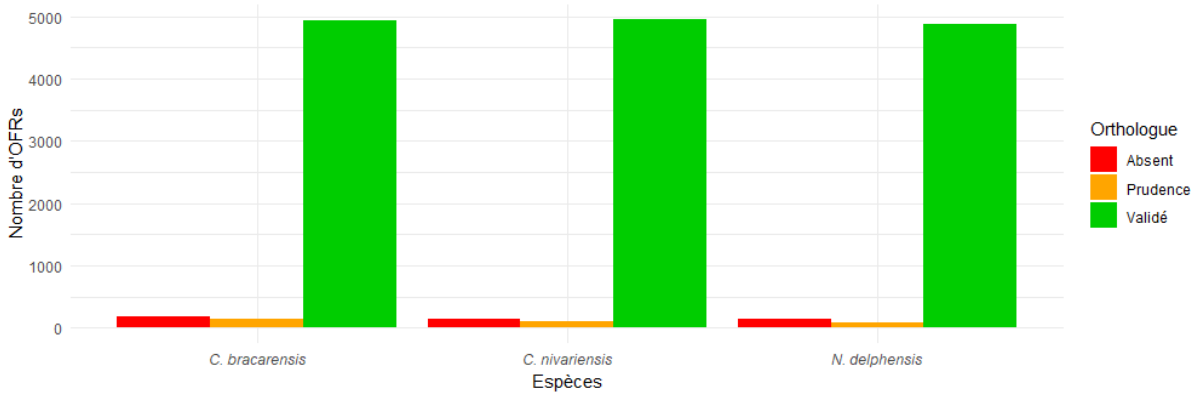


Figure 80 – Annotation associée aux orthologues trouvés entre C. glabrata et les 3 autres levures étudiées. La majorité des ORFs ont un orthologue associé.

Un constat est frappant : il y a une excellente conservation entre *C. glabrata* et ces 3 espèces. Un orthologue a été trouvé pour la grande majorité des gènes. Notre approche est donc validée. Néanmoins, avons-nous trouvé de nouveaux orthologues ? Oui et dans près de la moitié des cas, l'orthologue associé a été validé (Figure 81). Notre approche apporte donc une plus-value sur la recherche d'orthologue pour ces 4 levures.

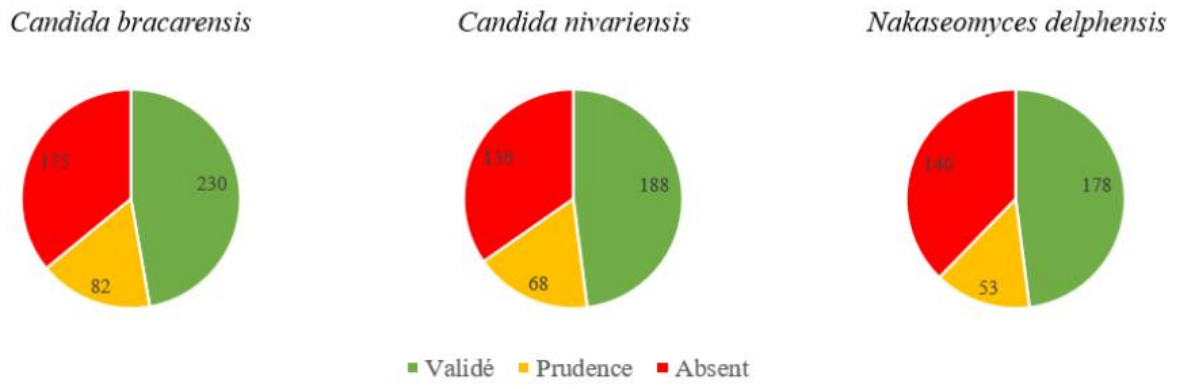


Figure 81 – Annotation des orthologues associés aux gènes qui n'avaient pas d'orthologue sur phylomeDB

Nous nous sommes ensuite demandés si cette observation était aussi valable en se focalisant sur les gènes réagissant au fer chez *C. glabrata*. Notre approche permet de trouver un orthologue pour presque la moitié des gènes qui n'en avaient pas par les tables d'orthologie de phylomeDB (Figure 82). De plus les orthologues associés sont d'un haut niveau de confiance (plus de 75% sont validés). Pour ceux qui n'ont toujours pas été associés, nous pouvons constater que très peu sont de Type I. Nous avions souligné l'importance de ces Type I dans l'homéostasie du fer chez *C. glabrata*.

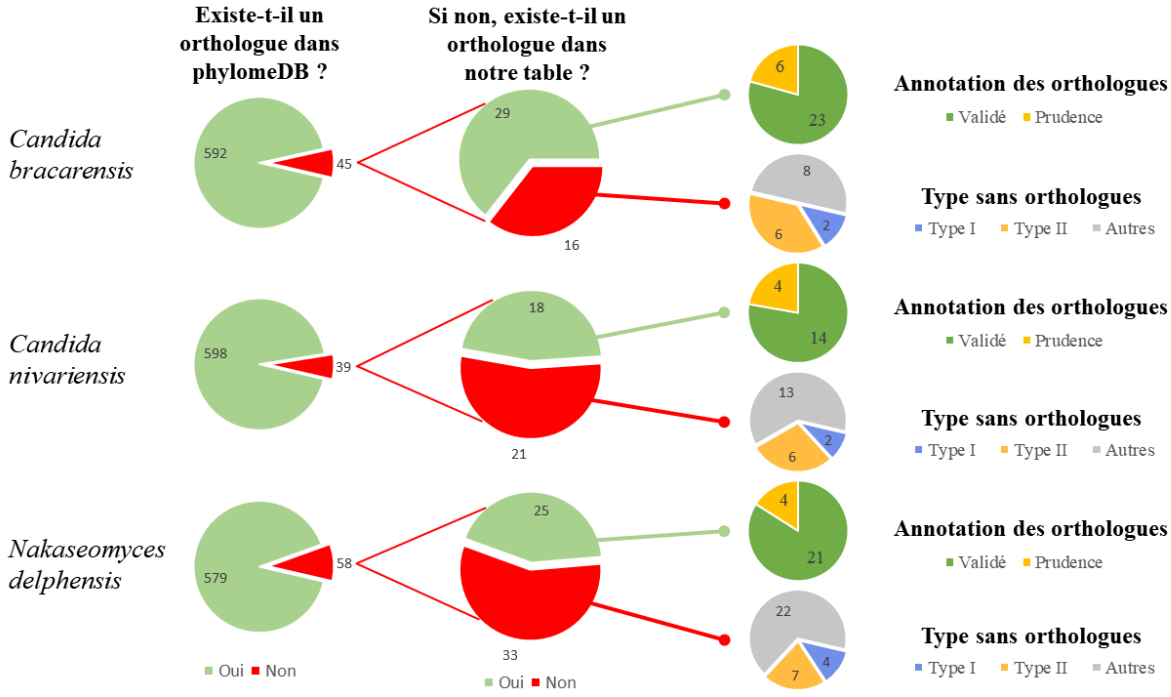


Figure 82 – Amélioration de la recherche d'orthologues dans le clade des Nakaseomyces

Dans la discussion des résultats de l'étude de l'homéostasie du fer chez *C. glabrata* (page 152), nous avions mis en évidence une liste de 73 gènes réagissant au fer chez *C. glabrata* et dont les orthologues réagissaient aussi au fer chez *C. nivariensis* et *C. bracarensis*. Pour s'assurer de la conservation, nous avons construit une table d'orthologie entre *C. bracarensis* et *C. nivariensis*. Ces 73 gènes sont bien orthologues entre ces deux espèces. Nous avons ainsi une confirmation que les gènes essentiels semblent conservés fortement au sein du clade des *Nakaseomyces*.

Pour finir cette exploration, nous souhaitions savoir si les gènes réagissant au fer chez *C. glabrata* n'ayant toujours pas d'orthologue n'avaient pas d'orthologue dans une ou dans plusieurs des espèces étudiées (Figure 83). Pour un tiers des gènes, aucun orthologue n'a été trouvé dans les 3 espèces (12 gènes / 37).

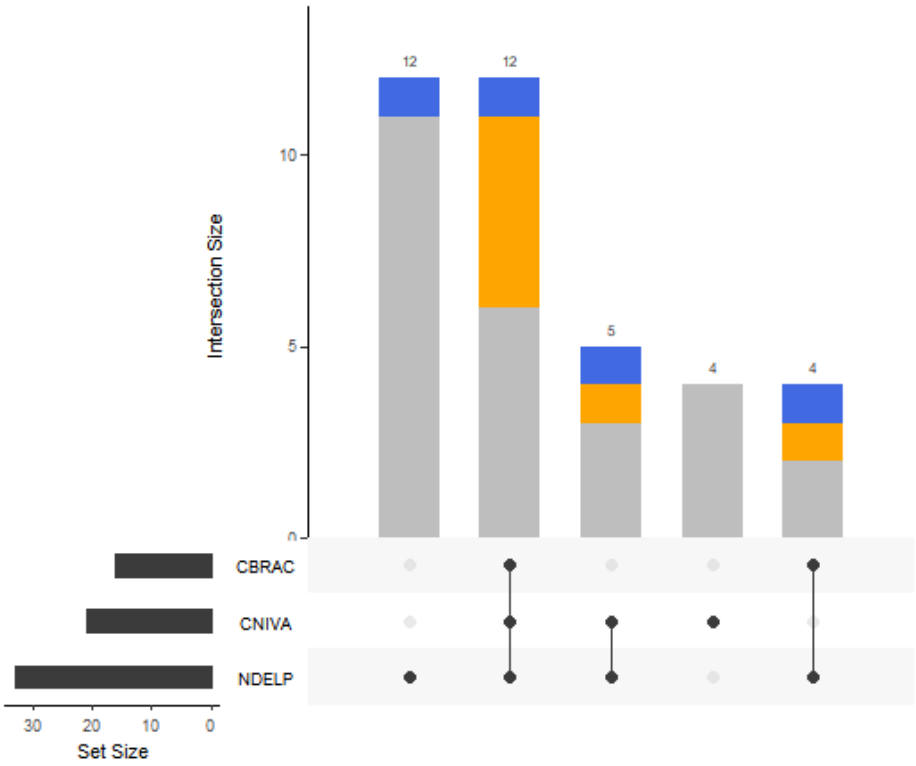


Figure 83 – Croisement des listes de gènes ne possédant toujours pas d'orthologues chez C. glabrata et faisant partie de la liste des gènes réagissant au fer. Les couleurs des barres présentes sur les barres correspondent aux types des gènes réagissant au fer chez C. glabrata (bleu pour les gènes de types I, orange pour les gènes de types II et gris pour les autres).

La liste de ces 12 gènes est présentée dans le Tableau 15. Nous pouvons faire plusieurs observations à partir de ce tableau. Pour commencer la majorité des gènes ne sont pas différentiellement exprimés en conditions de carence en fer à 30 °C (75%). Ensuite, nous les avons généralement classés dans la grande fonction *Others*. Enfin, l'annotation fonctionnelle

est peu présente pour ces gènes (dans plus de la moitié des cas la fonction protéique est inconnue). En résumé, nous savons que ces gènes sont impliqués dans l'homéostasie du fer mais les différentes informations à notre disposition ne permettent pas encore de déterminer pourquoi ces gènes n'ont pas d'orthologue au sein du clade des *Nakaseomyces*.

Feature name	Gene name	Description from CGD	logFC	P-value	General function
CAGL0A02277g		Protein of unknown function	-0.20	4.9e-01	Others
CAGL0B02970g	BMT5	Beta mannosyltransferase	0.15	3.5e-01	Others
CAGL0C02365g		Protein of unknown function	2.10	2.2e-05	Others
CAGL0G06050g		Protein of unknown function	0.19	4.7e-01	Others
CAGL0K07183g		Protein of unknown function	0.35	3.9e-02	Others
CAGL0L09537g		Has domain(s) with predicted FMN binding, catalytic activity, oxidoreductase activity and role in oxidation-reduction process	1.20	5.8e-04	Redox signaling
CAGL0J05390g		Protein of unknown function	0.21	1.8e-01	Others
CAGL0A02299g		Protein of unknown function	1.80	6.9e-05	Others
CAGL0B02904g	BMT6	Beta mannosyltransferase	1.30	5.1e-05	Others
CAGL0C00275g	HSP31	Putative cysteine protease; protein differentially expressed in azole resistant strain; gene is upregulated in azole-resistant strain	3.90	5.1e-06	Others
CAGL0D05082g		Protein of unknown function	2.60	3.0e-06	Metabolism
CAGL0I09702g		Ortholog(s) have riboflavin transmembrane transporter activity, role in riboflavin transport and plasma membrane localization	1.20	4.4e-05	Transport / trafficking

Tableau 15 – Liste de gènes n'ayant pas d'orthologue chez C. bracarensis, C. nivariensis et N. delphensis et faisant partie des gènes réagissant au fer chez C. glabrata

d. Discussion et perspectives

En conclusion, nous avons pu montrer à travers ces résultats préliminaires qu'il existe une forte conservation des gènes clés de l'homéostasie du fer au sein du clade des *Nakaseomyces*. Cette étude pourra aller plus loin lorsque l'annotation des levures étudiées sera plus complète. Au laboratoire, plusieurs perspectives sont envisageables pour ce travail. Pour commencer, au niveau computationnel, nous pourrions continuer à améliorer la recherche d'orthologues par exemple au niveau de la synténie. Actuellement, pour qu'un orthologue soit associé à un gène, il faut que les 2 gènes avant et les 2 gènes après soient dans le bon ordre et orthologues deux à deux. Nous pourrions imaginer passer à 6 gènes (3 avant et 3 après) pour permettre des « erreurs » (par exemple, un gène qui ne serait pas dans l'ordre ou absent). Nous pourrions aussi utiliser les données disponibles sur le MetaPhOrs 2.0, un outil qui calcule des orthologues et des paralogues[192] à partir de la phylogénie (Chorostecki et al. 2020). D'après les informations disponibles en ligne, les données sont en cours de calcul et seront bientôt disponibles au téléchargement. Une limite actuelle est qu'il n'y a pas encore de données pour *C. bracarensis* et *C. nivariensis*. Enfin, au niveau biologique, nous pourrions nous poser 3 questions :

1) Quels sont les gènes réagissant au fer chez les levures *C. bracarensis*, *C. nivariensis* et *N. delphensis* ? Nous pourrions, comme nous l'avons fait pour *C. glabrata*, mettre en avant une liste de gènes sans *a priori* avec les levures modèles et *C. glabrata*.
2) Existe-t-il un « core transcriptome » des gènes impliqués dans l'homéostasie du fer dans le clade des *Nakaseomyces* ? Nous pourrions pour l'identifier croiser les différentes listes obtenues à la première question et rechercher une conservation au sein du clade puis des spécificités liées aux levures.
3) Existe-t-il des gènes impliqués dans la pathogénicité et dans l'homéostasie du fer ? Nous avons pour répondre à cette question des données pour 3 levures pathogènes (*C. glabrata*, *C. bracarensis* et *C. nivariensis*) et une levure non pathogène (*N. delphensis*). Une hypothèse pourrait être que des gènes trouvés uniquement dans les 3 levures pathogènes et pas chez *N. delphensis* seraient impliqués dans la pathogénicité.

En résumé, il reste encore des questions scientifiques multiples à explorer.

[192] La notion de paralogue est présentée dans la ressource numérique de la thèse : https://thomasdenecker.github.io/thesisWebsite/annexes/orthologue/ [Accessible le 21/08/2020]

Bilan de thèse et Conclusion

Au cours de ma thèse, j'ai eu la chance de travailler dans le laboratoire de Bioinformatique Moléculaire d'Olivier Lespinet (I2BC) et dans des laboratoires de biologie expérimentale : en génomique et transcriptomique dans les équipes de Cécile Fairhead (IDEEV) et Fabienne Malagnac (I2BC) et en protéomique dans l'équipe de Jean-Michel Camadro (IJM). Ma position de « bioinformaticien infiltré » auprès des biologistes à la paillasse m'a permis de rester en contact étroit avec la réalité expérimentale et au plus proche des questions biologiques concrètes. J'ai eu l'occasion de m'impliquer à la fois dans le développement d'outils et de nouvelles approches méthodologiques, mais aussi dans l'analyse de données directement produites dans les laboratoires. Cette partie aura pour objectif de dresser un bilan de cette expérience de thèse.

I. Ce que nous avions prévu

Comme on dit : « *On sait où une thèse commence, on ne sait pas où elle termine* ». Lorsque j'ai postulé pour une bourse à l'école doctorale "Structure et Dynamique des Systèmes Vivants", le titre du projet était *Intégration de données multi-omiques pour une modélisation in silico du métabolisme du Fer des levures pathogènes* et était résumé de la façon suivante :

> *Dans de le cas d'infections fongiques chez l'Homme, l'accès à des ressources de fer est un élément critique de la relation hôte / pathogène. Les levures pathogènes Candida ont développé au cours de leurs évolutions des stratégies originales pour capter le fer de l'hôte et adapter leur métabolisme à des conditions de vie dans des milieux pauvres en Fer. Ce projet a pour objectif de réaliser une modélisation in silico du métabolisme du fer à partir de données expérimentales « multi-omiques » (génomique, transcriptomique, protéomique, etc.). Différentes espèces Candida seront étudiées afin d'identifier leurs spécificités d'action, au regard de leurs modes d'infections.*

Lors de l'audition, nous avons proposé un programme en 4 parties : (1) poursuivre le travail sur l'homéostasie du fer chez *C. glabrata* initié au cours du stage de Master 2 pour aboutir à un modèle de *in silico* de son métabolisme du fer, (2) réaliser une étude similaire chez *C. albicans* puis (3) chez d'autres levures pathogènes et enfin (4) comparer les modèles obtenus afin de mettre en évidence des éléments communs et spécifiques aux différentes levures étudiées (Figure 84).

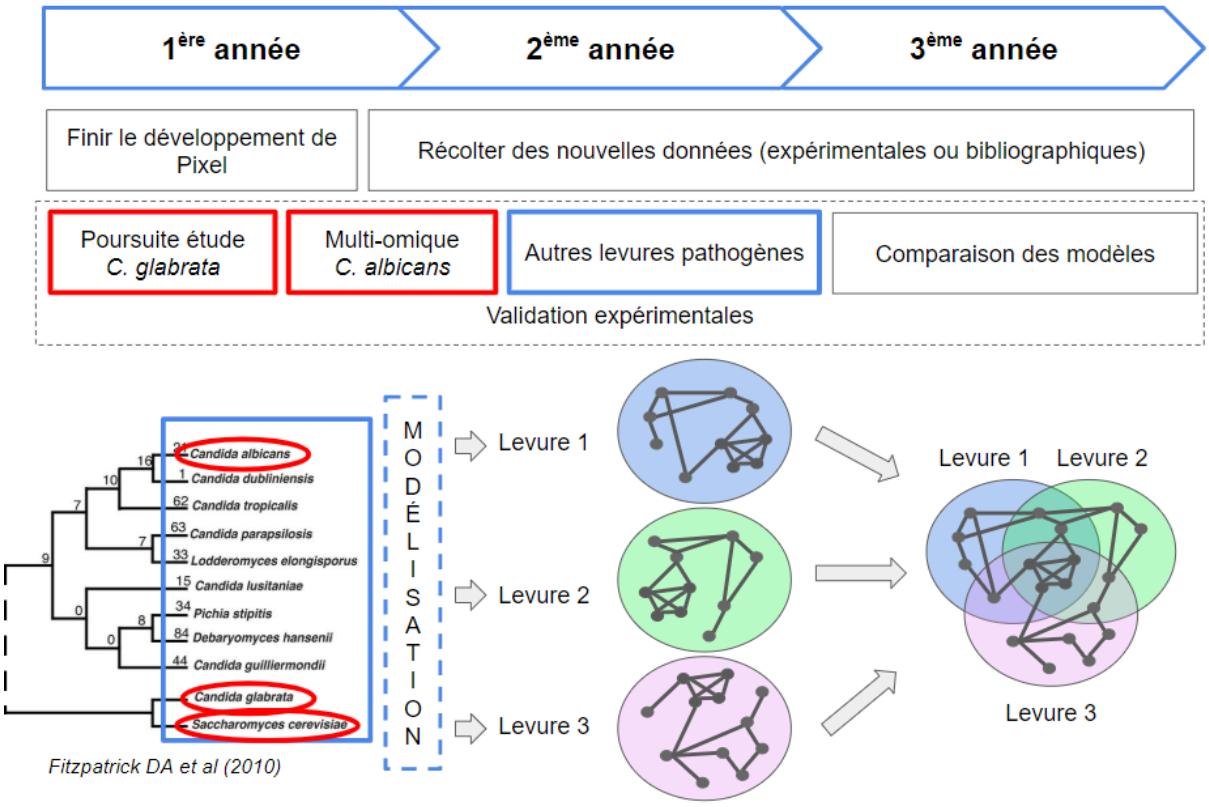


Figure 84 – Programme de thèse envisagé lors de l'audition pour l'obtention d'une bourse ministérielle à l'école doctorale "Structure et Dynamique des Systèmes Vivants" (diapositive extraite de la présentation).

Pour nous, le vrai défi de la thèse reposait sur le passage entre les approches descriptives et la modélisation prédictive (Figure 85). Nous devions dans un premier temps étudier de nombreux composants pour identifier les éléments pouvant intégrer le modèle. Ensuite, nous devions trouver les liens entre ces différents composants puis nous devions collecter un grand nombre de données spécifiques à ces composants afin de mettre en évidence une dynamique entre eux. La dernière étape consistait à proposer des équations mathématiques pour prédire le comportement de ces interactions. Notre souhait était de faire des aller-retours constants à chaque étape entre la biologie expérimentale et les modèles *in silico*.

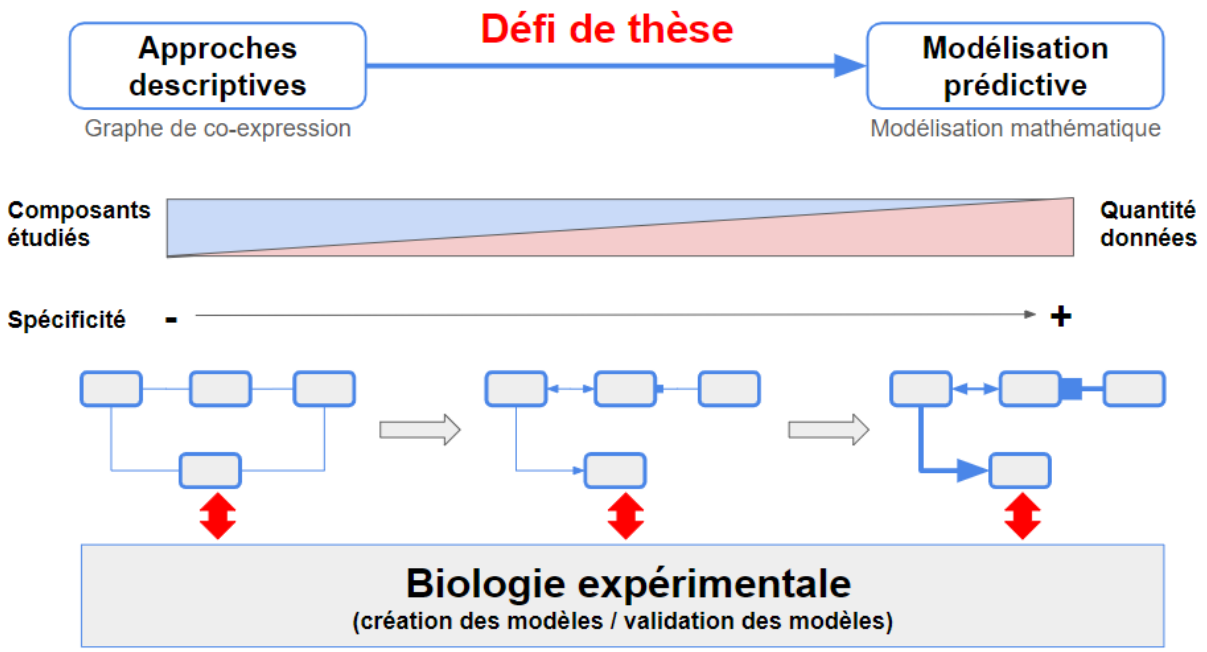


Figure 85 – Illustration de ce que nous considérions comme le défi le plus complexe de la thèse lors de l'audition pour l'obtention d'une bourse ministérielle à l'école doctorale "Structure et Dynamique des Systèmes Vivants" (la figure est extraite de la présentation).

