HAL open science

Développements d'algorithmes d'apprentissage machine pour l'analyse automatique de données biophysiques de haute résolution

Laura Duciel

▶ To cite this version:

Laura Duciel. Développements d'algorithmes d'apprentissage machine pour l'analyse automatique de données biophysiques de haute résolution. Biotechnologie. Université de Strasbourg, 2022. Français. NNT : 2022STRAJ065 . tel-04027801

HAL Id: tel-04027801

https://theses.hal.science/tel-04027801v1

Submitted on 14 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE STRASBOURG

ÉCOLE DOCTORALE des Sciences de la Vie et de la Santé

UMR 7104 - U 1258 IGBMC

THÈSE présentée par :

Laura DUCIEL

Soutenue le : **23 Septembre 2022**

Pour obtenir le grade de : **Docteur de l'Université de Strasbourg**

Discipline/ Spécialité : Sciences de la Vie et de la Santé / Biotechnologies

Développements d'algorithmes d'apprentissage machine pour l'analyse automatique de données biophysiques de haute résolution.

THÈSE dirigée par :

M. DELSUC Marc-André	Directeur de recherche, IGBMC, Université de Strasbourg

RAPPORTEURS :

Mme MALLIAVIN Thérèse	Directrice de recherche, LPCT, Université de Lorraine
M. VAN DER REST Guillaume	Directeur de recherche, LCP, Université Paris Sud

EXAMINATEURS :

Mme CARAPITO Christine	Chargée de recherche, IPHC, Université de Strasbourg

MEMBRES INVITES :

Mme BRIOT-DIETSCH Anne	Docteur, CASC4DE
M. KIEFFER Bruno	Directeur de recherche, IGBMC, Université de Strasbourg

REMERCIEMENTS

Ces années de doctorat en CIFRE ont été réalisées au sein de l'entreprise CASC4DE et du laboratoire de recherche IGBMC sous la direction de Marc-André Delsuc. Je tiens donc à exprimer ma gratitude à M. Bruno Kieffer, chef d'équipe à l'IGBMC, pour son accueil dans l'équipe de recherche, ainsi qu'à toutes les personnes ayant permis cette collaboration et mon intégration dans de très bonnes conditions dans l'entreprise CASC4DE, en particulier Aude Billon, Julia Asencio Hernández et Anne Briot-Dietsch.

Mes plus vifs remerciements vont à Marc-André qui m'a offert l'opportunité de réaliser ce travail et de porter ce projet scientifique innovant et intéressant au cours duquel j'ai beaucoup appris. Je le remercie pour ses nombreux conseils, les échanges enrichissants, le temps accordé et la confiance apportée tout au long de ces années de thèse.

J'adresse également mes remerciements aux membres de mon jury de thèse d'avoir accepté d'en faire partie pour évaluer mon travail. Je remercie Mme Thérèse Malliavin, Directrice de recherche au LPCT de l'Université de Lorraine et M. Guillaume Van Der Rest, directeur de recherche au LCP de l'Université Paris Sud d'avoir accepté d'être rapporteurs de mon manuscrit. Je remercie également Mme Christine Carapito, chargée de recherche à l'IPHC de l'Université de Strasbourg et M. Bruno KIEFFER, directeur de recherche à l'IGBMC de l'Université de Strasbourg d'avoir suivi mon travail en tant que membres du comité de suivi de thèse et d'accepter d'être membres du jury.

Je tiens également à remercier toutes les personnes de l'équipe de recherche et de CASC4DE pour leur aide et pour m'avoir accompagnée au cours de cette thèse, et plus particulièrement Afef, Anne, Camille et Julia.

La réalisation de certains des projets de cette thèse a été rendue possible grâce aux financements octroyés par l'ANR « ANR-18-CE44-0009-01 » et le programme de recherche et d'innovation H2020 de l'UE « INFRAIA-02-2017 », que je remercie.

Je remercie toute ma famille et en particulier mes parents ainsi que mes frères et sœur pour leurs encouragements, leur soutien et leur compréhension.

Enfin, toute mon affection et ma gratitude vont à Luis, l'homme qui partage ma vie, pour son support et son soutien sans faille au quotidien, d'autant plus au cours de ces derniers mois de rédaction. Merci de croire en moi et d'être toujours là, pour partager les bons comme les mauvais moments de la vie.

Merci à tous pour ces années de thèse, sans vous, rien de tout cela n'aurait été possible.

A mes parents Isabelle et Thierry,

A mes frères et ma sœur Lucas, Romain et Hélène,

A Luis,

TABLE DES MATIERES

Remerciements 3
Table des matières 7
Abréviations 9
Liste des figures 11
Liste des tables 15
Introduction 17
I. Problème des Big Data en biophysique et solutions apportées par le Machine et Deep Learning (ML & DL) . 17
II. Procédures classiques en Machine Learning 22
Choix des données 22
Préparation des données 23
Séparation des jeux de données 24
Modèles 26
Implémentation et optimisation 29
Evaluation du modèle 31
III. Méthodes générales utilisées au cours de la thèse 34
Python 34
SPIKE 35
Pandas 36
Scikit-Learn 37
Keras 38
Partie 1 – Application des techniques de ML à la Résonance Magnétique Nucléaire (RMN) 39
I. Projet Plasmodesma – Déconvolution pharmacophorique de substances naturelles 39
Introduction 39
Matériel et Méthode 42
Résultats 50
Conclusions et Perspectives 54
II. Projet Fluovial – Détection de polluants fluorés 56
Introduction 56
Projet 58

Matériel et Méthode 59
Résultats 73
Conclusions et Perspectives 75
III. Projet Rescue 3 – Attributions spectrales de protéines 77
Projet 77
Matériel et méthode 79
Résultats 93
Conclusions et Perspectives 99
Partie 2 – Application des techniques de ML à la Spectrométrie de Masse (MS) 103
I. Projet Européen Horizon 2020 « EU FTICR-MS » 106
Le projet 106
Métadonnées 107
II. Fouille des données du projet EU FT-ICR MS 114
Introduction 114
Matériel et Méthodes 119
Résultats 121
Conclusions et Perspectives 130
III. Déconvolution MS algorithmique 132
Historique et objectifs 132
Données et matériel 136
Méthodes 138
Résultats 141
Conclusions et Perspectives 144
Conclusions Générales 147
Références bibliographiques 151
Résumé 160
Résumé en anglais 160

ABREVIATIONS

ACP	*Analyse en Composante Principale*
API	*Interface de Programmation d'Application*
AUC	*Area Under Curve*
BMRB	*Biological Magnetic Resonance data Bank*
CNN	*Convolutional Neural Network*
COSY	*Correlated Spectroscopy*
CPU	*Central Processing Unit*
DDA	*Acquisition Dépendante des Données*
DIA	*Acquisition Indépendante des Données*
DL	*Deep Learning*
DMSO	*DiMethyl SulfOxide*
DOSY	*Diffusion-Ordered SpectroscopY*
DTD	*Définition de Type de Document*
ECD	*Electron-Capture Dissociation*
ESI	*Electrospray Ionization*
F.A.I.R.	*Findable Accessible Interoperable Reusable*
fid	*Free Induction Decay*
FTICR	*Fourier Transform Ion Cyclotron Resonance*
FTOH	*Fluorotelomer alcohol*
GC	*Gaz Chromatography*
GI	*Gini Index*
GNPS	*Global Natural Products Social Molecular Networking*
GPU	*Graphics Processing Unit*
GSH	*Glutathion*
HDF5	*Hierarchical Data Format 5*
HMBC	*Heteronuclear Multiple Bond Correlation*
HSQC	*Heteronuclear Single Quantum Correlation*
IA	*Intelligence Artificielle*
IC_{50}	*Concentration inhibitrice médiane*
IRMPD	*InfraRed MultiPhoton Dissociation*
k-NN	*K Nearest Neighbors*

LC	*Liquid Chromatography*
LOD	*Limit Of Detection*
LOQ	*Limit Of Quantification*
LR	*Linear Regression*
MALDI	*Matrix Assisted Laser Desorption Ionisation*
ML	*Machine Learning*
MS	*Mass Spectrometry/Spectrométrie de Masse*
NaN	*Not a Number*
NGS	*Séquençage de Nouvelle Génération*
NN	*Neural Network*
OvR	*One vs Rest*
PFAS	*Substances perfluoroalkylées*
PFBSF	*PerFluoroButane Sulfonyl Fluoride*
PFDA	*PerFluoroDecanoic Acid*
PFOA	*PerFluoroOctanoic Acid*
PFOS	*PerFluoroOctane Sulfonic acid*
PFTDA	*PerFluoroTetraDecanoic Acid*
PGD	*Plan de Gestion de Données*
PHATE	*Potential of Heat-diffusion for Affinity-based Trajectory Embedding*
POP	*Polluants Organiques Persistants*
PTFE	*PolyTetraFluoroEthylene*
ReLU	*Rectified Linear Unit*
RF	*Random Forest*
RFE	*Recursive Feature Elimination*
RMN	*Résonance Magnétique Nucléaire*
ROC	*Receiver Operating Characteristic*
SNR	*Signal to Noise Ratio*
SPIKE	*Spectrometry Processing Innovative Kernel*
SVM	*Support Vector Machine*
TFE	*TriFluoroEthanol*
TOCSY	*Total Correlation Spectroscopy*
t-SNE	*t-distributed stochastic neighbor embedding*
UVPD	*UltraViolet PhotoDissociation*
ZF	*Zero-Filling*

LISTE DES FIGURES

Figure 1 - Graphique d'aide à la sélection de modèles en fonction des besoins et des données disponibles. Source : Documentation Scikit-Learn 27

Figure 2 - Différentes topologies de réseaux de neurones « classiques » répertoriées par Fjodor van Veen de l'institut Asimov. 28

Figure 3 - Illustration d'une matrice de confusion. 31

Figure 4 - Illustration d'une courbe ROC. En gris est représentée l'aire sous la courbe. En bleu la courbe ROC elle-même. En vert une courbe ROC où l'AUC est égale à 0.5 et où l'algorithme est alors incapable de différencier une classe de l'autre. 33

Figure 5 - Illustration de la méthode Plasmodesma 41

Figure 6 - Les spectres RMN COSY des 6 fractions réalisées à partir de la série de données synthétiques complémentée en Artémisinine sont présentés sur le panneau de gauche. En vert la fraction avec la bioactivité mesurée la plus élevée, en rouge l'une de celle présentant la bioactivité la plus faible. A droite le spectre RMN COSY de la molécule de référence, l'artémisinine, responsable de la bioactivité du mélange. 43

Figure 7 – Les spectres RMN TOCSY des 6 fractions réalisées à partir de l'extrait brut d'écorce de quinquina sont présentés sur le panneau de gauche. En vert la fraction avec la bioactivité mesurée la plus élevée, en rouge l'une de celle présentant la bioactivité la plus faible. A droite les spectres RMN TOCSY des deux molécules de références attendues comme responsables de la bioactivité du mélange. 45

Figure 8 – Illustration schématique du bucketing (pointillés rouges) sur des spectres RMN 1D et 2D. Pour des raisons d'illustration et de lisibilité, les buckets dessinés ici sont bien plus gros qu'en réalité. 47

Figure 9 – Capture d'écran des résultats interactifs pour la série de données issue de l'extrait brut d'écorce de Cinchona. Est ici présenté la comparaison deux à deux de deux échantillons sélectionnés de la série (Fractions 3 et 6). Sur les deux panneaux du haut, les spectres seuls des deux échantillons sont affichés. Sur le panneau en bas à gauche est présenté le ratio des deux fractions (tâches violet foncé) ainsi que les références des molécules soupçonnées responsables de la bioactivité quinidine et cinchonine, tâches roses et vertes. 52

Figure 10 - Capture d'écran des résultats interactifs pour la série de données issue de l'extrait brut d'écorce de Cinchona. Est ici présenté l'analyse globale par RFE (à gauche) et RL (à droite) de la série d'échantillons. Les références sont indiquées en rose et vert sur les spectres reconstitués. Les tâches foncées correspondent ici à l'empreinte spectrale de la molécule bioactive. 53

Figure 11 - Illustration des différents artéfacts et bruit ajoutés synthétiquement dans les données disponibles. 62

Figure 12 - Résultats de l'application des méthodes de réduction de dimensionnalité ACP, t-SNE et PHATE sur les données de RMN des molécules fluorées représentées par différentes couleurs. La projection est réalisée sur une dimension en 2D en partant des données après bucketing et après une réduction de dimension par RandomForest régresseur, en réalisant (ligne haute) ou non (ligne basse) une normalisation des données au préalable. 65

Figure 13 - Ce graphique montre l'importance relative des caractéristiques des données utilisées pour classifier les spectres de molécules fluorées obtenue par une régression par Random Forest. Il n'est représenté ici

que le haut de cette liste. Le nom de chaque caractéristique est composé comme suit : [abréviation de la valeur statistique]bck[numéro du bucket dans le spectre]. 66

Figure 14 - Comparaison des taux de prédictions vraies de différents algorithmes de classification sur les données du projet FLUOVIAL. 67

Figure 15 - Exemple d'un arbre de décision issu d'une Random Forest appliquée aux données du projet FLUOVIAL. Cet arbre part de la valeur du bucket "kurtosisbck131" et classe les différentes données en fonction des valeurs consécutives de certains buckets. 69

Figure 16 - Illustration des recherches d'hyperparamètres aléatoire et par grille. En vert les jeux de paramètres testés, en gris les zones optimales des paramètres. L'étoile représente le jeu de paramètre qui serait selectionné dans chaque cas. Adapté de Bergstra et Bengio 2012. 72

Figure 17 - Matrices de confusion pour les algorithmes de (1) RF après recherche aléatoire des hyperparamètres, de (2) RF OvR sur les données du projet et (3) l'algorithme consensus entre RF après recherche aléatoire et RF OvR. 74

Figure 18 - Répartition des acides aminés 1. Naturellement (données récupérées à partir de la base de données Uniprot-TrEMBL du 25 mai 2022), 2. Au sein de la BMRB v2.0 et 3. Au sein de la base de données utilisée pour l'algorithme de prédiction Rescue3. 81

Figure 19 - Illustration d'une couche dense dans la structure d'un réseau de neurones artificiel. 89

Figure 20 – Graphe des résultats globalisés de chacun des scénarios testés, triés de la plus faible à la plus haute précision obtenue. 94

Figure 21 – Les matrices de confusion associées aux résultats de l'entraînement pour les différents scénarios évalués, présentant le taux de réussite pour les valeurs prédites par rapport aux valeurs réelles (plus on tend vers une diagonale foncée, meilleures sont les prédictions). 97

Figure 22 - Extrait de l'ontologie de spectrométrie de masse à droite dans un format web facilement lisible et accessible ici : https://www.ebi.ac.uk/ols/ontologies/ms. A gauche un extrait du format XML disponible entièrement à partir de cette adresse : https://raw.githubusercontent.com/HUPO-PSI/psi-ms-CV/master/psi-ms.owl. 109

Figure 23 - Capture d'écran du formulaire pour la génération de métadonnées de FT-ICR mis en place dans le cadre du projet H2020. 111

Figure 24 - Extrait de fichier « .meta » produit par le formulaire de génération de métadonnées mis en place. 112

Figure 25 - Spectre théorique de l'ion positif du Glutathion (cation GSH) obtenu avec la bibliothèque de simulation NeutronStar et sur la ligne du bas des zooms sur les massifs isotopiques 1.Cation GSH +1, 2.Cation GSH +2 et 3.Cation GSH +3. 115

Figure 26 - Fonction d'apodisation de Kaiser dite « décalée » pour différentes valeurs de β et maxi. 118

Figure 27 - Schéma de lecture d'un graphe violon en comparaison avec un diagramme en boîte. Sur chacun des diagrammes les informations de médiane et quartiles sont disponibles ainsi que le « minimum » et « maximum ». Les points aberrants sont également visibles à l'extérieur des points dits « minimum » et « maximum ». Généralement une valeur est considérée comme étant aberrante lorsqu'elle est en dehors de 1.5 fois l'intervalle interquartiles [Q1 – 1.5x(Q3-Q1) ; Q3 + 1.5x(Q3-Q1)]. 122

Figure 28 – Impact de la variation des paramètres Maxi et β associés aux apodisations sur (a) le SNR moyen, (b) la similarité de cosinus moyenne et (c) les distances cumulées entre les listes de pics après association par l'algorithme hongrois. 126

Figure 29 - Effet de l'application d'un zero-filling sur (a) la similarité de cosinus et (b) le rapport signal sur bruit (SNR). 127

Figure 30 - Effets des différents paramètres intrinsèques à l'acquisition sur, de gauche à droite : la similarité de cosinus, le rapport signal sur bruit (SNR) et la distance entre les listes de pics, après un traitement de données incluant zero-filling et apodisation. 1. Le type d'acquisition (Narrow et Broad Band) 2. Le champ magnétique B_0, 3. La taille du fid (en points), 4. La durée du fid (en secondes), et 5. Le nombre de scans (NS). 129

Figure 31 - Illustration de la distribution d'un motif isotopique pour l'exemple du peptide de l'ubiquitine "MQIFVKTLTGKTITLEVEPSDTIENVKAKIQDKEGIPPDQQRLIFAGKQLEDGRTLSDYNIQKESTLHLVLRLRGG" et dont la formule chimique associée est $C_{378}H_{629}N_{105}O_{118}S$. 133

Figure 32 - Zoom sur un motif isotopique observé en 2D FTICR-MS, issu ici du jeu de données d'extrait de levure complet. Les tâches (ou pics) sont séparées le long de chaque dimension par l'inverse de l'état de charge du précurseur z1 et du fragment z2. Le pic monoisotopique est la tâche en bas à gauche du motif..... 134

Figure 33 - Donnée de 2D FTICR sur un extrait de levure après digestion trypsique obtenu après transformée de Fourier et conversion en m/z. En bas, un zoom sur la zone d'intérêt porteuse d'information et sur laquelle l'analyse va se porter. 137

Figure 34 - Illustration de l'impact de la taille des morceaux d'analyse. En (1), chacun des morceaux fait 1024x256 points, en (2) les morceaux font 1536x256 points et en (3) 3072x256 points. 139

Figure 35 - Processus de déconvolution. a) une région zoomée issue de la donnée analysée, les étoiles indiquent les signaux des harmoniques repliées. b) la même région que dans a) représentée en 3D. c) la recherche des valeurs optimales en faisant varier les paramètres de largeur le long des deux axes, l'étoile indique la valeur utilisée pour l'ensemble de l'analyse ; d) le résultat de la déconvolution optimisée sur la même région pour les différentes paires (z1 ; z2). 140

Figure 36 - Résultat de la déconvolution complète après regroupement des différents morceaux pour la donnée 2D FTICR d'extrait de levure. a) Zoom sur la région d'intérêt de la donnée d'origine. b) Résultat d'un simple peak-picking sur la donnée d'origine avec un seuil de détection à 2E6 d'intensité. c) Résultat de la déconvolution après application de la procédure détaillée. Les différentes paires $z_1 - z_2$ sont superposées et différenciées par des couleurs différentes, en rouge : ($z1 = 1$, $z2 = 1$), en noir : ($z1 = 1$, $z2 = 2$), en vert : ($z1 = 2$, $z2 = 1$) et en bleu : ($z1 = 2$, $z2 = 2$). 143

Figure 37 - Zoom sur un motif isotopique 2D complexe, qui ne correspond pas à un modèle de motif diagonal. Image issue de: "Delsuc, M.-A.; Breuker, K.; van Agthoven, M.A. Phase Correction for Absorption Mode Two-Dimensional Mass Spectrometry. Molecules 2021, 26, 3388". 145

LISTE DES TABLES

Tableau 1 - Niveaux d'activité mesurés pour la série de donnée synthétique complémentée en Artémisinine.. 42

Tableau 2 - IC_{50} mesurées pour la série de données issue de l'extrait brut d'écorce de Cinchona. Les valeurs d'IC_{50} pour 5 molécules de références sont également indiquées..... 44

Tableau 3 - Tailles des buckets 1H et 13C ayant été testées pour l'optimisation du traitement par bucketing sur les données synthétiquement complémentées en artémisinine. 47

Tableau 4 - Table des molécules de la base de données initiale du projet FLUOVIAL et leur formule topologique. 59

Tableau 5 - Table des molécules présentes dans la base de données d'entraînement et leur formule topologique. 60

Tableau 6 – Tailles des buckets utilisés lors de l'application de Plasmodesma pour le prétraitement des données pour les noyaux ^{1}H, ^{13}C et ^{19}F pour les expériences RMN en 1D et 2D. 63

Tableau 7 - Exemple d'une entrée de la base de données d'origine telle que fournie par T. Malliavin pour le projet RESCUE 3. 80

Tableau 8 - Exemple d'une entrée de la base de données après une première réorganisation..... 82

Tableau 9 - Exemple d'une entrée de la base de données finale utilisée pour l'entrainement de l'algorithme de DL..... 82

Tableau 10 - Structure du réseau de neurones mis en place pour la prédiction des acides aminés en fonction des déplacements chimiques dans Rescue3. 88

Tableau 11 – Taux de réussite (en %) des prédictions de l'algorithme mis en place par acide aminé puis globalement, en fonction des scénarios. En rouge les prédictions sous 50%, où l'algorithme n'est pas capable de prédire correctement l'acide aminé. 95

Tableau 12 - Temps de calculs observés et estimés pour différentes tailles de morceaux choisis pour la découpe de la donnée de 2D FTICR MS. 144

INTRODUCTION

Ce travail de thèse s'inscrit dans un projet de thèse CIFRE en collaboration entre l'IGBMC et l'entreprise CASC4DE. Le sujet de la thèse portant sur la mise en place de méthodes d'analyse de données issues de techniques biophysiques par apprentissage automatique. Ceci a donc impliqué le traitement de données de type « Big Data » ainsi que l'utilisation d'algorithmes innovants de Machine Learning et de Deep Learning spécifiquement adaptés aux données à analyser.

I. Problème des Big Data en biophysique et solutions apportées par le Machine et Deep Learning (ML & DL)

Au fil des années la production de données, scientifiques ou non, n'a fait que s'accélérer d'une part grâce à l'amélioration des technologies permettant de les acquérir et d'autre part par la multiplication des projets nécessitant l'acquisition et le stockage de données. Ces données, produites en grand nombre et volumineuses, sont nommées « Big Data » et sont donc de plus en plus courantes.

L'utilisation du terme « Big Data » reste assez récente et est apparue, en biologie, avec les dernières avancées technologiques comme le séquençage de nouvelle génération (NGS) ou les nouvelles techniques d'imagerie comme l'imagerie MS FT-ICR qui fournit une énorme quantité d'informations dans des délais courts.

La définition des Big Data a, elle-même, évolué au fil des années et de leur génération. Elles peuvent être de sources complètement différentes, comme les données issues des réseaux sociaux, les données IoT (Internet of Things) comprenant les caméras de sécurités et autres capteurs, ou encore des données scientifiques. Elles partagent cependant des caractéristiques communes, parmi lesquelles les fameux « 5V» : Vélocité, Volume, Variété, Valeur et Véracité (Ward et Barker 2013). Les 3 premiers « V » décrivent les données en elles-mêmes, l'acquisition doit être très rapide et générer des volumes de données importants ainsi que présenter une grande variété au sein des jeux de données avec des formats différents, diverses sources ou encore des données structurées ou non. Les deux derniers « V », valeur et véracité, correspondent plutôt à l'analyse des données. En effet, ceci implique que l'on souhaite en extraire des informations utiles et utilisables et fournir la bonne information au bon moment.

La définition des « Big Data » inclus donc à la fois le type de données concernées mais également la façon de les analyser. En effet, ce type de données nécessite généralement des méthodes de traitement spécifiques puisqu'elles ne peuvent plus être interprétées « à la main », conduisant donc à des méthodes d'automatisation comme les méthodes de Machine Learning (ML), ainsi que des stockages NoSQL qui sont des bases de données non relationnelles, ou encore des techniques de MapReduce.
Les Big Data sont non seulement plus complexes à analyser et nécessitant des approches statistiques, mais elles contiennent aussi régulièrement plus de bruit et d'artefacts. Leurs annotations et leur préservation doivent donc être bien pensées pour faciliter leur utilisation et leurs interprétations.

Dans le cadre de ce travail de thèse des méthodes d'apprentissage automatique, regroupant Machine Learning et de Deep Learning, seront appliquées pour l'analyse de données de spectrométrie de masse (MS) et de résonance magnétique nucléaire (RMN). En effet, les big data sont de plus en plus présentes y compris dans ces domaines, et de nombreux projets en génèrent et nécessitent des techniques spécifiques pour leur analyse.

La spectrométrie de masse (MS) est un outil d'analyse utilisé dans de nombreux domaines scientifiques. Elle est très utilisée notamment en biotechnologie, en particulier depuis l'apparition de nouvelles techniques d'ionisation douce (ESI - MALDI) permettant l'analyse de molécules biologiques délicates ou de protéines qui ne résisteraient pas à d'autres conditions expérimentales de préparation. Par ailleurs, selon l'appareil et la méthode choisis, les possibilités sont très vastes, allant de la détermination de la masse d'une molécule aux données de structures. La MS est une technique très sensible et sélective, fournissant rapidement des informations qualitatives et quantitatives. De nombreuses techniques existent et sont utilisées en fonction des besoins des chercheurs et des scientifiques. Un grand avantage de la MS est qu'elle peut éventuellement être couplée à d'autres techniques pour des protocoles plus précis ou plus avancés.

La Résonance Magnétique Nucléaire (RMN) est une méthode d'analyse basée sur la mesure du signal électromagnétique produit, avec une fréquence caractéristique, lorsque des noyaux (communément ^{1}H, ^{13}C ou encore ^{19}F) sont placés dans un champ magnétique fort et constant et perturbés par un champ magnétique plus faible et oscillant. Cette méthode est utilisée, entre autres, en tant qu'imagerie

médicale, mais également en recherche dans le domaine de la biophysique pour élucider la structure de molécules par exemple.

Ces deux domaines étant en constante évolution, additionné à l'apparition de techniques d'analyses dite à haut-débit comme le criblage de candidats médicaments, ont engendré la génération de données de plus en plus nombreuses et volumineuses avec des besoins d'analyse différents, incluant des analyses plus globales (méta-analyses) ainsi que des besoins d'uniformisation et filtrage des données contenant, par exemple, des informations importantes pour ne garder que les variables informatives.
Par ailleurs, certaines des techniques mises en place aujourd'hui grâce à l'évolution des technologies, comme la 2D FT-ICR MS, génèrent des données volumineuses de haute résolution qui doivent être stockées dans des formats et structures particuliers et nécessitant également une analyse adaptée puisque toutes les informations contenues ne peuvent pas être observées « à la main ».

Les données massives et de haute résolution générées par les méthodes de MS et de RMN présentent des spécificités générant des obstacles pour leur analyse. En effet, il s'agit d'un type de Big Data, ne pouvant plus être analysées efficacement manuellement, même par des experts dans les domaines respectifs. Par ailleurs, des recoupements de différentes données peuvent également être nécessaires pour tirer le meilleur parti des données disponibles. Des méthodes spécifiques d'analyse sont donc requises et une des solutions permettant de faire face à ce type de problèmes est l'utilisation d'algorithmes d'apprentissage automatique, aussi appelé Machine Learning (ML) ou encore Deep Learning (DL).

Les techniques de ML et de DL sont des sous-groupes de ce qui est communément appelé « Intelligence Artificielle » ou IA. En effet, le terme d'intelligence artificielle regroupe l'idée théorique et tous les systèmes permettant d'imiter ou de reproduire une pensée ou une action humaine, comme la reconnaissance vocale ou textuelle ou encore la reconnaissance d'images.

Le ML est donc une sous-catégorie de ces systèmes, incluant tous les systèmes permettant d'apprendre à partir de grands ensembles de données (Mahesh 2019). Dans tous les cas, l'objectif est d'obtenir une décision et de donner des résultats sur une donnée à partir d'un entraînement sur un jeu de données type. Les algorithmes de ML sont généralement des algorithmes « classiques » de classification ou de

regroupement, et nécessitent des traitements de données et ajustements en amont pour être très efficaces.

Le DL est une des approches possibles en ML et est donc un sous-groupe des techniques que rassemble le ML (Charniak 2021). En DL, des réseaux de neurones profonds sont utilisés pour déduire une structure, un modèle à partir des données disponibles lors de l'entraînement et en fonction des sorties demandées. Dans ces algorithmes, aucune supervision humaine n'est réalisée au cours de l'apprentissage. L'idée des réseaux de neurones profonds utilisés en DL est de reproduire le fonctionnement du cerveau humain en utilisant une structure considérée comme similaire et fonctionnant par couches. Chaque couche du réseau est constituée de plusieurs neurones et va décomposer une partie du problème, en recoupant l'ensemble des informations extraites par les différentes couches il est alors possible d'obtenir un algorithme capable d'effectuer une tâche spécifique.

Les données à analyser vont généralement orienter le type d'algorithme (ML ou DL) utilisé, puisque de manière globale les algorithmes de DL vont plutôt être adaptés au traitement de données non-structurées comme un texte ou un son. Les algorithmes dits de « ML », en excluant ce qui est relatif au DL, vont plus généralement être utilisés pour traiter des données plus structurées (sous formes de tables, avec des variables bien définies par exemple) (Grinsztajn, Oyallon, et Varoquaux 2022). Le plus souvent lors de l'utilisation d'algorithmes de DL, les variables ou caractéristiques au sein des jeux de données ne sont pas présélectionnées, on laisse l'algorithme définir seul les informations utiles dans les données. A contrario, le plus souvent pour les algorithmes de ML, une extraction des variables est réalisée, complétée d'une réduction de dimension pour éliminer ce qui ne sera pas utile pour l'entraînement à la tâche à effectuer, une forme de supervision est donc présente pour les algorithmes de ML. L'utilisation d'algorithmes de DL est donc un gain de temps pour la mise en place du programme puisque l'étape de sélection et d'extraction des variables ou caractéristiques en ML est très chronophage.

Dans tous les cas, tant pour le ML que le DL, l'idée est d'utiliser un algorithme adapté à effectuer une tâche sans avoir à la programmer explicitement, les modèles sont construits par les algorithmes eux-mêmes. Le DL étant donc un sous-ensemble des approches du ML, lui-même étant un sous-ensemble de l'idée théorique de l'IA.

Les exemples d'applications d'algorithmes de DL sont nombreux, tant dans des domaines généraux que plus spécifiques. Nous pouvons citer ici, entre autres, les algorithmes de recommandations pour des plateformes de vidéos par exemple ou encore la reconnaissance d'images ou de son pouvant être implémentés dans les voitures autonomes (Grigorescu et al. 2020; S. Zhang et al. 2019). Le DL a également connu des développements dans le domaine pharmaceutique et de la médecine personnalisée avec des algorithmes permettant la personnalisation des traitements en fonction des pathologies et génomes des patients (Papadakis et al. 2019). En biophysique, les exemples d'applications d'algorithmes de DL sont également nombreux, notamment pour la modélisation des mécanismes moléculaires ou encore le fameux AlphaFold2 utilisé pour la prédiction de structure des protéines de manière très précise (AlQuraishi et Sorger 2021; Jumper et al. 2021).

Par ailleurs, les projets menés au cours de cette thèse sont tant que possible réalisés de manière à travailler en science ouverte. La recherche en science ouverte a de nombreux avantages, à commencer par une meilleure fiabilité puisque les méthodes mises en place pour l'obtention de résultats sont connues de tous et donc consultables et critiquables par les pairs. Cela permet également de manière générale une meilleure réplicabilité du travail effectué qui sera déposé et réutilisable par quiconque souhaiterait reproduire ou étendre la question abordée. Le fait de travailler en science ouverte permet généralement des collaborations interdisciplinaires facilitées ainsi que la mise en place de projets de recherche à plus large échelle. De plus, les ressources partagées en science ouverte sont disséminées bien plus facilement, les connaissances peuvent ainsi être utilisées plus rapidement, plus efficacement et cela évite la redondance potentielle de projets de recherche (National Academies of Sciences, Engineering and Medicine 2018). Une initiative nommée « ReScience » visant à reproduire des résultats scientifiques obtenus par des méthodes computationnelles a été mise en place, dans le but d'avoir les algorithmes dans des versions qui permettent une reproductibilité des résultats sur une plateforme de diffusion, ici GitHub, accessible publiquement (Rougier et al. 2017). Ce type d'initiatives montre l'intérêt et l'importance grandissante d'être capable de reproduire des résultats obtenus par d'autres équipes de recherche.

Différents problèmes sont présentés dans cette thèse tant avec des applications en RMN qu'en MS, liés à des données de grandes dimensions et pour lesquels des solutions ML ou DL additionnées de

traitements de données spécifiques ont été développées. Les programmes mis en place et résultats obtenus pour les différents projets sont dès que possible rendus disponibles sur GitHub dans l'idée de travailler en science ouverte.

II. Procédures classiques en Machine Learning

En Machine Learning, la façon de procéder est très souvent composée d'étapes précises et similaires d'un projet à l'autre, avec des éléments importants dans chacune des phases de développement d'un algorithme de ML. Ces méthodes seront retrouvées tout au long de ce travail de thèse et sont donc présentées ici.

En ML, il est important, avant toute chose, de commencer par saisir les tenants et les aboutissants du projet, comprendre à quoi on va être confronté, de quelles données on va pouvoir disposer, et ce dont on pourrait avoir besoin en supplément et qui puisse être accessible dans le cadre du sujet. La réflexion commence dès l'énoncé de la problématique et souvent certaines décisions peuvent déjà se dessiner ou bien la quantité de possibilités, pour le choix d'un algorithme par exemple, être réduite par la nature du projet elle-même.

Cette étape préliminaire d'évaluation au début d'un projet est très importante et permet souvent de gagner du temps sur les différentes étapes classiques décrites ci-après.

Choix des données

Dans un premier temps, il sera question de choisir les données à utiliser en vue de l'objectif à atteindre. Parfois les données seront fournies, parfois il faut récupérer et utiliser des jeux de données existants, plusieurs questions fondamentales apparaissent alors.

Tout d'abord, la quantité de données dont on dispose ou que l'on peut obtenir sera-t-elle suffisante dans le cadre de la problématique du projet.

Les données nécessitent-elles d'être étiquetées ou peut-on utiliser un algorithme non supervisé.

Il faut également s'assurer que la qualité des données soit suffisante, qu'il y ait une assez grande diversité pour pouvoir élaborer un modèle qui ne sera pas biaisé. Qu'il s'agisse de mettre en place un algorithme de clustering, de prédiction ou encore de data-mining, plus la quantité de données et leur variabilité sera élevée plus l'algorithme sera robuste à l'utilisation. Par ailleurs, l'équilibre des données en termes

de nombre d'exemples par classe disponibles peut avoir un impact important sur l'efficacité de certains algorithmes. Cette caractéristique du jeu de données est donc un élément clé à prendre en compte lors du choix et de l'évaluation du modèle mis en place.

Préparation des données

Une fois les données à utiliser sélectionnées et après vérification de leur qualité, vient l'étape de la récupération des données. Certaines données sont volumineuses, ou ne peuvent être récupérées que par une API qui implique des requêtes spécifiques sur des serveurs web.
Les données récupérées peuvent donc avoir des formats variés, qu'il faut aller prendre en charge, appréhender ou modifier pour les rendre utilisable dans le cadre de l'application d'un processus de traitement automatique de celles-ci. Les temps de traitement des données brutes, telles que disponibles en ligne ou fournies, sont d'autant plus longs que les manipulations à faire sont nombreuses et que la base de données est grande. Cette étape de préparation de données est indispensable et doit être effectuée minutieusement afin de faciliter l'application d'algorithme automatisés.
De manière générale, il est important d'avoir des identifiants uniques pour chaque élément du jeu de données, d'éviter les redondances et de maximiser les éléments porteurs d'informations. Dans l'idéal on ne voudra conserver que les informations qui seront utiles à l'objectif.

Lorsque les données sont trop volumineuses, ou contiennent plus d'informations que ce qu'il est nécessaire ou encore des informations inutiles, des techniques de réduction de dimensions peuvent être appliquées (Sorzano, Vargas, et Montano 2014; Reddy et al. 2020). Les techniques de réduction de dimensions sont nombreuses et doivent être choisies et adaptées en fonction du type de données ou de si une supervision est possible ou non, l'objectif étant de réduire le nombre de variables à analyser, pour ne conserver que les plus pertinentes et informatives.

Un enrichissement des données peut parfois être réalisé lorsque la base de données disponible est trop peu variée ou déséquilibrée par exemple ou encore lorsque des informations pertinentes sont manquantes. En effet, dans certains cas, l'augmentation des données en ajoutant des descripteurs statistiques, ou en dupliquant et transformant aléatoirement des éléments de la base de données pour ajouter une robustesse aux artéfacts et déformations permet de rendre la base de données plus complète et adaptée

au problème abordé. L'enrichissement de données peut aussi consister à compléter les données d'une base existante, lorsqu'elle est incomplète, avec des informations disponibles par d'autres moyens comme d'autres sources de données par exemple.

Une harmonisation des données est souvent nécessaire, d'autant plus lorsque les données sont issues de différentes sources (différentes bases de données, différentes acquisitions ou expérimentateurs). L'harmonisation consiste à utiliser ou générer un même format ou une même structure pour toutes les données, afin d'être en mesure de les comparer ou de les assembler en fonction des besoins de l'algorithme. Cette étape peut contenir des changements d'unités, des adaptations de structures de tables ou encore la création d'identifiants uniques pour la globalité des éléments de la base de données à utiliser pour l'analyse.

Le fait de générer une base de données nettoyées, harmonisées, contenant toutes les informations utiles et organisées de manière à pouvoir identifier de manière unique les éléments permet de pouvoir appliquer des méthodes automatiques sur l'ensemble des données afin de pouvoir les filtrer par exemple, ou bien les analyser globalement dans le cadre de fouille de données par exemple.

Séparation des jeux de données

Afin de pouvoir entrainer un modèle préalablement sélectionné, l'algorithme doit pouvoir être testé et validé.

La méthode la plus répandue dans le domaine est de découper le jeu de données de départ en différents sous-jeux de données ayant chacun un rôle spécifique au cours du développement de l'algorithme. Il faut au minimum deux sous-parties issues du jeu de données initial, une partie réservée à l'entrainement et une seconde au test de l'algorithme (Ripley 2007). Dans certain cas un troisième jeu de données, dit « de validation », est généré.

Le jeu de données d'entraînement est celui utilisé afin d'entraîner le modèle choisi. C'est la partie des données qui sera la plus conséquente. En effet, dans de nombreux algorithmes, plus on a de données pour l'entraînement de l'algorithme plus celui-ci est robuste. Cet effet a une limite et un entraînement

trop important peut mener à du surentrainement, aussi appelé over-training ou over-fitting (M. Kuhn et Johnson 2013; Ying 2019). Ce phénomène apparaît lorsque le modèle dépasse le niveau de généralisation du problème, mais se spécifie sur les données d'entraînement et n'est alors plus capable de généraliser le modèle sur des données qu'il n'a encore jamais vues. Cependant, l'over-training est plus souvent dû à la manière d'entraîner le modèle, incluant le nombre d'epoch par exemple, qu'à la quantité de données dans le jeu d'entraînement. Le nombre d'epoch d'un modèle est le nombre de passage au travers des données d'entraînement, ou d'un sous-ensemble de celles-ci, pour l'apprentissage de l'algorithme.

Le jeu de données dit de « Test » est nécessaire et utilisé afin de pouvoir réaliser une évaluation objective du modèle entrainé. Ce jeu de données n'est jamais vu par l'algorithme au cours du processus d'entraînement et doit permettre de quantifier l'efficacité du modèle pour chaque type de données présentes dans le jeu de données initial. Ce jeu de données doit donc être parfaitement échantillonné pour couvrir tous les cas possibles auxquels le modèles peut se retrouver confronté dans le cadre d'application à des cas concrets.

Le jeu de données de « validation » est utilisé pendant le développement du modèle afin d'évaluer les différentes versions entraînées d'un algorithme pour optimiser ses hyperparamètres. Le modèle va donc « voir » les données du jeu de validation pendant sa mise au point, même s'il ne va jamais apprendre ou être entrainé sur ces données spécifiques. Ce jeu de donnée a donc néanmoins un impact sur le choix final du modèle contrairement au jeu de données « test ».

Le pourcentage de répartition des données dans les différentes sous-parties est souvent du même ordre de grandeur entre différents entraînements et algorithmes. Cependant, un ajustement est tout de même nécessaire en fonction des cas et va dépendre de la variété et de la quantité des données dont on dispose, afin d'avoir une répartition représentative du jeu de données global dans chacune des sous-parties ainsi qu'une quantité suffisante de données pour un entraînement robuste et un test pertinent. Par ailleurs, la répartition entre les différents jeux dépend également du type de modèle à entraîner, en effet, certains modèles requièrent un plus grand nombre d'éléments pour un entraînement robuste.

Concrètement, le jeu de donnée de validation est souvent utilisé comme jeu de test, mais ce n'est pas une bonne pratique car il vaut mieux avoir comme jeu de test des données qui sont totalement étrangères à l'algorithme développé. En effet, même si le jeu de validation n'est utilisé que pour optimiser les paramètres, il a tout de même un impact sur le choix de l'algorithme final et n'est donc pas totalement exclu de la mise au point du modèle. Réaliser l'évaluation sur un jeu de donnée qui a servi à l'optimisation des paramètres ne fournit donc pas un résultat objectif et non biaisé de l'efficacité de l'algorithme.

Dans la plupart des implémentations disponibles, avec Keras par exemple, la validation croisée est utilisée (Refaeilzadeh, Tang, et Liu 2016). L'ensemble de données est divisé en deux, un jeu d'entraînement et un jeu de test. L'ensemble de test est mis de côté, et le travail est fait sur l'ensemble d'entraînement. Cet ensemble de travail se voit lui-même itérativement divisé en deux aléatoirement, avec un pourcentage fixe entre chaque itération d'environ 80% pour l'entraînement et 20% pour la validation. On génère donc différents ensembles d'entrainement-validation à partir du même sous-ensemble de départ qui sont utilisés pour entraîner et optimiser l'algorithme.
L'utilisation de la validation croisée permet d'éviter l'over-fitting et donc d'avoir des algorithmes plus résistants aux cas réels.

Modèles

Le choix du modèle en apprentissage automatique prend une part importante dans la réussite de la résolution du problème et l'efficacité de l'algorithme mis en place. Le modèle est généralement fortement lié au problème rencontré, aux résultats que l'on souhaite obtenir mais également aux données qui sont disponibles.
Les modèles de ML sont par ailleurs très nombreux, et peuvent être déclinés dans un grand nombre de possibilités, d'autant plus pour les algorithmes d'apprentissage profonds où les structures de réseaux sont à adapter à chaque situation.

Les besoins et les données disponibles, tant au niveau de leur quantité que des étiquettes présentes ou non, vont orienter le choix du modèle.

Lorsque les données sont étiquetées, un apprentissage supervisé peut être mis en place. Il s'agit d'un apprentissage où on donne les réponses à obtenir à l'algorithme lors de l'entraînement, par exemple la classe à attribuer à une donnée fournie. On parle alors d'algorithmes de classification ou de régression. Dans le cas contraire, on parle d'apprentissage non-supervisé, l'algorithme va alors générer lui-même les groupes ou classes de données en fonction des similitudes et différences qu'il va percevoir au sein du jeu de données. Il s'agit alors d'algorithmes de regroupements (clustering) ou d'associations.

La Figure 1 est un graphique issu de la documentation de la librairie Scikit-Learn qui a pour but d'aider à la décision lors du choix du modèle de ML à utiliser en fonction de la quantité de données disponibles et du type d'algorithme que l'on souhaite mettre en place. Cette figure démontre bien la grande variété de possibilités quant au choix d'un modèle, même lorsqu'il est restreint à un modèle d'apprentissage automatique « classique » non profond.

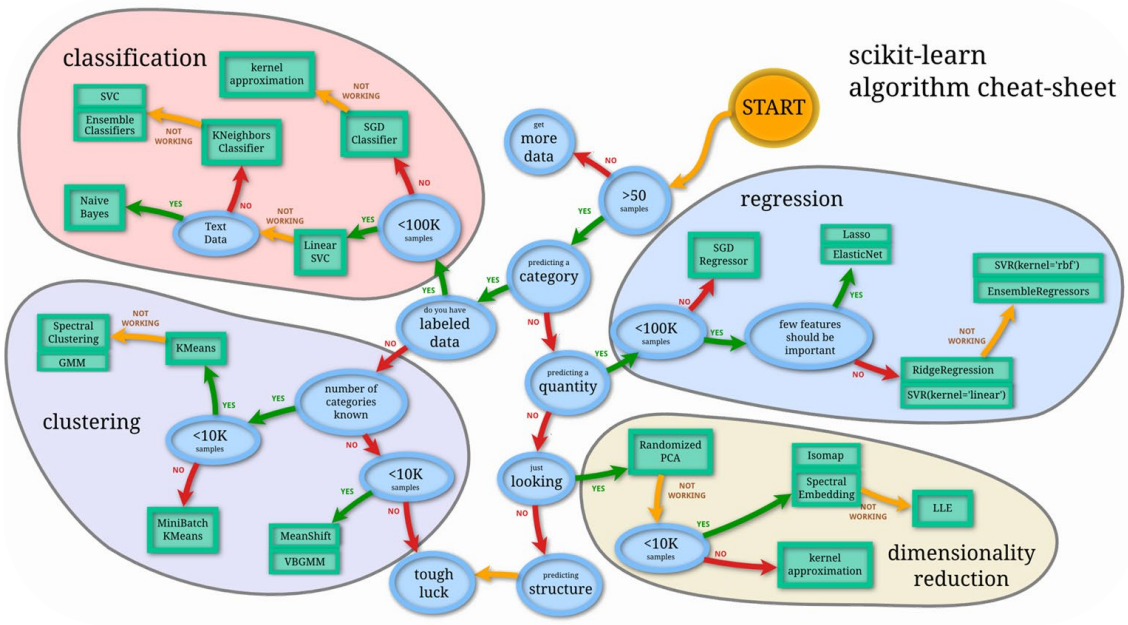


Figure 1 - Graphique d'aide à la sélection de modèles en fonction des besoins et des données disponibles. Source : Documentation Scikit-Learn

Dans le cas d'un réseau de neurone profond, les possibilités de structures sont également diverses. La Figure 2, réalisée par Fjodor van Veen de l'institut Asimov, illustre et regroupe les différentes structures classiques de réseaux de neurones qui sont utilisées couramment en apprentissage profond.

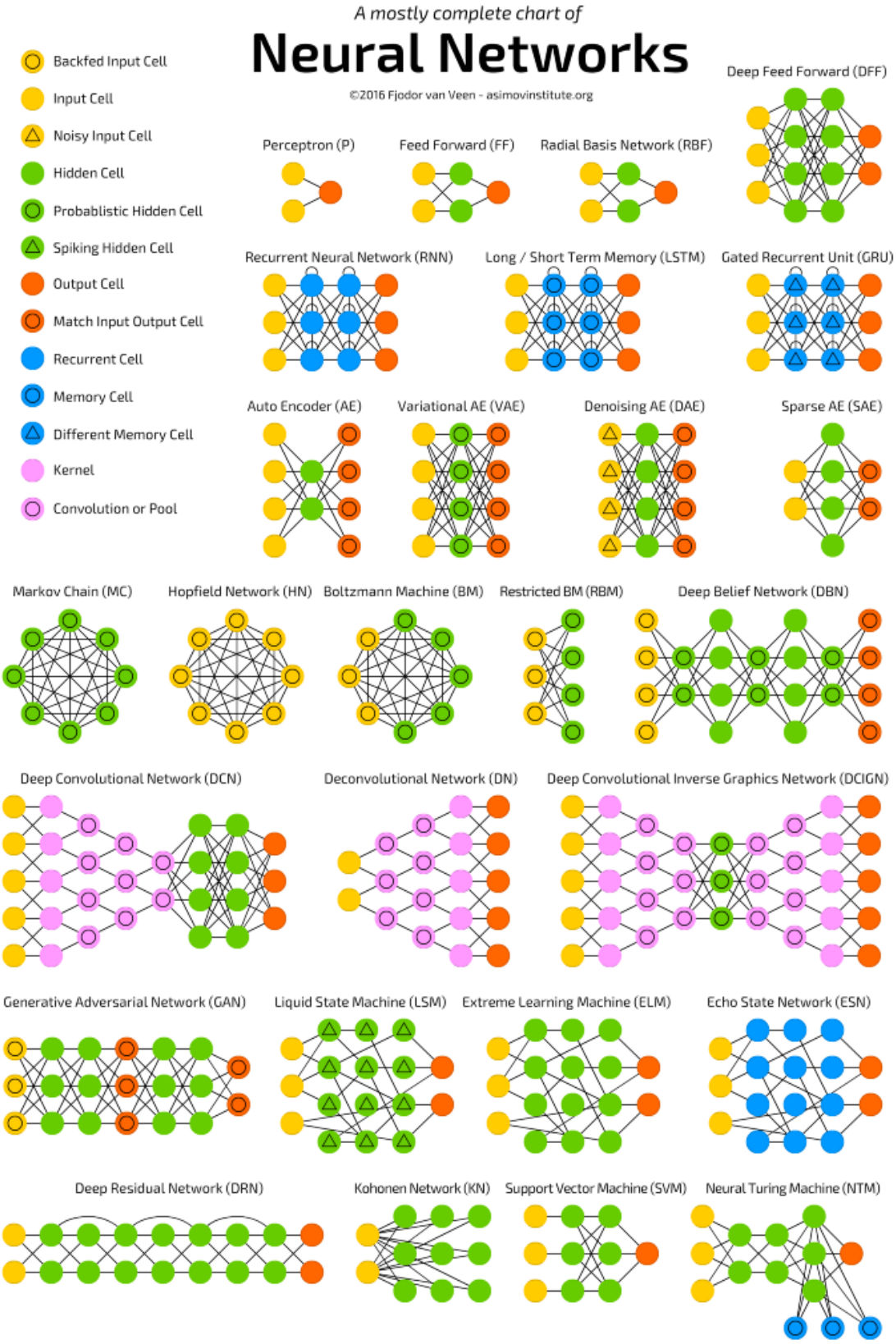


Figure 2 - Différentes topologies de réseaux de neurones « classiques » répertoriées par Fjodor van Veen de l'institut Asimov.

La variété de topologies disponibles présentées est révélatrice du nombre de modèles possibles qui peuvent être mis en place, et donc de l'importance et de la difficulté de bien choisir l'algorithme ou la structure utilisée pour résoudre un problème spécifique.

Par ailleurs, certains modèles sont plus ou moins résistants aux données manquantes, aux disparités au seins des données ou encore au déséquilibre entre les classes.

Les spécificités de chaque modèle sont donc à prendre en compte lors de l'élaboration de l'algorithme, la première étape étant de déterminer si un apprentissage profond (DL) est possible et nécessaire ou non. Chaque modèle ou topologie de réseau étant ensuite à adapter en fonction de la problématique en ajustant ses paramètres et hyperparamètres lors de l'implémentation.

Implémentation et optimisation

Dans le cadre de l'apprentissage automatique, deux types de paramètres vont être utilisés pour l'implémentation et l'optimisation de l'algorithme.

Les « paramètres » sont internes au modèle et ne vont pas être choisis ou optimisés par l'utilisateur. Ils vont être appris et estimés directement par l'algorithme pendant l'étape d'entraînement en fonction des données fournies et de la cible déclarée afin de construire un modèle permettant de passer de l'entrée à la sortie souhaitée. Les coefficients utilisés dans les modèles de régression ou encore les poids d'un réseau de neurones sont ce qu'on appelle des paramètres.

Par ailleurs, en plus des paramètres « classiques », un second type de paramètres existe, appelés les hyperparamètres. Ils sont quant à eux à choisir et optimiser par l'utilisateur. Ce sont les paramètres intrinsèques au choix et à la mise en place de l'algorithme, au modèle ou au processus d'apprentissage (Probst, Boulesteix, et Bischl 2019). De manière globale, tous les éléments (valeurs ou configurations de l'algorithme) fixés avant le début de l'entraînement et dont la valeur n'est pas changée au cours de celui-ci sont des hyperparamètres.

Ils ont généralement peu d'impact sur la performance du modèle mais vont souvent beaucoup influer sur la rapidité et la qualité de l'apprentissage, d'où l'intérêt de les optimiser afin de gagner en performances d'apprentissage. En fonction des algorithmes et modèles, certains hyperparamètres ont une importance plus ou moins grande sur l'efficacité d'apprentissage.

De manière générale, le ratio de données de test et d'entraînement au sein du jeu de données est un exemple d'hyperparamètre en ML.
En ML, pour un algorithme de forêt aléatoire (Random Forest) on peut, par exemple, répertorier le nombre d'arbres à utiliser ou encore la profondeur maximale de chacun des arbres comme étant des hyperparamètres.
Parmi les hyperparamètres d'un modèle d'apprentissage profond (DL) on peut citer la topologie, le nombre de couche et la taille de chacune d'entre-elles pour le réseau mis en place, mais également le taux d'apprentissage (ou learning rate en anglais) ou encore la taille des sous-ensembles de données utilisés à chaque epoch. Les fonctions d'optimisation ou d'activation choisies en sont d'autres exemples.

Ces paramètres peuvent être optimisés manuellement en observant l'évolution des éléments d'évaluation du modèle en fonction de la variation des paramètres et hyperparamètres. Cependant, de nouvelles méthodes d'implémentation simplifiées d'algorithmes de ML et DL sont de plus en plus mises en avant et disponibles pour les différentes bibliothèques de développements. Elles peuvent permettre d'accélérer le processus sur des cas habituels dans le domaine. Ces méthodes sont regroupées sous le terme « auto-ML » car, en effet, elles visent à automatiser une grande partie des étapes nécessaires à l'utilisation d'algorithmes ML (Truong et al. 2019).
Tant `Keras` que `Scikit-Learn` ou encore `PyTorch` mais encore bien d'autres plateformes proposent aujourd'hui des outils clés en main pour développer et utiliser le ML sur des projets « communs » sans la nécessité d'avoir des connaissances ou expertises en science de données avec des résultats rapides et efficaces.
A titre d'exemple, l'outil « Auto-sklearn », basé sur la librairie `Scikit-Learn` associée à Python, est spécialement adapté pour la mise en place de solutions ML supervisées. Auto-sklearn permet l'optimisation automatique d'hyperparamètres ou encore la recherche automatique d'ensembles de données adaptées au bon fonctionnement de l'algorithme. Il existe également « Auto-Keras », open-

source, qui se base sur la librairie `Keras` pour automatiser le choix de l'architecture du modèle de DL ainsi que l'optimisation des hyperparamètres. Par ailleurs, une librairie additionnelle nommée `Hyperopt` écrite en langage Python est également disponible et implémente différents algorithmes classiquement utilisés pour l'optimisation des hyperparamètres (Bergstra, Yamins, et Cox 2013). Cette dernière librairie est moins complète mais plus claire et accessible que les deux premières.

Evaluation du modèle

Il existe plusieurs méthodes quant à l'évaluation des modèles de Machine Learning, de manière générale dans le cas d'algorithmes de prédictions, des matrices de confusions ainsi que leur courbes ROC associées peuvent être générées. Ces deux éléments sont basés sur les taux de positifs et de négatifs obtenus sur le jeu de données test.

En effet, pour un modèle à deux classes, la matrice de confusion va montrer le nombre de vrais positifs et de vrais négatifs sur la diagonale et les faux négatifs et faux positifs sur l'antidiagonale comme indiqué sur la Figure 3. Dans ce cadre, plus la matrice s'approche d'une matrice diagonale, meilleur est le classificateur.

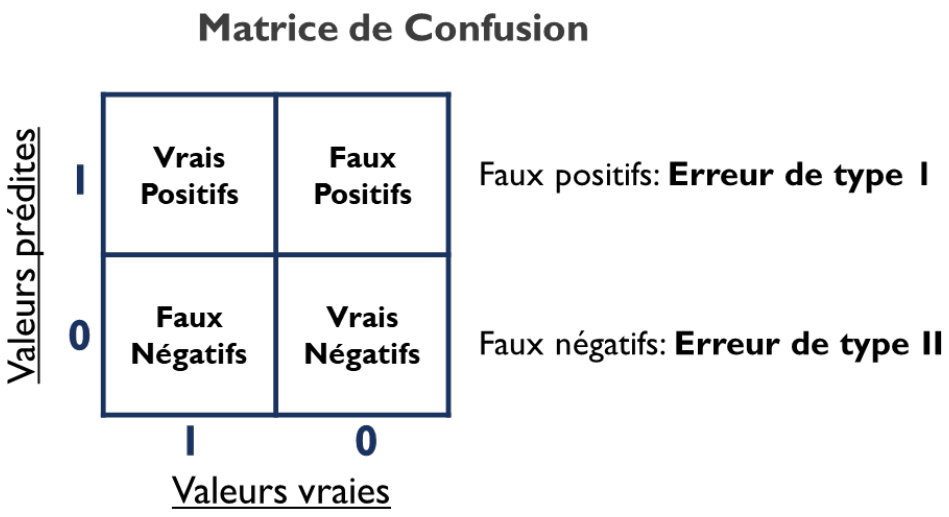


Figure 3 - Illustration d'une matrice de confusion.

Ce type de matrice permet également de calculer les erreurs de type I, cas dans lesquels la valeur prédite est positive alors que la valeur vraie est négative, et les erreurs de type II, cas dans lesquels la valeur prédite est négative alors que la valeur vraie est positive.

A partir des quatre quantités évoquées précédemment, on peut calculer la sensibilité, la spécificité, la précision et le rappel qui sont alors définies comme suit :

$\text{Précision} = \frac{Vrais\ Positifs}{Vrais\ Positifs + Faux\ Positifs}$	$\text{Sensibilité} = \frac{Vrais\ Positifs}{Vrais\ Positifs + Faux\ Négatifs}$
$\text{Rappel} = \frac{Vrais\ Positifs}{Vrais\ Positifs + Faux\ Négatifs}$	$\text{Spécificité} = \frac{Vrais\ Négatifs}{Vrais\ Négatifs + Faux\ Positifs}$

Dans le cadre d'un algorithme multi-classes, on va procéder au calcul comme indiqué précédemment pour chacune des classes puis on va réaliser une moyenne sur l'ensemble des classes.

La précision de l'algorithme représente la proportion de prédictions positives qui sont véritablement positives, un modèle sans faux positifs aura donc une précision de 1, il s'agit de la capacité à ne pas faire d'erreurs.