II. Ce que nous avons fait

La première année de ma thèse a donc consisté à collecter un maximum de données et à les organiser. Nous avons développé le logiciel Pixel. Aujourd'hui, l'instance Pixel dédiée à ma thèse contient plus de 250 000 couples gène/donnée spécifiques au fer et à *C. glabrata*. Nous avions initialement la volonté d'intégrer l'ensemble de ces données collectées, mais très rapidement, nous avons mieux cerné les différentes problématiques présentées dans ce manuscrit (bruit lié à l'accumulation de données, opposition de résultats entre deux expériences, etc.). Nous avons donc décidé de limiter les jeux de données afin de mettre en évidence une première liste de gènes d'intérêt puis de la confronter aux autres données. Les résultats obtenus par cette stratégie sont ceux présentés dans le chapitre « Étude transcriptomique pour l'exploration des mécanismes d'homéostasie du fer chez *Candida glabrata* » (page 128).

Pour approfondir mes connaissances en modélisation et proposer un modèle prédictif comme nous l'avions prévu, j'ai suivi une semaine de formation avec des mathématiciens et des informaticiens lors de l'école thématique du CNRS : Modélisation Formelle de Réseaux de Régulation Biologique[193] (été 2019). Cette formation a été une vraie prise de conscience des difficultés de ce domaine. Même s'il s'agit « du premier pas vers la connaissance »[194], je me suis posé de nombreuses questions sur la suite à donner à ma thèse. De plus, j'étais désireux de travailler sur tous les omiques et me confronter au *Big data* pour mettre en pratique mes compétences en informatique (acquises notamment lors de la collaboration avec TailorDev pour le développement de Pixel). Impliquée dans une ANR avec Jean-Michel Camadro, Gaëlle m'a alors proposé un projet qui alliait la protéomique et le *Big data*. Très motivé par ces problématiques et les défis proposés par la spectrométrie de masse, je suis allé travailler dans l'équipe de Jean-Michel. Les résultats obtenus sont présentés dans le chapitre « Étude systématique des modifications post-traductionnelles des protéines chez *Candida albicans* » (page 160). En parallèle de mes travaux de thèse, j'ai eu la liberté de travailler avec d'autres équipes de recherches sur de nouvelles thématiques (voir le chapitre « Contributions à d'autres projets », page 193).

[193] http://www.i3s.unice.fr/bioregul/ [Accessible le 03/06/2020]

[194] "To be conscious that you are ignorant is a great step to knowledge.", Benjamin Disraeli

Enfin, j'ai eu l'opportunité, en plus de ma charge d'enseignements au sein de l'IUT d'informatique d'Orsay, de pouvoir m'impliquer dans la formation au sein de la communauté des biologistes et des bioinformaticiens : (1) Par la création de la formation FAIR_Bioinfo (décembre 2018 - juillet 2019); (2) Par la rédaction d'une initiation au développement d'application web avec R et Shiny (avril 2019) ; (3) En suivant puis en intervenant dans le Diplôme Universitaire Création, analyse et valorisation de données biologiques Omiques (diplômé de la promotion 2028, j'ai aidé G. Lelandais avec la promotion 2019) ; (4) En étant tuteur à la 8ème école de bioinformatique AVIESAN-IFB : Initiation au traitement des données de génomique obtenues par séquençage à haut débit (atelier RNAseq, novembre 2019)[195].

En conclusion, au cours de ma thèse, j'ai travaillé à différents niveaux de omiques (génomique, transcriptomique et protéomique) à travers différentes études de génomiques fonctionnelles (Figure 86). Entre bioinformatique et analyses de données, nous avons généré de nouvelles informations qui aboutiront espérons-le à de nouvelles connaissances. Rythmée par des collaborations et des formations, cette thèse m'a permis de m'immerger au sein d'équipes de recherche expérimentale et de rencontrer des chercheurs de tous horizons. Initiée avec de nombreuses questions scientifiques, elle se conclut avec des réponses mais aussi de nouvelles questions passionnantes.

[195] https://www.france-bioinformatique.fr/fr/evenements/3955 [Accessible le 03/06/2020]

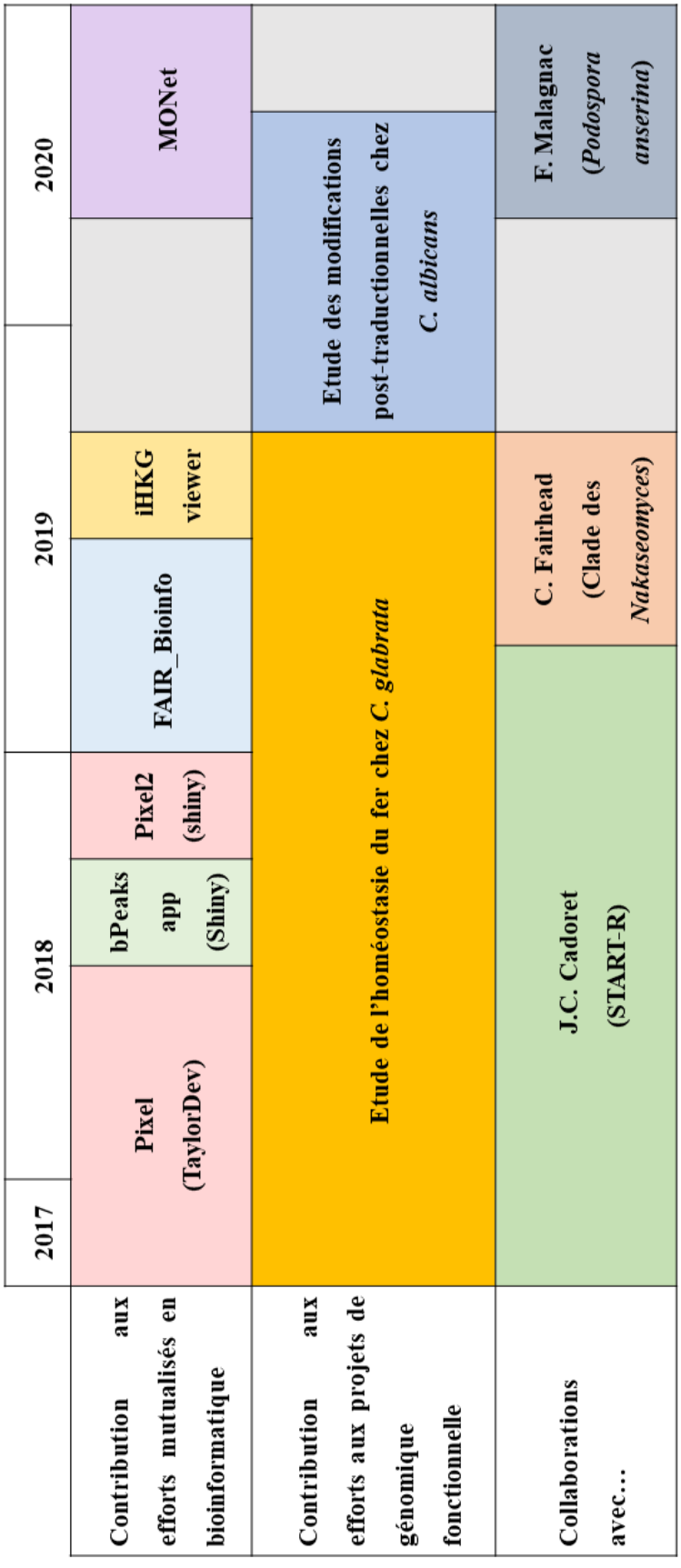


Figure 86 – Les différents projets menés au cours de cette thèse (2017-2020)

III. Ce que nous aurions pu ou pourrions faire

La rédaction de ce manuscrit de thèse aura été pour moi l'opportunité de prendre un peu de recul vis-à-vis de mon travail de scientifique. Si je pouvais revenir en arrière que ferais-je différemment ? Et pour la suite, dans quelle direction est-ce que je souhaite me diriger ? Une chose est sûre, j'ai au fil des années développé un intérêt grandissant pour les *buzz words* suivants : *Big data*, *data science* et *Intelligence artificielle*. Dans ce dernier chapitre, j'ai souhaité présenter ma vision de ces thèmes, dans un contexte de recherche en Biologie.

1. Le big data en biologie

a. Plus de données, pour quoi faire ?

Quelques chiffres

En 2017, l'EBI (European Bioinformatics Institute) gérait à travers ses différentes plateformes de l'ordre de 120 pétaoctets de données (Cook et al. 2017). Si ces données étaient écrites sur des disques Blu-ray d'une capacité de stockage de 25 Go chacun, une pile de près de 6 km pourrait être constituée. Il s'agit de l'altitude du Kilimandjaro ! La compagnie DOMO prévoyait en 2018 que chaque personne génèrerait 1.7 Mo de données par seconde en 2020[196]. Elle crée tous les ans une infographie illustrant la quantité de données produites au cours du temps (*Data never sleeps)*. Ainsi la dernière version[197] montre que chaque minute, 208 333 personnes participent à des réunions sur Zoom, presque 42 millions de messages sont distribués sur WhatsApp et 347 222 stories sont postées sur Instagram. Selon une étude du cabinet de recherche international dans le domaine des technologies IDC en collaboration avec Seagate (Reinsel et al. 2018), le volume de données stockées atteindra 175 Zettaoctets[198] en 2025, soit 5,3 fois plus que ce qui est stocké actuellement.

---

[196] https://www.socialmediatoday.com/news/how-much-data-is-generated-every-minute-infographic-1/525692/ [Accessible le 21/04/2020]

[197] La version 2020 est disponible en ligne à l'adresse suivante : https://www.domo.com/learn/data-never-sleeps-8 [Accessible le 24/08/2020]

[198] 175 000 000 000 Gigaoctets !

Plus de données, donc moins d'erreurs ?

Le volume des données augmentant de façon très rapide, l'automatisation de leur traitement est une stratégie utilisée pour réduire le temps d'analyse. Une problématique est alors de savoir si ces analyses automatiques sont fiables. S. Leonelli pose ainsi la question suivante : « Comment l'automatisation de l'analyse des données affecte-t-elle la fiabilité des résultats ? »[199] (Leonelli 2019). La conséquence de l'accumulation des données serait une diminution de la fiabilité des informations qui leur sont associées par l'accumulation d'erreurs d'analyses. Mais est-ce une problématique nouvelle ?

Les erreurs font partie intégrante du processus des découvertes scientifiques. En restant limité aux pensées et aux méthodes établies, l'avancement des connaissances serait fortement ralenti. Des erreurs peuvent mener à des découvertes ou tout simplement ouvrir une nouvelle réflexion. Alors pourquoi craindre autant les erreurs ? Dans un article publié dans PNAS, les auteurs classent les erreurs en différentes catégories (A. W. Brown et al. 2018) :

- Erreurs dans les choix analytiques – Il s'agit des erreurs liées aux choix réalisés lors de la conception et la réalisation d'une analyse de données (choix des données, choix des modèles statistiques, choix des modes de représentation, etc.). Nous pouvons y retrouver :
    - Les erreurs liées à la création des données – Nous souhaitons étudier l'effet d'une carence en fer par exemple, sur des cellules de levures, et nous oublions de mettre le chélateur[200] de fer dans le milieu de culture. Les données obtenues seront donc incorrectes et les informations associées erronées ;
    - Les erreurs liées à la manipulation des données – L'utilisation du logiciel Excel est sans doute l'exemple le plus populaire en bioinformatique. Des noms de gènes peuvent être modifiés automatiquement si l'utilisateur manque de vigilance (Ziemann et al. 2016) ;
    - Les erreurs lors de l'application des méthodes d'analyses statistiques – Ce point a déjà été largement discuté dans la première partie de ce manuscrit ;

[199] *"How does the automation of data analysis affect the reliability of results?"*

[200] Ligand pouvant former un complexe avec un cation métallique (ici le fer)

- Les erreurs de logique – Principalement fondées sur des opinions *a priori*. Ainsi certaines études sur les effets de produits tels que le tabac[201], le téflon[202] ou le radium[203] n'ont pas toujours été sans partis pris ou préjugés (le plus souvent pour des questions commerciales et économiques) ;
- Les erreurs de communication – Il s'agit par exemple d'une surinterprétation des observations sur un graphique. L'exemple du mélange entre « corrélation » et « causalité » est emblématique dont de très bonnes illustrations sont proposées par Tyler Vigen[204]. La problématique de *p-hacking*, évoquée dans la première partie de cette thèse est un autre exemple.

- Erreurs invalidantes – « *Il s'agit d'erreurs factuelles ou de déviations importantes par rapport à des procédures clairement acceptées qui, si elles sont corrigées, peuvent modifier les conclusions d'un document* »[205] (Allison et al. 2016) comme :
  - Des erreurs liées à l'ignorance – La communauté scientifique connaît une erreur ou un biais particulier, mais pas la personne qui réalise le travail. Ces erreurs sont souvent la cause d'un manque de préparation ou de communication ;
  - Des erreurs liées à un mauvais plan d'analyse – Ce point a été largement discuté dans la première partie ;
  - La pression de la publication – En recherche, cette pression peut être très forte… ;
  - L'enthousiasme – Les scientifiques sont des passionnés. L'excitation face à un résultat très attendu peut parfois l'emporter au détriment de la raison, conduisant par exemple à des surinterprétations ;

---

[201] Vous pourrez trouver sur le site https://www.vivelapub.fr/fumez-cest-bon-pour-votre-sante-selon-la-pub/ [Accessible le 23/04/2020] un ensemble de publicité pour le tabac. Celle de Lucky Stike est par exemple appuyée par 9651 docteurs !

[202] Le film Dark waters de Todd Haynes (2020) est un parfait résumé de la prise de conscience des dangers du téflon et de toutes les difficultés rencontrées pour le prouver.

[203] 15 publicités vantant les mérites du radium : http://www.topito.com/top-pubs-vantent-bienfaits-produits-radioactifs [Accessible le 23/04/2020]

[204] http://tylervigen.com/spurious-correlations [Accessible le 23/04/2020]

[205] "*These involve factual mistakes or veer substantially from clearly accepted procedures in ways that, if corrected, might alter a paper's conclusions.*"

- Les ressources disponibles – Faute d'accès à des experts, faute de temps, etc. des compromis peuvent être faits au prix de la perfection des analyses et donc potentiellement des erreurs.

Ces erreurs sont-elles amplifiées par le *Big data* ?

L'analyse de données, à l'échelle du *Big Data* nécessite de relever des défis spécifiques. Un petit biais avec peu de données peut aboutir à un grand biais avec beaucoup de données. C'est exactement la problématique des tests multiples qui a été évoquée dans la première partie de ce manuscrit (page 77). Mes différentes lectures et expériences m'ont permis d'identifier 5 problématiques majeures, en relation avec le *Big data* :

- Trouver du signal dans le bruit – S. Leonelli pose la question suivante : « *Quelle est la différence entre les données et le bruit, et quelles sont les données de premier plan* ? »[206] (Leonelli 2019). Il est souvent très difficile de conclure à l'absence d'un signal parce que deux explications sont possibles : (1) le signal est faible et il est perdu dans le bruit ou (2) il n'y a pas de signal à trouver. Rechercher un signal dans beaucoup de données est un peu comme « *chercher une aiguille dans une botte de foin* », mais sans savoir si l'aiguille existe ;
- Un fort biais de confirmation – Chercher dans un grand jeu de données, c'est aussi trouver à coup sûr au moins un indice qui confirme n'importe quel préjugé ou idée préconçue que l'on pourrait avoir. C'est ainsi que les *Fake News* se multiplient dans l'actualité ;
- Accumulation de données fausses – Selon le livre blanc proposé en 2016 par Experian (Experian 2016), 75% des entreprises pensent que les données de leurs clients sont fausses. Que penser des informations obtenues à partir de ces données ? Quelles connaissances pourront être produites ?
- Une intégration difficile – Le *Big data* n'est pas qu'une problématique de volume de données, mais c'est aussi une question de variabilité et d'hétérogénéité des données. Intégrer des données hétérogènes est sans doute un des plus importants défis pour les bioinformaticiens aujourd'hui ;

[206] "*What is the difference between data and noise, and what are data in the first place?"*

- Un manque de spécialistes en « analyse de données » – Un rapport de CapGemini révèle que 37% des entreprises ont du mal à trouver des analystes de données qualifiés. Actuellement, les entreprises recrutent puis forment elles-mêmes leur personnel. L'université fait évoluer en conséquence ses programmes de formation, et les Unités d'Enseignement en *Data science* sont créés. Actuellement, ces formations sont toutefois essentiellement proposées aux étudiants des filières en informatique. Un des enjeux des prochaines années sera de créer des enseignements adaptés aux bioinformaticiens et aux biologistes.

Ainsi, si la problématique des « erreurs » en science n'est pas une spécificité du *Big data* en biologie, elle y est toutefois amplifiée. L'expérience, l'expertise et le partage de ressources sont des éléments importants dans ce contexte.

b. Raisons pour générer toujours plus de données

Compte tenu des problématiques sous-jacentes au *Big data*, pourquoi les chercheurs continuent-ils à produire toujours plus de données ? N'est-il pas possible de simplement exploiter les données déjà disponibles ? Au moment de la rédaction de ce bilan de thèse, j'ai essayé de comprendre les raisons qui poussent à générer toujours plus de données. Voici ce que nos expériences m'ont appris :

- Les métadonnées n'ont globalement pas encore le niveau – Leur qualité globale est encore insuffisante pour en tirer pleinement parti et les informations fournies ne sont souvent pas standardisées, elles sont incomplètes et hétérogènes. Ainsi, si un ensemble de données n'est pas correctement décrit, alors il ne sera pas exploitable par d'autres.
- Il n'y a souvent pas de contrôle systématique des erreurs par des experts – Certaines bases de données le font, comme Swiss-Prot mais elles restent peu nombreuses. Et pour cause, comment vérifier toutes les données générées dans un contexte de *Data deluge* ? Le recours à l'automatisation est à la fois indispensable et source d'erreurs comme nous l'avons vu précédemment. Observées fortuitement, ces erreurs créent une perte de confiance globale pour la personne qui imagine re-exploiter un jeu de données. Or, la confiance est indispensable pour mener à son terme une analyse de données.
- Les données n'ont pas été générées exactement de la façon souhaitée – Lors de ma thèse, nous avons rencontré cette problématique au moment de la confrontation de nos résultats avec les données de la littérature. Si des différences sont observées, quelle en

est la cause ? La technique expérimentale, les conditions expérimentales, la méthode d'analyse, etc. ?

Toutes ces interrogations peuvent décourager une équipe de recherche qui préfèrera générer de nouvelles données plutôt que de « perdre » du temps avec des données déjà existantes. Dans le cadre de cette thèse, nous avons pris le parti d'exploiter autant que possible les données existantes. Pour tenir compte des risques évoqués précédemment, nous avons suivi le conseil *« Faites confiance mais vérifiez »*[207]. Ainsi, toutes les informations extraites d'un jeu de données devaient être retrouvées dans un autre jeu de données indépendant.

Enfin il me semble que l'on a souvent l'idée que « *plus il y a de données, mieux c'est* ». Pour les statisticiens, plus la taille des échantillons est grande, plus la puissance des tests statistiques est importante. Pour les *data scientists*, plus il y de données à consulter, plus les modèles prédictifs sont fiables. Toutefois, une autre tendance émerge actuellement : « *Plus de données n'est pas toujours mieux* »[208]. Les articles qui traitent du sujet évoquent notamment les problématiques du changement de métier entraînées par la confrontation aux données massives. L'analyse de données est un savoir-faire qui s'apprend et qui nécessite des compétences techniques particulières. Générer des données n'est souvent pas le travail le plus difficile. Le nouveau défi est d'être capable de les exploiter aussi utilement que possible. Il est important de se demander : comment les données seront utilisées ? Comment elles seront stockées ? Quelqu'un a-t-il déjà généré des données ressemblantes ? Comment faire pour que les autres équipes de recherche puissent à terme les exploiter également ?

En conclusion, le bioinformaticien a aujourd'hui un vrai travail d'équilibriste à effectuer. Il oscille constamment entre les besoins toujours plus gourmands en termes de quantité de données d'algorithmes complexes (voir ci-après le *machine learning*) et les problématiques liées au *Big data* en biologie que nous avons évoquées dans cette section.

---

[207] "*Trust, but verify*" Ronald Reagan.

[208] "*More Data isn't Always Better*". Cette phrase a été extraite des titres d'articles sur la thématique.

c. Plus d'automatisation des analyses, pour quels résultats ?

Exemple de l'annotation fonctionnelle automatique

Les données issues des projets NGS étant obtenues rapidement, beaucoup de séquences de gènes ou de protéines dans les bases de données publiques ne sont pas caractérisées expérimentalement, mais annotées à l'aide d'outils informatiques. Cela aboutit à une hétérogénéité de la qualité des annotations. En effet, l'annotation automatique est très rapide mais elle présente certaines limites. Lorsqu'un génome est séquencé ou lorsque la fonction d'un gène est recherchée, il est très fréquent d'utiliser l'annotation automatique par « transfert d'information ». Si deux gènes se ressemblent très fortement entre deux espèces proches, alors il est supposé qu'ils partagent la même fonction et l'annotation du gène « connu » est transférée au gène « inconnu ». Au cours de mon travail de thèse, j'ai souvent été confronté à des incohérences d'annotations fonctionnelles des gènes de *C. glabrata*. En effet, l'organisme le mieux annoté et le plus proche phylogénétiquement de *C. glabrata* est *S. cerevisiae*. Des transferts d'annotation ont donc été très largement réalisés entre ces deux levures. Parmi la liste des gènes réagissant au fer que nous avons mis en évidence chez *C. glabrata,* j'ai pu observer que le gène *HAP1* présentait un problème d'annotation (Denecker et al. 2020) : deux gènes possibles dans la *Candida Genome Database* alors qu'un seul est décrit dans la base de données GRYC. Une ressource numérique présente en détail comment ce problème a été mis en évidence et comment nous y avons remédié[209].