Le rappel démontre la capacité de l'algorithme à trouver tous les positifs, un modèle qui ne présente pas de faux négatifs aura donc un rappel de 1.

La spécificité d'un algorithme de prédiction indique le taux de vrais négatifs, tandis que la sensibilité de l'algorithme donne le taux de vrais positifs. La combinaison de ces deux valeurs permet donc d'avoir une indication sur la qualité de l'algorithme, car en effet, on souhaite avoir un algorithme qui va maximiser ces deux éléments, l'objectif étant de faire le moins d'erreurs de prédictions possible.

Enfin, on peut également calculer l'exactitude d'un modèle (accuracy en anglais), qui est différente de la précision. En effet, l'exactitude détermine la capacité de l'algorithme à réaliser les bonnes prédictions et se définit comme suit :

$$\text{Exactitude} = \frac{Vrais\ Positifs + Vrais\ Négatifs}{Vrais\ Positifs + Vrais\ Négatifs + Faux\ Positifs + Faux\ Négatifs}$$

Ainsi, plus l'exactitude d'un modèle est haute plus il est efficace dans la prédiction pour lequel il a été établit.

Par ailleurs, pour une représentation plus graphique de l'efficacité d'un modèle, il est possible d'utiliser une courbe ROC (Receiver Operating Characteristics) (Fan, Upadhye, et Worster 2006). Cette courbe, obtenue en faisant évoluer les paramètres d'un algorithme pour varier sa spécificité et illustrée sur la Figure 4, représente le taux de vrais positifs, ce qui équivaut à la sensibilité exprimée plus haut, en fonction du taux de faux positifs, correspondant à 1-Spécificité.

L'interprétation de cette courbe se fait de la manière suivante, plus la courbe tend à avoir une aire sous la courbe (ou AUC) égale à 1, meilleur est l'algorithme implémenté. Par exemple, si l'AUC est égale à 0.7, on a alors 70% de chance que l'algorithme prédise correctement la classe d'un élément. Si l'AUC vaut 0.5, on est alors en présence d'un algorithme qui n'est pas en capacité de différencier une classe de l'autre et qui va réaliser la mauvaise prédiction une fois sur deux. Dans le cas d'une AUC égale à 0, on a un algorithme qui prédit l'inverse de ce qu'il devrait, en effet, cela signifie qu'il va systématiquement prédire la classe opposée à celle dont il est en présence.

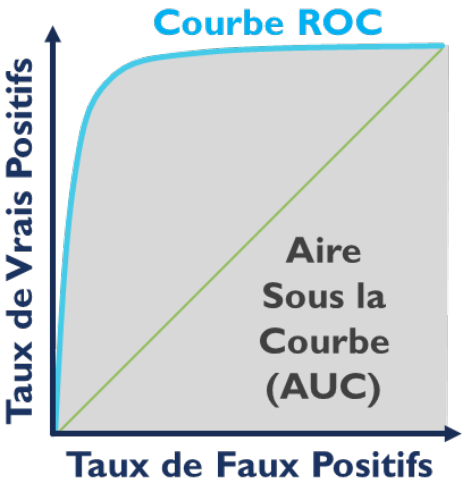


Figure 4 - Illustration d'une courbe ROC. En gris est représentée l'aire sous la courbe. En bleu la courbe ROC elle-même. En vert une courbe ROC où l'AUC est égale à 0.5 et où l'algorithme est alors incapable de différencier une classe de l'autre.

Dans le cadre de cette thèse, certains des algorithmes développés sont des algorithmes qui se prêtent bien aux évaluations par matrices de confusion et courbe ROC. Lorsque cela n'est pas possible, dans le cadre de développement d'algorithmes pour des projets différents, il est important de réfléchir à la mise en place de méthodes systématiques et les plus objectives possible pour l'évaluation des modèles développés afin de s'assurer de leur sensibilité et spécificité vis-à-vis de la problématique associée.

III. Méthodes générales utilisées au cours de la thèse

Python

Python est un langage de programmation spécifique conçu pour être puissant, polyvalent, mais aussi facile à apprendre et à appliquer, offrant de nombreuses possibilités de programmation (Langtangen 2009). De nombreuses fonctionnalités sont offertes par les modules natifs. De plus, avec le langage Python, il est facile d'étendre ces fonctionnalités de base en créant des bibliothèques ou en installant des bibliothèques supplémentaires. Par ailleurs, il s'agit d'un des langages de programmation les plus accessibles à la « lecture » pour des scientifiques peu initiés à la programmation informatique, puisque sa syntaxe est très légère par rapport à d'autres langages plus lourds et complexes syntaxiquement parlant, comme Java par exemple, et donc plus compréhensible au premier abord.

Ce langage est donc un outil très intéressant pour les personnes qui doivent s'attaquer à des problèmes informatiques et créer des scripts spécifiques pour répondre à des questions précises. Il permet de combiner de multiples contributions sur un même script ou algorithme et c'est un avantage significatif lorsqu'on travaille dans un environnement open-source puisqu'il est plus facile d'obtenir un code qui sera efficace, la plupart du temps testé et débogué par d'autres contributeurs que le rédacteur principal. C'est un langage de programmation très utilisé dans la communauté scientifique, notamment pour les biotechnologies, domaine dans lequel plusieurs librairies répondant à des besoins spécifiques ont été créées.

L'utilisation d'outils de programmation interactifs comme l'environnement `Jupyter Notebook` de la distribution Anaconda de Python ou encore l'environnement Google Colaboratory (`Colab`) permettent une collaboration lors de la rédaction de programmes mais également une distribution plus aisée de ceux-ci, au moins vers un public initié à la programmation. En effet, cela permet d'ajouter des titres, des commentaires mais également d'exécuter en interactif des parties d'un code pour en voir l'effet. Des outils permettent de figer les résultats et de les partager sous format HTML ou PDF. Par ailleurs, il est possible de distribuer un code fonctionnel via des méthodes comme `Binder`, où l'utilisateur va voir le notebook tel quel et pouvoir l'utiliser et le tester, ou bien encore avec la librairie `Voilà` qui permet de cacher le code et de créer des fenêtres interactives à partir d'un notebook `Jupyter`. Ces outils sont des ponts vers la production de science et de programmation ouverte.

Dans le cadre de cette thèse, Python est, pour toutes les raisons évoquées ci-dessus, le langage choisi pour le développement des algorithmes scientifique, en utilisant plusieurs bibliothèques références du langage telles que `Numpy`, `Scipy`, `Matplotlib`, `Seaborn`, `Bokeh` ou encore `Flask`.

`Numpy` et `Scipy` sont les librairies « classiques » de calculs utilisées avec le langage de programmation Python. Elles sont utiles dès lors que l'on a besoin d'effectuer des opérations sur les données et incluent tous les outils mathématiques nécessaires.
`Matplotlib`, `Seaborn` et `Bokeh` sont des librairies de visualisation de données. `Matplotlib` et `Seaborn` vont permettre de générer des graphiques, de les manipuler et de les exporter au format png par exemple. `Bokeh` est une librairie qui quant à elle permettra une visualisation interactive de données via un serveur web par exemple, avec la possibilité d'ajouter des outils d'interactivité spécifiques, d'ajouter des informations utiles au survol de la souris, grâce à une combinaison de scripts en langages `Python` et `Javascript`.

`Flask` est une plateforme de développement web open-source dépendante du langage Python (Grinberg 2018). Elle est très intuitive pour développer des applications web aussi bien simples que très complexes. Certains templates sont fournis mais peuvent être modifiés et adaptés à tous les besoins. Sa principale force est qu'elle est encore en développement très actif avec une documentation bien écrite et complète et une communauté réactive, ainsi qu'une liste de diffusion, pour l'assistance aux utilisateurs.

SPIKE

`SPIKE` est l'acronyme de « Spectrometry Processing Innovative Kernel » (Chiron et al. 2016). Il s'agit d'une librairie associée au langage Python dédiée à l'analyse de différents types de spectroscopies de Fourier tant en Résonance Magnétique Nucléaire (RMN) qu'en Spectrométrie de Masse (MS). `SPIKE` est open-source et il est facile de l'adapter à des besoins spécifiques si quelque chose n'a pas déjà été implémenté dans la libraire par un système de plugins qui peuvent être développés et ajoutés à volonté. La puissance de `SPIKE` est son implémentation innovante et très complète de nombreux outils de traitement et de visualisation de données volumineuses de RMN et MS. Les outils ont été écrits pour fonctionner en multiprocesseurs et avec une consommation faible de mémoire, ils s'avèrent donc efficaces et rapides.

SPIKE correspond au type de logiciels et librairies de programmation développés, en combinaison avec l'avancement technologique du matériel au cours des dernières décennies, pour faire face aux problèmes dus à l'augmentation des flux de productions de données (Marx 2013).

Grâce au développement de SPIKE, le temps nécessaire pour traiter un ensemble de données classique 2D FT-ICR MS avec un ordinateur comprenant 10 processeurs passe d'environ 11 heures à environ 8 minutes : une avancée considérable dans le traitement des big data.

Pandas

Pandas est une librairie du langage Python. Il s'agit d'une librairie open-source ayant pour but de permettre l'analyse et la gestion de données en Python (McKinney 2011). C'est un outil très performant et efficace, qui facilite grandement la manipulation des tables de données auxquelles on est souvent confrontés en science des données.

Cette librairie d'analyse de haut niveau va permettre de nombreuses transformations sur les données, avec l'idée de pouvoir structurer un jeu de données qui ne le serait pas, ou pas de la bonne manière, pour pouvoir l'utiliser comme entrée d'un algorithme de Machine Learning. Il est par exemple possible de charger différents fichiers csv, de les fusionner, de supprimer ou d'ajouter des colonnes ou des lignes, d'ajouter des index ou d'indexer sur une caractéristique spécifique.

Par ailleurs, du fait de la prise en main des fonctionnalités qui peut être délicate pour des utilisateurs peu informés, différentes librairies associées voient le jour et viennent compléter l'outil par des interfaces interactives. On a par exemple la librairie Bamboolib qui permet de générer très facilement des graphiques ou encore D-Tale qui ajoute une interface pour la manipulation des données, pour la suppression ou l'ajout d'éléments par exemple. Ce type de développements rend la gestion de données très accessible.

La librairie est assez rapide mais sa réactivité diminue avec la taille des données, une alternative dans ce cas est l'utilisation de la librairie PySpark qui présente une implémentation de Spark pour Python en embarquant directement une grande partie des fonctionnalités de Pandas, avec les mêmes notations pour des raisons de compatibilité. Il est donc très facile de passer d'une utilisation de Pandas à une utilisation de PySpark dans un algorithme donné.

La librairie python `Pyspark` est une interface en python du framework Apache `Spark` conçu pour traiter des bases de données importantes, en particulier dans le cadre d'applications dans le domaine des Big Data (Salloum et al. 2016). Ce framework est très optimisé, développé en `Scala` et basé sur le calcul distribué, il permet une très grande efficacité dans la gestion de données volumineuses. Bien que son efficacité soit maximale dans son langage d'origine, la version mise au point en python permet un gain important en termes de temps d'analyse et est une alternative efficace à la librairie classique `Pandas` lorsque les données deviennent trop importantes et les calculs trop chronophages. Comme indiqué, l'implémentation est très simple puisque le développement a été réalisé de manière à utiliser les mêmes notations que `Pandas`.

Scikit-Learn

`Scikit-Learn` (ou `sklearn`) est la librairie d'implémentation d'algorithmes de Machine Learning la plus répandue et complète du langage Python. Il s'agit d'une bibliothèque libre et développée en collaboration par de nombreux scientifiques, notamment issus du monde académique (Kramer 2016).

La librairie est conçue de manière très uniformisée, ce qui permet de passer d'une implémentation d'algorithme à une autre sans trop de complexité, les syntaxes utilisées par l'API étant très similaires. En effet, l'API de `sklearn` est très cohérente, les interfaces sont communes entre les différents objets, l'ensemble de méthodes applicables est limité au maximum. Par ailleurs, dès que possible des formats standardisés de représentation des données sont utilisés (`NumPy`, `SciPy` ou encore `Pandas`) et seuls les algorithmes sont inclus dans des classes de Python spécifiques, permettant une hiérarchie d'objets réduite.

En addition d'un grand nombre de modèles de Machine Learning, cette librairie permet également le traitement et la préparation des données en amont de l'implémentation des différents modèles. En effet, il est possible de séparer les jeux de données, mais surtout de gérer les valeurs manquantes ou encore de réduire la dimension des données.

L'outil dispose de très nombreux exemples et d'une documentation extrêmement fournie pour chacun des modèles disponibles.

L'utilisation de `sklearn` est très souvent composée des mêmes étapes : le choix du modèle, sa paramétrisation, l'entraînement sur un jeu d'apprentissage et le test sur des données que l'algorithme n'a pas encore vues.

Il s'agit d'une bibliothèque sous licence BSD, open-source, permettant ainsi à de nombreux collaborateurs de la faire évoluer de manière très régulière. La librairie continue donc de s'améliorer et de s'enrichir au fil du temps et des contributions, la rendant toujours plus complète et robuste.

Keras

`Keras` est une des librairies associées au langage Python permettant l'implémentation d'algorithmes de Deep Learning (Chollet 2018). Il existe d'autres librairies de ce type, comme `TensorFlow`, néanmoins dans le cadre de ce travail la librairie `Keras` a été choisie arbitrairement et est celle qui sera utilisée. Ces librairies sont assez similaires dans les outils proposés, la façon de les utiliser varie par contre légèrement. Dans le cas de besoins spécifiques, on peut être amené à utiliser plutôt l'une que l'autre pour des raisons d'optimisation ou de praticité d'implémentation.

L'API fournie par `Keras` est assez intuitive et accessible, leur mot d'ordre étant de proposer une librairie conçue pour les humains et non pour les machines. Les modèles prêts à l'emploi disponibles sont variés et il est par ailleurs très simple de générer un algorithme à façon en combinant les différentes structures proposées.

En effet, si des « recettes » préconçues sont proposées pour des applications classiques, les éléments indépendants peuvent être utilisés et combinés les uns aux autres en vue de construire l'algorithme désiré. Pour chaque couche de la structure les choix des différents hyperparamètres sont proposés.

De plus, la librairie permet des générer des jeux de données d'entrée à partir d'un jeu de données prétraité en `Pandas` par exemple. A partir du dataframe, structure de données de la librairie `Pandas`, on va être capable de produire trois jeux de données d'entrainement, de test et de validation, en choisissant la taille des différents jeux, par exemple en pourcentage par rapport au jeu de données complet. Il est également possible de transformer des labels texte en labels de type nombres entiers ce qui peut être requis dans certaines applications.

PARTIE 1 – APPLICATION DES TECHNIQUES DE ML A LA RESONANCE MAGNETIQUE NUCLEAIRE (RMN)

Cette première partie du manuscrit est orientée sur la présentation des projets réalisés portant sur des données issues de la RMN au cours de la thèse de doctorat. Chacun des projets aborde des données différentes, par des méthodes adaptées, qui seront présentées au fur et à mesure.

Les techniques de RMN sont des outils très couramment utilisés dans le cadre d'analyse biophysique, permettant en particulier l'étude de structures, de dynamiques de molécules, comme les protéines par exemple mais également l'analyse d'interactions moléculaires. Différents types de données peuvent être acquises par RMN, les plus courantes étant les données de déplacements chimiques sous forme de spectres 1D ou de cartographies 2D, comme il en sera utilisé dans les projets présentés ci-après.

I. Projet Plasmodesma – Déconvolution pharmacophorique de substances naturelles

Introduction

Au cours de l'évolution, les plantes ont acquis des caractères adaptatifs pour survivre dans leur environnement. Parmi ceux-ci, elles ont développé des « métabolites spécialisés » ayant pour rôle spécifique de défendre la plante contre des pathogènes biotiques ou abiotiques par exemple.
Ces produits naturels sont souvent bioactifs et constituent une source considérable pour la découverte de nouveaux médicaments (Newman et Cragg 2016).

En effet, on a des informations nombreuses sur l'efficacité de certaines plantes ou produits issus directement de la nature, utilisés depuis de nombreuses années, pour lutter contre certaines maladies. Ces connaissances « ancestrales » sont une source précieuse pour la découverte de nouveaux médicaments. La principale difficulté étant l'identification et la purification des composés bioactifs à partir des mélanges complexes issus des produits naturels dont nous avons l'information d'efficacité.

La méthode la plus couramment utilisée est l'isolement guidé par la bioactivité, qui se compose comme suit :

- L'extrait original est fractionné
- Les bio activités des différentes fractions sont testées
- Les fractions comportant des composés actifs sont à nouveau fractionnées
- Le cycle est répété jusqu'à ce que les composés bioactifs soient isolés et purifiés.

Ce processus étant lent et exigeant en ressources, avec un rendement assez faible la plupart du temps, des alternatives sont nécessaires.

Ces dernières années, l'utilisation de méthodes analytiques pour les études métabolomiques a augmenté, notamment dans le domaine de la dereplication, visant à identifier rapidement les composés connus dans un mélange. Ceci a été fait principalement avec la spectrométrie de masse, en raison de la sensibilité de la méthode.

Les méthodes de réseaux moléculaires se sont en particulier montrées très efficaces, notamment par l'utilisation de l'ensemble d'outils GNPS disponibles en ligne et permettant, entre autre, l'accès à une base de données brutes, traitées et annotées de spectrométrie de masse et ainsi qu'à des outils d'analyse comme les réseaux moléculaires (Wang et al. 2016). Ces derniers sont construits comme une représentation visuelle d'expériences de spectrométrie de masse en tandem (MS/MS). Cette méthode permet d'apparenter entre eux et donc de trouver des relations entre les molécules, même dans les cas où les molécules présentes dans les spectres sont inconnues. Chaque spectre de molécule disponible est un nœud du réseau, et un lien entre les différents nœuds est représenté lorsque des similitudes entre les spectres sont détectées. Des informations complémentaires sont parfois disponibles lorsqu'elles ont été indiquées lors du dépôt de la donnée, la couleur du nœud correspondant étant représentative de la quantité de données disponibles. Ce travail résulte en un réseau global entre tous les spectres présents dans la base de l'écosystème, permettant de déterminer une famille de molécule potentielle pour le spectre d'une molécule non identifiée.

En 2018, la méthode Plasmodesma a été développée pour la RMN pour la réalisation d'une analyse différentielle automatique afin d'obtenir rapidement l'empreinte moléculaire des composés bioactifs au sein d'un extrait naturel (Margueritte et al. 2018). En effet, de par la sensibilité de certaines techniques de RMN développées récemment, notamment grâce aux sondes cryogéniques, il était possible de développer une technique efficace pour identifier un composé bioactif au sein d'un mélange. La RMN en phase liquide étant déjà utilisée couramment sur des approches analytiques, notamment dans l'identification de composés bioactifs au sein de mélanges, l'idée était de mettre en place une méthode nécessitant peu d'acquisition de spectres RMN pour limiter le temps et le coût du processus.

Ainsi, le principe de Plasmodesma repose sur la volonté de réaliser une identification rapide d'un composé bioactif au sein d'un mélange par RMN. La méthode s'appuie sur la proportionnalité du signal RMN par rapport à la quantité de molécule, il est donc possible de corréler le signal RMN à l'activité de l'échantillon qui est une mesure indépendante et linéaire comme le signal acquis.

Une série d'environ une dizaine de fractions grossière est ainsi réalisée à partir du mélange présentant un intérêt et la bioactivité de chacune des fractions est mesurée. Un ensemble d'expériences RMN standardisées est ensuite réalisé pour chaque échantillon de la série de fractions apparentées et qui présentent donc une variation de bioactivité. Enfin une analyse automatique est appliquée aux données pour déterminer l'empreinte moléculaire de la molécule bioactive par analyse différentielle. Cette méthode est illustrée par la Figure 5.

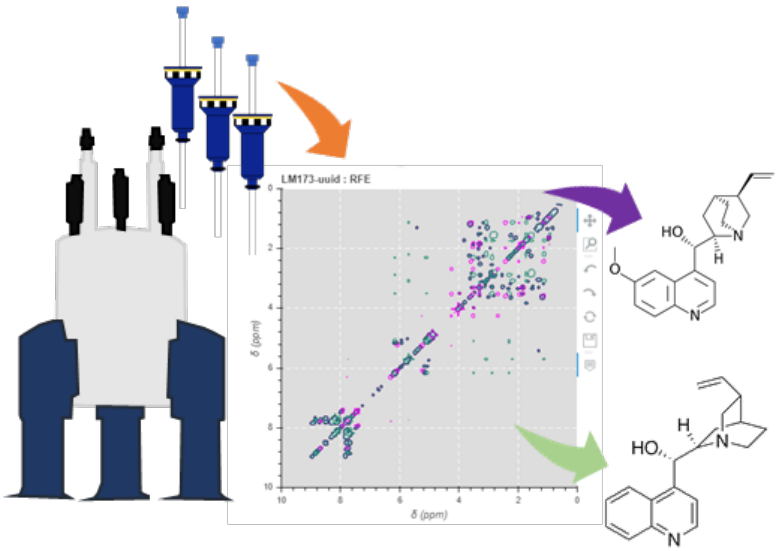


Figure 5 - Illustration de la méthode Plasmodesma

Le présent développement, réalisé en collaboration avec la faculté de pharmacie de l'Université de Strasbourg, est une extension de la méthode Plasmodesma, en vue d'une amélioration de l'analyse différentielle implémentée en ajoutant un algorithme d'analyse plus développé et de fournir un outil interactif d'observation des résultats afin d'obtenir l'empreinte spectrale de molécules responsables de bioactivité.

Matériel et Méthode

Données

L'algorithme développé est testé dans ce travail sur deux jeux de données expérimentales pour lesquels nous connaissons les molécules actives présentes dans les mélanges et responsables de la bioactivité des mélanges. Ces données expérimentales ont été produites et fournies par Laure Marguerite de la faculté de pharmacie de l'Université de Strasbourg, qui les a présentées plus en détails dans sa thèse.

Le premier jeu de données utilisé est une série de données synthétiques. Pour le réaliser, un extrait d'algues brut a été complémenté par une faible quantité d'artémisinine (Sigma-Aldrich), inhibiteur de l'agent responsable du paludisme. Un fractionnement de l'extrait a ensuite été effectué pour obtenir 6 fractions dont l'activité antipaludique a été testée. Les activités mesurées vont de 0% à 98% d'inhibition de la croissance de *P. falciparum*, responsable du paludisme (Tableau 1).

Tableau 1 - Niveaux d'activité mesurés pour la série de donnée synthétique complémentée en Artémisinine.

Fraction	**F1**	**F2**	**F3**	**F4**	**F5**	**F6**
Activité (% d'inhibition de P. falciparum)	11	98	0	0	0	40

Chacune des fractions a ensuite été diluée dans du D-méthanol afin de réaliser un spectre RMN 1D ^{1}H ainsi que des expériences COSY, TOCSY, DOSY et HSQC. La Figure 6 présente les spectres COSY pour chacune des fractions ainsi que pour la référence, l'artémisinine.

L'utilité de ce jeu de données est de pouvoir comparer les résultats obtenus à la théorie puisque dans le cas de cette série nous connaissons exactement à l'avance les résultats étant donné que la dose d'artémisinine ajoutée est connue, permettant ainsi de valider ou non la méthode développée.

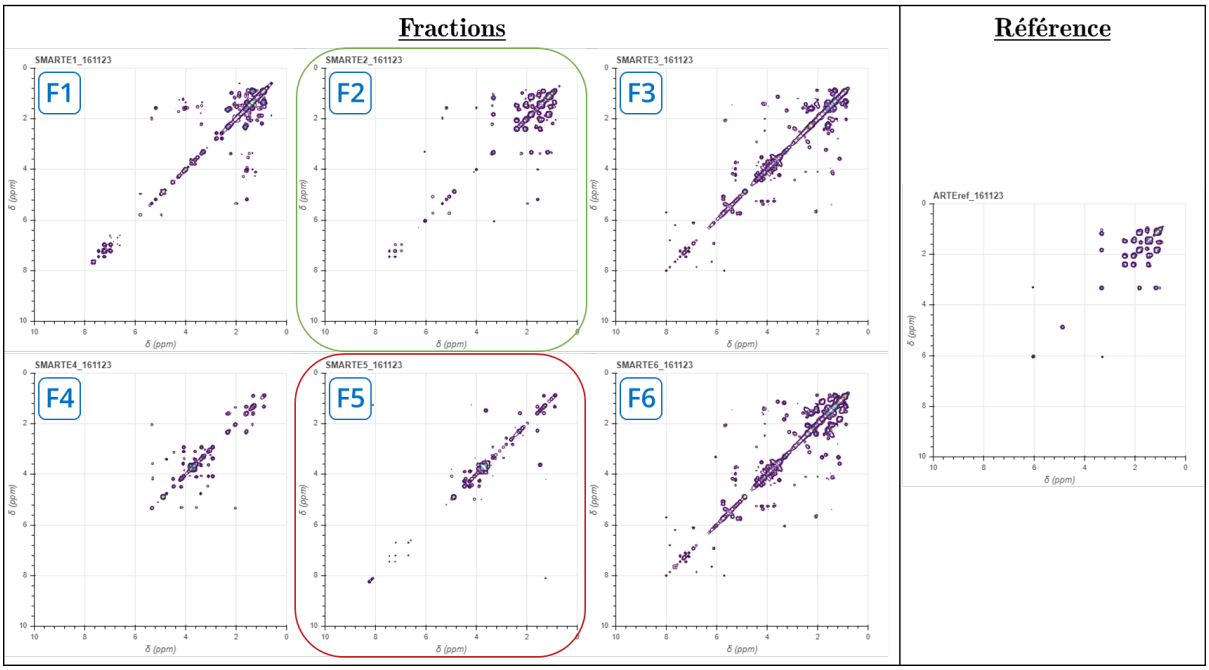


Figure 6 - Les spectres RMN COSY des 6 fractions réalisées à partir de la série de données synthétiques complémentée en Artémisinine sont présentés sur le panneau de gauche. En vert la fraction avec la bioactivité mesurée la plus élevée, en rouge l'une de celle présentant la bioactivité la plus faible. A droite le spectre RMN COSY de la molécule de référence, l'artémisinine, responsable de la bioactivité du mélange.

Le second jeu de données utilisé est un extrait brut d'écorce de Cinchona rouge. L'écorce de cet arbre est connue pour sa richesse en composés phénoliques, en acides organiques et en composés triterpéniques. Elle contient par ailleurs au minimum 6% d'alcaloïdes totaux dont une grande partie de type quinine. L'activité antipaludique, in vitro, d'une trentaine d'alcaloïdes extraits et identifiés de *Cinchona pubescens* a été démontrée. Il s'agit ici d'un cas d'étude complexe de par la quantité d'alcaloïdes présents ainsi que leurs similarités structurelles et leurs activités antipaludiques.

A partir d'un extrait hydroalcoolique d'écorce séchée de *Cinchona pubescens*, 6 fractions ont été réalisées. Pour chacune d'entre elles, l'activité antipaludique a été mesurée in vitro par IC_{50}. L'activité antipaludique de molécules de références a également été mesurée.

L'IC_{50}, ou concentration inhibitrice médiane, mesure la capacité d'un composé ou d'un mélange à inhiber la fonction biologique ou biochimique, ici le paludisme. Il s'agit d'une mesure quantitative qui permet de connaître la quantité de substance active nécessaire pour inhiber de 50% l'activité biologique in vitro. Par conséquent, plus la valeur de l'IC_{50} est élevée plus la bioactivité du composé est faible puisque cela signifie qu'il faudra une quantité plus importante du composé pour inhiber la fonction biologique de 50%.

Ainsi, d'après ces tests d'activité, les molécules de référence les plus actives sont donc la quinidine et la cinchonine. Chacune des fractions présente une activité antipaludique, les fractions 2 et 3 étant les plus actives et la fraction 6 étant la moins active (Tableau 2).

Tableau 2 - IC_{50} mesurées pour la série de données issue de l'extrait brut d'écorce de Cinchona. Les valeurs d'IC_{50} pour 5 molécules de références sont également indiquées.

Fraction	**F1**	**F2**	**F3**	**F4**	**F5**	**F6**	**Chloroquine**	**Quinine**	**Quinidine**	**Cinchonine**	**Cinchonidine**
IC_{50} (µg/ml)	0.23	0.095	0.095	0.166	6.17	8.53	0.1	0.24	0.06	0.06	0.24

Pour ces échantillons, seules des expériences TOCSY et HSQC ont été réalisées. En effet, les variations visibles au niveau des déplacements chimiques se trouvent sur les protons du noyau quinuclidine qui ressortent sur les spectres TOCSY. La zone aromatique ne présente pas beaucoup de variations entre les composés de référence, ne permettant donc pas de différencier les spectres.

Les variations de déplacements chimiques sur les spectres TOCSY sont donc ceux qui seront déterminants pour pouvoir identifier les différents alcaloïdes du mélange de départ issu de l'extrait brut de l'écorce de Cinchona. La Figure 7 montre les spectres RMN TOCSY des différentes fractions, ainsi que des composés de références, la quinidine et la cinchonine.

Pour chaque jeu de données, des fractions grossières ont été réalisées puis analysées en réalisant à chaque fois le même jeu d'expériences RMN standardisées en 1D et 2D, incluant des expériences de type COSY,

TOCSY, DOSY ou encore HSQC. Les acquisitions ont été réalisées à l'aide d'une cryosonde TCI sur un spectromètre Bruker Avance-III à 700MHz par Laure Marguerite. Les paramètres d'acquisitions ont été réglés pour le premier échantillon de la série puis conservés pour la suite.