Problématique généralisable

Même s'il est difficile d'évaluer la quantité de mauvaises annotations, plusieurs équipes ont travaillé sur la question. À partir des 4 principales bases données protéiques (UniProtKB / Swiss-Prot, GenBank NR, UniProtKB / TrEMBL et KEGG), A. Schnoes *et al* ont étudié les fonctions moléculaires de 37 familles d'enzymes (Schnoes et al. 2009). Le bilan est très intéressant. Exceptée la base de données Swiss-Prot pour laquelle le pourcentage d'erreurs d'annotations est proche de 0%, les autres bases de données peuvent avoir des pourcentages d'erreurs supérieurs à 80% avec une erreur moyenne comprise entre 5 et 63%. Quelle est la différence entre Swiss-Prot et les autres bases de données ? Swiss-Prot est une des rares bases de données qui a réussi à maintenir une étape de contrôle manuel experte. Les données n'entrent

[209] https://thomasdenecker.github.io/thesisWebsite/annexes/annotAuto/ [Accessible le 10/08/2020]

dans la base que si elles ont été vérifiées. Cette étude n'est pas la seule, puisqu'en 2007, C. Andorf *et al* ont montré que sur les 211 protéines kinases de souris annotées, 201 annotations GO proposées par l'outil par AmiGO étaient incompatibles avec les fonctions Uniport attribuées à leurs homologues humains (Andorf et al. 2007). En conclusion, face au déluge des données, l'automatisation des analyses est indispensable mais il faut néanmoins rester prudent comme nous le montrent ces exemples.

d. Travailler sous un déluge constant de nouvelles données, comment faire ?

La solution la plus simple serait d'ignorer toutes ces données ! Malheureusement, c'est une posture qui n'est pas tenable dans le temps. Aussi, j'ai pu expérimenter l'intérêt de suivre certains principes :

- Être prudent, sceptique et méthodique – Nous avons déjà vu que suivre un plan détaillé est indispensable lors d'une analyse de données. C'est d'autant plus vrai que le volume de données impliqué dans l'analyse est grand. Ensuite, il faut être prudent vis-à-vis de la qualité des données collectées et être vigilant dans l'interprétation des résultats obtenus ;
- Créer et collecter des données en anticipant l'intégration – Générer des données n'est plus une barrière. Pour être efficace et limiter un maximum la perte de données, penser à leur intégration lors de leur création est devenu indispensable ;
- Les données sont des outils et pas une ligne de conduite – Nous ne demandons pas à nos œufs comment faire un gâteau ? Les données ne doivent pas réfléchir à notre place mais être une aide à la réflexion ;
- Utiliser les outils adaptés pour analyser cette grande quantité de données. Il serait incompréhensible d'utiliser une brosse à dents pour nettoyer une voiture. Il en va de même avec le *Big data*. Utiliser une fenêtre glissante pour détecter des pics est très performant sur des génomes de petites tailles (c'est la stratégie du programme bPeaks), mais une vraie usine à gaz sur les grands génomes. À l'inverse, utiliser les outils des grands génomes sur les petits génomes entraînera une perte de sensibilité ;
- Tirer une réelle expérience de nos erreurs – A. W. Brown *et al* proposent des solutions très intéressantes : ne pas remettre en question l'auteur mais les résultats (« L'erreur est humaine »), mettre en place une procédure pour corriger les erreurs, utiliser des approches pédagogiques pour résoudre les problèmes (mettre en place des systèmes d'apprentissage optimisés pour limiter l'ignorance : « *Que dois-je savoir pour travailler*

efficacement avec cette équipe de recherche ? »), mettre en place un système de révisions des données (et pourquoi pas avoir des chercheurs à plein temps sur ces problématiques), changer notre culture scientifique (un article contenant une erreur qui aurait été trouvée *a posteriori* ne devrait plus être étiqueté comme « rétracté » mais plutôt accompagné de corrections et d'explications) (A. W. Brown et al. 2018).

Ainsi, travailler sous un déluge de données est une réelle compétence, un savoir-faire. Ce n'est pas simple. Une grande difficulté pour les chercheurs est d'accepter que tout ne soit pas parfait, ni compris. Le chemin entre données, informations et connaissances est sinueux.

2. L'explosion de la « data science »

D'après la définition proposée dans le livre *Process Mining : data science in action* (Van der Aalst 2016), la data science est « *un domaine interdisciplinaire visant à transformer les données en valeur réelle. Les données peuvent être structurées ou non structurées, grandes ou petites, statiques ou continues. La valeur peut être fournie sous forme de prédictions, de décisions automatisées, de modèles, etc. appris à partir des données, ou tout type de visualisation de données fournissant des aperçus.* ». Cette définition est accompagnée d'une illustration regroupant les principales disciplines la composant (Figure 87).

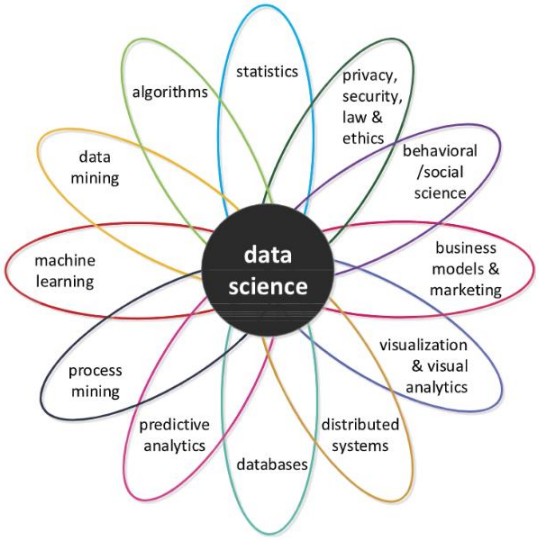


Figure 87 – Les différentes disciplines composant la data science. Il s'agit d'un domaine de recherche interdisciplinaire à l'intersection entre une douzaine de thématiques. Cette figure est extraite du livre de W. Van der Aalst (Van der Aalst 2016).

Au début de la rédaction de cette thèse, nous avons hésité avec G. Lelandais à utiliser la terminologie « science de données » plutôt que « analyse de données » dans le titre du manuscrit. Sans aucun doute, « science de données » (*data science*) aurait été plus à la mode ! En reprenant les différentes composantes de la *Data Science* (Figure 87), on peut d'ailleurs observer que la plupart ont servi de piliers à la réalisation des travaux de cette thèse (statistiques, algorithmes, *data mining*, organisation et stockage des données, visualisation et analyse visuelle, etc.). Toutefois deux composantes très importantes « *predictive analytics* » et « *machine learning* » n'ont pas été abordées. Celles-ci sont très fortement liées au domaine de l'intelligence artificielle.

3. L'intelligence artificielle a-t-elle une place en biologie ?

Aujourd'hui, on parle beaucoup d'intelligence artificielle. Entre peur et fascination, les points de vue divergent sur le sujet. Comment ce nouveau domaine se fait-il une place dans la société et dans la recherche en biologie ? Ce sont les questions que je me suis posées lors de la rédaction des paragraphes suivants.

a. Intelligence artificielle, *Machine Learning* et *Deep Learning*

Définitions

Les termes « intelligence artificielle », « *Machine Learning* » et « *Deep Learning* » sont souvent associés aux nouvelles technologies comme les voitures autonomes, les assistants virtuels (Alexa®, OK Google®, Siri®, Cortana®, etc.). La Figure 88 montre une inclusion classiquement faite de ces 3 termes. Le *Deep Learning* est un sous domaine du *Machine Learning* qui est un sous domaine de l'intelligence artificielle.

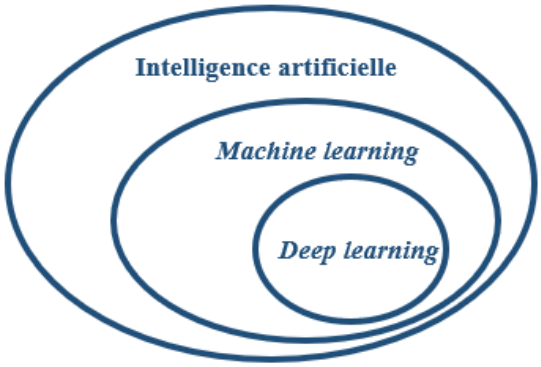


Figure 88 – Représentation schématique du lien entre les domaines de l'intelligence artificielle, le Marchine learning et le Deep learning.

L'objectif de l'intelligence artificielle est de créer des outils capables de reproduire le fonctionnement du cerveau humain. Pour cela, la machine a besoin d'une étape d'apprentissage. B. Thompson sépare l'intelligence artificielle en 2 types : l'intelligence artificielle générale, par exemple un ordinateur capable d'accomplir toute tâche humaine et l'intelligence artificielle étroite, par exemple un ordinateur expert dans la réalisation d'une tâche spécifique (jouer aux échecs)[210]. Dans ce contexte, A. Samuel définit le *Machine Learning* comme, « un domaine d'études qui donne aux ordinateurs la capacité d'apprendre sans être explicitement programmé » (Samuel 1959). Ainsi, les systèmes de *Machine Learning* apprennent comment combiner des entrées pour formuler des prédictions utiles. Cet apprentissage nécessite un grand nombre de données pour permettre une bonne prédiction. Enfin le *Deep Learning*, sous ensemble du *Machine Learning*, a la caractéristique de s'inspirer du modèle de traitement de l'information du cerveau humain. Schématiquement, le cerveau traite les observations qu'il reçoit en les étiquetant et en les triant dans différentes catégories. Lorsqu'une nouvelle observation arrive, le cerveau la compare avec les éléments qu'il connaît déjà puis lui donne une interprétation. Le *Deep learning* est capable de découvrir des nouvelles façons de classer les observations contrairement au *Machine Learning* où les méthodes de classement doivent être spécifiées manuellement. La contrepartie est un besoin accru de données et de performances informatiques. Les réseaux de neurones sont les exemples les plus classiques du *Deep Learning*. Le traducteur DeepL est basé sur du Deep Learning[211].

Débat de société

Comme le rappelle C. Villani lors de sa conférence TEDxSaclay[212], les mises en applications de l'IA conduisent à un débat de société. Il faut que le monde s'en imprègne afin d'éviter des situations de rejets. En effet, comme l'explique B. Falissard[213], l'IA est capable, en quelque sorte de « battre les humains ». Ainsi c'est une représentation plutôt effrayante de l'IA qui est présentée dans les médias. En octobre 2019, Forbes titrait « *Devrions-nous avoir peur de l'IA ?* »[214]. Faut-il pour autant rejeter l'Intelligence artificielle ? Je ne pense pas, puisque l'IA

---

[210] https://stratechery.com/2017/the-arrival-of-artificial-intelligence/ [Accessible le 16/04/2020]

[211] https://www.deepl.com/home [Accessible le 29/05/2020]

[212] https://www.youtube.com/watch?v=ten22DFl-Po [Accessible le 16/04/2020]

[213] https://www.youtube.com/watch?v=u5ILk10tJW0 [Accessible le 16/04/2020]

[214] "*Should We Be Afraid of AI?*" https://www.forbes.com/sites/cognitiveworld/2019/10/31/should-we-be-afraid-of-ai/ [Accessible le 16/04/2020]

est une aide incroyable pour faire avancer la connaissance notamment dans la Santé et en biologie comme nous allons le voir[215].

b. Applications en biologie

Dans le domaine de la santé

Il s'agit actuellement du domaine en biologie où l'intelligence artificielle est la plus utilisée. Fin 2019, B. Falissard nous rappelle que la technologie a toujours aidé la médecine. La question importante est donc comment l'IA peut-elle aider en médecine ? Pour éviter les débats vis-à-vis de la terminologie IA, il préfère parler d'Intelligence Augmentée. Ainsi, il explique que la première Intelligence Augmentée en médecine aura été le livre. Un livre est une mémoire augmentée, la mémoire étant une forme d'intelligence, le livre est donc une Intelligence Augmentée. Les mathématiques sont ensuite devenues une nouvelle forme d'Intelligence Augmentée pour leur capacité à conceptualiser les problématiques multidimensionnelles. Elles se basent également sur les statistiques pour se projeter dans la réalité. Comme le livre avant elles, les statistiques se heurtent à des problèmes de dimensions. Si les statistiques avaient permis de passer la barrière du 4D (impossible à représenter dans un livre) et bien qu'elles fonctionnent parfaitement dans une dizaine de dimensions, elles sont dépassées dans un ordre de grandeur de 100 dimensions. Pour ces types de calculs, une nouvelle forme d'Intelligence Augmentée est apportée par l'informatique. Les capacités actuelles des ordinateurs rendent possible la mise en application d'algorithmes très coûteux en termes de quantité de données et de puissance de calculs. La problématique actuelle est de savoir quel niveau de confiance il est raisonnable d'accorder aux résultats obtenus. En effet, il est beaucoup plus complexe d'appréhender un modèle construit à partir d'un million de variables que de 10. Concrètement aujourd'hui, où pouvons-nous trouver de l'IA dans la santé ? Nous pouvons la trouver dans des applications telles que :

- L'aide dans l'apprentissage et la formation des futurs médecins avec notamment la plateforme Epione[216] qui développe le e-patient pour une e-médecine ;

[215] Au cours de mes recherches pour écrire ce chapitre, j'ai relevé un ensemble de forces et de faiblesses associées à l'intelligence artificielle. Elles sont présentées dans la ressource numérique : https://thomasdenecker.github.io/thesisWebsite/annexes/IA/ [Accessible le 10/08/2020]

[216] https://team.inria.fr/epione/fr/ [Accessible le 16/04/2020]

- L'optimisation des prothèses capables de s'adapter au type de terrain[217] ;
- Les algorithmes de Cardiologs capables de faire un diagnostic automatique à partir d'un électrocardiogramme[218] ;
- Les algorithmes de reconnaissance automatique de cancer du sein à partir d'une mammographie de la *start up* Therapixel[219] ;
- Un algorithme capable de réaliser une reconstruction faciale à partir de la séquence ADN ou autrement dit capable de prédire un phénotype à partir du génotype[220] ;
- Des algorithmes de AHEAD[221] qui proposent une solution de diagnostic et d'aide à la décision clinique des cancers du sang ;
- Les algorithmes de reconnaissance d'images utilisés en radiographie pour le diagnostic du cancer du poumon par exemple (Hosny et al. 2018).

Enfin plus récemment, Y. Bengio spécialiste de l'IA et récompensé du Prix ACM A.M. Turing proposait une application pour optimiser la distanciation sociale lors de la pandémie à la COVID-19.[222]

Dans la recherche fondamentale

Si vous recherchez « *Deep learning* » sur Pubmed, le nombre de publications augmente ces dernières années (Figure 89). La majorité des publications concernent le domaine biomédical : « *Deep Learning in biomedicine.* », « *A guide to deep learning in healthcare.* », « *Deep learning in Neuroradiology* », « *Deep learning in Radiology* », « *Deep Learning in Pharmacogenomics: From Gene Regulation to Patient Stratification* » sont les 5 premiers résultats que j'ai obtenu lors de ma recherche. Ce résultat est cohérent puisque la santé se prête bien au *Deep learning* ou plus largement à l'Intelligence artificielle comme nous l'avons vu précédemment. Mais qu'en est-il en recherche fondamentale ? En 2005, Anne Carpenter, une bioinformaticienne au Broad Institute of MIT et Harvard à Cambridge, a publié un logiciel

---

[217] https://biomech.media.mit.edu/ [Accessible le 16/04/2020]

[218] https://cardiologs.com/ [Accessible le 16/04/2020]

[219] https://www.therapixel.com/ [Accessible le 16/04/2020]

[220] Riccardo Sabatini nous présente son algorithme lors de la conférence TED *How to read the genome and build a human being* https://www.youtube.com/watch?v=s6rJLXq1Re0 [Accessible le 16/04/2020]

[221] https://aheadmedicine.com/ [Accessible le 17/04/2020]

[222] https://yoshuabengio.org/fr/2020/03/25/depistage-pair-a-pair-de-la-covid-19-base-sur-lia/ [Accessible le 17/04/2020]

Open Source appelé CellProfiler[223] afin d'aider les biologistes à mesurer quantitativement les caractéristiques individuelles. Combien de colonies de levures sont présentes dans ce milieu de culture présenté à la Figure 90 ? Grâce à CellProfiler, il est possible de grouper et compter les colonies en fonction de la couleur, de l'aire, etc. (Bray et al. 2015).

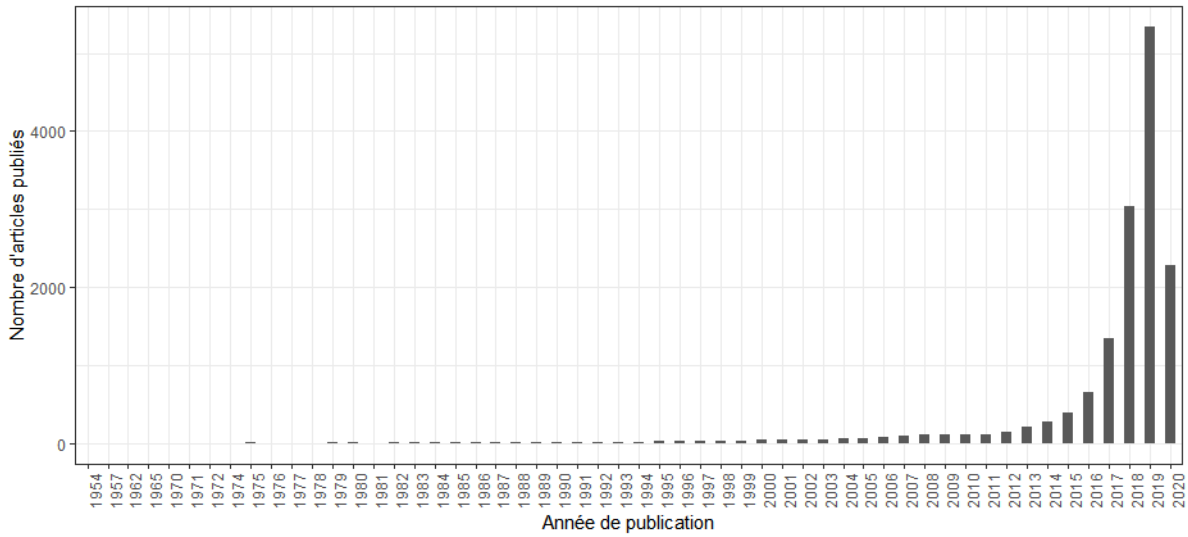


Figure 89 – Evolution du nombre de publications parues sur Pubmed concernant le Deep learning. Les données ont été obtenues grâce à l'outil timeline de Pubmed lors de la recherche « Deep learning » réalisée le 17/04/2020.

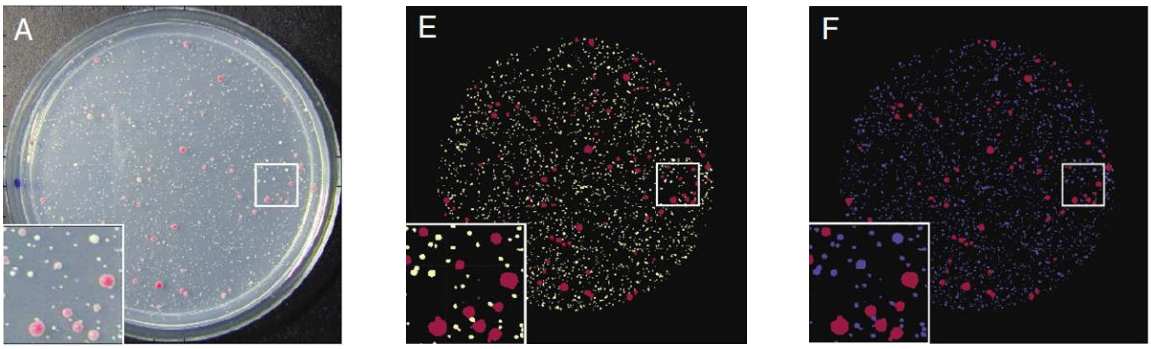


Figure 90 – Exemple de classifications de CellProfiler (Bray et al. 2015). *À partir d'une photographie initiale du milieu de culture (A), CellProfiler est capable de classer en fonction de l'aire (E) et la couleur de la colonie (F).*

De nombreuses découvertes ont pu être réalisées grâce au *Deep learning* en génomique. Une société de Google propose par exemple un logiciel appelé DeepVariant qui permet de détecter

[223] https://cellprofiler.org/ [Accessible le 17/04/2020]

des variants dans des séquences d'ADN[224]. La société Deep Genomics propose, à partir de données génomiques et transcriptomiques de cellules saines, des modèles prédictifs d'événements de traitement d'ARN tels que l'épissage, la transcription et la polyadénylation dans ces données[225]. Atomwise, une société de biotechnologie basée à San Francisco, a développé des algorithmes capables de convertir des molécules et des protéines en 3D et prédire quelles molécules sont susceptibles d'interagir avec une protéine testée[226]. Dans un article publié dans Nature Genetics, J. Zhou nous présente différentes applications du Deep Learning en génomique comme la prédiction de la structure d'une protéine à partir de sa séquence d'ADN ou d'ARN, l'identification des *long noncoding RNA*, la prédiction de la pathogénicité d'un micro-organisme à partir de son génome, etc. (Zhou et al. 2018). Il est aussi possible de retrouver de l'intelligence artificielle dans d'autres « omiques » comme en protéomique. Ces approches permettent par exemple d'améliorer l'identification des protéines à l'aide des spectres ou comme nous le présentent S. Iravani et T. Conrad, pour mieux classer certaines caractéristiques cellulaires à partir de données protéomiques (Holzinger et al. 2018).

c. Utopie réaliste ?

Même si des applications intéressantes existent, l'Intelligence Artificielle fait face à de nombreux défis pour être utilisée. Parmi les principales problématiques, nous pouvons citer :

- Un besoin d'une grande quantité de données d'une grande qualité – Comme nous le rappelle S. Webb, le choix du jeu de données est d'une importance capitale (Webb 2018). Les prédictions auront la qualité de la qualité des données utilisées pour y aboutir. Les algorithmes d'intelligence artificielle et particulièrement de *Deep Learning* nécessitent des jeux de données très grands et bien annotés. Comme la majorité des algorithmes utilisés actuellement reposent sur de l'apprentissage supervisé, il est particulièrement important d'avoir une annotation de grande qualité.
- Une difficulté lors de la prédiction des petits effets – M. Dimon, chercheuse à Google Accelerated Science, indique qu'il est très difficile de prédire des petites variations. Il

[224] Le projet Github est disponible ici : https://github.com/google/deepvariant et la documentation pour l'utiliser est disponible ici : https://cloud.google.com/life-sciences/docs/tutorials/deepvariant?hl=fr [Accessible le 17/04/2020]

[225] https://www.deepgenomics.com/ [Accessible le 17/04/2020]

[226] https://www.atomwise.com/ [Accessible le 17/04/2020]

est donc important de les intégrer lors de la création des données. Pour éviter la déception de ne pas les détecter, il est important « pour les biologistes et les informaticiens de travailler ensemble pour concevoir des expériences qui intègrent le Deep Learning »[227].

- Une boîte noire – Comme nous avons pu l'évoquer précédemment par l'utilisation d'un très grand nombre de variables dans ces nouveaux algorithmes, le sentiment de boîte noire est très fort actuellement. Ne pas comprendre exactement comment les prédictions sont données est une des problématiques fortes.
- Nous faisons face à nos propres limites – En 2019, pour A. Tugui, l'IA se heurte à deux limites : une limite technique et une limite biologique (Tugui et al. 2019). En reprenant la limite technique proposée par H. Dreyfus en 1976, nous n'avons pas encore les capacités computationnelles pour créer une IA capable d'imiter le cerveau humain. Pour franchir cette limite, un ordinateur devra être capable de traiter $10^{10^{10}}$ états. Concernant la limite biologique, comment reproduire ce que nous ne comprenons/connaissons pas totalement ? Aujourd'hui, le cerveau humain est encore un grand mystère dont nous explorons encore le potentiel.