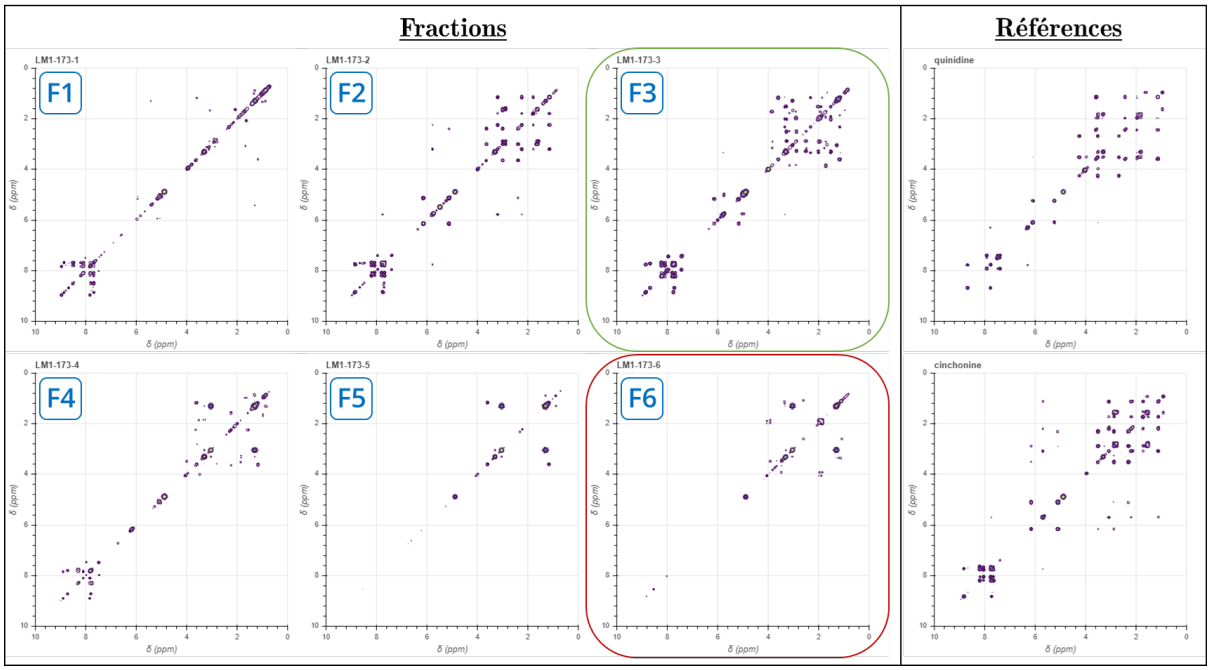


Figure 7 – Les spectres RMN TOCSY des 6 fractions réalisées à partir de l'extrait brut d'écorce de quinquina sont présentés sur le panneau de gauche. En vert la fraction avec la bioactivité mesurée la plus élevée, en rouge l'une de celle présentant la bioactivité la plus faible. A droite les spectres RMN TOCSY des deux molécules de références attendues comme responsables de la bioactivité du mélange.

Algorithmes de traitement des données

Traitement des données brutes de RMN

Nous partons donc, pour chaque jeu de données, d'une série d'expériences RMN standardisées. Chaque série d'expérience démontre une variation dans la bioactivité mesurée. Une transformée de Fourier optimisée, incluant du zéro-filling, un débruitage et une correction de ligne de base, est appliqué à chacune des expériences RMN.

Ce traitement est automatisé et optimisé de manière générale pour s'appliquer de la même manière à toutes les données, permettant ainsi de pouvoir comparer les résultats obtenus entre eux.

Différents types d'expériences RMN peuvent être réalisés et utilisés dans le cadre de l'analyse Plasmodesma. Les expériences COSY, TOCSY, DOSY, HSQC et HMBC sont actuellement implémentées dans l'algorithme.

Pour l'analyse par la méthode Plasmodesma, les concentrations des différentes fractions doivent être estimées à partir des niveaux d'activité. Dans le cas d'activité fournies en IC_{50}, s'il n'y a qu'un seul composé actif, la concentration considérée est l'inverse direct de la valeur d'IC_{50}. Une inversion de fonction sigmoïde, dont la pente et le décalage sont ajustables par l'utilisateur, est utilisée dans le cas où l'activité est fournie en pourcentage d'activité pour déterminer la concentration.

Bucketing

Le traitement classique par transformée de Fourier est complété par un traitement plus spécifique, appelé « bucketing », appliqué également avec les mêmes paramètres à toutes les fractions de chaque jeu de données.

Le bucketing consiste à analyser l'ensemble du spectre par petites zones spectrales de taille constante comme illustré sur la Figure 8.

Pour chacune des zones, certaines quantités sont calculées incluant la moyenne, le minimum, le maximum, le nombre de pics détectés ou l'écart-type. Des descripteurs supplémentaires sont calculés à partir des quantités indiquées précédemment, en particulier les valeurs logarithmiques, les sommes ou les différences. Ces quantités peuvent être utilisées en statistiques comme une approximation de l'intensité du signal dans la zone.

Cette méthode permet de réduire globalement la taille des données à analyser, tout en apportant un enrichissement de l'analyse locale. Elle protège également contre les variations locales dans les spectres entre les différents échantillons. Les buckets conservent la géométrie des spectres à partir desquels ils sont obtenus, ce qui permet de les représenter comme un spectre RMN classique.

La taille des buckets est un paramètre très important car ils doivent être assez petits pour conserver la résolution des données, mais assez gros pour permettre le lissage des variations locales et réduire la taille globale de la donnée

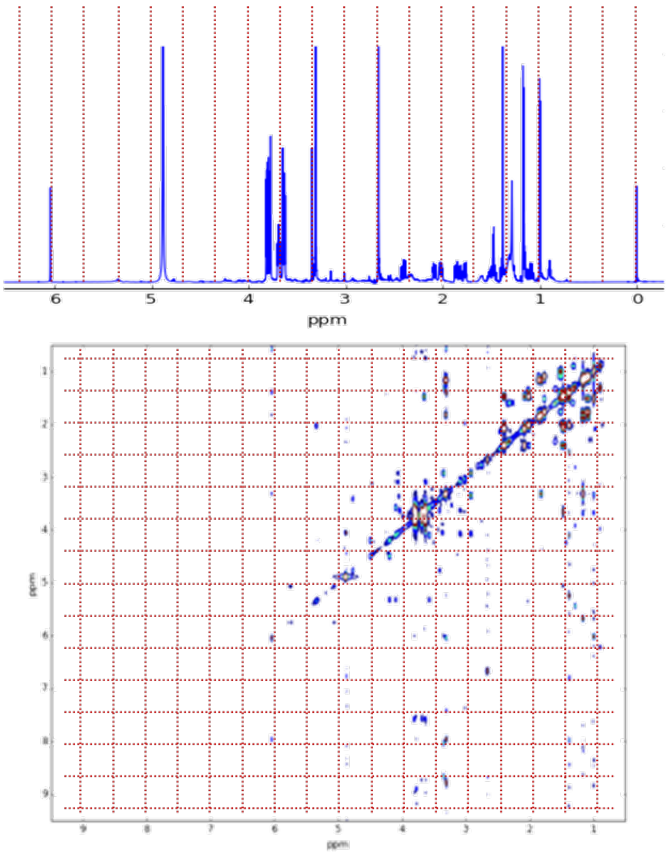


Figure 8 – Illustration schématique du bucketing (pointillés rouges) sur des spectres RMN 1D et 2D. Pour des raisons d'illustration et de lisibilité, les buckets dessinés ici sont bien plus gros qu'en réalité.

Ce paramètre a donc été optimisé dans le cadre de ce développement et différentes valeurs pour la taille des buckets ont été testées sur le jeu de données 2D synthétiquement complémenté en artémisinine. Les taille des bucket a été variée de 0.02 ppm à 0.1 ppm sur l'axe ^{1}H et de 0.5 ppm à 2.0 ppm sur l'axe ^{13}C, le Tableau 3 montrant les valeurs exactes ayant été testées.

Tableau 3 - Tailles des buckets 1H et 13C ayant été testées pour l'optimisation du traitement par bucketing sur les données synthétiquement complémentées en artémisinine.

Bucket size (ppm)	C1	C2	C3	C4	C5	C6
^{1}H 2D	0.02	0.02	0.03	0.05	0.04	0.1
^{13}C 2D	0.5	1	1	1	2	2

Pour chacun des jeux de tailles de buckets, un décompte du nombre de taches correctement proposées par la méthode a été réalisé par Laure Margueritte afin de déterminer les paramètres optimaux, pour lesquels le nombre de vrais positifs est maximal et le nombre de faux négatifs et de faux positifs sont minimisés. Ce décompte permet d'avoir les paramètres permettant une qualité de traitement des données optimale, pour lesquels le spectre proposé correspond au spectre théorique attendu de la molécule responsable de la bioactivité.

En amont du bucketing, les spectres sont débruités pour le bruit t_1, une suppression des artéfacts est réalisée ainsi qu'une symétrisation dans les cas où cela est possible et pertinent, en particulier pour les expériences 2D homonucléaires.

Analyse différentielle

L'analyse mise en place dans la première version de Plasmodesma était une analyse différentielle, réalisée entre les spectres de deux fractions en particulier. La méthode la plus efficace est de réaliser l'analyse différentielle entre les deux fractions ayant les bioactivités les plus éloignées (la fraction la plus active et la fraction la moins active). Ceci permet d'avoir un résultat plus net des différences entre les deux spectres et de faire ressortir les tâches du spectre liées aux composés responsables de la bioactivité de l'échantillon.

Cette analyse un à un a été réimplémentée dans l'outil mis en place et permet de réaliser des opérations entre deux fractions sélectionnées, en superposant une ou des références si elles sont disponibles. Deux méthodes différentes sont donc proposées, la différence et le ratio entre deux spectres.

La différence entre deux spectres consiste simplement à soustraire les canaux d'un élément statistique spécifique pour la liste de buckets respective de chaque spectre, en choisissant un spectre avec une forte activité et un spectre avec une faible activité. La méthode vise ici à éliminer les éléments présents au sein des deux spectres et à ne garder que ceux en lien avec la bioactivité de la molécule. Il s'agit d'une méthode assez peu sensible et qui donc présentait des limites lorsque la quantité de molécule active était faible.

Une seconde méthode, plus fine, a donc été testée. Il s'agit alors d'effectuer non pas une différence mais un ratio entre les deux listes de buckets pour un élément statistique défini, par exemple l'écart-type. La diagonale du spectre est alors perdue, sans que cela n'impacte grandement l'analyse puisqu'elle peut

être observée sur les différents spectres. On observe par contre une meilleure efficacité dans les cas où la concentration en molécule active est faible.

Ces deux méthodes permettent de récupérer une empreinte spectrale de la molécule active, mais ne tiennent compte que de deux des spectres de la série. On ne profite alors pas de toute l'information disponible. Il serait préférable, et la méthode en deviendrait plus robuste, d'utiliser l'entièreté de la série de données pour extraire une empreinte spectrale.

Analyse par séries de données

Une analyse plus poussée a donc été mise en place sur cette deuxième version de Plasmodesma, afin d'inclure les données globales de toutes les fractions pour l'analyse.

Il est nécessaire dans le cadre de ce type d'analyse, plus globale et incluant toute la série de données acquise, de disposer d'une mesure en lien avec la quantité de molécule active présente dans l'échantillon correspondant à chacun des spectres afin de pouvoir corréler les signaux présents dans les spectres avec l'empreinte de la molécule d'intérêt.

Régression linéaire

Une régression linéaire sur l'ensemble des spectres d'un même type est donc disponible. Ce modèle va chercher à établir une relation linéaire entre certaines variables, ici nous cherchons à relier l'activité biologique à certains descripteurs parmi les différents échantillons d'une même série. Ce processus vise à faire ressortir, après analyse, uniquement les descripteurs issus du bucketing qui vont varier linéairement en fonction de l'activité antipaludique mesurée. Le modèle est alors capable de fournir l'empreinte spectrale de la molécule responsable de la réponse antipaludique.

Elimination récursive de caractéristiques (RFE)

Tout comme pour la régression linéaire, la méthode d'élimination récursive de caractéristiques (RFE) est basée sur l'ensemble des spectres des fractions d'un même type disponible pour un mélange. L'algorithme va petit à petit réduire le nombre de caractéristiques des données à conserver en éliminant celles qui ont l'importance la plus faible vis-à-vis du modèle calculé préalablement par l'algorithme et adapté aux données. Cette méthode permet par conséquent d'éliminer de manière récursive les caractéristiques qui sont en corrélation les unes avec les autres ou encore qui ne sont pas pertinentes,

jusqu'à ne garder que le nombre de caractéristiques finales demandées par l'utilisateur. Avec ce modèle, toutes les dépendances et colinéarités présentes au sein des données vont donc être éliminées.

Dans le cas présent, les colinéarités et dépendances entre les différents spectres vont correspondre aux composés non bioactifs, qui ne sont pas en lien avec la bioactivité observée, et qui vont donc être présents dans toutes les fractions. Les tâches restantes sur le spectre, qui vont ressortir après l'application de l'analyse, correspondront donc à l'empreinte spectrale du ou des composés d'intérêt, responsables de la bioactivité du mélange de départ.
Le nombre de caractéristiques finales que l'on va désirer conserver est souvent compliqué à estimer par avance mais il est indispensable à la méthode, car il est l'objectif à atteindre. En effet, boucle après boucle l'algorithme va éliminer un petit nombre de caractéristiques et s'arrêter lorsqu'il a atteint le seuil. Pour lever cette difficulté, un curseur est disponible pour permettre à l'utilisateur de faire varier le nombre de caractéristiques à conserver, avec un résultat instantané, sur l'interface interactive mise en place et fournie sur un serveur web gratuitement. Ce curseur permet de régler cette valeur au mieux en fonction des données de l'utilisateur et des résultats observés.
Cette méthode, implémentée par la librairie scikit-learn disponible avec le langage python, est donc intégrée en tant qu'analyse interactive sur le serveur web.

Les deux méthodes de régression, RFE et régression linéaire, ont été implémentées à l'aide de la librairie `scikit-learn` associée au langage de programmation `python`. Cette implémentation permet d'inclure la totalité du jeu de données disponible, la régression étant effectuée sur l'ensemble des descripteurs disponibles après le bucketing entre chaque échantillon de la série. On a donc une approche plus globale et complète qu'avec l'approche où le ratio ou la différence est réalisée entre deux spectres choisis, idéalement le plus et le moins bioactif, pour un descripteur sélectionné.

Résultats

Nous avons donc, dans ce travail, étendu la méthode présentée précédemment en ajoutant des méthodes améliorées d'analyse des données disponibles afin d'obtenir l'empreinte spectrale de la molécule responsable de la bioactivité. Nous fournissons, par ailleurs, une analyse automatique par l'intermédiaire

d'un serveur web proposant le service et une visualisation interactive des résultats de l'analyse automatique.
Un utilisateur va donc pouvoir, simplement à travers son navigateur web, et sans aucun téléchargement ou installation de logiciel, réaliser une analyse de données lui permettant d'identifier l'empreinte spectrale de la molécule responsable de la bioactivité de son échantillon.

L'analyse se fait en deux étapes, l'utilisateur commence par déposer un dossier compressé contenant ses résultats d'analyse RMN ainsi qu'une adresse e-mail. Certains paramètres pour l'analyse de ses données sont à sélectionner au préalable afin d'appliquer l'algorithme Plasmodesma, s'appuyant sur la librairie `spike`, aux données brutes de RMN. Cette étape se fait en mode non-interactif, une fois les données déposées et les paramètres choisis, l'analyse peut être lancée. Une fois terminée les résultats d'analyses (listes de buckets, listes de pics ou encore rapports d'analyse) sont stockés sur notre serveur, pour une durée d'une quinzaine de jours.
L'utilisateur va alors recevoir un e-mail le redirigeant vers une page web privée vers une session d'analyse interactive fournissant les différents éléments évoqués précédemment appliqués aux résultats issus du traitement précédent.

Le temps requis pour effectuer la procédure énoncée va dépendre de la charge du serveur mais également de la taille des données à analyser, néanmoins, dans la majorité des cas, il sera de l'ordre de l'heure. La partie interactive est quant à elle sans délai, les changements de paramètres proposent des résultats quasi instantanés.
Les résultats des deux séries présentées sont disponibles sur le serveur web en mode interactif à titre d'exemple de cas d'utilisation.

L'analyse interactive fournit d'une part des comparaisons directes, soulignant les différences entre deux échantillons sélectionnés, comme une carte des différences (Figure 9). L'affichage des différents spectres se fait par leurs listes de buckets, qui sont utilisés comme des approximations du signal et permettent de représenter les spectres de la même manière que des spectres classiques. Cette représentation permet aux spécialistes de la RMN d'interpréter les données comme des données de RMN classiques.

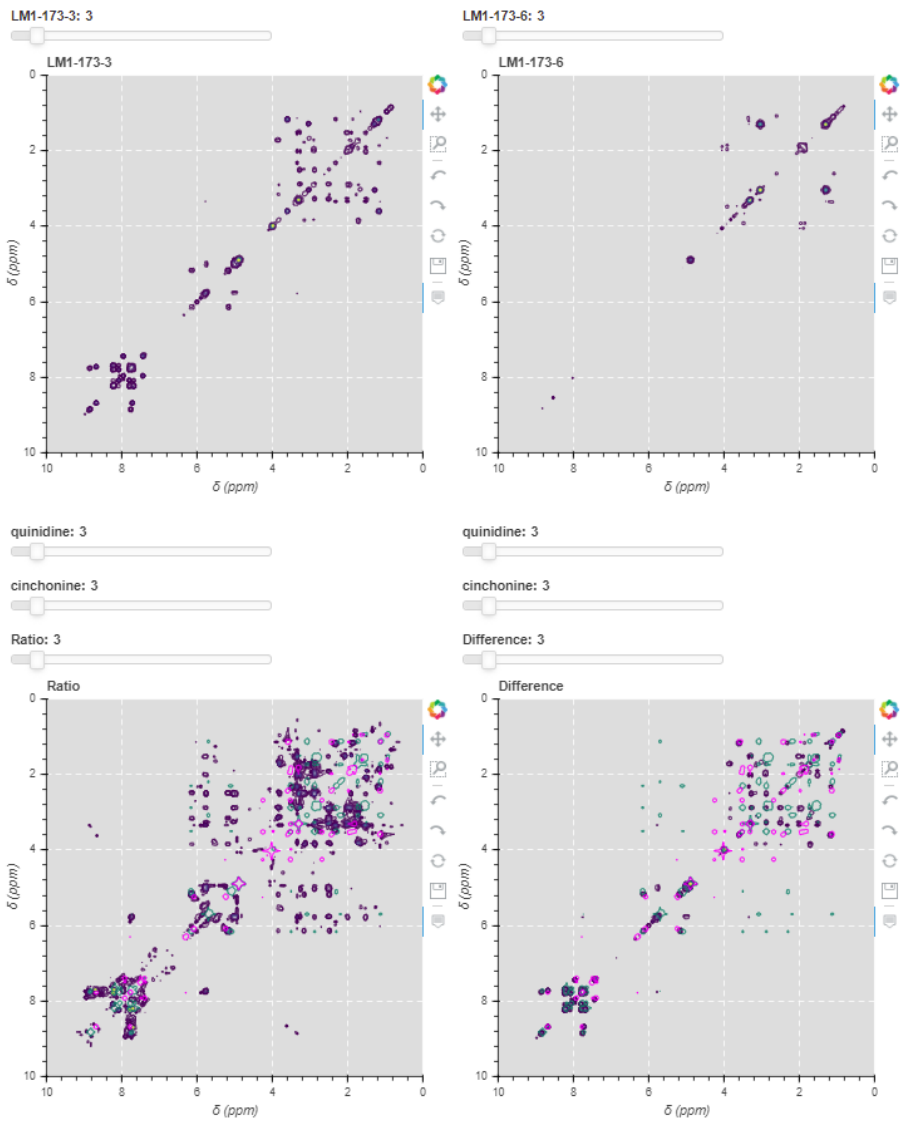


Figure 9 – Capture d'écran des résultats interactifs pour la série de données issue de l'extrait brut d'écorce de Cinchona. Est ici présenté la comparaison deux à deux de deux échantillons sélectionnés de la série (Fractions 3 et 6). Sur les deux panneaux du haut, les spectres seuls des deux échantillons sont affichés. Sur le panneau en bas à gauche est présenté le ratio des deux fractions (tâches violet foncé) ainsi que les références des molécules soupçonnées responsables de la bioactivité quinidine et cinchonine, tâches roses et vertes.

Une analyse en série complète est également disponible, en corrélant l'intensité du signal et la bioactivité de chaque fraction, en utilisant deux algorithmes d'apprentissage automatique différents : une régression linéaire classique (LR) ou une élimination récursive des caractéristiques (RFE) (Figure 10). Différents paramètres sont disponibles à la sélection par l'utilisateur soit avant l'affichage du panneau interactif, soit de manière interactive. En particulier, l'utilisateur choisis le type d'expériences à utiliser pour

réaliser l'analyse de la série (TOCSY, HSQC …) et le type d'information issue des buckets à prendre en compte (écart-type, minimum, maximum, minimum-maximum, etc…).

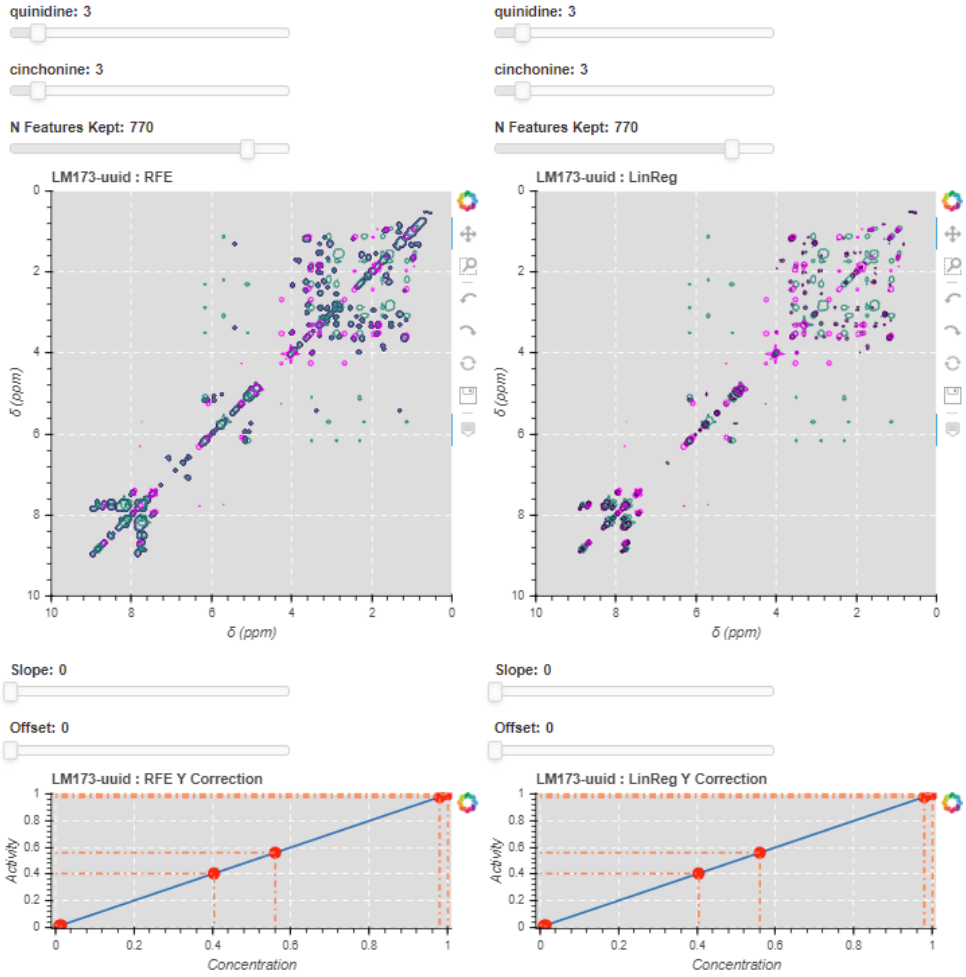


Figure 10 - Capture d'écran des résultats interactifs pour la série de données issue de l'extrait brut d'écorce de Cinchona. Est ici présenté l'analyse globale par RFE (à gauche) et RL (à droite) de la série d'échantillons. Les références sont indiquées en rose et vert sur les spectres reconstitués. Les tâches foncées correspondent ici à l'empreinte spectrale de la molécule bioactive.

Les résultats pour la série référence d'artémisinine sont présentés à titre d'exemple et visualisables de manière interactive sur le serveur web développé pour la méthode.

Le serveur web interactif développé est disponible à l'adresse suivante : https://plasmodesma.igbmc.science. Il permet donc une reconnaissance rapide de l'empreinte moléculaire des composés bioactifs dans un mélange.

Conclusions et Perspectives

La méthode Plasmodesma, permettant d'accélérer la déconvolution pharmacophorique tout en économisant les quantités d'échantillons utilisés et les manipulations nécessaire, a donc été mise en place sur un serveur web public accessible en incorporant de nouvelles méthodes d'analyse des résultats expérimentaux. Ces méthodes d'analyses de séries par ML ont montré des résultats efficaces sur les deux séries d'échantillons dont nous disposions et ont confirmé l'intérêt de cette méthode pour l'obtention rapide d'une empreinte spectrale de molécule active au sein d'un mélange.

Cette méthode pourrait notamment être utilisée en parallèle ou complément des méthodes proposées par GNPS, afin d'avoir une caractérisation rapide et relativement complète des molécules actives d'un mélange issu d'extraits naturels comprenant les molécules qui peuvent être similaires ainsi qu'une empreinte spectrale RMN.

Ce travail est présent dans le journal Faraday Discussions, dans le cadre de la collection thématique « Challenges in analysis of complex natural mixtures », sous la référence suivante :

- Automatised pharmacophoric deconvolution of plant extracts - application to Cinchona bark crude extract, Margueritte L.[‡], Duciel L.[‡], Bourjot M., Vonthron-Sénécheau C. et Delsuc M.-A. (2019) Discussions de Faraday, The Royal Society of Chemistry, 10.1039/C8FD00242H.

L'outil a donc été publié et a également été présenté dans le cadre d'une conférence de RMN à l'institut Pasteur "NMR, a tool for Biology" en Janvier 2019. Une démonstration a été réalisée et a suscité un intérêt, notamment dans le cadre d'analyses pharmacologiques.

Il présente donc une utilité dans le domaine et propose des fonctionnalités innovantes, faciles d'accès, qui pourraient être utilisées en pratique pour accélérer certains processus dans le cadre de la recherche de nouveaux médicaments.

Cependant, l'algorithme n'a pas encore été très promulgué et est resté assez peu présenté et ainsi méconnu des pairs qui pourraient en avoir l'utilité, un travail de diffusion du logiciel reste donc à faire afin d'augmenter son utilisation.

Par ailleurs, le logiciel est toujours en maintenance et des ajouts de fonctionnalités sont prévus. En particulier, il serait intéressant d'intégrer plusieurs types de spectres à la fois en fonction de ce qui est disponible (TOCSY, COSY ou encore DOSY) dans l'analyse globale de données (RFE & Régression Linéaire). En effet, dans la version actuelle du logiciel, l'analyse par RFE ou RL n'est effectuée que sur un type d'expérience choisie par l'utilisateur, l'inclusion de plusieurs types d'expériences permettrait de renforcer la robustesse et l'efficacité de l'algorithme dans l'extraction des signaux associés à la bioactivité de l'échantillon.

II. Projet Fluovial – Détection de polluants fluorés

Introduction

Le fluor est un élément commun mais non métabolisé dans les organismes vivants. Il fait partie de molécules artificielles largement utilisées dans l'industrie et les produits de consommation courante.

En effet, depuis 1938, l'entreprise américaine DuPont a commencé à commercialiser le polytétrafluoroéthylène (PTFE) d'abord comme matériau d'étanchéité puis sous la marque connue Téflon® à partir de 1949. D'après les informations des brevets, l'acide perfluorooctanoïque (PFOA) était utilisé comme agent dispersant historiquement dans la synthèse du PTFE (Beresniewicz 1986). Dès 1956 une autre société chimique, 3M, développe une solution antitache dont l'ingrédient principal est également une molécule fluorée, le perfluorooctane sulfonate (PFOS).

Les molécules perfluorées sont extrêmement résistantes et comme indiqué elles sont couramment utilisées dans la synthèse de produits industriels. Elles font partie des polluants de l'environnement et sont plus particulièrement connues sous le nom de POP (polluants organiques persistants).

Dans les années 1970, des études ont démontré la présence de composés fluorées dans l'organisme des ouvriers des entreprises évoquées précédemment ainsi que dans différentes sources environnementales comme des cours d'eau, les eaux souterraines ou les sols (Ubel, Sorenson, et Roach 1980; Giesy et Kannan 2001). La toxicité de la bioaccumulation des substances perfluoroalkylées (PFAS) a par la suite été largement démontrée dans différentes études, l'exposition à certains PFAS est suspectée d'engendrer des cancers, ou encore d'interférer avec le système endocrinien (Chang et al. 2014). Des études plus avancées sur le sujet sont toujours en cours aujourd'hui.