Au-delà de ces problématiques, nous pourrions nous demander si nous pouvons tous faire de l'intelligence artificielle ? Techniquement, oui. En 10 minutes en R, vous mettrez en place un algorithme de *Deep Learning* capable de faire de la reconnaissance d'image et prédire le chiffre écrit sur une photo[228]. Je pense que la vraie question est : tous les sujets sont-ils adaptés à l'Intelligence Artificielle ? Ce n'est pas faute d'avoir essayé d'utiliser l'IA dans l'étude de l'homéostasie du fer chez *C. glabrata* (j'en rêvais !), mais les questions que nous nous sommes posées ne s'y prêtaient pas. Nous n'avions pas forcément les données et les connaissances (biologiques, techniques, etc.) pour formaliser des problématiques traitables par IA. En revanche, je pense que les données utilisées lors de l'étude des modifications post-traductionnelles chez *C. albicans* pourraient être exploitées par l'IA. Nous pourrions par exemple essayer de prédire la position de modifications post-traductionnelles. Une perspective passionnante pour ce projet !

---

[227] « *This hazard underscores the importance of biologists and computer scientists working together to design experiments that incorporate deep learning* »

[228] https://r2018-rennes.sciencesconf.org/data/pages/Deep_with_R_1.pdf [Accessible le 20/04/2020]

Annexes

I. Les aperçus des différents formats évoqués dans cette thèse

1. BAM

BAM est l'acronyme de Binary Alignment/Map. Il s'agit d'un fichier au format SAM compressé au format de compression BGZF. L'avantage de ce format de compression est qu'il permet un accès efficace dans les fichiers indexés et qu'il est beaucoup plus léger qu'un fichier SAM. La contrepartie est que ce type de fichier n'est pas *human readable*. Ci-dessous, un extrait d'un fichier BAM :

```
‹      ÿ BC ;m"¿oÔ0Ç]…eât¨{ÿxŽã›-¤r-=`Aå©i ô'#9©^… %• þV&`aëÆÀŠÄÊÒ…›8åìc^}>Nœï{ÏW¯ÜZØ:•Ðê•uï^¿GBì
```

Extension des fichiers : .bam

Pour en savoir plus : https://samtools.github.io/hts-specs/SAMv1.pdf

2. BED

BED est l'acronyme de Browser Extensible Format. Il s'agit d'un fichier d'annotation composé de plusieurs colonnes. Les colonnes sont séparées par des tabulations. Chaque ligne est une annotation. Il est composé au minimum de 3 colonnes : le nom du chromosome, la position de début et la position de fin. 9 autres colonnes peuvent être ajoutées comme un nom, un score, une orientation du brin d'ADN (+ ou -), … Ci-dessous un exemple fictif de fichier BED dans sa composition minimale

```
chr2L  2740000    2860000
chr2L  2930000    0.0215
chr2L  3400000    3750000
chr2L  3830000    3950000
chr2L  4020000    5550000
```

Extension des fichiers : .bed

Pour en savoir plus : https://genome.ucsc.edu/FAQ/FAQformat.html

3. FASTA

Il s'agit d'un format de fichier contenant des séquences (nucléique ou peptidique). Une séquence au format FASTA commence par une description sur une seule ligne, suivie de lignes de données de séquences. La ligne de description commence toujours par un chevron : > . La séquence est au format texte avec un nombre de caractères limité par ligne. Le NCBI recommande moins de 80 caractères par ligne. Vous trouverez ci-dessous un exemple d'une séquence protéique au format FASTA (protéine Ftr1 codé par le gène *FTR1* de la levure pathogène *C. glabrata* CBS138) :

```
>CAGL0I06743g|FTR1 COORDS:ChrI C_glabrata_CBS138
MPNKVFNVAVFFVVFRECLEAVVIVSILLSFLKQAIGSKDIKLYRKLRKHVWIGVALGFF
ICLVIGAGFIGAYYSLQKDIFGSTEDLWEGIFCMIATTMISMMGIPMLRINKLQSKWRVK
LARSLVDIPKRKRDYFRIGYLTRRYAMFILPFITVLREGLEAVVFVAGAGITTKGSHASA
YPLPVVVGLIAGFIVGFLLYYGTSKSSMQIFLVISTSILYLIAAGLFSRGAWYFENYRFN
KATGGDASEGGDGNGSYNIAKSVYHVNCCNPELDNGWDIFNALLGWQNTGYLSSILCYNI
YWVVLIIVLGLMMHEERYGHLPFMRNVGMRHLNPGYWIKNKKKDELTDEQKAELLNRMDN
IQFNEEGDIVAHANEEEHDQESSLLRGNSNKMGSKEELNFKVTTTSSD*
```

Extension des fichiers : .fasta , .fa

Pour en savoir plus : http://genetics.bwh.harvard.edu/pph/FASTA.html

4. FASTQ

Le format FASTQ permet le stockage au format texte d'une séquence biologique (généralement nucléotidique) et les scores de qualité liés à cette séquence. Ce fichier est un fichier brut provenant des séquenceurs. Il se compose de 4 lignes au minimum :

1. La description qui débute par un @
2. La séquence
3. Un signe + qui sépare la séquence du score qualité
4. Le score qualité

Ci-dessous un exemple de fichier fictif au format FASTQ pour le gène FTR1 de la levure *C. glabrata* CBS138 :

```
@FTR1 Candida glabrata CBS138 ChrI
ATGCCTAACAAAGTATTTAACGTGGCAGTCTTCTTTGTCGTGTTCAGAGAATGTTTGGAAGCTGTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

Extension des fichiers : .fastq

Pour en savoir plus : https://emea.support.illumina.com/bulletins/2016/04/fastq-files-explained.html

5. GFF

GFF est l'acronyme de General Feature Format. Il s'agit d'un fichier texte composé de 9 colonnes, indiquant les positions génomiques de début et de fin d'éléments génomiques d'intérêts. Les 9 colonnes sont :

1. Seqname – Nom du chromosome ou équivalent. Les noms des chromosomes peuvent être donnés avec ou sans le préfixe "chr" ;
2. Source – Nom du programme qui a généré cet élément, ou de la source de données ;
3. Feature – Type de l'élément ;
4. Start – Position de départ de l'élément, avec numérotation séquentielle commençant à 1 ;
5. End – Position de fin de l'élément, avec numérotation séquentielle commençant à 1 ;
6. Score – Valeur de score (valeur numérique en flottant) ;
7. Strand – Sens de l'élément. Le symbole "+" est utilisé lorsque l'élément se trouve sur le même brin que la séquence de support et "-" sur le brin inverse ;
8. Frame - Une des valeurs "0", "1" ou "2" est ici ajoutée. Le "0" indique que la première base de l'élément est la première base d'un codon, le "1" que la deuxième base est la première base d'un codon, et ainsi de suite ;
9. Attribute – Liste de toutes les informations supplémentaires disponibles sur l'élément. Cette liste est séparée par des points-virgules.

Ci-dessous un exemple de fichier GFF disponible sur la CGD[229] pour la levure *C. glabrata*

```
##gff-version 3
# File name:
# Organism: Candida glabrata CBS138
# Genome version: s03-m01-r02
# Date created: Sun May  3 07:14:42 2020
# Created by: The Candida Genome Database (http://www.candidagenome.org/)
# Contact Email: candida-curator AT lists DOT stanford DOT edu
# Funding: NIDCR at US NIH, grant number 1-R01-DE015873-01
#
ChrA_C_glabrata_CBS138  CGD     chromosome      1       491328  .       .       .       ID=ChrA_C_glabrata_CBS138;Name=ChrA_C_glabrata_CBS138
ChrB_C_glabrata_CBS138  CGD     chromosome      1       502101  .       .       .       ID=ChrB_C_glabrata_CBS138;Name=ChrB_C_glabrata_CBS138
ChrC_C_glabrata_CBS138  CGD     chromosome      1       558804  .       .       .       ID=ChrC_C_glabrata_CBS138;Name=ChrC_C_glabrata_CBS138
ChrD_C_glabrata_CBS138  CGD     chromosome      1       651701  .       .       .       ID=ChrD_C_glabrata_CBS138;Name=ChrD_C_glabrata_CBS138
ChrE_C_glabrata_CBS138  CGD     chromosome      1       687738  .       .       .       ID=ChrE_C_glabrata_CBS138;Name=ChrE_C_glabrata_CBS138
ChrF_C_glabrata_CBS138  CGD     chromosome      1       927101  .       .       .       ID=ChrF_C_glabrata_CBS138;Name=ChrF_C_glabrata_CBS138
ChrG_C_glabrata_CBS138  CGD     chromosome      1       992211  .       .       .       ID=ChrG_C_glabrata_CBS138;Name=ChrG_C_glabrata_CBS138
ChrH_C_glabrata_CBS138  CGD     chromosome      1       1050361 .       .       .       ID=ChrH_C_glabrata_CBS138;Name=ChrH_C_glabrata_CBS138
ChrI_C_glabrata_CBS138  CGD     chromosome      1       1100349 .       .       .       ID=ChrI_C_glabrata_CBS138;Name=ChrI_C_glabrata_CBS138
ChrJ_C_glabrata_CBS138  CGD     chromosome      1       1195129 .       .       .       ID=ChrJ_C_glabrata_CBS138;Name=ChrJ_C_glabrata_CBS138
ChrK_C_glabrata_CBS138  CGD     chromosome      1       1302831 .       .       .       ID=ChrK_C_glabrata_CBS138;Name=ChrK_C_glabrata_CBS138
ChrL_C_glabrata_CBS138  CGD     chromosome      1       1455689 .       .       .       ID=ChrL_C_glabrata_CBS138;Name=ChrL_C_glabrata_CBS138
ChrM_C_glabrata_CBS138  CGD     chromosome      1       1402898 .       .       .       ID=ChrM_C_glabrata_CBS138;Name=ChrM_C_glabrata_CBS138
mito_C_glabrata_CBS138  CGD     chromosome      1       20063   .       .       .       ID=mito_C_glabrata_CBS138;Name=mito_C_glabrata_CBS138
```

Extension des fichiers : .gff

Pour en savoir plus : https://www.ensembl.org/info/website/upload/gff.html

6. HTML

HTML est l'acronyme de *HyperText Markup Language*. Il s'agit d'un format de balises pour créer et représenter le contenu d'une page web et sa structure. Ci-dessous un extrait d'un des rapports html généré par MONET :

```
<!DOCTYPE html>
<html>
      <head>
            <meta charset="utf-8" />
            <meta name="generator" content="pandoc" />
            <meta http-equiv="X-UA-Compatible" content="IE=EDGE" />
            <meta name="author" content="MONET" />
            <meta name="date" content="2020-05-04" />
            <title>Gene report</title>
      <head>
      <body>
            <!-- Le contenu du rapport -->
      </body>
</html>
```

[229] http://www.candidagenome.org/download/gff/C_glabrata_CBS138/C_glabrata_CBS138_current_features.gff [Accessible le 05/05/2020]

Extension des fichiers : .html

Pour en savoir plus : https://developer.mozilla.org/fr/docs/Web/HTML

7. idXML

Il s'agit d'un format XML pour les fichiers contenant des informations sur les identifications de peptides suite à une analyse des données de spectrométrie de masse.

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="https://www.openms.de/xml-stylesheet/IdXML.xsl" ?>
<IdXML version="1.5" xsi:noNamespaceSchemaLocation="https://www.openms.de/xml-schema/IdXML_1_5.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SearchParameters id="SP_0" db="/abi-data/sachsenb/OpenMS_IDE/src/tests/topp/THIRDPARTY/OPSSAAdapter_1.fasta" db_version="" taxonomy="" mass_type="monoisotopic" charges="" enzyme="trypsin" missed_cleavages="1"
      precursor_peak_tolerance="5" precursor_peak_tolerance_ppm="true" peak_mass_tolerance="0.3" peak_mass_tolerance_ppm="false" >
    <VariableModification name="dHex(1)Hex(2)HexNAc(2)NeuAc(1)Sulf(1) (M)" />
  </SearchParameters>
  <IdentificationRun date="2014-05-27T14:37:20" search_engine="XTandem" search_engine_version="" search_parameters_ref="SP_0" >
    <ProteinIdentification score_type="XTandem" higher_score_better="false" significance_threshold="0" >
      <ProteinHit id="PH_0" accession="BSA3 ##3" score="-15" sequence="" >
      </ProteinHit>
      <ProteinHit id="PH_1" accession="BSA2 ##2" score="-14.8999961185303" sequence="" >
      </ProteinHit>
      <ProteinHit id="PH_2" accession="test2_rev ##1" score="-10.3000001907349" sequence="" >
      </ProteinHit>
      <UserParam type="stringList" name="spectra_data" value="[file:///C:/testing/test_file_path/test_file_name.raw]"/>
    </ProteinIdentification>
    <PeptideIdentification score_type="XTandem" higher_score_better="true" significance_threshold="0" MZ="520.262817382812" RT="2655.095703125" spectrum_reference="spectrum=0" >
      <PeptideHit score="56.5" sequence="DFASSGEVVLHLHR" charge="3" aa_before="[" aa_after="E" start="0"  end="13" protein_refs="PH_2" >
        <UserParam type="float" name="nextscore" value="0"/>
        <UserParam type="float" name="delta" value="-0.0022"/>
        <UserParam type="float" name="mass" value="1558.7761"/>
        <UserParam type="float" name="E-Value" value="5.6e-11"/>
        <UserParam type="float" name="hyperscore" value="56.5"/>
        <UserParam type="float" name="b_score" value="7.7"/>
        <UserParam type="int" name="b_ions" value="2"/>
        <UserParam type="float" name="y_score" value="10.3"/>
        <UserParam type="int" name="y_ions" value="10"/>
      </PeptideHit>
    </PeptideIdentification>
    <PeptideIdentification score_type="XTandem" higher_score_better="true" significance_threshold="0" MZ="1063.20083886719" RT="4587.6689453125" spectrum_reference="spectrum=1" >
      <PeptideHit score="77.9" sequence="IALSRPHVEVVALNDPFITNDYAAYM(dHex(1)Hex(2)HexNAc(2)NeuAc(1)Sulf(1))FK" charge="3" aa_before="[" aa_after="E" start="0"  end="27" protein_refs="PH_1" >
        <UserParam type="float" name="nextscore" value="0"/>
        <UserParam type="float" name="delta" value="0.0017"/>
        <UserParam type="float" name="mass" value="3187.6132"/>
        <UserParam type="float" name="E-Value" value="1.4e-15"/>
        <UserParam type="float" name="hyperscore" value="77.9"/>
        <UserParam type="float" name="b_score" value="14.4"/>
        <UserParam type="int" name="b_ions" value="16"/>
        <UserParam type="float" name="y_score" value="11.1"/>
        <UserParam type="int" name="y_ions" value="18"/>
      </PeptideHit>
    </PeptideIdentification>
    <PeptideIdentification score_type="XTandem" higher_score_better="true" significance_threshold="0" MZ="775.387207031125" RT="4923.77734375" spectrum_reference="spectrum=2" >
      <PeptideHit score="86.6" sequence="RPGADSDIGGFGGLFDLAQAGFR" charge="3" aa_before="[" aa_after="A" start="0"  end="22" protein_refs="PH_0" >
        <UserParam type="float" name="nextscore" value="0"/>
        <UserParam type="float" name="delta" value="0.0053"/>
        <UserParam type="float" name="mass" value="2324.1418"/>
        <UserParam type="float" name="E-Value" value="1e-15"/>
        <UserParam type="float" name="hyperscore" value="86.6"/>
        <UserParam type="float" name="b_score" value="11.8"/>
        <UserParam type="int" name="b_ions" value="16"/>
        <UserParam type="float" name="y_score" value="12.1"/>
        <UserParam type="int" name="y_ions" value="13"/>
      </PeptideHit>
    </PeptideIdentification>
  </IdentificationRun>
</IdXML>
```


Extension des fichiers : .idXML

Pour en savoir plus :
http://bioportal.bioontology.org/ontologies/EDAM?p=classes&conceptid=format_3764

8. JSON

JSON est l'acronyme de JavaScript Object Notation (ou en français Notation Objet issue de JavaScript). Il s'agit d'un format léger d'échange de données sous forme structurée. L'Homme est capable de le lire et l'écrire facilement. Il est basé sur un sous-ensemble du langage de programmation JavaScript (JavaScript Programming Language, Standard ECMA-262 3rd Edition - December 1999). JSON est un format texte complètement indépendant de tout langage. JSON se base sur deux structures : (1) une collection de couples nom/valeur et (2) une liste de valeurs ordonnées. Ci-dessous, un exemple de fichier JSON provenant de l'API STRING pour la protéine Ftr1 de la levure *C. glabrata* (identique au fichier au format XML) :

```
[
  {
    "stringId":"5478.XP_447543.1",
    "preferredName":"FTR1",
    "taxonName":"Candida glabrata",
    "queryIndex":0,
    "annotation":"Uncharacterized protein; Highly similar to uniprot|P40088
    Saccharomyces cerevisiae YER145c FTR1",
    "ncbiTaxonId":5478
  }
]
```


Extension des fichiers : .json

Pour en savoir plus : https://www.json.org/json-fr.html

9. mzId (ou mzIdentML)

mzIdentML est le format standard d'identification des peptides en protéines à partir de spectres de masse. Ce format a été normalisé par HUPO PSI (Proteomic Standards Initiative). Ci-dessous un extrait d'un fichier mzId :


```
<?xml version="1.0" encoding="UTF-8"?>
<MzIdentML xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://psidev.info/psi/pi/mzIdentML/1.1 https://raw.githubusercontent.com/HUPO-PSI/mzIdentML/master/schema/mzIdentML1.1.0.xsd"
    xmlns="http://psidev.info/psi/pi/mzIdentML/1.1"
    version="1.1.0"
    id="OpenMS_1672870793250310304"
    creationDate="2019-10-28T11:41:37">
<cvList>
    <cv id="PSI-MS" fullName="Proteomics Standards Initiative Mass Spectrometry Vocabularies" uri="https://raw.githubusercontent.com/HUPO-PSI/psi-ms-CV/master/psi-ms.obo" version="3.15.0"></cv>
    <cv id="UNIMOD" fullName="UNIMOD" uri="http://www.unimod.org/obo/unimod.obo"></cv>
    <cv id="UO"     fullName="UNIT-ONTOLOGY" uri="https://raw.githubusercontent.com/bio-ontology-research-group/unit-ontology/master/unit.obo"></cv>
</cvList>
<AnalysisSoftwareList>
    <AnalysisSoftware version="" name="" id="SOF_11577580331039746761">
        <SoftwareName>
            <cvParam accession="" cvRef="PSI-MS" name=""/>
        </SoftwareName>
    </AnalysisSoftware>
    <AnalysisSoftware version="OpenMS TOPP v2.4.0-HEAD-2018-10-30" name="TOPP software" id="SOF_12750796922377228600">
        <SoftwareName>
            <cvParam accession="MS:1000752" cvRef="PSI-MS" name="TOPP software"/>
        </SoftwareName>
    </AnalysisSoftware>
</AnalysisSoftwareList>
<SequenceCollection>
    <DBSequence accession="A0A1D8PC38" searchDatabase_ref="SDB_17706004264719952703" id="PROT_18007565453051512991">
        <cvParam accession="MS:1001088" cvRef="PSI-MS" name="protein description" value="A0A1D8PC38"/>
    </DBSequence>
    <DBSequence accession="A0A1D8PC43" searchDatabase_ref="SDB_17706004264719952703" id="PROT_14036041027934367963">
        <cvParam accession="MS:1001088" cvRef="PSI-MS" name="protein description" value="A0A1D8PC43"/>
    </DBSequence>
    <DBSequence accession="A0A1D8PC56" searchDatabase_ref="SDB_17706004264719952703" id="PROT_15610898839100199918">
        <cvParam accession="MS:1001088" cvRef="PSI-MS" name="protein description" value="A0A1D8PC56"/>
    </DBSequence>
    <DBSequence accession="A0A1D8PC68" searchDatabase_ref="SDB_17706004264719952703" id="PROT_17223351414259817397">
        <cvParam accession="MS:1001088" cvRef="PSI-MS" name="protein description" value="A0A1D8PC68"/>
    </DBSequence>
    <DBSequence accession="A0A1D8PC71" searchDatabase_ref="SDB_17706004264719952703" id="PROT_14300932990547802291">
        <cvParam accession="MS:1001088" cvRef="PSI-MS" name="protein description" value="A0A1D8PC71"/>
    </DBSequence>
    <DBSequence accession="A0A1D8PC73" searchDatabase_ref="SDB_17706004264719952703" id="PROT_10668060119448348574">
        <cvParam accession="MS:1001088" cvRef="PSI-MS" name="protein description" value="A0A1D8PC73"/>
    </DBSequence>
    <DBSequence accession="A0A1D8PC75" searchDatabase_ref="SDB_17706004264719952703" id="PROT_76273657745011238836">
        <cvParam accession="MS:1001088" cvRef="PSI-MS" name="protein description" value="A0A1D8PC75"/>
    </DBSequence>
```


Extension des fichiers : .mzid

Pour en savoir plus : http://psidev.info/mzidentml

10.mzML

Le mzML est le format des données brutes issues des spectromètres de masse. Le format a été standardisé par HUPO PSI (Proteomic Standards Initiative). Il remplace le format mzData et le format mzXML. Ci-dessous, un extrait d'un fichier mzML :