Historiquement utilisés principalement pour l'étanchéité ou la propriété antiadhésive de produits du quotidien, et bien que plus contrôlés et régulés, des composés de la famille des PFAS sont toujours utilisés, pour leurs propriétés physico-chimiques, dans l'industrie et certains produits de consommation. Des composés fluorés sont également encore aujourd'hui présents dans une majorité de produits phytosanitaires et dans un grand nombre de médicaments.

La persistance des PFAS dans l'environnement implique que tant que leur rejet dans l'environnement continuera toutes les espèces vivantes seront exposées à des concentrations toujours plus élevées. Cela

signifie également que même si tout rejet s'arrêtait, les PFAS continueraient d'être retrouvés dans l'environnement et les êtres vivants pendant encore plusieurs générations.

Des normes et directives ont été mises en place pour limiter la quantité de molécules fluorées présentes dans l'environnement. L'UE a déjà restreint depuis plus de 10 ans l'utilisation de l'acide perfluorooctane Sulfonique (PFOS), de ses sels et fluorure de perfluorooctane sulfonyle dans le cadre du règlement concernant les POP. Par ailleurs, le PFOS et ses dérivés ont été inclus depuis 2009 dans la convention internationale de Stockholm afin d'éliminer leur utilisation. Cette convention interdit également l'utilisation du PFOA et de ses composés apparentés depuis le 4 Juillet 2020. Différentes restrictions ont été proposées et sont en cours d'étude pour être ajoutées à la réglementation REACH, de nouvelles propositions portées par différents pays sont également en cours de rédaction et seront proposées relativement rapidement. Les réglementations concernant ces substances sont donc en pleine évolution.

Néanmoins, à ce jour, aucune technique complète n'existe pour détecter, identifier et quantifier ces polluants persistants fluorés. La méthode utilisée classiquement couple la chromatographie liquide (LC) et la spectrométrie de masse (MS ou MS – MS). Bien que présentant une très bonne sensibilité, la méthode a aussi de nombreux inconvénients, en particulier la nécessité de standards de références, le besoin d'analyses ciblées mais aussi les spécificités de préparation de l'échantillon impliquant des traitements qui peuvent en altérer son contenu.

La RMN du fluor se présente comme une méthode alternative et complémentaire pour l'identification et la quantification de molécules fluorées. En effet, la RMN est une technique offrant une très bonne sensibilité (1 µM) et permettant de tout détecter de manière non ciblée, tous les fluors sont donc observables, mêmes s'ils ne sont pas spécifiquement recherchés.
Néanmoins, cette technique nécessite une attribution des spectres post-acquisition, ce qui est non trivial puisque d'une part les PFAS présentent de nombreuses raies différentes sur les spectres de RMN, et d'autre part la position de ces raies est dépendante de l'environnement et des conditions physico-chimiques lors de l'acquisition.
De plus en présence de mélanges de PFAS, ce qui est le cas pour la plupart des échantillons « pratiques » issus d'extraits de sols, les positions de chacune des raies des composés présents dans le mélange peuvent

être amenées à être influencées par les autres composés, et il faut être capable de dissocier les différents composés dans le spectre.

Une analyse assistée par le développement d'un algorithme de ML pourrait donc permettre une attribution plus efficace des spectres de RMN visant à détecter les polluants fluorés, dans un premier temps afin d'identifier les spectres de chacune des molécules fluorées, et dans un second temps pour attribuer les spectres de mélanges de PFAS.

Projet

Le projet FLUOVIAL, financé par l'ANR et mené conjointement par l'équipe de RMN de l'IGBMC et CASC4DE vise donc à caractériser et quantifier les molécules fluorées dans des échantillons environnementaux grâce à la RMN ^{19}F.

Ce projet sert de point de départ au projet IPANEMA, financé par l'ADEME ayant pour objectif de caractériser les mélanges de PFAS présents dans les sols d'une zone d'entraînement de pompiers, dont la localisation est tenue confidentielle, où des mousses anti-incendie à base d'hydrocarbures contenant des cocktails de PFAS ont été utilisées.

Ceci permettra de pouvoir prévoir leur devenir dans les sols et eaux ainsi que l'exposition des petits organismes vivants dans les sols comme les vers de terre.

Du point de vue de CASC4DE, le projet a été envisagé et réalisé en plusieurs phases :

- Enregistrement de jeux de données de RMN ^{19}F de composés fluorés connus
- Application de l'algorithme Plasmodesma spécifiquement adapté aux données fluorées
- Développement d'un algorithme d'apprentissage machine sur les spectres prétraités pour effectuer une classification des spectres en fonction de la molécule analysée
- Application de l'algorithme entraîné à des mélanges inconnus pour détecter et identifier les molécules fluorées présentes

Cette partie du projet mené par CASC4DE est complété par des compétences et éléments fournis et réalisés par les partenaires du projet, en particulier concernant la récupération d'échantillon ainsi que l'étude d'impact environnemental.

Matériel et Méthode

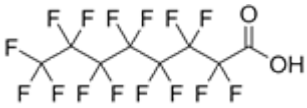
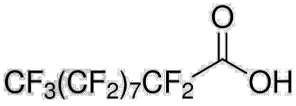
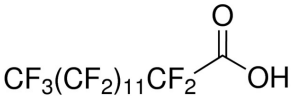
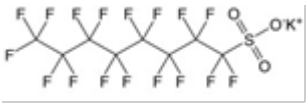
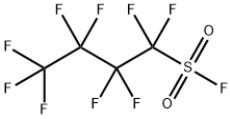
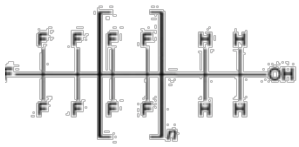
Dans le cadre de la thèse de doctorat, le travail a principalement porté sur le développement de l'algorithme qui est utilisé pour l'analyse, la détection et l'identification des composés fluorés.

Données

Un jeu de données variées a été acquis dans un premier temps afin d'avoir des résultats préliminaires démontrant l'efficacité de la méthode.

Ce jeu de données était composé de 6 molécules (Tableau 4) connues pour être des composés sources de molécules fluorées persistantes dans l'environnement : PFOA, PFTDA, PFBSF, PFDA, PFOS et FTOH.

Tableau 4 - Table des molécules de la base de données initiale du projet FLUOVIAL et leur formule topologique.

PFOA 95% *perfluorooctanoic acid* **335-67-1**		**PFDA** *Perfluorodecanoic acid* **335-76-2**	
PFTDA *Perfluorotetradecanoic acid* **376-06-7**		**PFOS 98%** *perfuorooctanesulfonate de K* **2795-39-3**	
PFBSF 90% *nonafluorobutansulfonyl fluoride* **375-72-4**		**FTOH** *Fluorotelomer alcohol*	

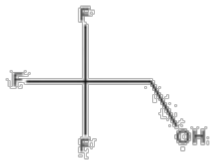
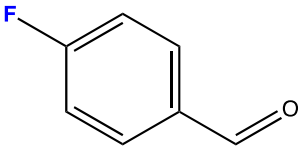
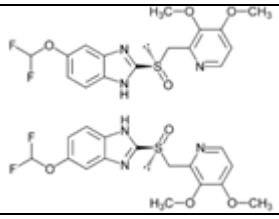
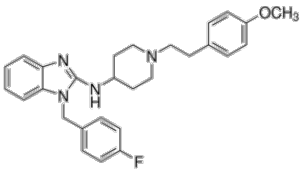
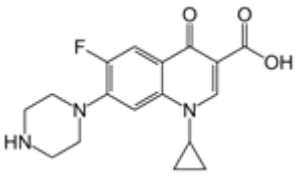
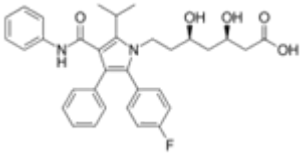
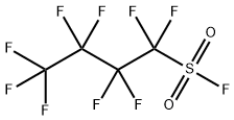
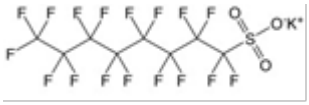
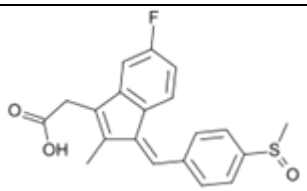
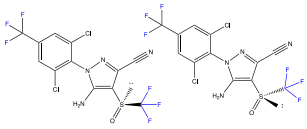
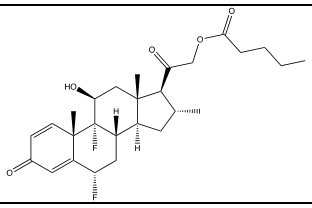
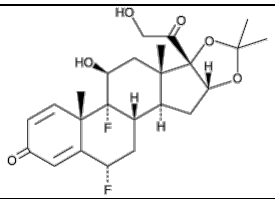
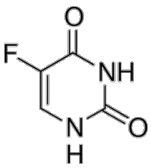
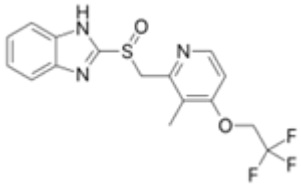
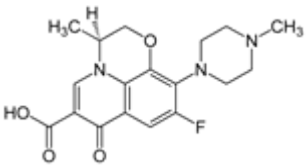
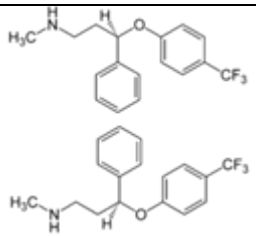
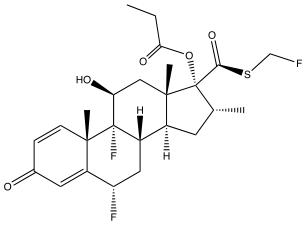
Les spectres RMN ont été effectués pour différentes concentrations allant de 10^{-2} à 10^{-6} M dans deux solvants : DMSO et H_2O/D_2O.

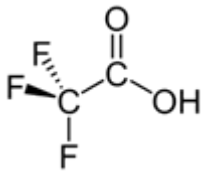
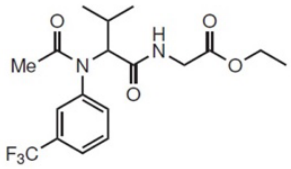
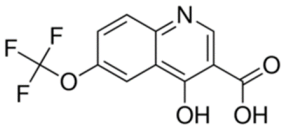
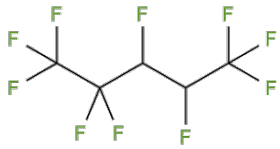
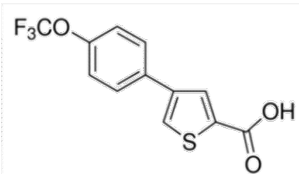
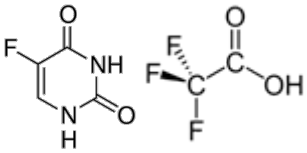
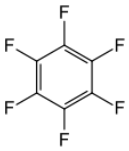
Les premières implémentations d'algorithmes de prédiction ayant démontré des résultats prometteurs, ce jeu de données relativement petit a été repris et augmenté.

Des acquisitions supplémentaires de spectres RMN sur les molécules additionnelles ont donc été réalisées par CASC4DE.

Les nouvelles molécules constituant la base de données utilisée par la suite et présentées dans le Tableau 5 comprennent différents composés connus comme responsables de pollution persistante fluorée.

Tableau 5 - Table des molécules présentes dans la base de données d'entraînement et leur formule topologique.

PFOA 95% *perfluorooctanoic acid* **335-67-1**		**TFE** *trifluoroéthanol* **75-89-8**	
4-Fluorobenzaldehyde **459-57-4**		**Pantoprazole** **102625-70-7**	
Astémizole **LP03** **68844-77-9**		**Ciprofloxacine** **85721-33-1**	
Atorvastatin /Tahor **134523-00-5**		**PFBSF 90%** *nonafluorobutansulfonyl fluoride* **375-72-4**	
PFOS 98% *perfuorooctanesulfonate de K* **2795-39-3**		**SULINDAC** **38194-50-2**	
Fipronil **120068-37-3**		**Diflucortolone valerate** **59198-70-8**	
Fluocinolone acetonide **67-73-2**		**5-fluorouracil** **51-21-8**	
Lanzoprazole **103577-45-3**		**Levofloxacine** **100986-85-4 (S(–))**	
Fluoxetine **54910-89-3 (racémique)**		**Fluticasone propionate** **80474-14-2**	

TFA *Acide trifluoroacétique* **76-05-1**		**Glycine, N-acetyl-N-[3-(trifluoro methyl)phenyl]valyl-, ethyl ester** **379685-94-6**	
4-hydroxy-6-(trifluoromethoxy) quinoline-3-carboxylic acid **175203-86-8**		**2H,3H-perfluoropentane** **138495-42-8**	
4-[4-(Trifluoromethoxy) phenyl]-2-thiophenecarboxylic acid **666721-06-8**		**MIX : Fluorouracile/TFA**	
C6F6 **Hexafluorobenzene** **392-56-3**			

Les acquisitions des spectres RMN ont été réalisées sur un spectromètre Bruker Avance 600, équipé d'une cryosonde QCI-F et opérant à 600.16 MHz, associé au logiciel topspin 2.1.

Certaines des données utilisées sont des données non acquises expérimentalement dans le cadre du projet mais issues de la littérature dans le domaine. Les constantes de couplages ainsi que les déplacements chimiques détectés ont été relevés et enregistrés au format `csv` puis des fid de spectres RMN ont été resimulés dans un format Bruker classique à l'aide de la bibliothèque `spike`. Les spectres ainsi obtenus sont utilisables de la même manière que des spectres de RMN classiques et peuvent donc être traités et inclus dans la base de données.

Une des difficultés expérimentales était de pouvoir réaliser l'excitation du spectre entier en une seule fois. Ceci n'est pas possible par l'utilisation d'une séquence d'excitation standard sur le spectromètre Bruker 600MHz dont nous disposions et qui a été utilisé pour générer les données. Cependant la parution de la séquence d'acquisition « OPERA » a permis de lever cette difficulté en travaillant sur une très grande largeur spectrale en une seule expérience (Coote, Bermel, et Arthanari 2021). Cette technique a été utilisée pour réaliser une nouvelle acquisition de spectres pour les molécules dont nous disposions afin d'uniformiser toutes les données.

A. Belqasmi a réalisé un stage de master 2, ayant permis de réaliser une augmentation de la base de données en ajoutant synthétiquement différents types d'artéfacts et du bruit aléatoire aux données disponibles pour en créer de nouvelles et enrichir la base disponible. En effet, 3 types d'ajouts ont été réalisés, comme illustré sur la Figure 11.

Dans un premier temps, un artéfact dans le fid a été implémenté sous forme d'un pic « aberrant » ayant pour effet après transformée de Fourier de faire osciller la ligne de base dans le spectre final.

Ensuite, des artéfacts dans le spectre lui-même sont implémentés sous forme de pics plus ou moins intenses supplémentaires dans le spectre.

Enfin, le bruit blanc gaussien naturellement présent dans les spectres de RMN a été augmenté artificiellement sur certains fid de la base de données, menant à des spectres plus bruités qu'à l'origine, permettant ainsi à l'algorithme d'être robuste à des spectres avec un taux de bruit important.

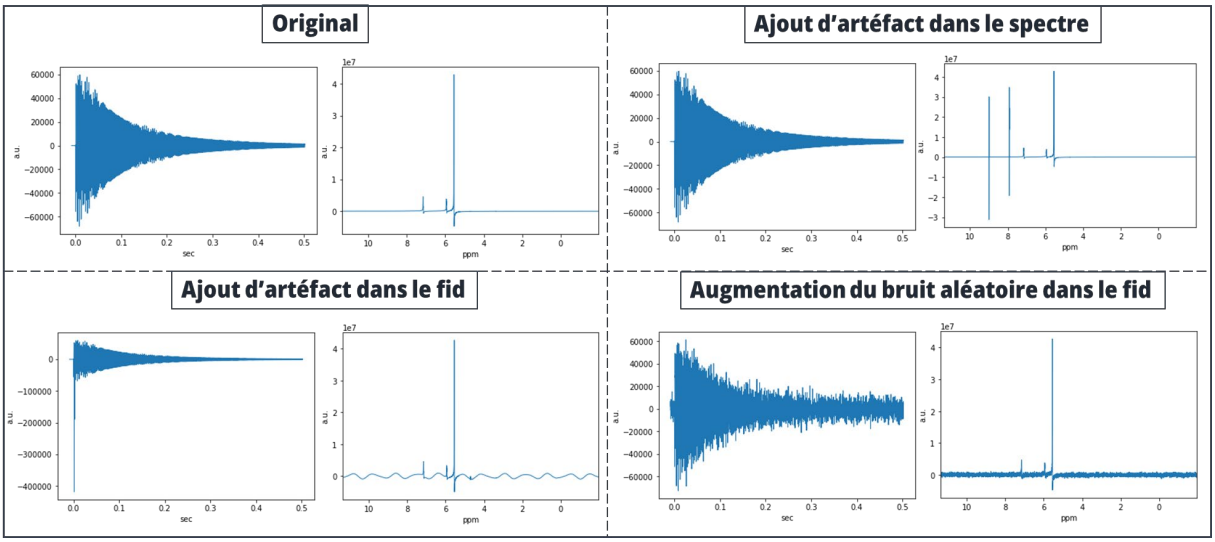


Figure 11 - Illustration des différents artéfacts et bruit ajoutés synthétiquement dans les données disponibles.

Cette base de données reste néanmoins une base de données déséquilibrée, le nombre d'éléments par classes est relativement variable, cet aspect est donc à prendre en compte pour le choix et l'évaluation du modèle de ML appliqué. Elle continue par ailleurs d'évoluer au fil de l'acquisition de nouveaux spectres et de la récupération de données de littérature, permettant d'améliorer la robustesse et la capacité de l'algorithme à reconnaître une plus grande variété de molécules.

Traitement des données

Les données sont ensuite traitées en prévision des nécessités de l'algorithme d'apprentissage machine qui sera mis en place.

Une version alternative de l'algorithme Plasmodesma présenté précédemment, spécialement adaptée aux données RMN de molécules fluorées, est dans un premier temps appliquée aux données brutes permettant une harmonisation des données. Le bucketing appliqué réduit une première fois la taille globale des données, tout en les enrichissant puisque nous disposons alors de plusieurs données statistiques pour chacun des buckets.

Dans le cadre de l'application de l'algorithme Plasmodesma sur les données associées à ce projet, la taille des buckets réalisés a été ajustée afin de les optimiser au mieux. Cette optimisation a été réalisée en évaluant la qualité du spectre reconstruit à l'aide d'une expertise dans le domaine RMN. Les tailles de buckets utilisées sont présentées dans le Tableau 6.

Tableau 6 – Tailles des buckets utilisés lors de l'application de Plasmodesma pour le prétraitement des données pour les noyaux ^{1}H, ^{13}C et ^{19}F pour les expériences RMN en 1D et 2D.

	^{1}H 1D	^{1}H 2D	^{13}C 1D	^{13}C 2D	^{19}F 1D	^{19}F 2D
Bucket size (ppm)	0.01	0.03	0.03	1.0	1.0	1.0

Des descripteurs supplémentaires ont été ajoutés à ceux décrits précédemment dans la partie dédiée à Plasmodesma. Ces descripteurs sont les coefficients d'asymétrie (skewness) et d'aplatissement (kurtosis). Ces moments statistiques permettent de décrire la forme de la distribution au sein d'un bucket, la skewness permettant de décrire l'asymétrie de la queue de la distribution et d'indiquer si elle est plus allongée à droite ou à gauche et le kurtosis décrit la forme plus ou moins en pointe de la distribution.

Ces descripteurs enrichissent encore le jeu de données dont on dispose et sont des éléments qui, de par leur ajout d'informations sur la distribution des données dans un bucket, peuvent permettre à l'algorithme de prise de décision d'être plus efficace. En effet, plus les données sont détaillées, plus il est facile de discriminer différentes classes de molécules par leur spectre.

Ils sont par ailleurs des descripteurs sensibles à l'ordre des points dans le bucket, ce qui n'est pas le cas des autres descripteurs dont nous disposions jusqu'alors. Cette spécificité les rend donc porteurs d'une information importante, qui était perdue jusqu'à présent dans l'étape de traitement de données.

Pour permettre une analyse plus efficace des données, une étape supplémentaire de réduction de dimension est effectuée. La réduction de dimensionnalité comprend un ensemble de techniques qui permettent de projeter un ensemble de données de grande dimension dans un espace plus restreint, cela réduit donc le nombre de données qui seront à fournir en entrée de l'algorithme d'apprentissage automatique. Différentes techniques ont été testées afin de déterminer laquelle était la plus adaptée au jeu de données dont nous disposons.

En particulier, des algorithmes d'analyse en composante principale (ACP), t-SNE (t-distributed stochastic neighbor embedding) et PHATE (Potential of Heat diffusion for Affinity-based Transition Embedding) ont été appliqués aux données.

L'ACP est une technique de réduction de dimension linéaire basée sur la décomposition en valeur principale. Cette méthode cherche des combinaisons linéaires de la matrice des caractéristiques des données d'origines pour en construire une représentation significative de dimension réduite. L'ACP permet donc de conserver la structure globale du jeu de donnée d'origine.

La méthode t-SNE est également non supervisée mais cependant très différente d'algorithmes comme l'ACP. Elle est non-linéaire et se base sur la construction d'une distribution de probabilité définie sur les paires de données d'origine telle que des données similaires donnent une forte probabilité dans la distribution et des données très différentes donneront une probabilité très faible (Maaten et Hinton 2008). Une seconde distribution est également générée pour l'espace réduit. L'algorithme cherche ensuite à minimiser la divergence entre les deux distributions en rapport avec les données fournies. Cette technique permet donc de conserver la structure locale des données de grandes dimensions, parfois au prix de la structure globale.

L'algorithme PHATE est, comme t-SNE, une méthode non linéaire et non supervisée (Moon et al. 2019). Néanmoins, à l'inverse de t-SNE, va permettre de conserver à la fois les relations locales et globales des données d'origine. Pour réaliser cela, l'algorithme se base sur le calcul d'affinités entre les différents points du jeu de données (plus ils sont similaires, plus l'affinité est haute et inversement). L'affinité capture donc les structures locales. Par ailleurs, un calcul des probabilités de diffusion de chaque donnée dans le jeu de données est calculé pour déterminer une structure plus globale.

Après avoir testé ces trois méthodes sur le jeu de donnée disponible, avec ou sans normalisation des échantillons au sein du jeu de données, aucune d'entre elles ne s'est montrée véritablement efficace pour séparer les différentes classes de molécules présentes (Figure 12).

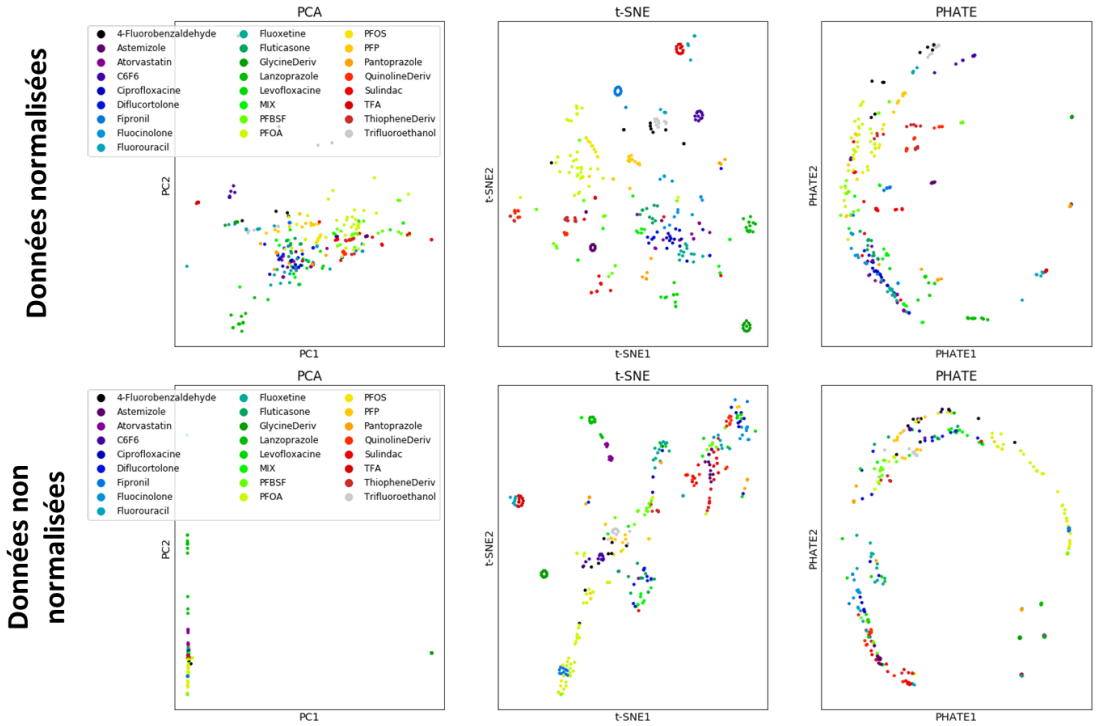


Figure 12 - Résultats de l'application des méthodes de réduction de dimensionnalité ACP, t-SNE et PHATE sur les données de RMN des molécules fluorées représentées par différentes couleurs. La projection est réalisée sur une dimension en 2D en partant des données après bucketing et après une réduction de dimension par RandomForest régresseur, en réalisant (ligne haute) ou non (ligne basse) une normalisation des données au préalable.

Une régression par Random Forest (RF) a finalement été réalisée, afin de déterminer l'importance des caractéristiques des données. Dans le cas présent il s'agissait de déterminer quelle valeur statistique issue d'un bucket particulier a une importance considérable dans la classification supervisée (par molécule)

des données. Une classification supervisée consiste à avoir des données d'entrée et de sortie qui sont étiquetées, contrairement à la classification non supervisée où l'on n'a pas d'information sur les types de données à classifier.

Une fois la classification des caractéristiques effectuée, nous n'en gardons qu'une liste réduite comme entrée pour l'algorithme d'apprentissage automatique mis en place ensuite (Figure 13). Il est a noter que dans la liste de descripteurs les plus utilisés par l'algorithme pour l'identification de molécules, la skewness et le kurtosis sont beaucoup représentés. Ceci est un indicateur de la pertinence de ces descripteurs et montre qu'ils portent des informations importantes dans la caractérisation du spectre RMN de la molécule.

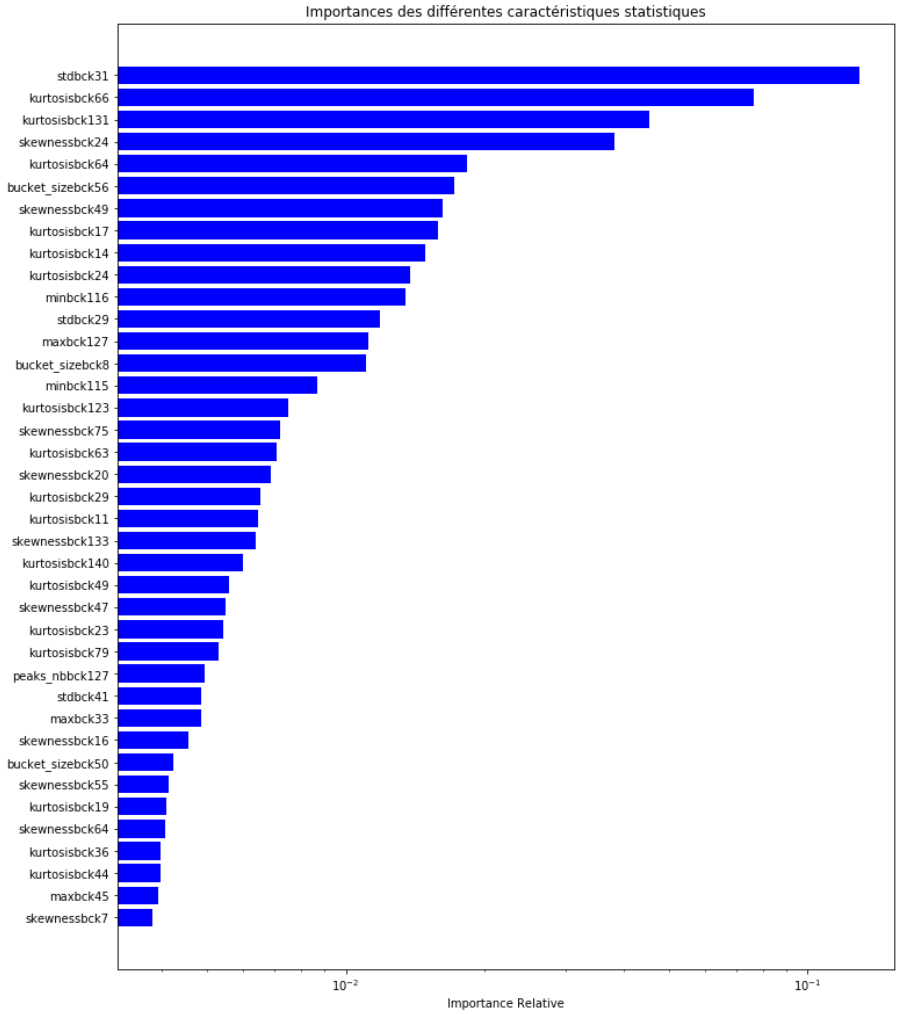


Figure 13 - Ce graphique montre l'importance relative des caractéristiques des données utilisées pour classifier les spectres de molécules fluorées obtenue par une régression par Random Forest. Il n'est représenté ici que le haut de cette liste. Le nom de chaque caractéristique est composé comme suit : [abréviation de la valeur statistique]bck[numéro du bucket dans le spectre].