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<mzML xmlns="http://psi.hupo.org/ms/mzml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://psi.hupo.org/ms/mzml http://psidev.info/files/ms/mzML/xsd/mzML1.1.0.xsd" accession="" version="1.1.0">
    <cvList count="2">
        <cv id="MS" fullName="Proteomics Standards Initiative Mass Spectrometry Ontology" URI="http://psidev.cvs.sourceforge.net/*checkout*/psidev/psi/psi-ms/mzML/controlledVocabulary/psi-ms.obo"/>
        <cv id="UO" fullName="Unit Ontology" URI="http://obo.cvs.sourceforge.net/obo/obo/ontology/phenotype/unit.obo"/>
    </cvList>
    <fileDescription>
        <fileContent>
            <cvParam cvRef="MS" accession="MS:1000294" name="mass spectrum" />
        </fileContent>
    </fileDescription>
    <sampleList count="1">
        <sample id="sa_0" name="">
            <cvParam cvRef="MS" accession="MS:1000004" name="sample mass" value="0"  unitAccession="UO:0000021" unitName="gram" unitCvRef="UO" />
            <cvParam cvRef="MS" accession="MS:1000005" name="sample volume" value="0" unitAccession="UO:0000098" unitName="milliliter" unitCvRef="UO" />
            <cvParam cvRef="MS" accession="MS:1000006" name="sample concentration" value="0" unitAccession="UO:0000175" unitName="gram per liter" unitCvRef="UO" />
        </sample>
    </sampleList>
    <softwareList count="3">
        <software id="so_in_0" version="" >
            <cvParam cvRef="MS" accession="MS:1000799" name="custom unreleased software tool" value="" />
        </software>
        <software id="so_default" version="" >
            <cvParam cvRef="MS" accession="MS:1000799" name="custom unreleased software tool" value="" />
        </software>
        <software id="so_dp_sp_0_pm_0" version="1.8.0" >
            <cvParam cvRef="MS" accession="MS:1000756" name="FileConverter" />
        </software>
    </softwareList>
    <instrumentConfigurationList count="1">
        <instrumentConfiguration id="ic_0">
            <cvParam cvRef="MS" accession="MS:1000031" name="instrument model" />
            <softwareRef ref="so_in_0" />
        </instrumentConfiguration>
    </instrumentConfigurationList>
    <dataProcessingList count="1">
        <dataProcessing id="dp_sp_0">
            <processingMethod order="0" softwareRef="so_dp_sp_0_pm_0">
                <cvParam cvRef="MS" accession="MS:1000544" name="Conversion to mzML" />
                <cvParam cvRef="MS" accession="MS:1000747" name="completion time" value="2011-02-17+10:46" />
                <userParam name="parameter: in" type="xsd:string" value="neg_mz.mgf"/>
                <userParam name="parameter: out" type="xsd:string" value="neg_mz.mzML"/>
                <userParam name="parameter: in_type" type="xsd:string" value=""/>
                <userParam name="parameter: out_type" type="xsd:string" value=""/>
                <userParam name="parameter: log" type="xsd:string" value=""/>
                <userParam name="parameter: debug" type="xsd:integer" value="0"/>
                <userParam name="parameter: threads" type="xsd:integer" value="1"/>
                <userParam name="parameter: no_progress" type="xsd:string" value="false"/>
                <userParam name="parameter: test" type="xsd:string" value="false"/>
            </processingMethod>
        </dataProcessing>
    </dataProcessingList>
```


Extension des fichiers : .mzml

Pour en savoir plus : http://psidev.info/mzML

11. pepXML

Il s'agit d'un format permettant de stocker les identifications des peptides en protéine. Il est utilisable par de nombreux outils disponibles en ligne. Ci-dessous, un extrait d'un fichier

<?xml version="1.0" encoding="utf-8"?>
<msms_pipeline_analysis xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:schemaLocation="http://regis-web.systemsbiology.net/pepXML /home/dfermin/apps/tpp/schema/pepXML_v117.xsd"
  <analysis_summary analysis="Proteome Discoverer" time="2018-04-26T09:56:46.0000000+02:00" version=" 2.2.0.388">
    <summary xmlns="http://portal.thermo-brims.com">
      <search_summary>
Result name: 1613001-F1_160331204806
Description: Processing workflow for precursor area quantification. Ion trap-detected HCD spectra using SequestHTwith Percolator validation. Specify the FASTA database and any additional modifications.
Workflow based on template: PWF_Fusion_Homo_Sapiens_IT HCD_Mascot_Percolator_ptmRS
Creation date: 26/04/2018 09:56:46

The pipeline tree:

  |-(5) Spectrum Files
    |-(1) Spectrum Selector
      |-(7) Mascot
        |-(3) Percolator
          |-(6) ptmRS

Processing node 5: Spectrum Files

Input Data:
- File Name(s): G:\1-SLIM_Labeling\1613001_Gamme_12C-100-75-50-25-0-NC_Calb\1613001_mzML\1613001-F1_160331204806.mzML

Processing node 1: Spectrum Selector
----
1. General Settings:
- Precursor Selection: Use MS1 Precursor
- Use New Precursor Reevaluation: True
- Use Isotope Pattern in Precursor Reevaluation: True

2. Spectrum Properties Filter:
- Lower RT Limit: 0
- Upper RT Limit: 0
- First Scan: 0
- Last Scan: 0
- Lowest Charge State: 0
- Highest Charge State: 0
- Min. Precursor Mass: 350 Da
- Max. Precursor Mass: 5000 Da
- Total Intensity Threshold: 0
- Minimum Peak Count: 1

3. Scan Event Filters:
- MS Order: Is Not MS1
- Min. Collision Energy: 0
- Max. Collision Energy: 1000
- Scan Type: Is Full

Extension des fichiers : .pepxml

Pour en savoir plus :
http://sashimi.sourceforge.net/schema_revision/pepXML/pepXML_v118.xsd (schéma)

12. PDF

Sans doute le plus connu de cette liste. Le PDF est l'accronyme de Portable Document Format. Il a été créé en 1992 par la société Adobe Systems et il est devenu une norme ISO en 2008. Sa particularité est de préserver la mise en page du document à l'identique par rapport au document sauvegardé initialement (police de caractère, taille et forme des images, des figures, la pagination, …). Il peut être lu à l'aide de nombreux logiciels sur tous les périphériques (ordinateurs, smartphones, …) sans altération. Il est enfin très souvent conseillé de convertir un document en PDF avant l'impression.

Extension des fichiers : .pdf

Pour en savoir plus : https://acrobat.adobe.com/fr/fr/acrobat/about-adobe-pdf.html

13.RAW

Un fichier RAW est un fichier brut sortant du spectromètre de masse. Il contient l'ensemble des spectres de masse qui ont été générés par le spectromètre de masse. Il est au format binaire

(illisible pour l'Homme mais optimal pour l'ordinateur). Dans ce fichier, il est possible de trouver les informations suivantes pour chaque spectre de masse analysé :

- Le spectre (intensité et MZ de tous les pics) ;
- Base peak m/z ;
- Base peak intensity ;
- Le nombre d'ions ;
- La plus petite et la plus grande valeur de m/z du spectre de masse ;
- Un numéro de scan ;
- Le niveau de MS (MS1 ou MS2) ;
- Le nombre de pics ;
- Le type de scan ;
- Le temps de rétention ;
- …

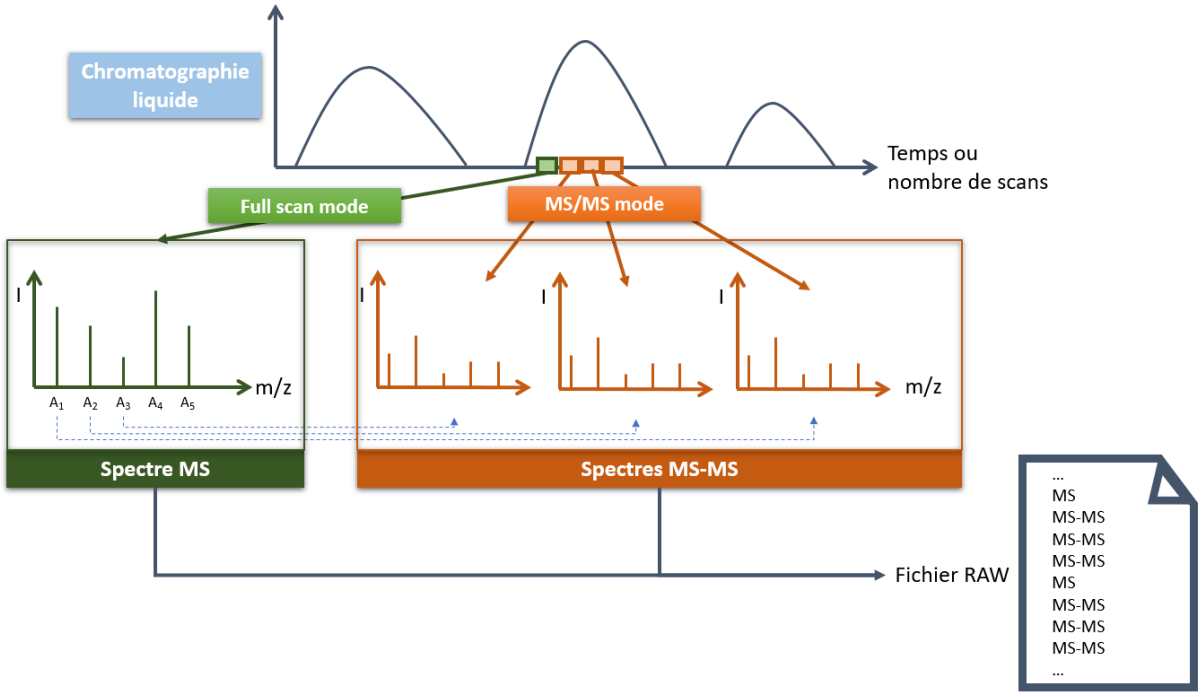


Figure 91 – Illustration du remplissage et de la composition d'un fichier RAW généré lors d'une analyse de spectrométrie de masse en tandem LC-MS/MS par l'approche Bottom up.

Extension des fichiers : .raw

Pour en savoir plus : L'article de E. Deutsch (Deutsch 2012) ou les livres co-écrits par L. Martens (Eidhammer et al. 2013; Oveland et al. 2015)

14. SAM

SAM est l'acronyme de *Sequence Alignment/Map*. Il s'agit d'un format de texte délimité par des tabulations représentant des alignements de séquences. Chaque ligne d'alignement comporte 11 champs obligatoires pour les informations essentielles à l'alignement, comme le nom de la séquence requête, le nom de la séquence référence, la qualité de l'alignement, …. D'autres champs optionnels peuvent être ajoutés. Ci-dessous un exemple de fichier SAM

```
SRR3099585.16     16          NC_014443.2        249781    42          100M       *           0           0
TCCCGCGTCTGGAAAGCCGCGCGATGAGATCCTCGAGGAGGTTGGAGCGATGCTCGATAAGTTTGACGACGTCATGAGTGCGGAAGAGAACGACGATGAG
>B>@DDDDDEDDCBDDBD@<BCCACCDDDDDDDDDDEC>C@CBFHA:GHIGGEAGHEIIIGIGIIIHDJJIJJJJGIIJHHGHEIHHHHGHFFFFFBBB  AS:i:0
XN:i:0   XM:i:0   XO:i:0   XG:i:0   NM:i:0   MD:Z:100        YT:Z:UU
```

Extension des fichiers : .sam

Pour en savoir plus : https://samtools.github.io/hts-specs/SAMv1.pdf

15. XML

XML provient de *eXtensible Markup Language*. Il s'agit d'un langage de balisage générique qui permet de définir des documents contenant à la fois les données et des indications sur la structure de ces données. Ces balises ne sont pas prédéfinies, d'où la nature extensible du langage. Les balises se reconnaissent facilement puisque qu'elles commencent par un < et se termine par un > . L'objectif principal du format XML est de faciliter au maximum les échanges de données. Ci-dessous, un exemple de fichier XML provenant de l'API STRING pour la protéine Ftr1 de la levure *C. glabrata* (identique au fichier au format JSON) :