Algorithme

La construction de l'algorithme de classification des données s'est réalisée en plusieurs étapes.

Il a tout d'abord été question de comparer les différents algorithmes de classification potentiels pour l'analyse de l'ensemble de données.

Des versions générales, sans optimisation des paramètres, d'algorithmes de classification ont donc été mises en place et testées sur une partie des données présentées ci-dessus.

En effet, seules les molécules pour lesquelles au moins 10 exemples étaient disponibles ont été conservées lors de cette étape car pour moins d'éléments, les résultats risquent de ne pas être fiables.

En particulier, nous avons comparé les méthodes suivantes : k-Nearest Neighbors (k-NN), Support Vector Machine (SVM), Régression logistique, Random Forest, Gradient Boosting et enfin une combinaison de ces différentes méthodes (Figure 14).

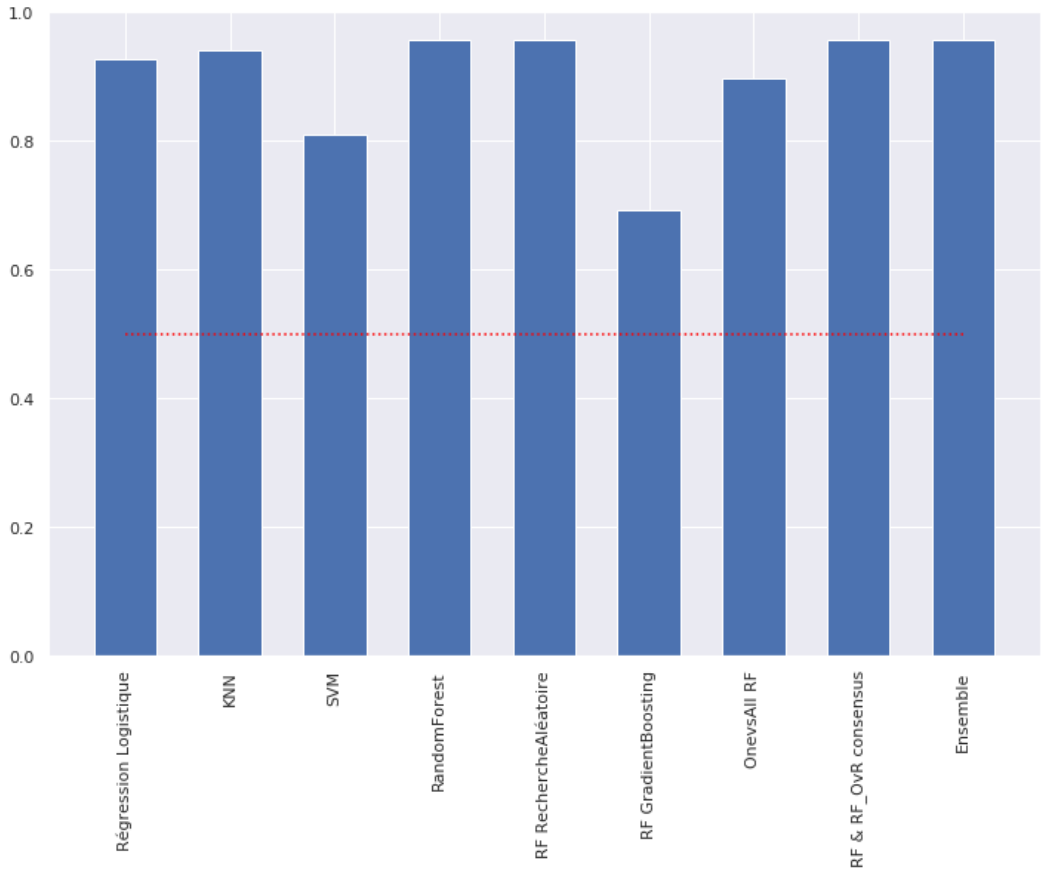


Figure 14 - Comparaison des taux de prédictions vraies de différents algorithmes de classification sur les données du projet FLUOVIAL.

L'algorithme "k-Nearest Neighbors", ou k-NN, est une technique de classification non paramétrique qui peut être utilisée en tant que classifieur ou régresseur en fonction des besoins (M.-L. Zhang et Zhou 2007). Dans le cadre d'une classification, la méthode consiste à classer un élément nouveau en fonction des k éléments les plus proches en termes de distance.
La classe attribuée est donc celle qui est majoritaire parmi les k éléments proche. Le choix du nombre de voisins à prendre en compte pour le vote de la classe est donc le paramètre central de la technique. L'algorithme peut être très impacté si les données ne sont pas normalisées, ce qui n'est pas le cas ici. Par ailleurs, la précision de l'algorithme peut beaucoup varier en cas de présence de bruit par exemple. Un déséquilibre de la quantité d'éléments par classe au sein du jeu de donnée peut également entraîner des erreurs de prédictions.

La Support Vector Machine, ou SVM, est une méthode de classification non-probabilistique et qui peut ou non être supervisée (Mathur et Foody 2008). Dans le cas ici présent, nous l'utilisons de manière supervisée. L'algorithme va construire un modèle pour séparer les différentes classes dans une hypersurface de grande dimension. L'objectif étant de construire ce modèle en maximisant les marges entre les différentes classes. Pour chaque nouvel élément à classifier, ce dernier sera placé dans l'espace et sa classe déduite de sa position dans l'espace parmi les éléments connus.

La régression logistique est quant à elle également un modèle linéaire, couramment utilisé en apprentissage automatique pour classifier (Tolles et Meurer 2016). Cette méthode consiste à obtenir à partir des données un modèle qui va s'adapter au mieux au jeu de donnée et permettre de définir des seuils pour décider de la classe de l'élément à prédire.

La Random Forest, ou RF, est un algorithme d'apprentissage supervisé, avec tous les éléments du jeu de données étiquetés (Ho 1995). L'algorithme est basé sur de multiples arbres de décisions à partir desquels une prédiction finale est faite par un vote entre chaque arbre individuel. Un arbre de décision est construit de nœuds de décision. Pour chacun de ces nœuds, la valeur d'une des caractéristiques des données est testée. En fonction de si celle-ci est en dessous ou au-dessus d'une valeur donnée on passe au nœud suivant correspondant. L'impureté de Gini, ou index de Gini, (GI) est utilisée pour déterminer la caractéristique à utiliser pour le nœud suivant. L'impureté de Gini représente la probabilité qu'une

caractéristique choisie aléatoirement soit mal classée. La GI varie entre 0 et 1, donc si GI=0 la classification est dite pure, signifiant que chaque élément appartient à une classe spécifique. Pour le nœud suivant, la séparation avec la plus petite valeur de GI est donc sélectionnée, jusqu'à ce que GI=0 et que la classe soit déterminée par l'arbre ou lorsqu'on a atteint la profondeur maximum de l'arbre en fonction des paramètres choisis (Figure 15). Une Random Forest est donc composée d'un certain nombre d'arbres de décisions, également déterminé par les paramètres de l'algorithme, chaque arbre ayant un nœud de départ différent défini aléatoirement. La classe de l'élément est ensuite déterminée par un vote à la majorité parmi tous les arbres de la Random Forest.

La RF permet d'associer explicitement des caractéristiques spécifiques qui déterminent la classification, impliquant une explicabilité relativement facilement exploitable des résultats de la classification par apprentissage automatique.

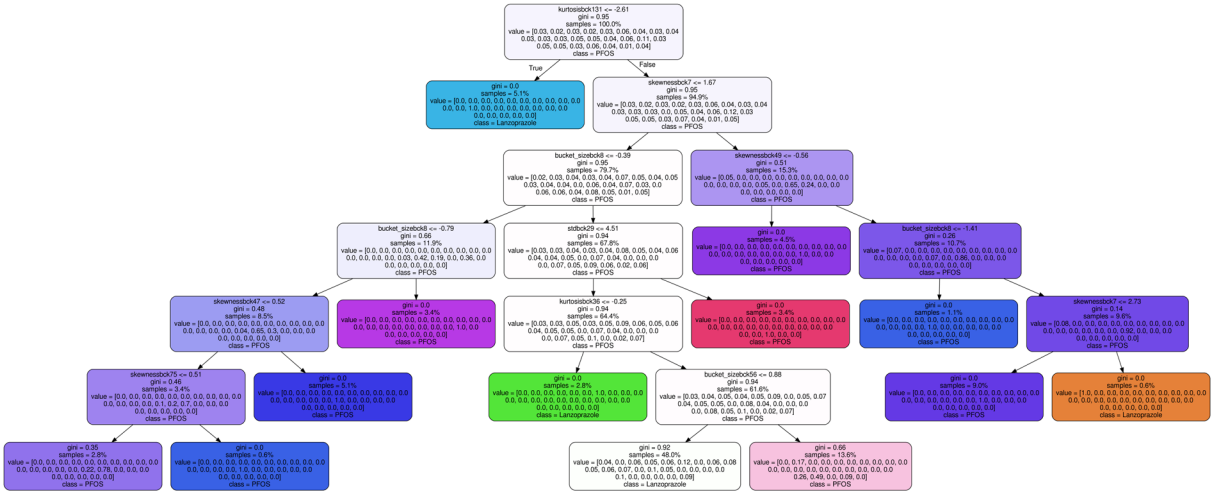


Figure 15 - Exemple d'un arbre de décision issu d'une Random Forest appliquée aux données du projet FLUOVIAL. Cet arbre part de la valeur du bucket "kurtosisbck131" et classe les différentes données en fonction des valeurs consécutives de certains buckets.

Une méthode par « Gradient Boosting » RF a également été testée. Le principe est d'ajouter une minimisation de la fonction de perte par descente de gradient (Natekin et Knoll 2013).

Contrairement à la RF « classique » qui génère les arbres indépendamment les uns des autres et procède à un vote final, avec une méthode par Gradient Boosting les arbres sont construits un à la fois, en combinant les résultats au fur et à mesure, permettant d'améliorer progressivement les arbres au fil de l'entraînement. Les méthodes de Gradient Boosting peuvent permettre après une bonne optimisation d'obtenir de meilleurs résultats.

Cependant, elle est peu résistante au bruit qui entraînera rapidement de l'overfitting et leur paramétrage est plus influant sur les performances de l'algorithmes.
La combinaison d'algorithme, aussi appelé « Ensemble », consiste simplement à combiner plusieurs algorithmes de classification et à réaliser une décision par un mécanisme de vote consensus en prenant en compte chacun des résultats des algorithmes.

L'algorithme Random Forest, RF, a finalement été choisi dans le cas présent car il a fourni de bons résultats préliminaires et qu'il permet par ailleurs de tirer des explications à la classification réalisée.
Par ailleurs, nous sommes en présence d'une base de données non équilibrée, en effet, toutes les classes ne sont pas représentées dans la même quantité. L'avantage de la RF vis-à-vis de ce problème est qu'elle y est assez résistante. Ce type d'algorithme permet même de corriger légèrement le problème par construction. En effet, la RF sélectionne aléatoirement les caractéristiques à tester pour chaque séparation, en optimisant le point de séparation pour chaque nœud. Ce mécanisme implique une probabilité pour que la RF génère des points de séparation tenant compte du déséquilibre des données.

La RF est un algorithme classique de classification multi-classes qui est entrainé pour reconnaitre différentes classes à la fois. Cependant, pour les cas non binaires, plus le nombre de classes est élevés, plus il y a de risques de confusions de l'algorithme.
En effet, dans le cas d'algorithme de classifications multi-classes, il y a d'autres possibilités supplémentaires pour l'algorithme à mettre en place, en particulier la technique un contre tous ou un contre le reste, aussi appelé OvR pour « One vs. Rest » en anglais (Ramírez et al. 2018).

Cette méthode consiste à mettre en place différents classificateurs au lieu d'un seul, pour renforcer la reconnaissance des différentes classes spécifiques. Dans le cas du « un contre tous », il va alors falloir générer autant de classificateurs binaires que de classes à reconnaitre. Chaque classificateur est dans ce cas entrainé à déterminer si l'élément fait partie d'une classe spécifique ou non.

Ce type d'algorithme se trouve être très adapté à la problématique finale du projet qui vise à être capable d'identifier les molécules présentes dans des mélanges.

En ayant donc à notre disposition un set de classificateurs binaires entrainés à reconnaitre si oui ou non une molécule spécifique est présente dans un spectre, il suffit de les appliquer de la même manière sur un mélange que sur un spectre de molécule seule et de réaliser l'apprentissage sur des mélanges, puis de simplement revoir légèrement la manière d'interpréter la sortie de l'algorithme.

Deux algorithmes ont été mis en place et comparés sur la base de données du projet FLUOVIAL. Un algorithme de RF « classique » et un algorithme de RF par méthode OvR.
En effet, la librairie scikit-learn permet de réaliser une méthode « un contre tous » en utilisant comme modèle de départ de chaque classificateur un modèle de machine learning spécifique.
Ici donc une RF a été développée pour la reconnaissance de l'ensemble des différentes classes de molécules.
Ce modèle a été utilisé pour la mise en place de la méthode OvR où chacun des classificateurs reconnait donc une seules des molécules de la base de données, avec un vote final permettant d'obtenir une prédiction.

Optimisation

L'étape suivante est l'optimisation de l'algorithme choisi pour en tirer les meilleurs résultats, qui peut revêtir de multiples facettes, allant du choix des données à inclure dans l'apprentissage à l'augmentation de la base de données, mais avant tout en optimisant les hyperparamètres de l'algorithme.

Les hyperparamètres d'un algorithme d'apprentissage automatique sont ceux utilisés pour contrôler le processus d'apprentissage, pour une forêt aléatoire il y a par exemple le nombre maximum d'arbres ou la profondeur maximum pour chaque arbre.
Au niveau de l'optimisation, l'idée est de faire varier les valeurs des paramètres à travers une grille ou une recherche aléatoire (Bergstra et Bengio 2012). Cela permet de trouver une combinaison donnant les meilleurs résultats pour le jeu de données testé (Figure 16).

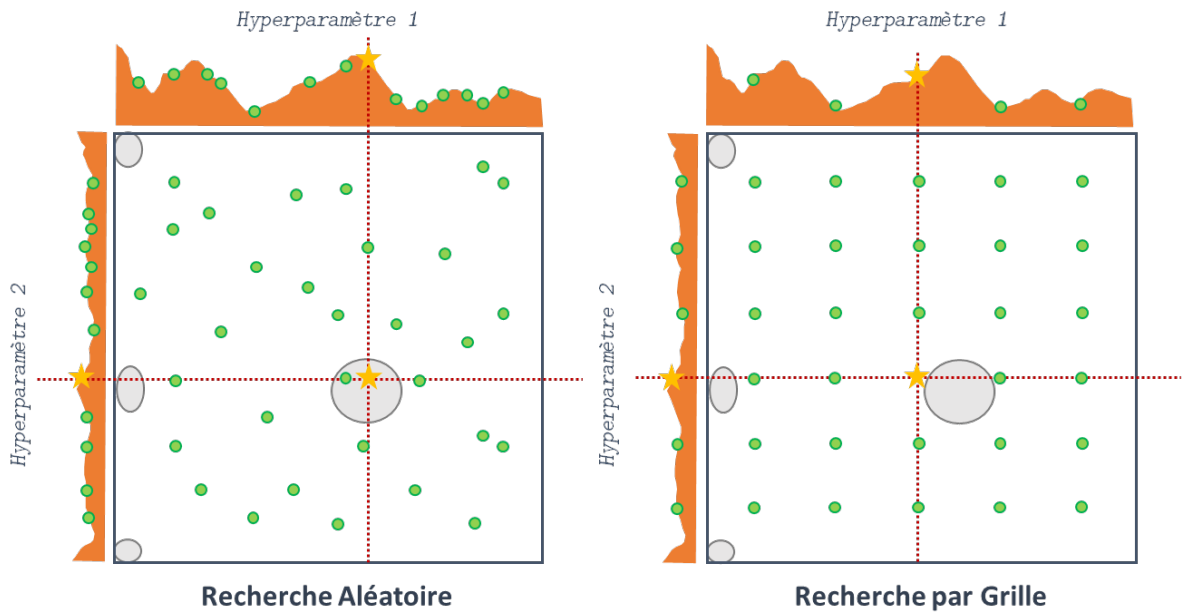


Figure 16 - Illustration des recherches d'hyperparamètres aléatoire et par grille. En vert les jeux de paramètres testés, en gris les zones optimales des paramètres. L'étoile représente le jeu de paramètre qui serait selectionné dans chaque cas. Adapté de Bergstra et Bengio 2012.

La recherche aléatoire d'hyperparamètres consiste à tester des jeux de paramètres sélectionnés aléatoirement, on va donc pouvoir trouver des hyperparamètres donnant de meilleurs résultats, mais nous pouvons possiblement passer à côté des paramètres optimaux, mais aussi tomber dessus très rapidement avec peu de jeux de paramètres testés. La méthode demande d'entrer des séries de paramètres à tester, et n'en sélectionnera que certaines paires qui seront effectivement testées.

La recherche sur grille est souvent plus lente, car on va devoir choisir des jeux de paramètres nous permettant de balayer une surface de possibilité complète afin de trouver un optimum, il nous faudra donc beaucoup de jeux de paramètres pour s'assurer de ne pas passer à côté de la zone optimale. La recherche par grille est donc plus précise car elle nous permet de sélectionner précisément les jeux de paramètres qui vont être explorés.

Une optimisation par recherche sur grille et recherche aléatoire a été mise en place et testée lors d'un stage de master 2, réalisé par F. Fatmaoui, permettant l'obtention de résultats de prédiction à plus de 90%. La méthode par recherche aléatoire étant beaucoup plus rapide que la recherche par grille.

Résultats

Après l'application d'un prétraitement des données impliquant une version adaptée de l'algorithme Plasmodesma, ainsi qu'une récupération et une réorganisation des données, une réduction de dimension par sélection des caractéristiques statistiques à l'aide d'un régresseur basé sur une RF (avec 100 estimateurs d'une profondeur maximale de 10) a été réalisée et les 40 caractéristiques les plus importantes ont été conservées pour la suite.
Des algorithmes de classification ont ensuite été testés et appliqués sur ces données en séparant le jeu de données en un jeu d'entraînement et un jeu de test contenant respectivement 80% et 20% des données.

Deux algorithmes de RF ont, en particulier, été choisis pour leur efficacité et leurs caractéristiques. La recherche aléatoire a permis de générer un algorithme de Random Forest optimisé avec 100 arbres (ou estimateurs) et une profondeur maximale de 16.
Par ailleurs, les paramètres « min_samples_leaf » et « min_samples_split » ont été fixés à 1 et 2, signifiant qu'une séparation n'est possible que lorsque deux éléments forment un nœud interne, et qu'au moins un échantillon doit aller dans chaque feuille nouvellement créée.

Un algorithme de RF OvR a également été appliqué, avec comme modèle initial pour chacun des classificateurs le modèle obtenu précédemment par recherche aléatoire et donc un classificateur est généré par classe présente dans les données.

Le temps de préparation et pré-traitement des données est d'environ 5 à 10 min, le temps d'entraînement des algorithmes est quant à lui très rapide, de l'ordre d'une minute, sur un ordinateur équipé d'un processeur Intel® Core™ i7-8565U à 4 cœurs cadencés à 1.80GHz et 16Go de RAM.

La Figure 17 présente les matrices de confusion des résultats de la RF optimisée par recherche aléatoire et de la RF OvR.
Cette représentation permet d'observer le taux de bonnes prédictions réalisées (correspondant aux valeurs réelles), plus les résultats sont concentrés sur la diagonale meilleurs sont les algorithmes.

Cette méthode d'appréciation des résultats permet également d'observer l'efficacité classe par classe du classificateur et en particulier ici on observe que les erreurs de prédictions sont principalement sur les molécules suivantes : 4-Fluorobenzaldéhyde, Ciprofloxacine, Fluorouracil et Diflucortolone. L'algorithme consenus des deux méthodes de RF mises en place permet en particulier de réduire les erreurs de prédiction sur le Diflucortolone. Il semble, par ailleurs, que le 4-Fluorobenzaldéhyde a tendance à être confondu avec le Trifluoroéthanol (TFE).

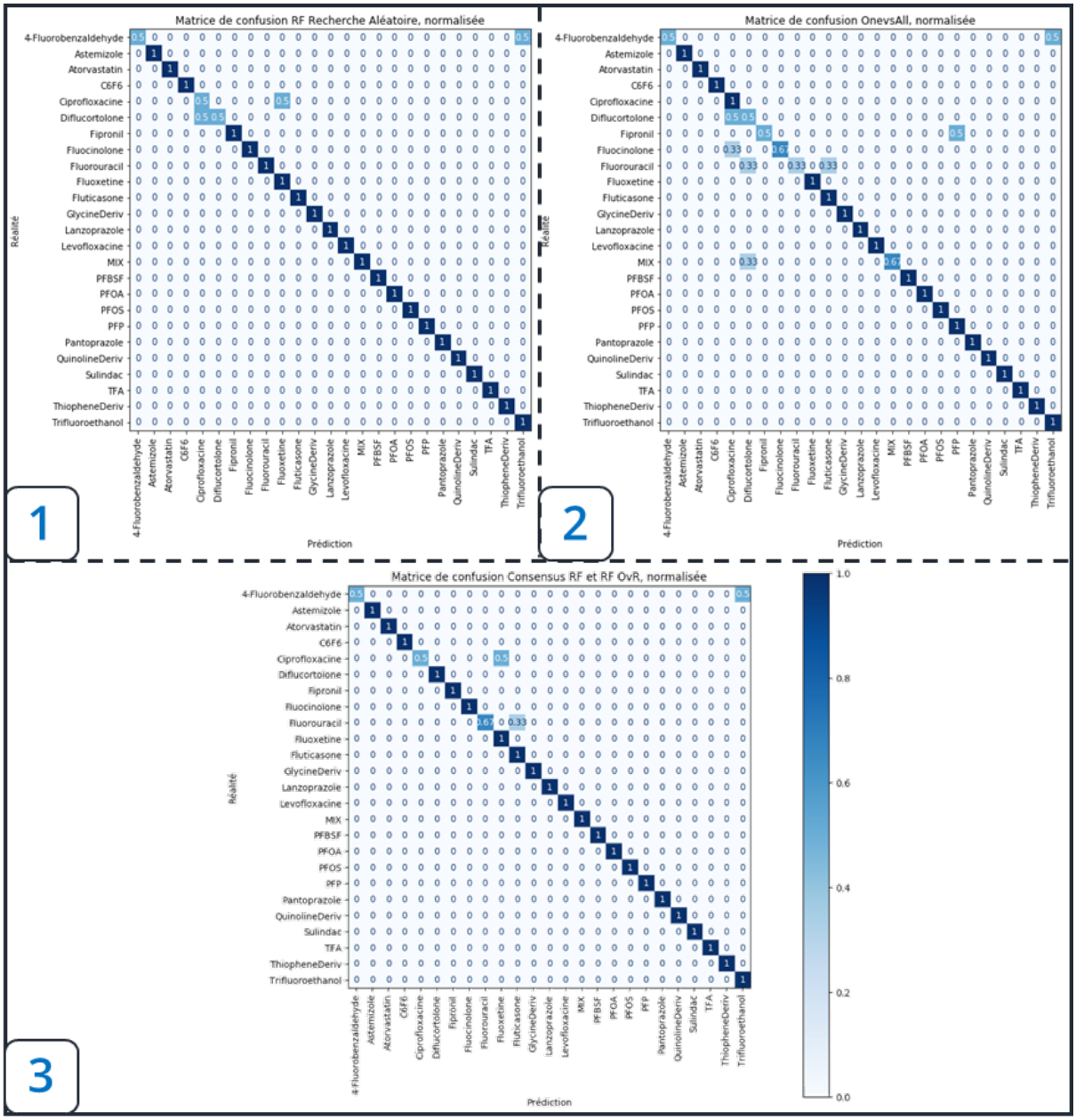


Figure 17 - Matrices de confusion pour les algorithmes de (1) RF après recherche aléatoire des hyperparamètres, de (2) RF OvR sur les données du projet et (3) l'algorithme consensus entre RF après recherche aléatoire et RF OvR.

De manière plus générale, les algorithmes développés et conservés présentent de bons résultats sur les exemples de la base de données, atteignant plus de 95% de prédictions vraies pour la RF optimisée par recherche aléatoire et 90% pour la version RF OvR.

Conclusions et Perspectives

Le projet FLUOVIAL propose donc une nouvelle approche pour la surveillance quantitative, robuste et non ciblée des polluants fluorés.

Cette approche a été mise en place en suivant différentes étapes, en réalisant dans un premier temps l'acquisition d'ensembles de données de RMN ^{19}F provenant de composés fluorés connus, puis en les traitant en appliquant une version adaptée de Plasmodesma permettant de générer des descripteurs statistiques spécifiques porteurs d'informations. Le travail effectué ici permet, en effet, de montrer que la détection et l'identification de polluants fluorés par RMN et analyse par apprentissage automatique est possible et efficace. Les algorithmes développés permettant de classifier 25 molécules avec plus de 90% de précision en un temps très court.

Les résultats obtenus restent néanmoins à confirmer et étendre avec une base de données plus grande tant en nombre de molécules différentes disponibles ainsi qu'en quantité d'exemples par classe de molécules. Ceci permettra une plus grande robustesse de l'algorithme aux variations qui peuvent se produire lors de l'acquisition des spectres mais également d'adapter l'algorithme a un plus grand nombre de cas réels.

Une amélioration pourra de plus être apportée en ajoutant une explication des prédictions de l'algorithme. En effet, la librairie Python `treeinterpreter` permet d'indiquer la contribution de chacune des caractéristiques statistiques dans la prédiction de l'algorithme, et il sera alors possible d'indiquer sur le spectre les zones (buckets) responsables de l'attribution de la classe. Cette étape permettra de s'assurer que le choix est « logique » physiquement parlant et qu'il ne s'agit pas d'un biais de l'algorithme dû à la reconnaissance d'un pic de bruit récurrent sur une classe en particulier.

Par ailleurs, les premières estimations réalisées concernant la performance de la méthode en termes de limites de détection et de quantification (LOD/LOQ) sont prometteuses. En effet, les PFAS peuvent être détectés à partir de 10 ppb et quantifiés à partir de 30 ppb de fluor total en 1 heure d'acquisition. De plus les méthodes de RMN sont en constante évolution, améliorant leur sensibilité, ce qui permettra une efficacité d'autant plus haute. Ces LOD/LOQ faibles peuvent permettre à la méthode d'être fiable et utilisée dans le cadre de normes de qualité, telles que les normes ISO par exemple, concernant les molécules ou polluants fluorés.

Ces résultats sont des résultats préliminaires et sont une première étape pour une approche en cours d'industrialisation.

En effet, le projet IPANEMA a pour but d'identifier des polluants fluorés afin de prévoir le devenir et la biodisponibilité des PFAS dans les sols ainsi que leur toxicité sur les organismes vivants dans ces sols pollués.

Les méthodes développées ici à moindre échelle seront donc appliquées sur des échantillons de sols prélevés sur un ancien site industriel dépollué où a eu lieu une utilisation de mousses extinctrices de feux d'hydrocarbures, chargées en PFAS. Ces échantillons contiendront alors des mélanges de molécules fluorées et nécessiteront donc des évolutions de l'algorithme de prédiction, en partant principalement de la version OvR qui peut s'adapter à des classifications multiples en modifiant, en autre, le type de sorties de l'algorithme.

III. Projet Rescue 3 – Attributions spectrales de protéines

Projet

Le développement de techniques permettant d'automatiser les attributions spectrales des protéines en RMN est important. En effet, encore aujourd'hui, cela nécessite généralement une expertise importante dans le domaine et un long et difficile travail souvent fait « à la main ».

Généralement, pour des spectres ^{1}H, une analyse des contraintes logiques impliquées par des jeux de données 2D et 3D J-corrélés est réalisée. Cette analyse est ensuite recoupée avec les données des squelettes des chaînes secondaires. Il est possible, par ailleurs, d'utiliser des spectres de corrélation NOE pour lier séquentiellement les acides aminés bien que cette technique soit peu utilisée. D'autres expériences de RMN 3D basées sur l'acquisition de spectres ^{13}C et ^{15}N, telles que les HNCA ou HNCO, sont couramment utilisées et permettent par ailleurs de lever beaucoup d'ambiguïtés et de réaliser des attributions plus complètes (Bax et Ikura 1991).

Des outils d'automatisation de ce processus, basés sur la topologie ou les déplacements chimiques, ont donc été développés afin d'accélérer les analyses dans les cas où l'attribution de spectres est nécessaire, comme pour la détermination de structures par RMN, permettant alors d'obtenir un débit plus élevé de traitement des données. Cependant, les outils existants prenant les déplacements chimiques comme point de départ sont généralement basés sur des statistiques simples issues de la BMRB (Biological Magnetic Resonance Data Bank).

L'objectif est donc de développer une méthode plus complète et efficace utilisant de manière optimisée les données disponibles dans la BMRB.

Le but étant par la suite de pouvoir appliquer l'algorithme lors d'études d'interactions. Dans ces études nous disposons généralement d'une protéine avec une structure relativement connue, pour laquelle aucune étude RMN n'existe et donc pour laquelle il n'y a pas d'attribution réalisée. L'idée étant de pouvoir identifier les éléments liés aux déplacements des taches sur les spectres de RMN lors de l'interaction avec la protéine.