```
<Get_string_idsResult>
  <record>
    <queryIndex>0</queryIndex>
    <stringId>5478.XP_447543.1</stringId>
    <ncbiTaxonId>5478</ncbiTaxonId>
    <taxonName>Candida glabrata</taxonName>
    <preferredName>FTR1</preferredName>
    <annotation>
      Uncharacterized protein; Highly similar to uniprot|P40088
      Saccharomyces cerevisiae YER145c FTR1
    </annotation>
  </record>
</Get_string_idsResult>
```

Extension des fichiers : .xml

Pour en savoir plus : https://www.w3.org/XML/

Accessibilité

Tous les sites proposés dans cette annexe étaient accessibles le 11/08/2020.

II. Recette du fameux gâteau au chocolat

1. Liste des réactifs

- 200 g de chocolat noir (de préférence corsé (65%) ou absolu (70%), et dans l'idéal en pastilles, plus simples à fondre au bain-marie)
- 4 œufs
- 125 g de beurre doux
- 200 g de sucre en poudre
- 100 g de farine
- 1 sachet de levure chimique
- Une pincée de fleur de sel

2. Le protocole expérimental

1. Préchauffez le four à 180°C.
2. Faites fondre le chocolat au bain-marie.

   Astuce – Pour ne pas brûler le chocolat, l'eau de doit pas toucher le contenant du chocolat. La vapeur doit être la seule source de chaleur.

   Recommandation – Le bain marie donne les meilleurs résultats pour tempérer du chocolat. Pour les plus pressés, vous pouvez le faire fondre au micro-ondes. Dans ce cas, ajoutez le beurre et faites fondre à faible puissance. Il ne faut surtout pas cuire le chocolat (formation d'une pâte compacte).
3. Ajoutez le beurre au chocolat fondu et fouettez jusqu'à homogénéisation.
4. Dans un saladier, fouettez les œufs et le sucre jusqu'à ce que le mélange blanchisse et incorporez la levure puis la farine.

   Astuce – Vous pouvez ajouter une pincée de fleur de sel. Le sel est un formidable exhausteur de goût et se marie très bien avec le chocolat. Plus courageux ? Vous pouvez la remplacer par du piment. Plus classique ? Une pointe de cannelle est un vrai régal. Plus original ? Un sachet de café soluble ! L'association café – chocolat est le duo gagnant qui surprendra vos convives !

   Information – Bien fouetter le gâteau permettra d'avoir un meilleur moelleux par l'incorporation d'air.

5. Versez le mélange chocolat - beurre dans la préparation puis homogénéisez jusqu'à l'obtention d'une pâte homogène.

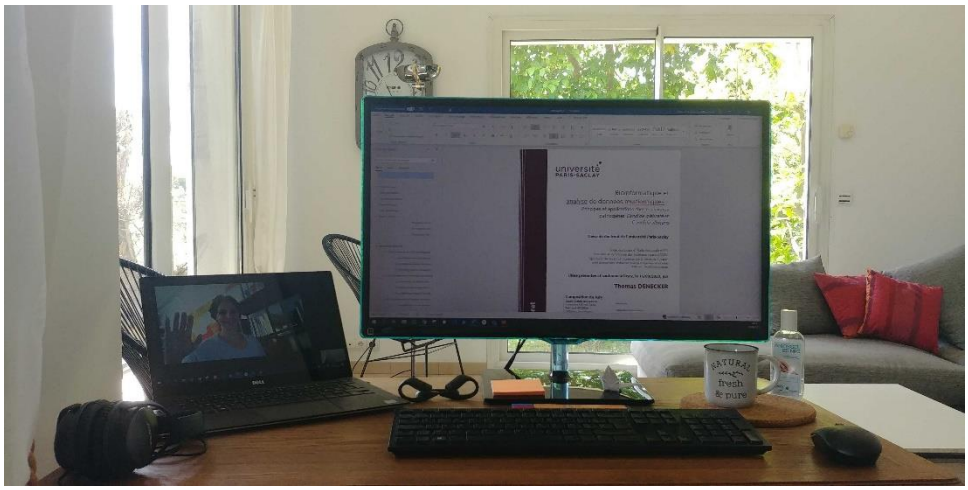
   Attention – Le mélange chocolat – beurre ne doit pas être trop chaud pour ne pas activer la levure chimique.
6. Versez la préparation dans un moule beurré et fariné.

   Astuce – Pour ne pas avoir une coloration blanche sur votre gâteau à cause de la farine, vous pouvez la remplacer par du cacao non sucré (comme du Van Houten)
7. Faites cuire environ 25 minutes.

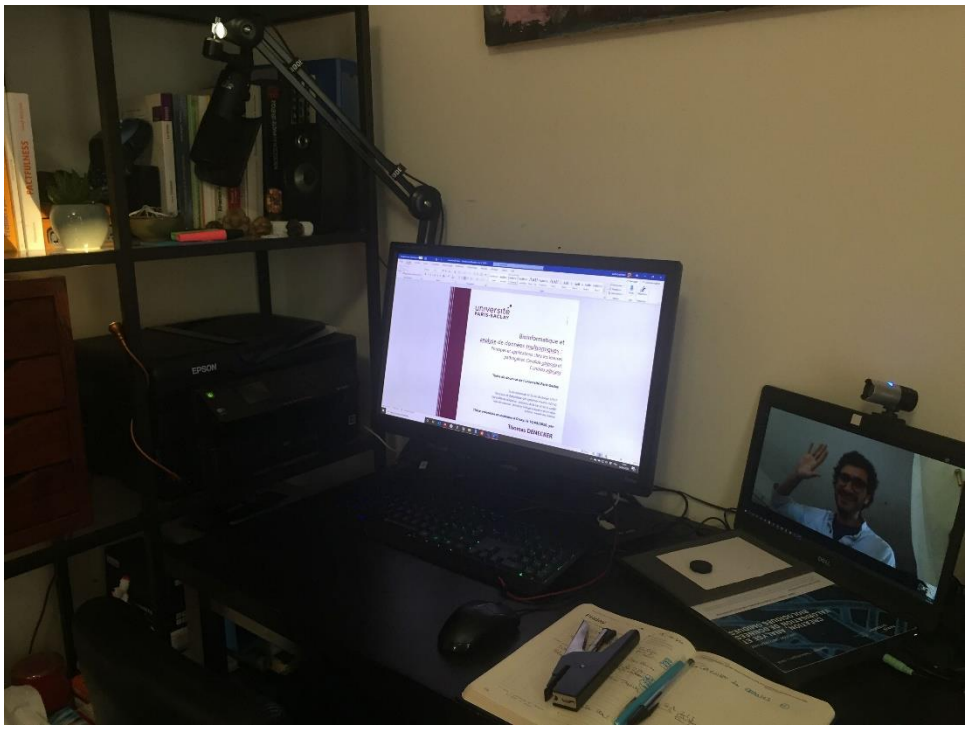
   Astuce – Utilisez la pointe d'un couteau pour tester la cuisson de votre gâteau. Vous aimez le cœur coulant, la pointe doit ressortir humide. Vous aimez votre gâteau cuit à cœur ? La pointe doit ressortir sèche.

III. Le cadre de travail

Pendant l'écriture de ma thèse, j'ai lu le livre *How to write a lot* (Silvia 2007). J'ai particulièrement apprécié une illustration proposée par l'auteur : une photo de son bureau où il a écrit son livre. En m'inspirant de cette idée, voici les bureaux où cette thèse a été écrite :



Bureau « dynamite » de Thomas



Bureau « pinceau » de Gaëlle

Références bibliographiques

Aalst, Wil Van der. 2016. *Process Mining: Data Science in Action*. *Process Mining: Data Science in Action*. https://doi.org/10.1007/978-3-662-49851-4.

Abbaspour, Nazanin, Richard Hurrell, and Roya Kelishadi. 2014. "Review on Iron and Its Importance for Human Health." *Journal of Research in Medical Sciences : The Official Journal of Isfahan University of Medical Sciences*.

Ahmad, Khadija M., Janez Kokošar, Xiaoxian Guo, Zhenglong Gu, Olena P. Ishchuk, and Jure Piškur. 2014. "Genome Structure and Dynamics of the Yeast Pathogen Candida Glabrata." *FEMS Yeast Research* 14 (4): 529–35. https://doi.org/10.1111/1567-1364.12145.

Allison, David B., Andrew W. Brown, Brandon J. George, and Kathryn A. Kaiser. 2016. "Reproducibility: A Tragedy of Errors." *Nature*. Nature Publishing Group. https://doi.org/10.1038/530027a.

Almende B.V., Benoit Thieurmel, and Titouan Robert. 2019. "VisNetwork: Network Visualization Using 'vis.Js' Library." https://cran.r-project.org/package=visNetwork.

Alon, Uri. 2020. *An Introduction to Systems Biology. Nowotwory*. Vol. 62.

Altenhoff, Adrian M., Romain A. Studer, Marc Robinson-Rechavi, and Christophe Dessimoz. 2012. "Resolving the Ortholog Conjecture: Orthologs Tend to Be Weakly, but Significantly, More Similar in Function than Paralogs." *PLoS Computational Biology* 8 (5). https://doi.org/10.1371/journal.pcbi.1002514.

Altman, R. B., and J. M. Dugan. 2003. "Defining Bioinformatics and Structural Bioinformatics. Methods of Biochemical Analysis." *Structural Bioinformatics*, 1–14. https://doi.org/10.5040/9781474213981.part-001.

Alves, Carlos Tiago, Xiao Qing Wei, Sónia Silva, Joana Azeredo, Mariana Henriques, and David W. Williams. 2014. "Candida Albicans Promotes Invasion and Colonisation of Candida Glabrata in a Reconstituted Human Vaginal Epithelium." *Journal of Infection* 69 (4): 396–407. https://doi.org/10.1016/j.jinf.2014.06.002.

Alves, Gelio, Aleksey Y. Ogurtsov, and Yi Kuo Yu. 2007. "RAId_DbS: Peptide Identification Using Database Searches with Realistic Statistics." *Biology Direct* 2: 1–20. https://doi.org/10.1186/1745-6150-2-25.

———. 2008. "RAId_DbS: Mass-Spectrometry Based Peptide Identification Web Server with Knowledge Integration." *BMC Genomics* 9: 1–12. https://doi.org/10.1186/1471-2164-9-505.

———. 2010. "RAId_aPS: MS/MS Analysis with Multiple Scoring Functions and Spectrum-Specific Statistics." *PLoS ONE* 5 (11). https://doi.org/10.1371/journal.pone.0015438.

Alves, Gelio, and Yi Kuo Yu. 2005. "Robust Accurate Identification of Peptides (RAId): Deciphering MS2 Data Using a Structured Library Search with de Novo Based Statistics." *Bioinformatics* 21 (19): 3726–32. https://doi.org/10.1093/bioinformatics/bti620.

Amrhein, Valentin, and Sander Greenland. 2018. "Remove, Rather than Redefine, Statistical Significance." *Nature Human Behaviour* 2 (1): 4–4. https://doi.org/10.1038/s41562-017-0224-0.

Amrhein, Valentin, Sander Greenland, and Blake McShane. 2019. “Scientists Rise up against Statistical Significance.” *Nature*. Nature Publishing Group. https://doi.org/10.1038/d41586-019-00857-9.

Amrhein, Valentin, Fränzi Korner-Nievergelt, and Tobias Roth. 2017. “The Earth Is Flat (p > 0:05): Significance Thresholds and the Crisis of Unreplicable Research.” *PeerJ* 2017 (7): 1–40. https://doi.org/10.7717/peerj.3544.

Amrhein, Valentin, David Trafimow, and Sander Greenland. 2019. “Inferential Statistics as Descriptive Statistics: There Is No Replication Crisis If We Don’t Expect Replication.” *American Statistician* 73 (sup1): 262–70. https://doi.org/10.1080/00031305.2018.1543137.

Andorf, Carson, Drena Dobbs, and Vasant Honavar. 2007. “Exploring Inconsistencies in Genome-Wide Protein Function Annotations: A Machine Learning Approach.” *BMC Bioinformatics* 8 (1): 284. https://doi.org/10.1186/1471-2105-8-284.

Angoulvant, A., J. Guitard, and C. Hennequin. 2016. “Old and New Pathogenic Nakaseomyces Species: Epidemiology, Biology, Identification, Pathogenicity and Antifungal Resistance.” *FEMS Yeast Research* 16 (2): 1–13. https://doi.org/10.1093/femsyr/fov114.

Aparicio, Manuela, and Carlos J. Costa. 2014. “Data Visualization.” *Communication Design Quarterly*. https://doi.org/10.1007/978-3-319-68837-4_5.

Attali, Dean. 2017. “Colourpicker: A Colour Picker Tool for Shiny and for Selecting Colours in Plots.” https://cran.r-project.org/package=colourpicker.

———. 2020. “Shinyjs: Easily Improve the User Experience of Your Shiny Apps in Seconds.” https://cran.r-project.org/package=shinyjs.

Attali, Dean, and Tristan Edwards. 2018. “Shinyalert: Easily Create Pretty Popup Messages (Modals) in ‘Shiny.’” https://cran.r-project.org/package=shinyalert.

Bai, Chen, Xiao Li Xu, Fong Yee Chan, Raymond Teck Ho Lee, and Yue Wang. 2006. “MNN5 Encodes an Iron-Regulated α-1,2-Mannosyltransferase Important for Protein Glycosylation, Cell Wall Integrity, Morphogenesis, and Virulence in Candida Albicans.” *Eukaryotic Cell* 5 (2): 238–47. https://doi.org/10.1128/EC.5.2.238-247.2006.

Baker, Monya, and Dan Penny. 2016. “Is There a Reproducibility Crisis?” *Nature*. Nature Publishing Group. https://doi.org/10.1038/533452A.

Bao, Wei Guo, Bernard Guiard, Zi An Fang, Claudia Donnini, Michel Gervais, Flavia M. Lopes Passos, Iliana Ferrero, Hiroshi Fukuhara, and Monique Bolotin-Fukuhara. 2008. “Oxygen-Dependent Transcriptional Regulator Hap1p Limits Glucose Uptake by Repressing the Expression of the Major Glucose Transporter Gene RAG1 in Kluyveromyces Lactis.” *Eukaryotic Cell* 7 (11): 1895–1905. https://doi.org/10.1128/EC.00018-08.

Bartalanffy, Ludwig Von. 1969. “General System Theory : Foundations, Development, Applications.” *Arch Gen Psychiatry* 21: 251–52.

Bastian, Mathieu, Sebastien Heymann, and Mathieu Jacomy. 2009. “Gephi: An Open Source Software for Exploring and Manipulating Networks Visualization and Exploration of Large Graphs.” *Third International AAAI Conference on Weblogs and Social Media*. www.aaai.org.

Bell, Gordon, Tony Hey, and Alex Szalay. 2009. “Computer Science: Beyond the Data

Deluge." *Science*. https://doi.org/10.1126/science.1170411.

Benjamin, Daniel J., James O. Berger, Magnus Johannesson, Brian A. Nosek, E. J. Wagenmakers, Richard Berk, Kenneth A. Bollen, et al. 2018. "Redefine Statistical Significance." *Nature Human Behaviour* 2 (1): 6–10. https://doi.org/10.1038/s41562-017-0189-z.

Berinato, Scott. 2016. *Good Charts - The HBR Guide to Making Smarter, More Persuasive Data Visualizations*. Harvard Business Review Press.

Bertin, Jacques. 1968. "Sémiologie Graphique. Les Diagrammes. Les Réseaux. Les Cartes." *Archives de Sociologie Des Religions* 26 (1): 176–77.

Bilotta, Mariaconcetta, Giuseppe Tradigo, and Pierangelo Veltri. 2018. *Bioinformatics Data Models, Representation and Storage*. *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*. Vol. 1–3. Elsevier Ltd. https://doi.org/10.1016/B978-0-12-809633-8.20410-X.

Bishop, Dorothy. 2019. "Rein in the Four Horsemen of Irreproducibility." *Nature*. Nature Research. https://doi.org/10.1038/d41586-019-01307-2.

Boubaker, Leila, Leila Mellal, and Mébarek Djebabra. 2010. "Modèle DIC (Données ? Informations ? Connaissances) Outil Support Pour Le Développement Des Mémoires Projets." *La Revue Des Sciences de Gestion* 243–244 (3): 153. https://doi.org/10.3917/rsg.243.0153.

Boussier, Jérémy. 2019. "BEWARE OF P-VALUES." In .

Bray, Mark Anthony, Martha S. Vokes, and Anne E. Carpenter. 2015. "Using Cellprofiler for Automatic Identification and Measurement of Biological Objects in Images." *Current Protocols in Molecular Biology* 2015 (January): 14.17.1-14.17.13. https://doi.org/10.1002/0471142727.mb1417s109.

Breakey, Alicia A, Katie Hinde, Claudia R Valeggia, Allison Sinofsky, and Peter T Ellison. 2015. "Illness in Breastfeeding Infants Relates to Concentration of Lactoferrin and Secretory Immunoglobulin A in Mother ' s Milk." *Evolution, Medicine, and Public Health*, 21–31. https://doi.org/10.1093/emph/eov002.

Breitling, Rainer. 2010. "What Is Systems Biology?" *Frontiers in Physiology* 1 MAY. https://doi.org/10.3389/fphys.2010.00009.

Brena, Sonia, Jonathan Cabezas-Olcoz, María D. Moragues, Iñigo Fernández De Larrinoa, Angel Domínguez, Guillermo Quindós, and José Pontón. 2011. "Fungicidal Monoclonal Antibody C7 Interferes with Iron Acquisition in Candida Albicans." *Antimicrobial Agents and Chemotherapy* 55 (7): 3156–63. https://doi.org/10.1128/AAC.00892-10.

Brown, Andrew W., Kathryn A. Kaiser, and David B. Allison. 2018. "Issues with Data and Analyses: Errors, Underlying Themes, and Potential Solutions." *Proceedings of the National Academy of Sciences of the United States of America* 115 (11): 2563–70. https://doi.org/10.1073/pnas.1708279115.

Brown, Gordon D., David W. Denning, Neil A.R. Gow, Stuart M. Levitz, Mihai G. Netea, and Theodore C. White. 2012. "Hidden Killers: Human Fungal Infections." *Science Translational Medicine* 4 (165). https://doi.org/10.1126/scitranslmed.3004404.

Brunke, Sascha, and Bernhard Hube. 2013. "Two Unlike Cousins: Candida Albicans and

C.Glabrata Infection Strategies." *Cellular Microbiology*. https://doi.org/10.1111/cmi.12091.

Bunnik, Evelien M., and Karine G. Le Roch. 2013. "An Introduction to Functional Genomics and Systems Biology." *Advances in Wound Care* 2 (9): 490–98. https://doi.org/10.1089/wound.2012.0379.

Bürkle, A. 2001. "Posttranslational Modification." *Encyclopedia of Genetics*, 1533. https://doi.org/10.1006/rwgn.2001.1022.

Burley, Stephen K., Helen M. Berman, Charmi Bhikadiya, Chunxiao Bi, Li Chen, Luigi Di Costanzo, Cole Christie, et al. 2019. "RCSB Protein Data Bank: Biological Macromolecular Structures Enabling Research and Education in Fundamental Biology, Biomedicine, Biotechnology and Energy." *Nucleic Acids Research* 47 (D1): D464–74. https://doi.org/10.1093/nar/gky1004.

Campos-Garcia, Lizbeth, Rocio Jimena Jimenez-Valdes, Romel Hernandez-Bello, Jose Palma-Nicolas, Gloria Maria Gonzalez, and Alejandro Sanchez-Gonzalez. 2019. "Candida Albicans and Non-Albicans Isolates from Bloodstream Have Different Capacities to Induce Neutrophil Extracellular Traps." *Journal of Fungi* 5 (2). https://doi.org/10.3390/jof5020028.

Carbon, S., E. Douglass, N. Dunn, B. Good, N. L. Harris, S. E. Lewis, C. J. Mungall, et al. 2019. "The Gene Ontology Resource: 20 Years and Still GOing Strong." *Nucleic Acids Research* 47 (D1): D330–38. https://doi.org/10.1093/nar/gky1055.

Carter, Matt, and Jennifer Shieh. 2015. *Biochemical Assays and Intracellular Signaling*. *Guide to Research Techniques in Neuroscience*. https://doi.org/10.1016/b978-0-12-800511-8.00015-0.

Cassat, James E., and Eric P. Skaar. 2013. "Iron in Infection and Immunity." *Cell Host and Microbe*. Cell Press. https://doi.org/10.1016/j.chom.2013.04.010.

Chang, Winston, and Barbara Borges Ribeiro. 2018. "Shinydashboard: Create Dashboards with 'Shiny.'" https://cran.r-project.org/package=shinydashboard.

Chang, Winston, Joe Cheng, J J Allaire, Yihui Xie, and Jonathan McPherson. 2019. "Shiny: Web Application Framework for R." https://cran.r-project.org/package=shiny.

Charif, D, and J R Lobry. 2007. "Seqin{R} 1.0-2: A Contributed Package to the {R} Project for Statistical Computing Devoted to Biological Sequences Retrieval and Analysis." In *Structural Approaches to Sequence Evolution: Molecules, Networks, Populations*, edited by U Bastolla, M Porto, H E Roman, and M Vendruscolo, 207–32. Biological and Medical Physics, Biomedical Engineering. New York: Springer Verlag.

Chaudet, Bruno. 2009. "Donnée, Information, Connaissance | Logiques Processuelles." 2009. https://brunochaudet.wordpress.com/2009/03/30/donnee-information-connaissance/.

Cherry, J Michael, Eurie L Hong, Craig Amundsen, Rama Balakrishnan, Gail Binkley, Esther T Chan, Karen R Christie, et al. 2012. "Saccharomyces Genome Database : The Genomics Resource of Budding Yeast." *Nucleic Acids Research* 40 (November 2011): 700–705. https://doi.org/10.1093/nar/gkr1029.

Cho, Ilseung, and Martin J. Blaser. 2012. "The Human Microbiome: At the Interface of Health and Disease." *Nature Reviews Genetics* 13 (4): 260–70. https://doi.org/10.1038/nrg3182.

Chorostecki, Uciel, Manuel Molina, Leszek P Pryszcz, and Toni Gabaldon. 2020. “MetaPhOrs 2.0: Integrative, Phylogeny-Based Inference of Orthology and Paralogy across the Tree of Life.” *Nucleic Acids Research*, 1–5. https://doi.org/10.1093/nar/gkaa282.

Collberg, Christian, Todd Proebsting, and Alex M Warren. 2015. “Repeatability and Benefaction in Computer Systems Research.” http://cs.brown.edu/.

Consortium, The UniProt. 2019. “UniProt: A Worldwide Hub of Protein Knowledge.” *Nucleic Acids Research* 47.

Cook, Charles E, Mary T Bergman, Guy Cochrane, Rolf Apweiler, and Ewan Birney. 2017. “The European Bioinformatics Institute in 2017: Data Coordination and Integration.” *Nucleic Acids Research* 46: 21–29. https://doi.org/10.1093/nar/gkx1154.

CrowdFlower. 2016. “Data Science Report.”

Crowley, D. E., Y. C. Wang, C. P.P. Reid, and P. J. Szaniszlo. 1991. “Mechanisms of Iron Acquisition from Siderophores by Microorganisms and Plants.” *Plant and Soil* 130 (1–2): 179–98. https://doi.org/10.1007/BF00011873.

Csank, Csilla, and Ken Haynes. 2000. “Candida Glabrata Displays Pseudohyphal Growth.” *FEMS Microbiology Letters* 189 (1): 115–20. https://doi.org/10.1016/S0378-1097(00)00241-X.

Csardi, Gabor, and Tamas Nepusz. 2006. “The Igraph Software Package for Complex Network Research.” *InterJournal* Complex Sy: 1695. http://igraph.org.

Csizmok, Veronika, and Julie D. Forman-Kay. 2018. “Complex Regulatory Mechanisms Mediated by the Interplay of Multiple Post-Translational Modifications.” *Current Opinion in Structural Biology* 48: 58–67. https://doi.org/10.1016/j.sbi.2017.10.013.

Cui, Lijia, Alison Morris, and Elodie Ghedin. 2013. “The Human Mycobiome in Health and Disease.” *Genome Medicine* 5 (7): 1–12. https://doi.org/10.1186/gm467.

David, Jean. 2019. “Trop d’informations, Pas Assez de Connaissances.” Vitaloweb. 2019. https://vitaloweb.org/2017/01/26/trop-informations-pas-assez-de-connaissances-prise-de-decision-pyramide-connaissance/.

Debrouwere, Stijn, and Els Goetghebeur. 2014. “The Statistical Crisis in Science.”

Delamare, Garnier. 2009. *Dictionnaire Illustré Des Termes de Médecine*. Edited by MALOINE.

Denecker, Thomas, Youfang Zhou Li, Cécile Fairhead, Karine Budin, Jean-Michel Camadro, Monique Bolotin-Fukuhara, Adela Angoulvant, and Gaëlle Lelandais. 2020. “Functional Networks of Co-Expressed Genes to Explore Iron Homeostasis Processes in the Pathogenic Yeast Candida Glabrata.” *NAR Genomics and Bioinformatics* 2 (2): 1–14. https://doi.org/10.1093/nargab/lqaa027.

Desai, Jigar V, Aaron P Mitchell, and David R Andes. 2014. “And Recurrent Infection,” 1–18.

Deutsch, Eric W. 2012. “File Formats Commonly Used in Mass Spectrometry Proteomics.” *Molecular and Cellular Proteomics* 11 (12): 1612–21. https://doi.org/10.1074/mcp.R112.019695.

Dev, Som, and Jodie L. Babitt. 2017. “Overview of Iron Metabolism in Health and Disease.”

Hemodialysis International. Blackwell Publishing Inc. https://doi.org/10.1111/hdi.12542.

Devaux, Frédéric, and Antonin Thiébaut. 2019. "The Regulation of Iron Homeostasis in the Fungal Human Pathogen Candida Glabrata." *Microbiology (United Kingdom)* 165 (10): 1041–60. https://doi.org/10.1099/mic.0.000807.

Diniz, W. J.S., and F. Canduri. 2017. "Bioinformatics: An Overview and Its Applications." *Genetics and Molecular Research* 16 (1): 1–21. https://doi.org/10.4238/gmr16019645.

Doi, André Mario, Antonio Carlos Campos Pignatari, Michael B. Edmond, Alexandre Rodrigues Marra, Luis Fernando Aranha Camargo, Ricardo Andreotti Siqueira, Vivian Pereira Da Mota, and Arnaldo Lopes Colombo. 2016. "Epidemiology and Microbiologic Characterization of Nosocomial Candidemia from a Brazilian National Surveillance Program." *PLoS ONE* 11 (1). https://doi.org/10.1371/journal.pone.0146909.

Downing, Gregory, Florence Haseltine, Belinda Seto, and Yuan Liu. 2000. "NIH WORKING DEFINITION OF BIOINFORMATICS AND COMPUTATIONAL BIOLOGY."

Dujon, Bernard. 2010. "Yeast Evolutionary Genomics." *Nature Reviews. Genetics* 11 VN-r (7): 512–24. https://doi.org/10.1038/nrg2811.

———. 2019. "My Route to the Intimacy of Genomes." *FEMS Yeast Research* 19 (3). https://doi.org/10.1093/femsyr/foz023.

Dujon, Bernard, David Sherman, Gilles Fischer, Pascal Durrens, Serge Casaregela, Ingrid Lafentaine, Jacky De Montigny, et al. 2004. "Genome Evolution in Yeasts." *Nature* 430 (6995): 35–44. https://doi.org/10.1038/nature02579.

Dutta, Amit Kumar, and Ragib Hasan. 2013. "How Much Does Storage Really Cost ? Towards," 29–43.

Ebanks, Roger O., Kenneth Chisholm, Stewart McKinnon, Malcolm Whiteway, and Devanand M. Pinto. 2006. "Proteomic Analysis of Candida Albicans Yeast and Hyphal Cell Wall and Associated Proteins." *Proteomics* 6 (7): 2147–56. https://doi.org/10.1002/pmic.200500100.

Eidhammer, Ingvar, Harald Barsnes, Geir Egil Eide, and Lennart Martens. 2013. *Computational and Statistical Methods for Protein Quantification by Mass Spectrometry*. *Computational and Statistical Methods for Protein Quantification by Mass Spectrometry*. https://doi.org/10.1002/9781118494042.

El-Gebali, Sara, Jaina Mistry, Alex Bateman, Sean R. Eddy, Aurélien Luciani, Simon C. Potter, Matloob Qureshi, et al. 2019. "The Pfam Protein Families Database in 2019." *Nucleic Acids Research* 47 (D1): D427–32. https://doi.org/10.1093/nar/gky995.

Elgabry, Omar. 2019. "The Ultimate Guide to Data Cleaning." Towards Data Science. 2019. https://towardsdatascience.com/the-ultimate-guide-to-data-cleaning-3969843991d4.

Elmasri, Ramez, and Shamkant B. Navathe. 2010. *FUNDAMENTALS OF Database Systems*. *Database Systems*.

Emms, David M., and Steven Kelly. 2019. "OrthoFinder: Phylogenetic Orthology Inference for Comparative Genomics." *Genome Biology* 20 (1): 1–14. https://doi.org/10.1186/s13059-019-1832-y.

Experian. 2016. "The Impact of Contact Data Quality," no. August.

Fecher, Benedikt, and Sascha Friesike. 2013. “Open Science: One Term, Five Schools of Thought.” *SSRN Electronic Journal*, June. https://doi.org/10.2139/ssrn.2272036.

Fedier, Andre, Konstantin J. Dedes, Patrick Imesch, Andre O. Von Bueren, and Daniel Fink. 2007. “The Histone Deacetylase Inhibitors Suberoylanilide Hydroxamic (Vorinostat) and Valproic Acid Induce Irreversible and MDR1-Independent Resistance in Human Colon Cancer Cells.” *International Journal of Oncology* 31 (3): 633–41. https://doi.org/10.3892/ijo.31.3.633.

Fenton, HJ..H. 1894. “Oxydation of Tartaric Acid in Presence of Iron.” *J. Chem. Soc.* 65: 899–910.

Fernández, Rosa, Toni Gabaldón, and Christophe Dessimoz. 2019. “Orthology: Definitions, Inference, and Impact on Species Phylogeny Inference.” *ArXiv*, 1–17. http://arxiv.org/abs/1903.04530.

Fidel, P. L. 2006. “Candida-Host Interactions in HIV Disease: Relationships in Oropharyngeal Candidiasis.” *Advances in Dental Research*. https://doi.org/10.1177/154407370601900116.

Fiori, Alessandro, and Patrick Van Dijck. 2012. “Potent Synergistic Effect of Doxycycline with Fluconazole against Candida Albicans Is Mediated by Interference with Iron Homeostasis.” *Antimicrobial Agents and Chemotherapy* 56 (7): 3785–96. https://doi.org/10.1128/AAC.06017-11.

Fitzpatrick, David A., Peadar O’Gaora, Kevin P. Byrne, and Geraldine Butler. 2010. “Analysis of Gene Evolution and Metabolic Pathways Using the Candida Gene Order Browser.” *BMC Genomics* 11 (1). https://doi.org/10.1186/1471-2164-11-290.

Fitzpatrick, David A, Mary E Logue, Jason E Stajich, and Geraldine Butler. 2006. “A Fungal Phylogeny Based on 42 Complete Genomes Derived from Supertree and Combined Gene Analysis.” *BMC Evolutionary Biology* 15: 1–15. https://doi.org/10.1186/1471-2148-6-99.

Fourie, Ruan, Oluwasegun O. Kuloyo, Bonang M. Mochochoko, Jacobus Albertyn, and Carolina H. Pohl. 2018. “Iron at the Centre of Candida Albicans Interactions.” *Frontiers in Cellular and Infection Microbiology*. Frontiers Media S.A. https://doi.org/10.3389/fcimb.2018.00185.

Furlaneto, Márcia C., Helena P. Góes, Hugo F. Perini, Renan C. dos Santos, and Luciana Furlaneto-Maia. 2018. “How Much Do We Know about Hemolytic Capability of Pathogenic Candida Species?” *Folia Microbiologica*. Springer Netherlands. https://doi.org/10.1007/s12223-018-0584-5.

Gabaldón, Toni, and Laia Carreté. 2016. “The Birth of a Deadly Yeast: Tracing the Evolutionary Emergence of Virulence Traits in Candida Glabrata.” *FEMS Yeast Research* 16 (2): 1–9. https://doi.org/10.1093/femsyr/fov110.

Gabaldón, Toni, and Cécile Fairhead. 2019. “Genomes Shed Light on the Secret Life of Candida Glabrata: Not so Asexual, Not so Commensal.” *Current Genetics* 65 (1): 93–98. https://doi.org/10.1007/s00294-018-0867-z.

Gabaldón, Toni, and Eugene V. Koonin. 2013. “Functional and Evolutionary Implications of Gene Orthology.” *Nat Rev Genet.* https://doi.org/10.1016/j.physbeh.2017.03.040.

Gabaldón, Toni, Tiphaine Martin, Marina Marcet-Houben, Pascal Durrens, Monique Bolotin-

Fukuhara, Olivier Lespinet, Sylvie Arnaise, et al. 2013. “Comparative Genomics of Emerging Pathogens in the Candida Glabrata Clade.” *BMC Genomics* 14 (1). https://doi.org/10.1186/1471-2164-14-623.

Galocha, Mónica, Pedro Pais, Mafalda Cavalheiro, Diana Pereira, Romeu Viana, and Miguel C. Teixeira. 2019. “Divergent Approaches to Virulence in C. Albicans and C. Glabrata: Two Sides of the Same Coin.” *International Journal of Molecular Sciences* 20 (9). https://doi.org/10.3390/ijms20092345.

Ganz, Tomas, and Elizabeta Nemeth. 2012. “Hepcidin and Iron Homeostasis.” *Biochimica et Biophysica Acta - Molecular Cell Research*. https://doi.org/10.1016/j.bbamcr.2012.01.014.

———. 2015. “Iron Homeostasis in Host Defence and Inflammation.” *Nat Rev Immunol*.

Gayon, Jean. 2016. “De Mendel à l’épigénétique : Histoire de La Génétique.” *Comptes Rendus - Biologies* 339 (7–8): 225–30. https://doi.org/10.1016/j.crvi.2016.05.009.

Gehlenborg, Nils. 2019. “UpSetR: A More Scalable Alternative to Venn and Euler Diagrams for Visualizing Intersecting Sets.” https://cran.r-project.org/package=UpSetR.

Gergondey, R., C. Garcia, V. Serre, J. M. Camadro, and F. Auchère. 2016. “The Adaptive Metabolic Response Involves Specific Protein Glutathionylation during the Filamentation Process in the Pathogen Candida Albicans.” *Biochimica et Biophysica Acta - Molecular Basis of Disease* 1862 (7): 1309–23. https://doi.org/10.1016/j.bbadis.2016.04.004.

Gerts, E. Michael, Yi Kuo Yu, Richa Agarwala, Alejandro A. Schäffer, and Stephen F. Altschul. 2006. “Composition-Based Statistics and Translated Nucleotide Searches: Improving the TBLASTN Module of BLAST.” *BMC Biology* 4: 1–14. https://doi.org/10.1186/1741-7007-4-41.

Gerwien, Franziska, Abu Safyan, Stephanie Wisgott, Sascha Brunke, Lydia Kasper, and Bernhard Hube. 2017. “The Fungal Pathogen Candida Glabrata Does Not Depend on Surface Ferric Reductases for Iron Acquisition.” *Frontiers in Microbiology* 8 (JUN). https://doi.org/10.3389/fmicb.2017.01055.

Gerwien, Franziska, Abu Safyan, Stephanie Wisgott, Fabrice Hille, Philipp Kaemmer, Jörg Linde, Sascha Brunke, Lydia Kasper, and Bernhard Hube. 2016. “A Novel Hybrid Iron Regulation Network Combines Features from Pathogenic and Nonpathogenic Yeasts.” *MBio* 7 (5). https://doi.org/10.1128/mBio.01782-16.

Gerwien, Franziska, Volha Skrahina, Lydia Kasper, Bernhard Hube, and Sascha Brunke. 2018. “Metals in Fungal Virulence.” *FEMS Microbiology Reviews* 42 (1): 1–21. https://doi.org/10.1093/femsre/fux050.

Gesmann, Markus, and Diego de Castillo. 2011. “Using the Google Visualisation Api with R.” *R Journal* 3 (2): 40–44. https://doi.org/10.32614/RJ-2011-017.

Gibbons, Jean D, and John W Pratt. 1975. “P-Values: Interpretation and Methodolog.” *The American Statisticia* 29 (1): 829–34.

GO FAIR. 2020. “FAIR Principles.” 2020. https://www.go-fair.org/fair-principles/.

Granjon, David. 2019. “ShinydashboardPlus: Add More ‘AdminLTE2’ Components to ‘Shinydashboard.’” https://cran.r-project.org/package=shinydashboardPlus.

Grant, Gregory R., Elisabetta Manduchi, and Christian J. Stoeckert. 2007. "Analysis and Management of Microarray Gene Expression Data." *Current Protocols in Molecular Biology / Edited by Frederick M. Ausubel ... [et Al.]* Chapter 19. https://doi.org/10.1002/0471142727.mb1906s77.

Greenland, Sander. 2017. "Invited Commentary: The Need for Cognitive Science in Methodology." *American Journal of Epidemiology* 186 (6): 639–45. https://doi.org/10.1093/aje/kwx259.

Grüning, Björn, John Chilton, Johannes Köster, Ryan Dale, Nicola Soranzo, Marius van den Beek, Jeremy Goecks, Rolf Backofen, Anton Nekrutenko, and James Taylor. 2018. "Practical Computational Reproducibility in the Life Sciences." *Cell Systems*. Cell Press. https://doi.org/10.1016/j.cels.2018.03.014.

Guinea, J. 2014. "Global Trends in the Distribution of Candida Species Causing Candidemia." *Clinical Microbiology and Infection*. Elsevier B.V. https://doi.org/10.1111/1469-0691.12539.

Han, Yongmoon. 2014. "The Identification of Surface Interaction of Apotransferrin with Candida Albicans." *Archives of Pharmacal Research* 37 (10): 1301–7. https://doi.org/10.1007/s12272-013-0301-5.

Head, Megan L., Luke Holman, Rob Lanfear, Andrew T. Kahn, and Michael D. Jennions. 2015. "The Extent and Consequences of P-Hacking in Science." *PLOS Biology* 13 (3): e1002106. https://doi.org/10.1371/journal.pbio.1002106.

Higson, Frank K., Ron Kohen, and Mordechai Chevion. 1988. "Iron Enhancement of Ascorbate Toxicity." *Free Radical Research* 5 (2): 107–15. https://doi.org/10.3109/10715768809066918.

Hirai, Kazuyuki, Tatsuya Inukai, and Hironobu Nakayama. 2016. "Promising Therapies for Fungal Infection Based on the Study to Elucidate Mechanisms to Cope with Stress in Candida Species." *Medical Mycology Journal* 57 (4): J163–70. https://doi.org/10.3314/mmj.16.007.

Ho, Brandon, Anastasia Baryshnikova, and Grant W. Brown. 2018. "Unification of Protein Abundance Datasets Yields a Quantitative Saccharomyces Cerevisiae Proteome." *Cell Systems* 6 (2): 192-205.e3. https://doi.org/10.1016/j.cels.2017.12.004.

Hogeweg, Paulien. 2011. "The Roots of Bioinformatics in Theoretical Biology." Edited by David B. Searls. *PLoS Computational Biology* 7 (3): e1002021. https://doi.org/10.1371/journal.pcbi.1002021.

Holzinger, Andreas, Peter Kieseberg, A Min Tjoa, and Edgar Weippl. 2018. *Machine Learning and Knowledge Extraction*. *Springer*. https://doi.org/10.1007/978-3-030-29726-8.

Hosny, Ahmed, Chintan Parmar, John Quackenbush, Lawrence H. Schwartz, and Hugo J.W.L. Aerts. 2018. "Artificial Intelligence in Radiology." *Nature Reviews Cancer*. Nature Publishing Group. https://doi.org/10.1038/s41568-018-0016-5.

Hu, Zhenjun, Yi Chien Chang, Yan Wang, Chia Ling Huang, Yang Liu, Feng Tian, Brian Granger, and Charles Delisi. 2013. "VisANT 4.0: Integrative Network Platform to Connect Genes, Drugs, Diseases and Therapies." *Nucleic Acids Research* 41 (Web Server issue): 225–31. https://doi.org/10.1093/nar/gkt401.

Huerta-Cepas, Jaime, Salvador Capella-Gutiérrez, Leszek P. Pryszcz, Marina Marcet-Houben, and Toni Gabaldón. 2014. "PhylomeDB v4: Zooming into the Plurality of Evolutionary Histories of a Genome." *Nucleic Acids Research* 42 (D1): 897–902. https://doi.org/10.1093/nar/gkt1177.

Hurlbert, Stuart H., Richard A. Levine, and Jessica Utts. 2019. "Coup de Grâce for a Tough Old Bull: 'Statistically Significant' Expires." *American Statistician* 73 (sup1): 352–57. https://doi.org/10.1080/00031305.2018.1543616.

Ihrig, Jessica, Anja Hausmann, Anika Hain, Nadine Richter, Iqbal Hamza, Roland Lill, and Ulrich Mühlenhoff. 2010. "Iron Regulation through the Back Door: Iron-Dependent Metabolite Levels Contribute to Transcriptional Adaptation to Iron Deprivation in Saccharomyces Cerevisiae." *Eukaryotic Cell* 9 (3): 460–71. https://doi.org/10.1128/EC.00213-09.

Jafari, Mohieddin, and Naser Ansari-Pour. 2019. "Why, When and How to Adjust Your P Values?" *Cell Journal* 20 (4): 604–7. https://doi.org/10.22074/cellj.2019.5992.

Jaillette, Emmanuelle, Christophe Girault, Guillaume Brunin, Farid Zerimech, Arnaud Chiche, Céline Broucqsault-Dedrie, Cyril Fayolle, et al. 2016. "French Intensive Care Society, International Congress – Réanimation 2016." *Annals of Intensive Care* 6 (S1). https://doi.org/10.1186/s13613-016-0114-z.

Jassal, Bijay, Lisa Matthews, Guilherme Viteri, Chuqiao Gong, Pascual Lorente, Antonio Fabregat, Konstantinos Sidiropoulos, et al. 2020. "The Reactome Pathway Knowledgebase." *Nucleic Acids Research* 48 (D1): D498–503. https://doi.org/10.1093/nar/gkz1031.

Jones, Ted, Nancy A. Federspiel, Hiroji Chibana, Jan Dungan, Sue Kalman, B. B. Magee, George Newport, et al. 2004. "The Diploid Genome Sequence of Candida Albicans." *Proceedings of the National Academy of Sciences of the United States of America* 101 (19): 7329–34. https://doi.org/10.1073/pnas.0401648101.

Joyce, Brendan, Danny Lee, Alex Rubio, Aleksey Ogurtsov, Gelio Alves, and Yi Kuo Yu. 2018. "A Graphical User Interface for RAId, a Knowledge Integrated Proteomics Analysis Suite with Accurate Statistics." *BMC Research Notes* 11 (1): 1–7. https://doi.org/10.1186/s13104-018-3289-6.

Juan M. Vaquerizas, Sarah A. Teichmann, and Nicholas M. Luscombe, and Abstract. 2012. "How Do You Find Transcription Factors? Computational Approaches to Compile and Annotate Repertoires of Regulators for Any Genome." In *Gene Regulatory Networks: Methods and Protocols, Methods in Molecular Biology*. Vol. 786. https://doi.org/10.1007/978-1-61779-292-2.

Kanehisa, Minoru, and Susumu Goto. 2000. "KEGG: Kyoto Encyclopedia of Genes and Genomes." *Nucleic Acids Research*.

Kasper, Lydia, Katja Seider, and Bernhard Hube. 2015. "Intracellular Survival of Candida Glabrata in Macrophages." *FEMS Yeast Res.*

Kaur, Rupinder, Renee Domergue, Margaret L. Zupancic, and Brendan P. Cormack. 2005. "A Yeast by Any Other Name: Candida Glabrata and Its Interaction with the Host." *Current Opinion in Microbiology* 8 (4): 378–84. https://doi.org/10.1016/j.mib.2005.06.012.

Kavanagh, Kevin. 2007. *New Insights in Medical Mycology*. *New Insights in Medical*

Mycology. https://doi.org/10.1007/978-1-4020-6397-8.

Koonin, Eugene V, and Michael Y Galperin. 2003. "Genome Annotation and Analysis - Sequence - Evolution - Function - NCBI Bookshelf." 2003. https://www-ncbi-nlm-nih-gov.insb.bib.cnrs.fr/books/NBK20253/.

Kuipers, Mirjam E., Jannet Heegsma, Hester I. Bakker, Dick K.F. Meijer, Pieter J. Swart, Erik W. Frijlink, Anco C. Eissens, Hilly G. De Vries-Hospers, and Jeroen J.M. Van Den Berg. 2002. "Design and Fungicidal Activity of Mucoadhesive Lactoferrin Tablets for the Treatment of Oropharyngeal Candidosis." *Drug Delivery: Journal of Delivery and Targeting of Therapeutic Agents* 9 (1): 31–38. https://doi.org/10.1080/107175402753413154.

Kumar, Vishal, and V. P. Choudhry. 2010. "Iron Deficiency and Infection." *Indian Journal of Pediatrics* 77 (7): 789–93. https://doi.org/10.1007/s12098-010-0120-3.

Kuo, Tien Chueh, Tze Feng Tian, and Yufeng J. Tseng. 2013. "3Omics: A Web-Based Systems Biology Tool for Analysis, Integration and Visualization of Human Transcriptomic, Proteomic and Metabolomic Data." *BMC Systems Biology* 7 (1): 1. https://doi.org/10.1186/1752-0509-7-64.

Laney, Doug. 2001. "3D Data Management: Controlling Data Volume, Velocity, and Variety."

Lee, J. H., Y. G. Kim, and J. Lee. 2018. "Inhibition of Candida Albicans Biofilm and Hyphae Formation by Biocompatible Oligomers." *Letters in Applied Microbiology* 67 (2): 123–29. https://doi.org/10.1111/lam.13016.

Lehmann, C, S Islam, S Jarosch, J Zhou, D Hoskin, A Greenshields, N Al-Banna, et al. 2015. "The Utility of Iron Chelators in the Management of Inflammatory Disorders." *Mediators of Inflammation* 2015: 516740. https://doi.org/10.1155/2015/516740.

Lelandais, Gaëlle, Thomas Denecker, Camille Garcia, Nicolas Danila, Thibaut Léger, and Jean Michel Camadro. 2019a. "Label-Free Quantitative Proteomics in Candida Yeast Species: Technical and Biological Replicates to Assess Data Reproducibility." *BMC Research Notes* 12 (1): 470. https://doi.org/10.1186/s13104-019-4505-8.

———. 2019b. "Label-Free Quantitative Proteomics in Candida Yeast Species: Technical and Biological Replicates to Assess Data Reproducibility." *BMC Research Notes* 12 (1): 470. https://doi.org/10.1186/s13104-019-4505-8.

Lelandais, Gaëlle, Ivo Scheiber, Javier Paz-Yepes, Jean Claude Lozano, Hugo Botebol, Jana Pilátová, Vojtěch Žárský, et al. 2016. "Ostreococcus Tauri Is a New Model Green Alga for Studying Iron Metabolism in Eukaryotic Phytoplankton." *BMC Genomics* 17 (1): 319. https://doi.org/10.1186/s12864-016-2666-6.

Leonelli, Sabina. 2019. "The Challenges of Big Data Biology." *ELife* 8 (April). https://doi.org/10.7554/eLife.47381.

Leroy, Olivier, Sébastien Bailly, Jean Pierre Gangneux, Jean Paul Mira, Patrick Devos, Hervé Dupont, Philippe Montravers, et al. 2016. "Systemic Antifungal Therapy for Proven or Suspected Invasive Candidiasis: The AmarCAND 2 Study." *Annals of Intensive Care* 6 (1): 1–11. https://doi.org/10.1186/s13613-015-0103-7.

Leroy, Olivier, Jean Pierre Gangneux, Philippe Montravers, Jean Paul Mira, François Gouin, Jean Pierre Sollet, Jean Carlet, et al. 2009. "Epidemiology, Management, and Risk Factors

for Death of Invasive Candida Infections in Critical Care: A Multicenter, Prospective, Observational Study in France (2005-2006)." *Critical Care Medicine* 37 (5): 1612–18. https://doi.org/10.1097/CCM.0b013e31819efac0.

Leroy, Olivier, J Mira, P Montravers, J Gangneux, F Gouin, and J Sollet. 2008. "Invasive Candidiasis in ICU: Analysis of Antifungal Treatments in the French Study AmarCand." *Elsevier* 27: 999–1007. https://doi.org/10.1016/j.annfar.2008.10.004.

Lesne, Annick. 2009. "Biologie Des Systèmes - L'organisation Multiéchelle Des Systèmes Vivants." *Médecine/Sciences* 25 (6–7): 585–87. https://doi.org/10.1051/medsci/2009256-7585.

Letunic, Ivica, and Peer Bork. 2018. "20 Years of the SMART Protein Domain Annotation Resource." *Nucleic Acids Research* 46 (D1): D493–96. https://doi.org/10.1093/nar/gkx922.

Lin, Lin, Paul Pantapalangkoor, Brandon Tan, Kevin W. Bruhn, Tiffany Ho, Travis Nielsen, Eric P. Skaar, et al. 2014. "Transferrin Iron Starvation Therapy for Lethal Bacterial and Fungal Infections." *Journal of Infectious Diseases* 210 (2): 254–64. https://doi.org/10.1093/infdis/jiu049.

Lortholary, Olivier, Marie Desnos-Ollivier, Karine Sitbon, Arnaud Fontanet, Stéphane Bretagne, Françoise Dromer, C. Bouges-Michel, et al. 2011. "Recent Exposure to Caspofungin or Fluconazole Influences the Epidemiology of Candidemia: A Prospective Multicenter Study Involving 2,441 Patients." *Antimicrobial Agents and Chemotherapy* 55 (2): 532–38. https://doi.org/10.1128/AAC.01128-10.

Lu, Shin Yu. 2016. "Perception of Iron Deficiency from Oral Mucosa Alterations That Show a High Prevalence of Candida Infection." *Journal of the Formosan Medical Association* 115 (8): 619–27. https://doi.org/10.1016/j.jfma.2016.03.011.

Luo, G., L. P. Samaranayake, B. P.K. Cheung, and G. Tang. 2004. "Reverse Transcriptase Polymerase Chain Reaction (RT-PCR) Detection of HLP Gene Expression in Candida Glabrata and Its Possible Role in in Vitro Haemolysin Production." *APMIS* 112 (4–5): 283–90. https://doi.org/10.1111/j.1600-0463.2004.apm11204-0509.x.

Luo, G., L. P. Samaranayake, and J. Y.Y. Yau. 2001. "Candida Species Exhibit Differential in Vitro Hemolytic Activities." *Journal of Clinical Microbiology* 39 (8): 2971–74. https://doi.org/10.1128/JCM.39.8.2971-2974.2001.

Luscombe, N. M., D. Greenbaum, and M. Gerstein. 2001. "What Is Bioinformatics? A Proposed Definition and Overview of the Field." *Methods of Information in Medicine* 40 (4): 346–58. https://doi.org/10.1055/s-0038-1634431.

Magill, S. S., E. O'Leary, S. J. Janelle, D. L. Thompson, G. Dumyati, J. Nadle, L. E. Wilson, et al. 2018. "Changes in Prevalence of Health Care-Associated Infections in U.S. Hospitals." *New England Journal of Medicine* 379 (18): 1732–44. https://doi.org/10.1056/NEJMoa1801550.

Małek, Marianna, Paulina Mrowiec, Karolina Klesiewicz, Iwona Skiba-Kurek, Adrian Szczepański, Joanna Białecka, Iwona Żak, et al. 2018. "Prevalence of Human Pathogens of the Clade Nakaseomyces in a Culture Collection—the First Report on Candida Bracarensis in Poland." *Folia Microbiologica*, 307–12. https://doi.org/10.1007/s12223-018-0655-7.

Mbengue, Ababacar. 2010. “Faut-Il Brûler Les Tests de Signification Statistique ?” *Management (France)* 13 (2): 99–127. https://doi.org/10.3917/mana.132.0100.

McCall, Andrew D., Ruvini U. Pathirana, Aditi Prabhakar, Paul J. Cullen, and Mira Edgerton. 2019. “Candida Albicans Biofilm Development Is Governed by Cooperative Attachment and Adhesion Maintenance Proteins.” *Npj Biofilms and Microbiomes* 5 (1). https://doi.org/10.1038/s41522-019-0094-5.

McCandless, David. 2000. *Information Is Beautiful*. Collins.

———. 2009. *Visual Miscellaneum_ The Bestselling Classic, Revised and Updated_ A Colorful Guide to the World’s Most Consequential Trivia*. Harper Design.

———. 2014. *Knowledge Is Beautiful*.

McShane, Blakeley B., David Gal, Andrew Gelman, Christian Robert, and Jennifer L. Tackett. 2019. “Abandon Statistical Significance.” *American Statistician* 73 (sup1): 235–45. https://doi.org/10.1080/00031305.2018.1527253.

Mehta, Devang. 2019. “Highlight Negative Results to Improve Science.” *Nature*, October. https://doi.org/10.1038/d41586-019-02960-3.

Mendel, Gregor. 1865. “EXPERIMENTS IN PLANT HYBRIDIZATION (1865).” http://www.netspace.org./MendelWeb/.

Merhej, Jawad, Amandine Frigo, Stéphane Le Crom, Jean-Michel Camadro, Frédéric Devaux, and Gaëlle Lelandais. 2014. “BPeaks: A Bioinformatics Tool to Detect Transcription Factor Binding Sites from ChIPseq Data in Yeasts and Other Organisms with Small Genomes.” *Yeast* 31 (10): 375–91. https://doi.org/10.1002/yea.3031.

Meyer, Debra. 2006. “Iron Chelation as Therapy for HIV and Mycobacterium Tuberculosis Co-Infection Under Conditions of Iron Overload.” *Current Pharmaceutical Design* 12 (16): 1943–47. https://doi.org/10.2174/138161206777442164.

Ministère de l’enseignement supérieur de la recherche et de l’innovation. 2018. “Plan National Pour La Science Ouverte.”

Mitchell, Alex L., Teresa K. Attwood, Patricia C. Babbitt, Matthias Blum, Peer Bork, Alan Bridge, Shoshana D. Brown, et al. 2019. “InterPro in 2019: Improving Coverage, Classification and Access to Protein Sequence Annotations.” *Nucleic Acids Research* 47 (D1): D351–60. https://doi.org/10.1093/nar/gky1100.

Nagi, Minoru, Koichi Tanabe, Hironobu Nakayama, Keigo Ueno, Satoshi Yamagoe, Takashi Umeyama, Hideaki Ohno, and Yoshitsugu Miyazaki. 2016. “Iron-Depletion Promotes Mitophagy to Maintain Mitochondrial Integrity in Pathogenic Yeast Candida Glabrata.” *Autophagy* 12 (8): 1259–71. https://doi.org/10.1080/15548627.2016.1183080.

Nairz, Manfred, Andrea Schroll, Thomas Sonnweber, and Günter Weiss. 2010. “The Struggle for Iron - a Metal at the Host-Pathogen Interface.” *Cellular Microbiology* 12 (12): 1691–1702. https://doi.org/10.1111/j.1462-5822.2010.01529.x.

Nakano, Felipe Kenji, Mathias Lietaert, and Celine Vens. 2019. “Machine Learning for Discovering Missing or Wrong Protein Function Annotations.” *BMC Bioinformatics* 20 (1): 485. https://doi.org/10.1186/s12859-019-3060-6.

National academy of sciences, National academy of engeineering and Institute of Medicine.

2009. *Ensuring the Integrity, Accessibility, and Stewardship of Research Data in the Digital Age*. http://www.nap.edu/catalog.php?record_id=12615.

Nemeth, Elizabeta, Marie S. Tuttle, Julie Powelson, Michael D. Vaughn, Adriana Donovan, Diane Mc Vey Ward, Tomas Ganz, and Jerry Kaplan. 2004. "Hepcidin Regulates Cellular Iron Efflux by Binding to Ferroportin and Inducing Its Internalization." *Science* 306 (5704): 2090–93. https://doi.org/10.1126/science.1104742.

Nevitt, Tracy, and Dennis J. Thiele. 2011. "Host Iron Withholding Demands Siderophore Utilization for Candida Glabrata to Survive Macrophage Killing." *PLoS Pathogens* 7 (3): e1001322. https://doi.org/10.1371/journal.ppat.1001322.

Nichio, Bruno T.L., Jeroniza Nunes Marchaukoski, and Roberto Tadeu Raittz. 2017. "New Tools in Orthology Analysis: A Brief Review of Promising Perspectives." *Frontiers in Genetics* 8 (OCT): 1–12. https://doi.org/10.3389/fgene.2017.00165.

O'Leary, Nuala A., Mathew W. Wright, J. Rodney Brister, Stacy Ciufo, Diana Haddad, Rich McVeigh, Bhanu Rajput, et al. 2016. "Reference Sequence (RefSeq) Database at NCBI: Current Status, Taxonomic Expansion, and Functional Annotation." *Nucleic Acids Research* 44 (D1): D733–45. https://doi.org/10.1093/nar/gkv1189.

OECD. 2007. "Principles and Guidelines for Access to Research Data from Public Funding."

Ogurtsov, Aleksey Y., Gelio Alves, and Yi Kuo Yu. 2019. "RAId: Knowledge-Integrated Proteomics Web Service with Accurate Statistical Significance Assignment." *Proteomics* 19 (14): 1–6. https://doi.org/10.1002/pmic.201800367.

Ooms, Jeroen. 2014. "The Jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects." *ArXiv:1403.2805 [Stat.CO]*. https://arxiv.org/abs/1403.2805.

Otasek, David, John H. Morris, Jorge Bouças, Alexander R. Pico, and Barry Demchak. 2019. "Cytoscape Automation: Empowering Workflow-Based Network Analysis." *Genome Biology* 20 (1): 1–15. https://doi.org/10.1186/s13059-019-1758-4.

Oveland, Eystein, Thilo Muth, Erdmann Rapp, Lennart Martens, Frode S. Berven, and Harald Barsnes. 2015. "Viewing the Proteome: How to Visualize Proteomics Data?" *Proteomics* 15 (8): 1341–55. https://doi.org/10.1002/pmic.201400412.