RESCUE est un outil initialement développé en 1999 afin d'effectuer l'attribution spectrale RMN de spectres ^{1}H des protéines par le biais d'un réseau neuronal artificiel, de type perceptron, à partir des données de déplacements chimiques (Pons et Delsuc 1999).

Le jeu de données utilisé était un extrait non-redondant de la BMRB de Juillet 1996, représentant un total de 100 000 déplacements chimiques uniques de ^{1}H, issus de 1169 protéines ou peptides. Cette première version présentait deux approches pour le réseau de neurones artificiel développé. La première méthode consistait à distinguer les 20 principaux acides aminés indépendamment. Cette méthode ne permettait de réaliser une prédiction vraie que dans 63% des cas, avec des erreurs systématiques de confusion sur des acides aminés proches.

La seconde méthode, s'apparentant à une méthode de DL, plus efficace et fournissant jusqu'à 80% de prédictions vraies, se compose de deux étapes. Un premier réseau est entrainé à reconnaitre des groupes d'acides aminés et un second réseau permet de différencier les acides aminés parmi le sous-groupe déterminé par le premier réseau.

Dans les deux cas, un perceptron simple était utilisé en tant que réseau de neurones, ce qui correspondait à l'option la plus performante à l'époque. Un perceptron est le type de réseau de neurones le plus simple, il s'agit d'un classifieur linéaire, sans cycle et à propagation vers l'avant.

En 2004, de nouveaux développements ont été apportés pour améliorer l'efficacité des prédictions donnant naissance à RESCUE 2.

Depuis 1999 et la première version de RESCUE, des méthodes d'attribution automatique ont vu le jour mais prenant souvent comme point de départ les corrélations et non les déplacements chimiques. Quelques méthodes utilisent néanmoins les déplacements chimiques en vue de réaliser de l'attribution automatique d'acides aminés, en particulier les méthodes PROTYP (Grzesiek et Bax 1993), PLATON (Labudde et al. 2003) et RESCUE présentée précédemment. La méthode ARTINA développée récemment permet quant à elle, en s'appuyant sur une méthode de ML, de réaliser l'attribution de spectre avec une efficacité de 91% ainsi que de proposer une prédiction de structure avec 1.44Å de RMSD médian à partir des déplacements chimiques et de la séquence de la protéine (Klukowski, Riek, et Güntert 2022).

Le renouvellement, nommé RESCUE 2, se base à nouveau sur les données de la BMRB d'octobre 2002 contenant 2300 séquences de protéines, à partir desquelles les données statistiques des déplacements chimiques sont extraites. Ce renouvellement vise à améliorer ce qui avait été proposé précédemment. Un filtre très poussé des données avait été réalisé pour avoir la base de données la plus adaptée et seules 783 séquences avaient été conservées.

Bien qu'il s'agisse d'une suite du travail RESCUE, celui-ci ne base pas son calcul sur la même méthode sous-jacente et n'utilise cette fois pas de réseau de neurones pour réaliser la prédiction et l'attribution automatique.

L'algorithme était alors basé sur un modèle probabilistique (« Naive Bayes ») pour modéliser les corrélations deux par deux des déplacements chimiques, permettant ainsi de prédire le type d'acide aminé à partir d'un ensemble de déplacements chimiques donnés (Marin et al. 2004). Le modèle présenté permet à la fois de réaliser l'attribution spectrale à partir de déplacements chimiques lorsque cela est nécessaire, fournissant des informations importantes dans le processus de détermination de structure, mais également un schéma général pour le traitement d'informations spectrales.

RESCUE2 présentait une efficacité de 70% dans des conditions de données typiques de ce que l'on peut acquérir en pratique et des performances 10% supérieures aux méthodes concurrentes disponibles dont PROTYP et PLATON.

Suite à un besoin évoqué d'un outil pour l'attribution automatique de spectres et en collaboration avec l'équipe de biologie structurale de Gif-sur-Yvette, l'idée est venue de renouveler le code de cet outil qui reste celui utilisé aujourd'hui et qui a une bonne marge d'amélioration.

En effet, la base de données BMRB disponible aujourd'hui est bien plus fournie que pour les versions précédentes et de nouvelles techniques d'apprentissage automatique ont été développées et peuvent être implémentées en vue d'obtenir de meilleures prédictions afin de renouveler cet outil qui permet un véritable apport de gain de temps pour le traitement de spectres et dans le cadre d'élucidation de structures en particulier.

Matériel et méthode

Données

Nous avons travaillé avec une base de données nettoyée (suppression des doublons et des séquences non pertinentes) de déplacements chimiques extraits de la BMRB fournie par Thérèse Malliavin, de l'Université de Lorraine.

Pour construire cette base de déplacements chimiques, les fichiers FASTA de toutes les séquences polypeptidiques de la BMRB ont été récupérés. Seules les séquences uniques ont ensuite été conservées et leurs déplacements chimiques associés récupérés.

Nous disposons donc d'une liste de déplacements chimiques issus de séquences uniques, chaque élément de cette liste comprend : l'identifiant BMRB, le nom de la molécule, la référence dans la PDB, un ID d'entité, le numéro du résidu, le nom du résidu, le nom de l'atome, le type d'atome, le type d'isotope, le déplacement chimique et l'erreur sur le déplacement chimique si elle est disponible. Un exemple d'entrée du tableau est donné sur le Tableau 7.

Tableau 7 - Exemple d'une entrée de la base de données d'origine telle que fournie par T. Malliavin pour le projet RESCUE 3.

ID BMRB	Nom de la molécule	ID PDB	ID d' entité	N° résidu	Nom du résidu	Nom de l' atome	Type d' atome	Type d' isotope	Déplacement chimique	Erreur
bmr10001	MP-X	6214	1	1	ILE	CA	C	13	59.2	0.6

La base de données d'origine, après suppression des valeurs de déplacements chimiques aberrantes, contient 4 618 970 lignes de déplacements chimiques, pour un total de 5507 entrées uniques de la BMRB.

Les 20 acides aminés principaux sont tous représentés dans la base de données (Figure 18 - 3).

Cependant ils ne sont pas représentés à nombre de données égales. Il est possible d'adapter la base de données afin d'avoir une représentation équilibrée des différentes classes, mais cela n'a pas été réalisé car la base de données déséquilibrée permet de mieux représenter la réalité biologique.

En effet, naturellement, la répartition des acides aminés n'est pas équilibrée et certains sont majoritaires par rapport à d'autres (Figure 18 – 1). En laissant cette répartition dans la base de données, l'algorithme entraîné sera optimisé par rapport à la réalité biologique et donc légèrement biaisé pour mieux prédire les cas auxquels il sera confronté.

Par ailleurs, la Figure 18 montre que la répartition de la base de données utilisée pour cette version de Rescue3 est assez proche de la répartition observée pour la BMRB entière sans filtre des données, mais

également de la répartition naturelle des acides aminés dans l'ensemble de ce qui est répertorié au sein de la base de données Uniprot-TrEMBL.

Le biais induit en conservant cette disparité entre les proportions d'acides aminés est donc cohérent avec la réalité expérimentale pour les prédictions à réaliser, l'algorithme aura donc tendance à prédire un acide aminé apparaissant naturellement à plus haut pourcentage s'il y a une hésitation entre deux prédictions possibles. Ceci aura donc l'avantage de fournir une meilleure prédiction dans le cas général, pour des protéines « classiques » avec une répartition naturelle d'acides aminés. Cependant pour des protéines plus atypiques, avec des répartitions moins habituelles, il y a un risque d'avoir une plus grande difficulté à réaliser une attribution correcte.

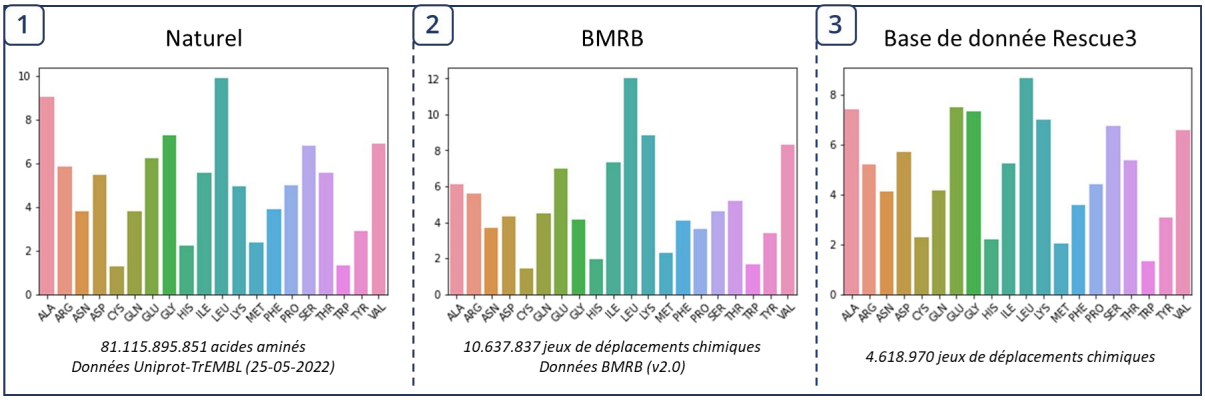


Figure 18 - Répartition des acides aminés 1. Naturellement (données récupérées à partir de la base de données Uniprot-TrEMBL du 25 mai 2022), 2. Au sein de la BMRB v2.0 et 3. Au sein de la base de données utilisée pour l'algorithme de prédiction Rescue3.

Une adaptation de la base de données est réalisée afin de pouvoir l'utiliser facilement et simplement pour l'entrainement de l'algorithme de DL par la suite.

En particulier et dans un premier temps, certaines des colonnes n'étant pas pertinentes ont été mises de côté pour ne conserver, comme présenté dans le Tableau 8, que les colonnes suivantes : l'ID BMRB, le numéro de résidu, le nom de l'atome, le déplacement chimique et le Nom du résidu (qui correspond à la classe dans le cas présent).

Tableau 8 - Exemple d'une entrée de la base de données après une première réorganisation.

ID BMRB	N° résidu	Nom du résidu	Nom de l' atome	Déplacement chimique
bmr10001	1	ILE	CA	59.2

Une fonction est ensuite appliquée afin que pour chaque entrée unique de la BMRB et pour chaque numéro de résidu par entrée de la BMRB, les informations de déplacements chimiques concernant chacun des acides aminés soient concaténées.

Cela permet ainsi, comme illustré par le Tableau 9, d'avoir des jeux de déplacements chimiques pour chaque acide aminé de chaque résidu par entrée unique de la BMRB. Des « NaN » sont ajoutés pour les déplacements chimiques manquants.

Tableau 9 - Exemple d'une entrée de la base de données finale utilisée pour l'entrainement de l'algorithme de DL.

Classe	C	CA	CB	CD1	CG1	CG2	N	...	NE	CD2	CE2	CE3	CH2	NH1	NH2
ILE	172	59.2	37.5	12.9	27.5	15.7	36.9	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN

La base de données maximale pouvant être utilisée pour l'entrainement de l'algorithme de DL est donc finalement composée d'une table de 70 colonnes par 485 264 lignes, correspondant au nombre d'acides aminés différents pour lesquels un jeu de déplacements chimiques est disponible. Il est alors possible de filtrer les données par type d'acide aminé ou par types de déplacements chimiques en fonction des expériences à simuler.

Les éléments « NaN », pour « Not a Number », visibles dans le Tableau 9 correspondent à des données manquantes dans le jeu de déplacements chimiques. En effet, toutes les données ne sont pas disponibles

pour chaque jeu de déplacements chimiques associé à un acide aminé, les valeurs inconnues ont donc été remplacées par des « NaN ».

Ces valeurs peuvent être inconnues pour deux raisons, d'une part cela peut être dû au fait qu'elles n'aient simplement pas été mesurées expérimentalement et d'autre part car la nomenclature pour un acide aminé peut être différente de celle d'un autre, en effet on peut citer l'exemple des CB et CG qui peuvent changer de noms en fonction des résidus.

Dans le cas d'une valeur inconnue par manque de mesure expérimentale, il peut être décidé de les laisser dans le jeu de données, mais les algorithmes de DL gèrent généralement mal les informations manquantes, ou il est possible d'imputer les valeurs manquantes, consistant à les remplacer systématiquement par une valeur. La valeur de remplacement est généralement la valeur la plus probable ou la valeur moyenne pour la classe. Lorsque cette information n'est pas accessible, une autre possibilité est d'utiliser une valeur hors de l'intervalle de valeurs possibles que peuvent prendre les données, mais ce n'est pas une option idéale. Une transformation des données permettant d'intégrer les données manquantes peut par ailleurs être mise en place et éviter d'avoir à les remplacer par une valeur arbitraire comme évoqué précédemment.

L'utilisation d'une transformation des données, qui sera présenté par la suite, permet quant à elle de résoudre le problème d'encodage variable entre les différents résidus.

Pré-traitement des données réorganisées

Une fois les données réorganisées, il est préférable pour l'entrainement de DL d'avoir des vecteurs d'entrée de taille constante et qui soient insensibles aux permutations. Les vecteurs d'entrée dans leur état actuel ne remplissent pas ces conditions. En effet, il s'agit à chaque fois d'un vecteur rassemblant un nombre N de déplacements chimiques correspondants à une mesure RMN.

Nous allons donc mettre en place des transformations invariantes pour la permutation et pour la longueur des vecteurs d'entrée afin de transformer ces vecteurs d'entrées variables en vecteurs de tailles constantes et qui permettront de représenter le jeu de déplacement chimique malgré les potentielles permutations.

L'invariance est un élément important dans le domaine du ML, en effet il s'agit d'avoir un modèle qui va résister aux variations possibles sur les données à reconnaitre. En particulier, dans le domaine de la reconnaissance d'image, l'objectif est d'être capable de reconnaitre un motif, malgré les possibles translations ou rotations qui peuvent s'appliquer. L'invariance est donc la capacité à reconnaître un élément en lui-même, quelques soit les variations qui peuvent s'y appliquer. L'objectif est donc d'avoir la même efficacité de prédiction lorsqu'une transformation est appliquée à la donnée d'entrée ou non. Il s'agit en partie de la raison pour laquelle les réseaux de neurones convolutionnels (CNN) sont si populaires pour l'analyse d'image. En effet, ce type de NN est intrinsèquement invariant aux translations par leur structure composée de combinaisons de couches convolutives et de couche dite de « max-pooling », les CNN sont donc particulièrement adaptés à cette application.

Pour rendre notre modèle invariant au nombre et à l'ordre des valeurs de déplacements chimiques disponibles, deux méthodes ont été mises en place et sont comparées afin de transformer le vecteur initial en vecteurs invariants aux permutations ou variations de taille qui peuvent s'appliquer au vecteur initial : une transformation dite « fuzzy » et une transformation en moments statistiques. Ce type de transformation est souvent utilisée en traitement d'images par ML car les moments, notamment les moments centrés, permettent de décrire une image en gardant une invariance par rapport aux rotations et autres déformations possibles (Nasrudin et al. 2021).

➔ La transformation dite « fuzzy » issue de la version de 1999 du programme RESCUE. Cette transformation permet d'obtenir un nombre de paramètres d'entrée pour le réseau identique pour chaque jeu de déplacement chimiques. La grille associée à cette logique fuzzy va donc échantillonner l'axe de déplacement chimique de manière régulière, avec un pas adapté à chaque noyau H, C et N.
L'équation suivante est utilisée pour calculer la valeur post-fuzzy y_k pour chaque zone de déplacement chimique échantillonnée :

$$y_k = \sum_l \left(\max\left(1 - \frac{\|\delta_k - x_l\|}{\Delta\delta}, 0 \right) \right)$$

Avec $\Delta\delta$ le pas d'échantillonnage, δ_k les positions exactes des n entrées de la grille de logique fuzzy échantillonnée avec le pas $\Delta\delta$ et x_l les valeurs de déplacements chimiques de départ.

Pour les protons, l'intervalle balayé va de -5 à +15 ppm par pas de 0.2 ppm.

Pour les carbones, la gamme balayée va de -15 à +200 ppm par pas de 1 ppm.

Pour les azotes, le balayage se fait également de +90 à +180 ppm, par pas de 2 ppm.

Tous les déplacements chimiques se trouvant en dehors de ces intervalles sont éliminés de la base de données.

Pour des raisons pratiques, la base de données de déplacements chimiques est donc séparée en 3 tables (par noyaux : ^{1}H – ^{13}C – ^{15}N), gardant les mêmes indices de lignes pour ne pas mélanger les jeux de déplacements chimiques. Chacune des 3 tables est traitée par logique fuzzy séparément pour pouvoir appliquer les différents paramètres évoqués précédemment. Les 3 tables sont alors regroupées pour reformer l'ensemble des jeux de déplacements chimiques.

➔ La transformation en moments statistiques qui permettent de décrire les propriétés d'une distribution statistiques, déjà vue et utilisée dans le chapitre sur le projet Fluovial. Pour une variable aléatoire X données, les moments statistiques sont définis comme les n espérances mathématique de cette variable : $E[X^n]$. Les 4 moments les plus connus et utilisés étant la moyenne, la variance, l'asymétrie et le kurtosis.

Dans le cas des déplacements chimiques, ces moments ne changent pas si certaines valeurs sont permutées puisque la distribution en elle-même en sera peu affectée. Ils sont donc invariants aux possibles permutations au sein du vecteur. En effet, dans le cas d'un vecteur de déplacement chimique, la principale transformation à laquelle on peut être confronté est la permutation de certaines valeurs.

Cette seconde transformation consiste donc à transformer le vecteur de déplacements chimiques en un vecteur des 8 premiers moments statistiques associés à la distribution du jeu de déplacements chimiques en question. Chaque ligne de la base de données est donc transformée en un vecteur à 8 éléments qui sert d'entrée à l'algorithme de DL.

Après différents tests, cette transformation fournissait, sur notre jeu de données, de moins bons résultats que la transformation fuzzy. Elle permettait donc probablement de décrire moins en profondeur les données. Le choix des moments utilisés n'était peut-être pas le bon, et des combinaisons de moments

pour mieux décrire les déplacements chimiques seraient nécessaires. Cette transformation n'est donc pas celle ayant été appliquée systématiquement pour la suite du travail.

Par ailleurs, l'utilisation de l'une ou l'autre de ces deux méthodes de transformation des données permet de ne plus avoir de données manquantes, ou « NaN », dans le jeu de données final. Cette procédure permet donc de lever un des problèmes rencontrés dans la mise en place de l'algorithme.

Algorithme

Nous avons ensuite construit des scénarios pour filtrer les ensembles de déplacements chimiques en fonction de ce qui peut être acquis dans la réalité expérimentale. Cela nous permet d'entraîner les algorithmes d'apprentissage automatique uniquement sur des ensembles similaires à ceux qui vont être mesurés.

Un réseau neuronal séquentiel à 7 couches denses, dont la structure est présentée dans le Tableau 10, a été mis en place avec Keras, une API d'algorithme de Deep Learning en langage python et a été appliqué aux différents scénarios présentés ci-dessous.
Un optimiseur « AdaMax » et une fonction de perte (ou « loss function ») « categorical_crossentropy » ont été utilisés, avec un entraînement sur 25 epoch.

L'optimiseur « AdaMax » est une extension de l'optimiseur « Adam ». Il s'agit d'une méthode d'optimisation par descente de gradient avec un pas de courbe d'apprentissage différent et adapté à chaque variable d'entrée. La version Adam utilise une normalisation L2. La version AdaMax permet d'accélérer le processus d'optimisation en généralisant l'approche à la norme infinie. La norme L2 est la norme dite Euclidienne, qui représente la distance la plus courte pour aller d'un point à un autre. Avec la norme infinie, on calcule la plus grande amplitude existante entre chaque élément du vecteur, et la minimisation par norme infinie consiste alors à réduire la distance entre les points les plus éloignés uniquement, accélérant donc le processus de minimisation.

De manière générale, une fonction de perte est la fonction utilisée pour calculer la quantité à minimiser par le modèle pendant le processus d'apprentissage. Pour les modèles prédictifs, l'entropie croisée, ou « cross-entropy », est la fonction de perte couramment mise en place.
La fonction « categorical_crossentropy » de Keras est utilisée pour les modèles de classification multi-classes, comme ici où on a 20 acides aminés à prédire. Les 20 étiquettes ou classes sont alors encodées en catégories sous formes de 0 et 1.
Pour chaque calcul de probabilité d'appartenance à une classe, la classe prédite est comparée à la valeur réelle de la classe (encodée en 0 er 1) et un score/une perte est calculée et pénalise la probabilité en fonction de sa distance par rapport à la classe réelle.

Le nombre d'epoch correspond au nombre de passages dans les données pour l'entraînement de l'algorithme. A chaque nouveau passage l'algorithme va affiner son modèle généralisé correspondant aux données d'apprentissage. Le choix du nombre d'epoch utilisé est important car un nombre de passage dans les données d'entraînement trop important va mener à de l'overfitting conduisant ainsi l'algorithme à ne plus avoir un modèle généralisé mais un modèle spécifique aux données fournies pour l'entrainement, il ne sera alors pas capable de donner de bons résultats sur des données de test ou inconnues jusqu'alors.
Ici, 25 epoch ont été choisies après avoir observé les capacités de l'algorithme à prédire sur les données d'entraînement et de test au fur et à mesure du nombre d'epoch. En effet, en suivant l'entraînement on observe une augmentation de la précision de l'algorithme à la fois sur les données d'entraînement mais également sur des données de test qui ne sont pas utilisées lors de la construction du modèle par l'algorithme. Lorsque la précision sur les données de test cesse d'augmenter, le nombre d'epoch est optimal. Tout gain de précision sur le jeu de données d'entraînement correspond alors à de l'overfitting.

La fonction d'activation en DL est la fonction qui va permettre de décider si l'on a une réponse ou non du neurone, parallèlement à ce qui peut se produire biologiquement dans le corps humain pour les réponses des neurones. Cette fonction a également pour rôle de normaliser les données en sortie de couche, puisqu'elles fournissent des résultats bornés (souvent entre 0 et 1).
Les fonctions d'activation utilisées sont généralement non-linéaires, permettant d'avoir un modèle de DL qui ne l'est pas non plus ce qui est indispensable pour pouvoir construire un modèle complexe non-

linéaire capable de représenter finement des jeux de données. Un neurone dans un réseau ne va faire qu'appliquer : $X = \sum(entrée \times poids) + biais$, sur chaque entrée puis la fonction d'activation s'applique ensuite sur X. Cette fonction est spécifique à chaque couche et transforme les données de manière à les présenter sous un nouvel angle à chaque étape.

On utilise ici pour toutes les couches intermédiaires la fonction ReLU, pour Rectified Linear Unit, qui est la plus couramment utilisée dans les réseaux de neurones profonds. Cette fonction conserve X si $X > 0$ et 0 sinon, elle s'exprime donc comme suit : $ReLU(X) = \max(X, 0)$. On applique donc un filtre sur les données, ne laissant passer que les positives.

La fonction Softmax est utilisée pour la couche de sortie et permet de transformer le vecteur réel en vecteur de probabilité. Elle est très pertinente sur la couche de sortie d'un modèle de classification multiclasse.

Différentes structures ont été testées et la taille des différentes couches ainsi que leur nombre a été optimisé pour répresenter la totalité du problème. Le fait d'augmenter d'abord le nombre de paramètres permet à l'algorithme de représenter l'ensemble des degrès de liberté, puis la taille est progressivement réduite pour éliminer les informations « inutiles » ou corrélées entre elles pour ne garder que 20 descripteurs correspondants aux 20 acides aminés classiques à prédire.

Tableau 10 - Structure du réseau de neurones mis en place pour la prédiction des acides aminés en fonction des déplacements chimiques dans Rescue3.

	Type de couche	Taille	Fonction d'activation	Nombre de paramètres
Couche 1	Dense	300	Relu	61500
Couche 2	Dense	500	Relu	150500
Couche 3	Dense	1000	Relu	501000
Couche 4	Dense	2000	Relu	2002000
Couche 5	Dense	1000	Relu	2001000
Couche 6	Dense	500	Relu	500500
Couche 8 - Sortie	Dense	20	Softmax	10020
				Total : 5 226 520

Une couche dense est une couche complètement connectée à la couche précédente (Figure 19). Il s'agit du type de couche le plus couramment utilisé dans les réseaux de neurones artificiels. Chaque neurone d'une couche dense dans un modèle reçoit donc la sortie de chaque neurone de la couche précédente.

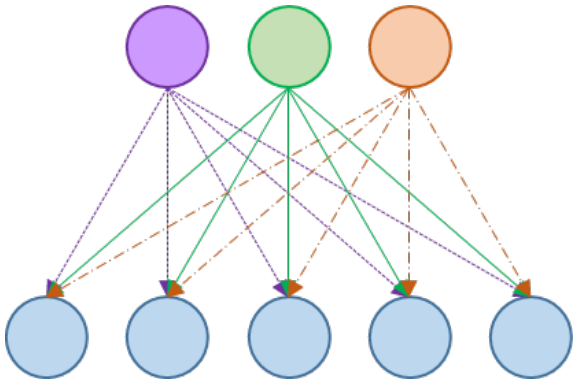


Figure 19 - Illustration d'une couche dense dans la structure d'un réseau de neurones artificiel.

Il est à noter que la couche d'entrée précédant la couche 1 dans cette structure n'est pas illustrée, sa taille va varier et dépendre du vecteur d'entrée. Par exemple, pour un vecteur issu d'une transformation par logique fuzzy, le vecteur d'entrée sera plus grand que pour une transformation en moments statistiques par exemple où on on aurait un vecteur de taille 8. Le reste du réseau conserve néanmoins la même structure quelque soit le vecteur d'entrée.

Scénarios

Des scénarios ont ensuite été élaborés afin de filtrer les données et de conserver chaque fois un jeu de données final correspondant au mieux à une réalité expérimentale possible.

Un scénario général qui conserve toutes les données, ainsi que des scénarios classiques utilisés dans les précédents travaux ont été réalisés afin de pouvoir comparer les résultats obtenus avec ce programme aux résultats des versions antérieures de ce dernier ainsi qu'aux méthodes alternatives. Des scénarios plus spécifiques aux expériences réalisées par nos partenaires ont également été testés afin d'évaluer la pertinence du renouvellement dans ces cas particuliers.

1. Général

Ce premier scénario détermine l'efficacité de la prédiction de l'algorithme pour l'ensemble des déplacements chimiques qui sont à notre disposition. C'est un calcul théorique au vu des possibilités d'acquisitions réelles en laboratoire qui ne permettent jamais d'avoir la totalité des informations de déplacements chimiques concernant tous les noyaux d'une protéine.

Ce scénario ne vise donc qu'à montrer théoriquement la puissance de prédiction possible avec un maximum d'informations.

Il est cependant à noter que la base de données de déplacements chimiques utilisée n'est pas complète. En effet, tous les déplacements chimiques pour un acide aminé d'une protéine ne sont pas connus, il y a un grand nombre de données manquantes dû à des manques de mesures expérimentales, qui risquent donc d'impacter les résultats de l'algorithme, bien que le rendant plus robuste à des cas concrets où toutes les informations ne sont pas disponibles pour l'attribution.

2. TOCSY (^{1}H) – RESCUE

Une acquisition TOCSY, pour « Total Correlation SpectroscopY » est une expérience classique en RMN. Il s'agit d'une 2D homonucléaire où le couplage entre deux noyaux d'hydrogène génère un pic dans la carte 2D. La TOCSY n'est pas limitée à la détection des couplages entre voisins les plus proches, elle permet de détecter les couplages entre toutes les paires d'hydrogène d'une chaine. Les protons d'une chaîne ininterrompue de spins couplés sont donc liés par des tâches corrélées dans le spectre TOCSY 2D.

Pour ce scénario, qui correspond au scénario mis en place dans la première version de RESCUE de 1999, seuls les déplacements chimiques « H » sont conservés sauf ceux liés à des aromatiques. Pour simuler cela, les acides aminés ont été séparés en trois sous-groupes :

- AMX : ASN TRP SER CYS ASP PHE TYR HIS
- EQM : GLU GLN MET
- Autres : ILE GLY ALA LEU PRO VAL SER THR

Pour chacun des groupes, les déplacements chimiques conservés sont adaptés en fonction de ce qui est observable en TOCSY de la manière suivante :

- AMX : H HA HB2 HB3
- EQM : H HA HB2 HB3 HG2 HG3

- Autres : tous les H – sauf HG SER et HG1 THR qui ne sont jamais observés

3. HSQC-TOCSY (^{1}H ^{15}N) – RESCUE2

La HSQC-TOCSY permet d'élucider les pics croisés que l'on peut observer en TOCSY classique. Ce scénario est donc la représentation d'une technique classique de RMN, souvent utilisée et plus efficace que la TOCSY puisqu'il y a moins de superpositions et donc une plus grande facilité pour l'attribution des spectres. Cette méthode requiert cependant que la protéine soit marquée à l'azote ^{15}N.