Peck, Roxy, Chris Olsen, and Jay L. Devore. 2016. *Introduction to Statistics and Data Analysis*. *Technometrics*. https://doi.org/10.1198/tech.2002.s664.

Pereira, Cécile, Alain Denise, and Olivier Lespinet. 2014. "A Meta-Approach for Improving the Prediction and the Functional Annotation of Ortholog Groups." *BMC Genomics* 15 (6): 1–8. https://doi.org/10.1186/1471-2164-15-S6-S16.

Perlroth, Joshua, Bryan Choi, and Brad Spellberg. 2007. "Nosocomial Fungal Infections: Epidemiology, Diagnosis, and Treatment." *Medical Mycology*. https://doi.org/10.1080/13693780701218689.

Perrier, Victor, Fanny Meyer, and David Granjon. 2020. "ShinyWidgets: Custom Inputs Widgets for Shiny." https://cran.r-project.org/package=shinyWidgets.

Pevsner, EJonathan. 2015. *Bioinformatics and Functional Genomics*. *Briefings in Functional Genomics and Proteomics*.

Playfair, William. 1801. *THE COMMERCIAL AND POLITICAL ATLAS Representing, by*

Means of STAINED COPPER-PLATE CHARTS, THE PROGRESS OF THE COMMERCE, REVENUES, EXPENDITURE, AND DEBTS OF ENGLAND, DURING THE WHOLE OF THE EIGHTEENTH CENTURY.

Plesser, Hans E. 2018. "Reproducibility vs. Replicability: A Brief History of a Confused Terminology." *Frontiers in Neuroinformatics* 11 (January): 76. https://doi.org/10.3389/fninf.2017.00076.

Pons, Tirso, Luis A. Montero, and Juan P. Febles. 2007. "Computational Biology in Cuba: An Opportunity to Promote Science in a Developing Country." *PLoS Computational Biology* 3 (11): e227. https://doi.org/10.1371/journal.pcbi.0030227.

Prakash, Amol, Brian Piening, Jeff Whiteaker, Heidi Zhang, Scott A. Shaffer, Daniel Martin, Laura Hohmann, et al. 2007. "Assessing Bias in Experiment Design for Large Scale Mass Spectrometry-Based Quantitative Proteomics." *Molecular and Cellular Proteomics* 6 (10): 1741–48. https://doi.org/10.1074/mcp.M600470-MCP200.

Prasad, Tulika, Aparna Chandra, Chinmay K. Mukhopadhyay, and Rajendra Prasad. 2006. "Unexpected Link between Iron and Drug Resistance of Candida Spp.: Iron Depletion Enhances Membrane Fluidity and Drug Diffusion, Leading to Drug-Susceptible Cells." *Antimicrobial Agents and Chemotherapy* 50 (11): 3597–3606. https://doi.org/10.1128/AAC.00653-06.

Puri, Sumant, Rohitashw Kumar, Isolde G. Rojas, Ornella Salvatori, and Mira Edgerton. 2019. "Iron Chelator Deferasirox Reduces Candida Albicans Invasion of Oral Epithelial Cells and Infection Levels in Murine Oropharyngeal Candidiasis." *Antimicrobial Agents and Chemotherapy* 63 (4). https://doi.org/10.1128/AAC.02152-18.

Quotidien, A U. 2019. "LA FACE CACHÉE DU NUMÉRIQUE." www.ademe.fr/contact.

Reinsel, David, John Gantz, and John Rydning. 2018. "The Digitization of the World - From Edge to Core." *IDC White Paper*, no. November: US44413318. https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf.

Reproducibility and Replicability in Science. 2019. *Reproducibility and Replicability in Science*. National Academies Press. https://doi.org/10.17226/25303.

Rhee, Seung Yon. 2005. "Bioinformatics. Current Limitations and Insights for the Future." *Plant Physiology*. American Society of Plant Biologists. https://doi.org/10.1104/pp.104.900153.

Richards, Alicia L., Alexander S. Hebert, Arne Ulbrich, Derek J. Bailey, Emma E. Coughlin, Michael S. Westphall, and Joshua J. Coon. 2015. "One-Hour Proteome Analysis in Yeast." *Nature Protocols* 10 (5): 701–14. https://doi.org/10.1038/nprot.2015.040.

Ritchie, Matthew E., Belinda Phipson, Di Wu, Yifang Hu, Charity W. Law, Wei Shi, and Gordon K. Smyth. 2015. "Limma Powers Differential Expression Analyses for RNA-Sequencing and Microarray Studies." *Nucleic Acids Research* 43 (7): e47. https://doi.org/10.1093/nar/gkv007.

Roetzer, Andreas, Toni Gabaldón, and Christoph Schüller. 2011. "From Saccharomyces Cerevisiae to Candida Glabrata in a Few Easy Steps: Important Adaptations for an Opportunistic Pathogen." *FEMS Microbiology Letters* 314 (1): 1–9. https://doi.org/10.1111/j.1574-6968.2010.02102.x.

Rogozin, Igor B., David Managadze, Svetlana A. Shabalina, and Eugene V. Koonin. 2014. "Gene Family Level Comparative Analysis of Gene Expression n Mammals Validates the Ortholog Conjecture." *Genome Biology and Evolution* 6 (4): 754–62. https://doi.org/10.1093/gbe/evu051.

Rose, Alexander S., Anthony R. Bradley, Yana Valasatava, Jose M. Duarte, Andreas Prlic, and Peter W. Rose. 2018. "NGL Viewer: Web-Based Molecular Graphics for Large Complexes." *Bioinformatics* 34 (21): 3755–58. https://doi.org/10.1093/bioinformatics/bty419.

Rose, Alexander S., and Peter W. Hildebrand. 2015. "NGL Viewer: A Web Application for Molecular Visualization." *Nucleic Acids Research* 43 (W1): W576–79. https://doi.org/10.1093/nar/gkv402.

Rosling, Hans, Ola Rosling, and Anna Rosling Rönnlund. 2018. *Factfulness : Ten Reasons We're Wrong About the World and Why Things Are Better Than You Think*. *Sceptre*.

Rossoni, Rodnei Dennis, Júnia Oliveira Barbosa, Simone Furgeri Godinho Vilela, Antonio Olavo Cardoso Jorge, and Juliana Campos Junqueira. 2013. "Comparison of the Hemolytic Activity between C. Albicans and Non-Albicans Candida Species." *Brazilian Oral Research* 27 (6): 484–89. https://doi.org/10.1590/S1806-83242013000600007.

RR, Sarkar. 2016. "The Big Data Deluge in Biology: Challenges and Solutions." *Journal of Informatics and Data Mining* 1 (2). https://doi.org/10.21767/2472-1956.100014.

Sali, Andras, and Dean Attali. 2020. "Shinycssloaders: Add CSS Loading Animations to 'shiny' Outputs." https://cran.r-project.org/package=shinycssloaders.

Samuel, A. L. 1959. "Some Studies in Machine Learning Using the Game of Checkers. II-Recent Progress." *Annual Review in Automatic Programming* 6 (PART 1): 1–36. https://doi.org/10.1016/0066-4138(69)90004-4.

Santos, Ana L., and Ariel B. Lindner. 2017. "Protein Posttranslational Modifications: Roles in Aging and Age-Related Disease." *Oxidative Medicine and Cellular Longevity* 2017. https://doi.org/10.1155/2017/5716409.

Santos, Giselle C.de Oliveira, Cleydlenne C. Vasconcelos, Alberto J.O. Lopes, Maria do S.de Sousa Cartágenes, Allan K.D.B. Filho, Flávia R.F. do Nascimento, Ricardo M. Ramos, et al. 2018. "Candida Infections and Therapeutic Strategies: Mechanisms of Action for Traditional and Alternative Agents." *Frontiers in Microbiology*. Frontiers Media S.A. https://doi.org/10.3389/fmicb.2018.01351.

Sasse, Christoph, Mike Hasenberg, Michael Weyler, Matthias Gunzer, and Joachim Morschhäuser. 2013. "White-Opaque Switching of Candida Albicans Allows Immune Evasion in an Environment-Dependent Fashion." *Eukaryotic Cell* 12 (1): 50–58. https://doi.org/10.1128/EC.00266-12.

Savage, Kimberley A., Maria del Carmen Parquet, David S. Allan, Ross J. Davidson, Bruce E. Holbein, Elizabeth A. Lilly, and Paul L. Fidel. 2018. "Iron Restriction to Clinical Isolates of Candida Albicans by the Novel Chelator Dibi Inhibits Growth and Increases Sensitivity to Azoles in Vitro and in Vivo in a Murine Model of Experimental Vaginitis." *Antimicrobial Agents and Chemotherapy* 62 (8). https://doi.org/10.1128/AAC.02576-17.

Schnoes, Alexandra M., Shoshana D. Brown, Igor Dodevski, and Patricia C. Babbitt. 2009. "Annotation Error in Public Databases: Misannotation of Molecular Function in Enzyme

Superfamilies." *PLoS Computational Biology* 5 (12). https://doi.org/10.1371/journal.pcbi.1000605.

Schwarzmüller, Tobias, Biao Ma, Ekkehard Hiller, Fabian Istel, Michael Tscherner, Sascha Brunke, Lauren Ames, et al. 2014. "Systematic Phenotyping of a Large-Scale Candida Glabrata Deletion Collection Reveals Novel Antifungal Tolerance Genes." *PLoS Pathogens* 10 (6). https://doi.org/10.1371/journal.ppat.1004211.

Scientific Data. 2017. "Recommended Data Repositories : Scientific Data." Scientific Data. 2017. http://www.nature.com/sdata/policies/repositories.

Seider, Katja, Franziska Gerwien, Lydia Kasper, Stefanie Allert, Sascha Brunke, Nadja Jablonowski, Tobias Schwarzmüller, et al. 2014. "Immune Evasion, Stress Resistance, and Efficient Nutrient Acquisition Are Crucial for Intracellular Survival of Candida Glabrata within Macrophages." *Eukaryotic Cell* 13 (1): 170–83. https://doi.org/10.1128/EC.00262-13.

Seneviratne, Chaminda J., Suhasini Rajan, Sarah S.W. Wong, Dominic N.C. Tsang, Christopher K.C. Lai, Lakshman P. Samaranayake, and Lijian Jin. 2016. "Antifungal Susceptibility in Serum and Virulence Determinants of Candida Bloodstream Isolates from Hong Kong." *Frontiers in Microbiology* 7 (FEB). https://doi.org/10.3389/fmicb.2016.00216.

Shannon, Paul, Andrew Markiel, Owen Ozier, Nitin S. Baliga, Jonathan T. Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. 2003. "Cytoscape: A Software Environment for Integrated Models." *Genome Research*. https://doi.org/10.1101/gr.1239303.metabolite.

Sharma, Vandana, Rajaram Purushotham, and Rupinder Kaur. 2016. "The Phosphoinositide 3-Kinase Regulates Retrograde Trafficking of the Iron Permease CgFtr1 and Iron Homeostasis in Candida Glabrata." *Journal of Biological Chemistry* 291 (47): 24715–34. https://doi.org/10.1074/jbc.M116.751529.

Shen-Orr, Shai S., Ron Milo, Shmoolik Mangan, and Uri Alon. 2002. "Network Motifs in the Transcriptional Regulation Network of Escherichia Coli." *Nature Genetics* 31 (1): 64–68. https://doi.org/10.1038/ng881.

Sievert, Carson. 2018. "Plotly for R." https://plotly-r.com.

Silvia, Paul. 2007. *How to Write a Lot - A Pratical Guide to Productive Academic Writing*. *American Psychological Association*.

Singh, Vijender, Indranil Sinha, and Parag P. Sadhale. 2005. "Global Analysis of Altered Gene Expression during Morphogenesis of Candida Albicans in Vitro." *Biochemical and Biophysical Research Communications* 334 (4): 1149–58. https://doi.org/10.1016/j.bbrc.2005.07.018.

Skrzypek, Marek S, Jonathan Binkley, Gail Binkley, Stuart R Miyasato, Matt Simison, Gavin Sherlock, The Candida, and Genome Database. 2017. "The Candida Genome Database (CGD): Incorporation of Assembly 22 , Systematic Identifiers and Visualization of High Throughput Sequencing Data." *Nucleic Acids Research* 45 (October 2016): 592–96. https://doi.org/10.1093/nar/gkw924.

Slowikowski, Kamil. 2020. "Ggrepel: Automatically Position Non-Overlapping Text Labels with 'Ggplot2.'" https://cran.r-project.org/package=ggrepel.

Sonnhammer, Erik L.L., Toni Gabaldon, Alan W. Sousa Da Silva, Maria Martin, Marc Robinson-Rechavi, Brigitte Boeckmann, Paul D. Thomas, and Christophe Dessimoz. 2014. "Big Data and Other Challenges in the Quest for Orthologs." *Bioinformatics* 30 (21): 2993–98. https://doi.org/10.1093/bioinformatics/btu492.

Soon, Wendy Weijia, Manoj Hariharan, and Michael P. Snyder. 2013. "High-Throughput Sequencing for Biology and Medicine." *Molecular Systems Biology* 9 (640): 1–14. https://doi.org/10.1038/msb.2012.61.

Srivastava, Vivek Kumar, Korivi Jyothiraj Suneetha, and Rupinder Kaur. 2014. "A Systematic Analysis Reveals an Essential Role for High-Affinity Iron Uptake System, Haemolysin and CFEM Domain-Containing Protein in Iron Homoeostasis and Virulence in Candida Glabrata." *Biochemical Journal* 463 (1): 103–14. https://doi.org/10.1042/BJ20140598.

Stelzer, Gil, Naomi Rosen, Inbar Plaschkes, Shahar Zimmerman, Michal Twik, Simon Fishilevich, Tsippi Iny Stein, et al. 2016. "The GeneCards Suite: From Gene Data Mining to Disease Genome Sequence Analyses." *Current Protocols in Bioinformatics* 2016 (June): 1.30.1-1.30.33. https://doi.org/10.1002/cpbi.5.

Sudbery, Peter E. 2011. "Growth of Candida Albicans Hyphae." *Nature Reviews Microbiology*. Nature Publishing Group. https://doi.org/10.1038/nrmicro2636.

Tarca, Adi L., Roberto Romero, and Sorin Draghici. 2006. "Analysis of Microarray Experiments of Gene Expression Profiling." *American Journal of Obstetrics and Gynecology* 195 (2): 373–88. https://doi.org/10.1016/j.ajog.2006.07.001.

Tati, Swetha, Peter Davidow, Andrew McCall, Elizabeth Hwang-Wong, Isolde G. Rojas, Brendan Cormack, and Mira Edgerton. 2016. "Candida Glabrata Binding to Candida Albicans Hyphae Enables Its Development in Oropharyngeal Candidiasis." Edited by Mairi C Noverr. *PLOS Pathogens* 12 (3): e1005522. https://doi.org/10.1371/journal.ppat.1005522.

Thermo Fisher Scientific. 2015. "Protein Expression Handbook Recombinant Protein Expression and Purification Technologies." *Educational Purposes Only.*, 112.

Thiébaut, Antonin, Thierry Delaveau, Médine Benchouaia, Julia Boeri, Mathilde Garcia, Gaëlle Lelandais, and Frédéric Devaux. 2017. "The CCAAT-Binding Complex Controls Respiratory Gene Expression and Iron Homeostasis in Candida Glabrata." *Scientific Reports* 7 (1). https://doi.org/10.1038/s41598-017-03750-5.

Thompson, Mitchell G., Brendan W. Corey, Yuanzheng Si, David W. Craft, and Daniel V. Zurawski. 2012. "Antibacterial Activities of Iron Chelators against Common Nosocomial Pathogens." *Antimicrobial Agents and Chemotherapy* 56 (10): 5419–21. https://doi.org/10.1128/AAC.01197-12.

Touati, Danièle. 2000. "Iron and Oxidative Stress in Bacteria." *Archives of Biochemistry and Biophysics* 373 (1): 1–6. https://doi.org/10.1006/abbi.1999.1518.

Tufte, Edward R. 2001. *The Visual Display of Quantitative Information*. *Graphics Press*. https://doi.org/10.1075/idj.4.3.12cos.

Tugui, Alexandru, Daniela Danciulescu, and Mihaela Simona Subtirelu. 2019. "The Biological as a Double Limit for Artificial Intelligence: Review and Futuristic Debate." *International Journal of Computers, Communications and Control* 14 (2): 253–71. https://doi.org/10.15837/ijccc.2019.2.3536.

Underhill, David M., and Iliyan D. Iliev. 2014. “The Mycobiota: Interactions between Commensal Fungi and the Host Immune System.” *Nature Reviews Immunology* 14 (6): 405–16. https://doi.org/10.1038/nri3684.

Velliyagounder, K., W. Alsaedi, W. Alabdulmohsen, K. Markowitz, and D. H. Fine. 2015. “Oral Lactoferrin Protects against Experimental Candidiasis in Mice.” *Journal of Applied Microbiology* 118 (1): 212–21. https://doi.org/10.1111/jam.12666.

Walsh, Christopher T., Sylvie Garneau-Tsodikova, and Gregory J. Gatto. 2005. “Protein Posttranslational Modifications: The Chemistry of Proteome Diversifications.” *Angewandte Chemie - International Edition* 44 (45): 7342–72. https://doi.org/10.1002/anie.200501023.

Warren, Thomas A., Lisa McTaggart, Susan E. Richardson, and Sean X. Zhang. 2010. “Candida Bracarensis Bloodstream Infection in an Immunocompromised Patient.” *Journal of Clinical Microbiology* 48 (12): 4677–79. https://doi.org/10.1128/JCM.01447-10.

Webb, Sarah. 2018. “Deep Learning for Biology.” *Nature* 554 (7693): 555–57. https://doi.org/10.1038/d41586-018-02174-z.

Wickham, Hadley. 2007. “Reshaping Data with the {reshape} Package.” *Journal of Statistical Software* 21 (12): 1–20. http://www.jstatsoft.org/v21/i12/.

———. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. https://ggplot2.tidyverse.org.

———. 2019. “Httr: Tools for Working with URLs and HTTP.” https://cran.r-project.org/package=httr.

Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2020. “Dplyr: A Grammar of Data Manipulation.” https://cran.r-project.org/package=dplyr.

Wickham, Hadley, James Hester, and Jeroen Ooms. 2018. “Xml2: Parse XML.” https://cran.r-project.org/package=xml2.

Wilkinson, Mark D., Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, et al. 2016. “Comment: The FAIR Guiding Principles for Scientific Data Management and Stewardship.” *Scientific Data* 3 (1): 1–9. https://doi.org/10.1038/sdata.2016.18.

Wilkinson, Mark D. 2016. “Comment: The FAIR Guiding Principles for Scientific Data Management and Stewardship.” *Nature Publishing Group*. https://doi.org/10.1038/sdata.2016.18.

Wisplinghoff, H., T. Bischoff, S. M. Tallent, H. Seifert, R. P. Wenzel, and M. B. Edmond. 2004. “Nosocomial Bloodstream Infections in US Hospitals: Analysis of 24,179 Cases from a Prospective Nationwide Surveillance Study.” *Clinical Infectious Diseases* 39 (3): 309–17. https://doi.org/10.1086/421946.

World Health Organization. 2002. “Genomics and World Health.”

Xie, Yihui, Joe Cheng, and Xianying Tan. 2019. “DT: A Wrapper of the JavaScript Library ‘DataTables.’” https://cran.r-project.org/package=DT.

Xu, Xiaobin, Yu Huang, Hao Pan, Rosalynn Molden, Haibo Qiu, Thomas J. Daly, and Ning Li. 2019. “Quantitation and Modeling of Posttranslational Modifications in a Therapeutic

Monoclonal Antibody from Single- And Multiple-Dose Monkey Pharmacokinetic Studies Using Mass Spectrometry." *PLoS ONE* 14 (10): 1–26. https://doi.org/10.1371/journal.pone.0223899.

Yates, Andrew D., Premanand Achuthan, Wasiu Akanni, James Allen, Jamie Allen, Jorge Alvarez-Jarreta, M. Ridwan Amode, et al. 2020. "Ensembl 2020." *Nucleic Acids Research* 48 (D1): D682–88. https://doi.org/10.1093/nar/gkz966.

Zahn-Zabal, Monique, Pierre André Michel, Alain Gateau, Frédéric Nikitin, Mathieu Schaeffer, Estelle Audot, Pascale Gaudet, et al. 2020. "The NeXtProt Knowledgebase in 2020: Data, Tools and Usability Improvements." *Nucleic Acids Research* 48 (D1): D328–34. https://doi.org/10.1093/nar/gkz995.

Zaken, Ministerie van Buitenlandse. 2016. "Amsterdam Call for Action on Open Science - Publication - EU2016.Nl." http://english.eu2016.nl/documents/reports/2016/04/04/amsterdam-call-for-action-on-open-science.

Zhou, Jian, Chandra L. Theesfeld, Kevin Yao, Kathleen M. Chen, Aaron K. Wong, and Olga G. Troyanskaya. 2018. "Deep Learning Sequence-Based Ab Initio Prediction of Variant Effects on Expression and Disease Risk." *Nature Genetics* 50 (8): 1171–79. https://doi.org/10.1038/s41588-018-0160-6.

Ziemann, Mark, Yotam Eren, and Assam El-Osta. 2016. "Gene Name Errors Are Widespread in the Scientific Literature." *Genome Biology*. BioMed Central Ltd. https://doi.org/10.1186/s13059-016-1044-7.

Zipf, George Kingsley. 1935. "The Psycho-Biology of Language; an Introduction to Dynamic Philology." Boston: Houghton Mifflin company. file://catalog.hathitrust.org/Record/000359461.

université PARIS-SACLAY

Ecole doctorale n° 577

Structure et dynamique des systèmes vivants (SDSV)

Titre : Bioinformatique et analyse de données multiomiques : principes et applications chez les levures pathogènes *Candida glabrata* et *Candida albicans*

Mots clés : Bioinformatique ; analyse de données ; Technologies multi-omiques ; levures *Candida* ; Génomique fonctionnelle

Résumé :

Plusieurs évolutions sont constatées dans la recherche en biologie. Tout d'abord, les études menées reposent souvent sur des approches expérimentales quantitatives. L'analyse et l'interprétation des résultats requièrent l'utilisation de l'informatique et des statistiques. Également, en complément des études centrées sur des objets biologiques isolés, les technologies expérimentales haut débit permettent l'étude des systèmes (caractérisation des composants du système ainsi que des interactions entre ces composants). De très grandes quantités de données sont disponibles dans les bases de données publiques, librement réutilisables pour de nouvelles problématiques. Enfin, les données utiles pour les recherches en biologie sont très hétérogènes (données numériques, de textes, images, séquences biologiques, etc.) et conservées sur des supports d'information également très hétérogènes (papiers ou numériques). Ainsi « l'analyse de données » s'est petit à petit imposée comme une problématique de recherche à part entière et en seulement une dizaine d'années, le domaine de la « Bioinformatique » s'est en conséquence totalement réinventé.

Disposer d'une grande quantité de données pour répondre à un questionnement biologique n'est souvent pas le défi principal. La vraie difficulté est la capacité des chercheurs à convertir les données en information, puis en connaissance. Dans ce contexte, plusieurs problématiques de recherche en biologie ont été abordées lors de cette thèse. La première concerne l'étude de l'homéostasie du fer chez la levure pathogène *Candida glabrata*. La seconde concerne l'étude systématique des modifications post-traductionnelles des protéines chez la levure pathogène *Candida albicans*. Pour ces deux projets, des données « omiques » ont été exploitées : transcriptomiques et protéomiques. Des outils bioinformatiques et des outils d'analyses ont été implémentés en parallèle conduisant à l'émergence de nouvelles hypothèses de recherche en biologie. Une attention particulière et constante a aussi été portée sur les problématiques de reproductibilité et de partage des résultats avec la communauté scientifique.

Title: Bioinformatics and multiomics data analysis: principles and applications in the pathogenic yeast species *Candida glabrata* and *Candida albicans*

Keywords: Bioinformatics ; data analysis ; Multi-omics technologies ; Yeasts species Candida ; Functional genomics

Abstract:

Biological research is changing. First, studies are often based on quantitative experimental approaches. The analysis and the interpretation of the obtained results thus need computer science and statistics. Also, together with studies focused on isolated biological objects, high throughput experimental technologies allow to capture the functioning of biological systems (identification of components as well as the interactions between them). Very large amounts of data are also available in public databases, freely reusable to solve new open questions. Finally, the data in biological research are heterogeneous (digital data, texts, images, biological sequences, etc.) and stored on multiple supports (paper or digital). Thus, "data analysis" has gradually emerged as a key research issue, and in only ten years, the field of "Bioinformatics" has been significantly changed.

Having a large amount of data to answer a biological question is often not the main challenge. The real challenge is the ability of researchers to convert the data into information and then into knowledge. In this context, several biological research projects were addressed in this thesis. The first concerns the study of iron homeostasis in the pathogenic yeast Candida glabrata. The second concerns the systematic investigation of post-translational modifications of proteins in the pathogenic yeast Candida albicans. In these two projects, omics data were used: transcriptomics and proteomics. Appropriate bioinformatics and analysis tools were developed, leading to the emergence of new research hypotheses. Particular and constant attention has also been paid to the question of data reproducibility and sharing of results with the scientific community.