Dans ce scénario la proline est invisible car ne contenant pas de proton amide, on ne conserve donc pas cet acide aminé dans la base de données. De la même manière que pour le scénario TOCSY, les acides aminés ont été séparés en trois sous-groupes pour adapter les déplacements chimiques conservés. Le filtrage est fait similairement à celui du scénario précédent, à la différence que les déplacements chimiques « N » sont conservés en plus de ceux de la TOCSY, ce qui donne donc :

- AMX : N H HA HB2 HB3
- EQM : N H HA HB2 HB3 HG2 HG3
- Autres (sans PRO) : N, tous les H – sauf HG SER et HG1 THR qui ne sont jamais observés

4. 3D RMN

Différents scénarios visant à reproduire les situations et les données pouvant être obtenues par des expériences de RMN en 3D ont également été évalués. De la même manière que pour le scénario précédent, la proline n'est pas visible dans ces expériences. Pour ces acquisitions les protéines doivent être doublement marquées à l'azote ^{15}N et au carbone ^{13}C.

L'expérience historique « HNCA » a été modélisée, ainsi que sa version alternative « HNCO », ces deux expériences restent néanmoins ambiguës et ne permettent pas des attributions évidentes (Kay et al. 1990). Des combinaisons et extensions de ces expériences historiques existent et sont plus couramment utilisées en RMN car beaucoup moins ambigües puisque l'on dispose de plus d'informations sur les déplacements chimiques. La HNCA et la HNCO ainsi que leurs expériences dérivées ont donc été évaluées produisant ainsi 5 scénarios incluant les déplacements chimiques suivants :

- « HNCA » : H N CA
- « HNCO » : H N C
- « HNCOCA » : H N C CA

- « HNCOCACB » : H N C CA CB
- « HNCOCACBCG » : H N C CA CB CG

Les déplacements chimiques ne correspondant pas au scénario modélisé ne sont pas conservés.

5. H(C)CH3-TOCSY et (H)CCH3-TOCSY

Ces deux scénarios sont également issus d'expériences de RMN 3D, plus spécifiques et particulières que celles présentées ci-dessus. Les expériences H(C)CH3-TOCSY et (H)CCH3-TOCSY permettent de réaliser l'attribution des chaînes latérales d'acides aminés méthylés (Uhrín et al. 2000). Ce type d'expérience est intéressante et souvent utilisée car elle n'est quasiment pas limitée par la masse grâce aux marquages spécifiques qui sont utilisés.

Dans les deux cas, les acides aminés observables sont les suivants, car il s'agit d'acides aminés méthylés : isoleucine (ILE), alanine (ALA), méthionine (MET), leucine (LEU), valine (VAL) et thréonine (THR), ils sont donc les seuls à être conservés dans la base de données. La base de données associée contient alors 171 294 jeux de déplacements chimiques.

Pour la mise en place du scénario H(C)CH3-TOCSY, le type de déplacement chimique a été filtré en fonction de l'expérience, sont donc conservés, s'ils existent, tous les déplacements H et C associés aux méthyles, ainsi que tous les autres C sauf le « C » (carboxyle) :

- HB HG HE HD11 HD12 HD13 HG21 HG22 HG23 HG11 HG12 HG13
- CB CG CE CD1 CG1 CG2 MD CD MG

De la même manière pour le scénario (H)CCH3-TOCSY, les déplacements chimiques ont été filtrés, en conservant, s'ils existent, tous les déplacements H et C associés aux méthyles listés ci-dessus, ainsi que tous les autres H sauf le « H » (amide).

Chacun des scénarios se met en place en filtrant simplement le jeu de données obtenu après pré-traitement et en enlevant donc les jeux de déplacements chimiques qui ne peuvent pas être récupérés par l'expérience de RMN en question pour le scénario.

Résultats

L'algorithme développé a donc été testé dans plusieurs situations reproduisant des scénarios envisageables et permettant d'évaluer l'efficacité du programme dans différents cas de figure.

Pour chacun des scénarios, le jeu de données final après filtre est séparé en deux, 70% pour produire un jeu de données d'entraînement et 30% sont conservés en tant que jeu de données de test.

L'entraînement de l'algorithme pour un scénario est de l'ordre de quinze à vingt minutes dans un environnement Colab muni de 2 vCPU (virtual Central Processing Unit) Intel® Xeon® cadencés à 2.20GHz et possédant 13Go de RAM et 80Go d'espace disque.
Une carte graphique, ou « GPU » pour Graphical Processing Unit, est disponible mais n'a pas été utilisée. Cela pourrait encore accélérer le calcul mais des problèmes liés à la mémoire dédiée de la carte graphique pourraient apparaître.

Différents résultats sont obtenus en fonction du scénario testé puisque les données conservées dans chacun des cas sont différentes. Nous nous attendons à ce que le plus de données contenant de l'information unique sont conservées, meilleurs soient les résultats fournis par l'algorithme.
Tous les résultats présentés ici sont ceux obtenus avec une transformation en logique fuzzy, la transformation en moments statistiques fournissant des résultats de moins bonne qualité.

La Figure 20 présente la précision obtenue globalement sur les données d'entraînement (en bleu) et sur les données de test (en orange) en fonction des différents scénarios mis en place.
Ces résultats permettent d'identifier les scénarios pour lesquels l'algorithme mis en place n'est globalement pas capable de prédire correctement les acides aminés par les valeurs de déplacements chimiques restreintes au scénario.

En particulier, les scénarios HNCO, HNCA et HNCOCA ne fournissent pas de bons résultats, ce à quoi l'on pouvait s'attendre au vu du peu de déplacements chimiques disponibles dans ces expériences, induisant des ambiguïtés et donc des confusions entres les différents acides aminés.

Par ailleurs, certains des scénarios montrent des précisions sur les prédictions clairement meilleures que d'autres, en particulier les scénarios HNCOCACBCG, (H)CCH3-TOCSY et H(C)CH3-TOCSY permettent d'obtenir plus de 80% de prédictions correctes.

Cependant ces expériences ne peuvent élucider qu'un nombre restreint d'acides aminés. Il est ainsi possible d'imaginer les combiner afin d'obtenir un scénario plus complet, réalisable expérimentalement, fournissant assez d'informations de déplacements chimiques pour obtenir des prédictions d'attribution encore plus précises. Ceci peut s'avérer très intéressant pour les études d'interaction en particulier.

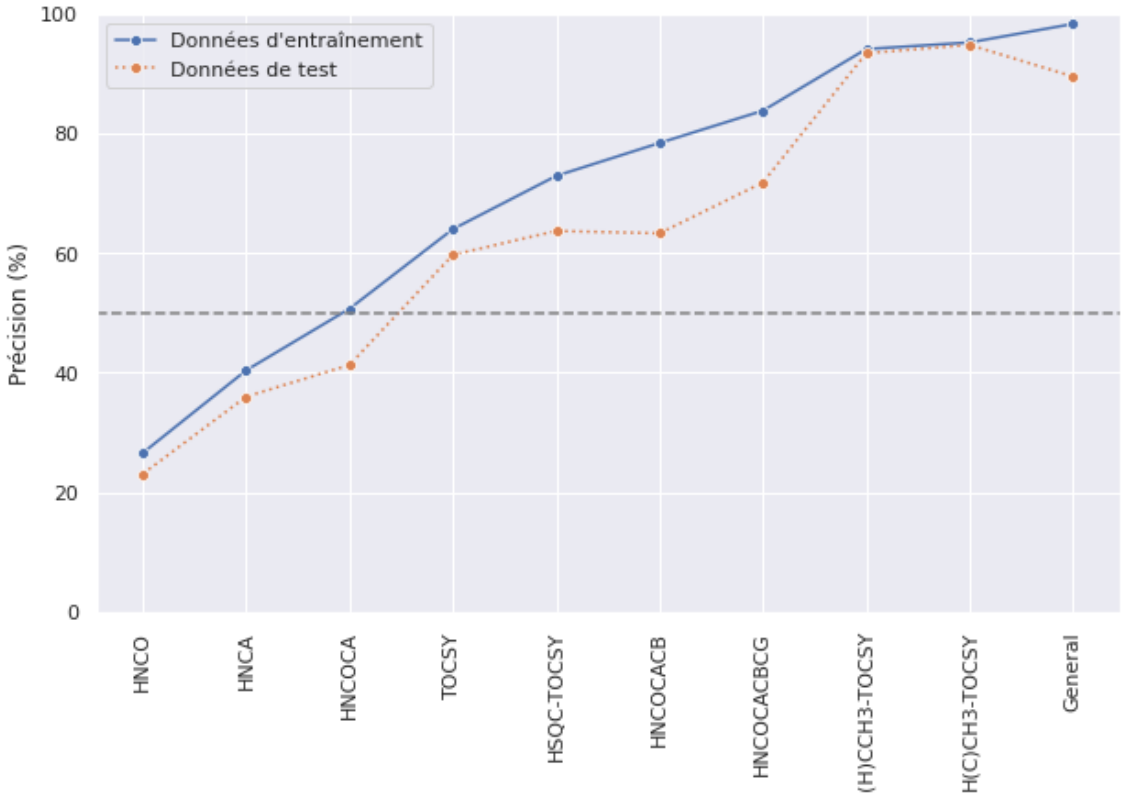


Figure 20 – Graphe des résultats globalisés de chacun des scénarios testés, triés de la plus faible à la plus haute précision obtenue.

Le Tableau 11 regroupe les résultats, sur les données d'entraînement et les données de test, des différents scénarios mis en place par acide aminé et reporte également la précision globale de chaque scénario.

Les résultats obtenus pour le scénario « Général » permettent de déduire de l'efficacité potentielle de la méthode. En effet, en conservant toutes les informations de déplacements chimiques, le Tableau 11 montre que la précision des prédictions de l'algorithme est très élevée et peu de confusion entre les différents acides aminés est observée (Figure 21).

En conservant donc toutes les informations disponibles dans la BMRB pour un acide aminé d'une protéine, la prédiction d'attribution obtenue par cette méthode est donc plutôt fiable et précise.

Tableau 11 – Taux de réussite (en %) des prédictions de l'algorithme mis en place par acide aminé puis globalement, en fonction des scénarios. En rouge les prédictions sous 50%, où l'algorithme n'est pas capable de prédire correctement l'acide aminé.

	General	TOCSY	HSQC-TOCSY	HNCA	HNCO	HNCOCA	HNCOCACB	HNCOCACBCG	H(C)CH3-TOCSY	(H)CCH3-TOCSY
ALA	97.6	77.4	87.5	61.6	47.2	72.6	91.7	91.9	99.2	99.5
ARG	84.6	-	-	2.8	1.0	6.4	28.4	61.2	-	-
ASN	81.4	23.9	32.8	42.8	9.5	41.8	63.6	68.9	-	-
ASP	85.3	60.5	54.4	18.3	3.6	27.9	54.6	62.6	-	-
CYS	77.9	10.1	26.3	10.0	10.9	13.0	34.5	40.0	-	-
GLN	79.4	23.7	34.1	8.9	4.2	12.9	43.6	56.2	-	-
GLU	87.6	64.2	62.5	29.3	26.4	36.9	52.2	69.2	-	-
GLY	98.4	72.3	90.9	94.0	79.9	94.3	94.0	94.6	-	-
HIS	76.9	21.9	22.0	7.2	2.7	15.0	22.9	34.5	-	-
ILE	90.5	48.8	51.6	34.6	5.4	36.4	70.8	79.4	88.6	91.5
LEU	92.2	68.3	71.9	42.6	27.6	45.9	66.5	75.0	94.4	95.3
LYS	87.7	-	-	23.6	18.8	27.5	62.3	72.9	-	-
MET	78.5	35.6	39.9	5.3	1.3	8.6	22.3	49.5	74.1	83.2
PHE	79.9	26.2	23.0	2.4	1.0	10.7	30.3	38.1	-	-
PRO	98.4	-	-	-	-	-	-	-	-	-
SER	94.7	90.8	90.8	55.4	53.9	59.2	88.2	89.0	-	-
THR	96.2	72.6	73.8	51.6	14.5	64.7	87.8	88.4	95.3	96.5
TRP	84.4	14.8	23.9	0.4	0.4	4.6	17.3	27.6	-	-
TYR	80.5	14.8	20.6	4.0	1.5	12.3	31.7	38.4	-	-
VAL	94.9	74.6	73.1	46.7	17.1	51.2	81.4	86.0	93.8	93.6
Précision globale - données d'entraînement	98.3	64.0	72.9	40.4	26.6	50.7	78.4	83.8	95.2	94.1
Précision globale - données de test	89.5	59.7	63.7	36.0	23.1	41.3	63.3	71.8	94.8	93.4
Comparaison Rescue2	-	51	70	-	-	-	-	-	-	-

Les scénarios TOCSY et HSQC-TOCSY fournissent des résultats corrects mais avec néanmoins des confusions et erreurs de prédictions sur certains acides aminés, notamment le groupe des AMX. Ceci s'explique en partie par le fait que les informations disponibles pour ces acides aminés avec ce type d'expérience est plus restreint que pour le reste des acides aminés.

Certains acides aminés ne sont pas présents dans les résultats de ces scénarios. Cela est dû au fait qu'après le filtre des données, les jeux de déplacements chimiques ne contenant aucun déplacement qui puisse être mesuré par le scénario sont éliminées, et certains acides aminés (ici arginine et lysine) n'ont alors plus de représentants.

Le passage du scénario TOCSY au scénario HSQC-TOCSY améliore tout de même de 4% au global les prédictions, ce qui semble cohérent puisque les informations sur les déplacements chimiques « N » sont disponibles, permettant de réduire certaines ambigüités.

Les résultats associés aux expériences de RMN 3D permettent de montrer que les expériences historiques HNCA et HNCO sont trop pauvres en informations pour pouvoir réaliser une attribution efficace des acides aminés.

En revanche, les extensions de ces méthodes, en particulier lorsque les informations sur les C_β et les $C\gamma$ sont disponibles (HNCOCACB et HNCOCACBCG), contiennent davantage d'informations et permettent des attributions moins ambiguës des acides aminés en fonction des déplacements chimiques. Les acides aminés du groupe AMX ainsi que la méthionine restent cependant difficiles à prédire efficacement avec les scénarios mis en place et les données disponibles issues de la BMRB.

Les scénarios H(C)CH3-TOCSY et (H)CCH3-TOCSY sont des types d'expériences RMN assez récentes et fréquemment utilisées car permettant d'obtenir et d'observer un grand nombre de déplacements chimiques différents. On observe ici qu'avec ce type d'expérience, les acides aminés méthylés pouvant être élucidés le sont très efficacement.

Les deux scénarios présentent des résultats similaires, ce qui semble indiquer que les informations principales utilisées pour les prédictions sont associées aux méthyles, bien que les informations portées soit par les H ou C non méthyles permettent probablement de lever des incertitudes s'il y en a.

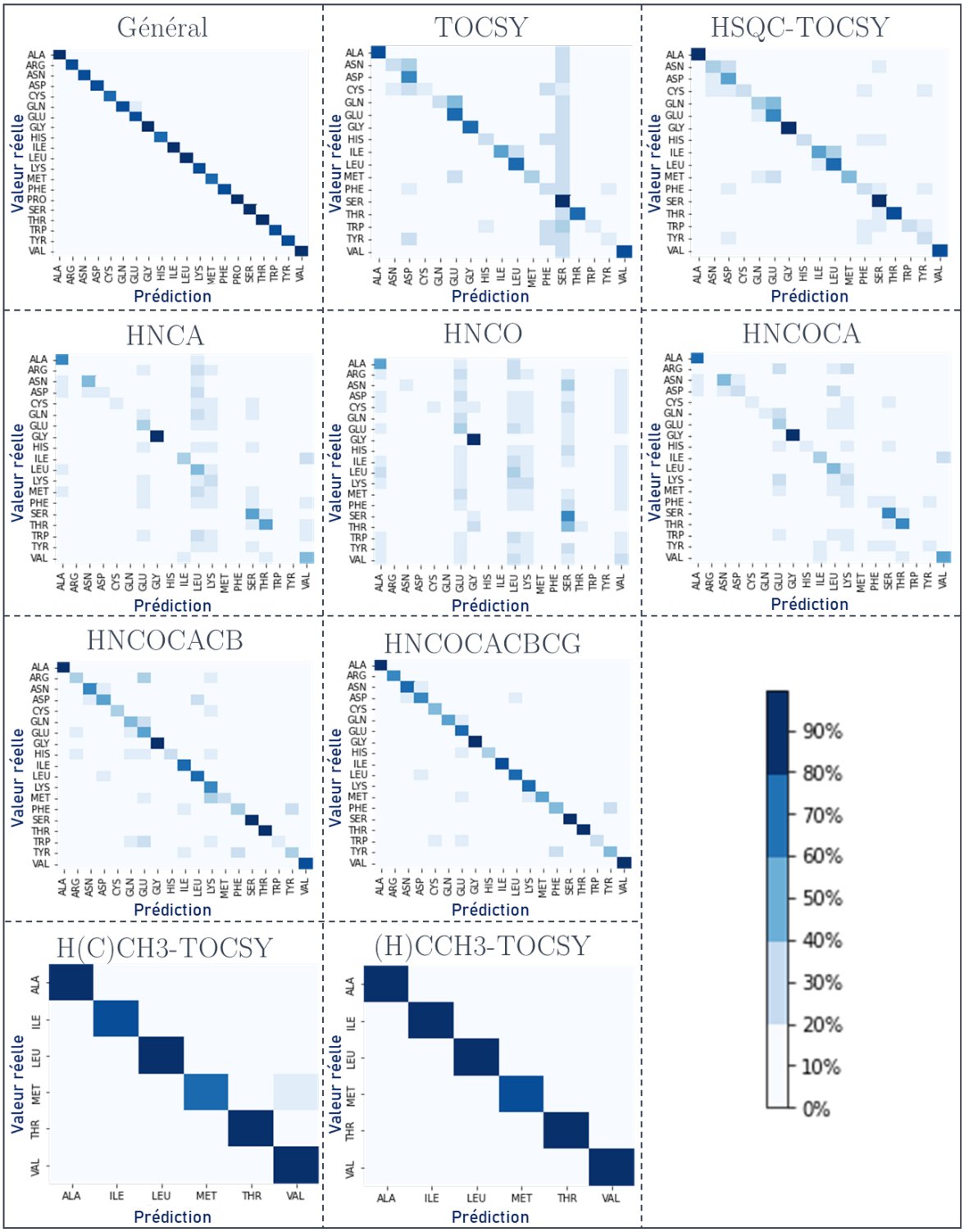


Figure 21 – Les matrices de confusion associées aux résultats de l'entraînement pour les différents scénarios évalués, présentant le taux de réussite pour les valeurs prédites par rapport aux valeurs réelles (plus on tend vers une diagonale foncée, meilleures sont les prédictions).

Certains acides aminés, notamment faisant partie des AMX, semblent toutefois rester difficiles à prédire, quel que soit le scénario mis en place dans ce travail, en particulier CYS qui comporte un thiol qui peut être la cause de la difficulté d'attribution. Par ailleurs, HIS, PHE, TRP et TYR font partie des résidus aromatiques et sont donc spéciaux également, pouvant rendre les prédictions plus compliquées. Davantage de scénarios permettant de les attribuer avec plus d'efficacité pourraient donc être mis en place pour fournir un algorithme plus complet pour tous les types d'acides aminés.

Enfin, les versions précédentes de RESCUE fournissaient déjà des résultats intéressants avec une précision générale sur les prédictions pour l'ensemble des acides de 63.5% pour RESCUE 1 avec un jeu de données équivalent au scénario TOCSY mis en place dans ce travail. RESCUE 2 permettait d'atteindre 70% de précision sur les prédictions pour une expérience de type HSQC-TOCSY.
Les résultats obtenus avec cette 3ème version de RESCUE sont donc équivalents à ce qui était possible avec les versions précédentes, ils sont cependant obtenus dans des temps beaucoup plus restreints que ce qui était possible auparavant.

Le modèle de RESCUE 3 permet de tenir compte des différentes corrélations qui peuvent exister entre les déplacements chimiques ce qui n'était pas le cas pour RESCUE 1 qui utilisait un réseau de neurones de base assez restreint pour des raisons techniques, qui n'avait alors pas assez de degrés de liberté pour représenter toutes les corrélations.
Ici, la transformation fuzzy est utilisée pour l'invariance afin d'être compatible avec un modèle de réseau de neurones, et cela permet par ailleurs de solutionner le problème des données manquantes. Cependant, cela transforme les données et il ne s'agit peut-être pas de la méthode la plus optimale. Une méthode alternative pour l'imputation des données manquantes et l'utilisation d'une transformation invariante différente pourraient être mises en place afin d'améliorer les performances de RESCUE 3.
Néanmoins, l'avantage principal de ce renouvellement est de présenter plusieurs scénarios et une adaptabilité de l'algorithme en conservant de bons taux de réussites sur les prédictions effectuées. En fonction des scénarios, et donc des données disponibles, la précision sur les prédictions effectuées peut aller jusqu'à plus de 94%, fournissant donc des attributions avec un très faible taux d'erreur.

Conclusions et Perspectives

Ce renouvellement de la méthode RESCUE pour une 3ème version permet, grâce aux évolutions technologiques et avec l'augmentation des données disponibles, d'étendre les versions précédentes et d'adapter l'algorithme aux besoins et aux expériences de RMN actuels. Différents algorithmes ont donc été mis en place, d'une part pour transformer les données brutes issues de la BMRB en données utilisables comme jeux d'entrée pour l'algorithme de ML, puis pour réaliser le filtre des données en fonction des scénarios choisis et enfin pour la mise en place du réseau de neurone permettant de prédire le type d'acide aminé en fonction des déplacements chimiques disponibles.

Le système de scénarios mis en place permet une adaptation très complète et relativement simple et rapide du programme aux différents cas réels pouvant se présenter. La méthode peut donc facilement être étendue en fonction des besoins exprimés, d'autres scénarios potentiels peuvent donc être imaginés et testés.

Le calcul étant assez rapide, cela permet ainsi de réentraîner de nouveaux scénarios à volonté. L'application de l'algorithme pour réaliser une prédiction est quant à elle très rapide, quasiment instantanée, et ne demande que très peu de ressources une fois l'entraînement réalisé. Il s'agit de l'avantage principal de ce genre de méthode de DL, les besoins en temps et en ressources informatiques sont limités à l'entraînement de l'algorithme, avec une application très légère et réalisable sur presque n'importe quel support.

Par ailleurs, certaines des méthodes disponibles pour la prédiction d'attribution du type d'acide aminé requièrent des jeux de déplacements chimiques complets, et ne sont donc pas compatibles avec des données incomplètes. La méthode présentée dans ce travail est compatible et peut être appliquées à des données partielles, et y est même très adaptée puisque la base de données d'entraînement contient elle-même des jeux de déplacements chimiques partiels.

Des essais de regroupements des acides aminés, comme cela avait pu être fait pour les versions précédentes de RESCUE, ont été réalisés. Cependant, cela n'apportait pas une amélioration suffisante pour présenter un intérêt. En effet, si la précision en avait été largement augmentée il aurait pu être préférable de distinguer précisément un groupe d'acide aminé plutôt que de prédire un acide aminé en particulier avec une mauvaise précision et donc une haute probabilité d'erreur.

Il a ici été conservé une seule optimisation pour la structure du réseau de neurones artificiel pour tous les scénarios. Ce choix a été fait pour permettre une comparaison globale de l'efficacité des différents scénarios. Néanmoins, il ne s'agit pas de la méthode la plus optimale pour obtenir les meilleurs taux de précision sur les prédictions possibles. En effet, chaque scénario présentant des données différentes, la structure du réseau pourrait être réoptimisée et adaptée à chacun d'entre eux. Ceci permettrait d'améliorer les résultats obtenus pour chacun des scénarios.

Le pas de la transformation fuzzy utilisé dans chacun des scénarios pourrait aussi être optimisé en réalisant des tests pour différentes valeurs et en observant l'efficacité des prédictions des algorithmes mis en place.

Par ailleurs, seule la méthode d'analyse par DL (réseau de neurones ici) a été évaluée, il est possible que d'autres méthodes de ML puissent être appliquées avec efficacité pour ce problème, en particulier par exemple un algorithme de forêt aléatoire.

L'imputation des données manquantes est également une amélioration possible pour la méthode développée ici. En effet, les jeux de déplacements chimiques disponibles sur la BMRB par acide aminé de chaque protéine sont largement incomplets, rendant les différentes données d'entraînement relativement variables, malgré le passage par la transformation fuzzy qui permet de mettre de côté le problème de valeurs manquantes. Il est probable qu'en imputant les valeurs manquantes pour obtenir des jeux de déplacements chimiques d'entraînement puisse améliorer les performances de prédiction de l'algorithme. L'inconvénient étant qu'il sera alors moins robuste pour prédire des jeux de déplacements chimiques incomplets.

Les méthodes de ML se développent de plus en plus dans tous les domaines et notamment pour l'aide à l'attribution d'acides aminés ou de structures secondaires en RMN à partir de différentes informations dont les déplacements chimiques mais également les similarités de séquences (Cheung et al. 2010; Hafsa et Wishart 2014; Shen et Bax 2013). Tous ces développements présentent des méthodes innovantes et peuvent être complémentaires. Les sources d'améliorations de RESCUE 3 sont donc nombreuses, à commencer par la structure du réseau utilisé, ou encore le type d'algorithme de ML mis en place, mais également le traitement du jeu de données et les informations qui en sont extraites.

Enfin, pour que la méthode développée puisse être utilisée facilement par des utilisateurs issus du domaine de la RMN, une interface utilisateur simple d'accès disponible en téléchargement ou sous forme de serveur web serait idéale et permettrait une distribution et une utilisation plus large de l'algorithme.

Partie 2 – Application des techniques de ML a la Spectrometrie de Masse (MS)

Cette seconde partie du manuscrit porte sur les projets d'analyse de données liés à la spectrométrie de masse, en particulier la technique FT-ICR, menés durant le doctorat.

La spectrométrie de masse (MS) est un outil d'analyse très puissant utilisé dans de nombreux domaines scientifiques, notamment en biotechnologies (Aebersold et Mann 2003). Les applications possibles sont très variées grâce à un large panel de techniques, allant de la détermination de la masse d'une molécule à l'établissement de structures tridimensionnelles (3D). La MS bénéficie d'une sensibilité et d'une sélectivité importante permettant de fournir des données qualitatives et quantitatives assez rapidement. De nombreuses techniques sont déjà développées et utilisées pour répondre aux besoins des chercheurs et des scientifiques (Lee et al. 2012). Un avantage de la MS est qu'elle peut éventuellement être couplée à d'autres techniques pour des protocoles plus précis ou plus avancés. Les applications biotechnologiques de la MS ont augmenté après l'invention de techniques d'ionisation douce comme l'ionisation par Electro-Spray (ESI) ou encore la désorption/ionisation par laser assistée par matrice (MALDI), permettant l'analyse de molécules biologiques délicates ou de protéines qui ne résisteraient pas dans d'autres conditions (Domon et Aebersold 2006).

Nous nous intéresserons, dans le cadre de ce travail, en particulier à la spectrométrie de masse à résonance cyclotronique ionique par transformée de Fourier (FT-ICR) qui est une technique de spectrométrie de masse à analyse par piégeage, basée sur le confinement des ions sur des orbites circulaires cyclotroniques de fréquence $\omega = \frac{qB}{m}$, dans une chambre IC, à l'aide d'un puissant champ magnétique uniforme. En effet, l'effet cyclotron indique que, placé dans un champ magnétique B statique et uniforme, le mouvement d'un électron est dirigé par la force de Lorentz et dépendant du champ magnétique B, de la masse m et la charge q de l'ion et est donc circulaire.

En FT-ICR MS, les ions à analyser sont piégés entre deux plaques dans une cellule dans laquelle un champ magnétique perpendiculaire aux plaques est appliqué. Les ions entrent alors en oscillation entre

les deux plaques à une fréquence qui leur est propre. Lors de l'excitation, une impulsion à la fréquence cyclonique de l'ion est effectuée, induisant un phénomène d'entrée en résonance de l'ion suite à l'absorption de l'énergie par ce dernier, ce qui augmente alors son énergie cinétique et par conséquent son rayon d'oscillation et sa vitesse.

Chacun des ions, lorsqu'il est excité par un champ électromagnétique de radiofréquence, génère alors un potentiel électrique dépendant du temps qui peut être mesuré (Comisarow et Marshall 1974).

Une transformée de Fourier permet d'obtenir les fréquences orbitales correspondantes et les spectres MS sont obtenus à l'aide d'une fonction de calibration : $m/q = f(\omega)$, pour compenser des irrégularités expérimentales, dans laquelle m est la masse de la molécule, q est sa charge et ω la fréquence en Hertz enregistrée pendant l'expérience. Cette technique offre une résolution et une précision de la masse plus élevées que toutes les autres, et présente également une bonne sensibilité. Ceci a un avantage important pour des applications où l'on ne dispose que de peu de matériaux ou lorsque l'on désire des spectres plus informatifs. La méthode de MS par FT-ICR permet ainsi d'identifier efficacement les ions dans les complexes protéiques ou de caractériser des mélanges complexes. Cependant, la FT-ICR MS 1-Dimensionnelle (1D) présente l'inconvénient d'un temps de mesure très lent.

La méthode FT-ICR offre également une approche 2D très spécifique et différente de l'approche tandem MS/MS classique. En effet, avec les approches MS/MS classiques, un premier spectre est réalisé avec des ions précurseurs dont certains sont choisis pour être fragmentés lors de la seconde analyse. Ceci a pour conséquence de réduire les informations obtenues et les quantités d'échantillons nécessaires sont importantes car une étape de chromatographie est requise au préalable.

Le principe de la 2D FT-ICR MS est apparu en 1987 avec l'idée de contourner les limitations de la 1D FT-ICR (Pfändler et al. 1987). Cette technique est l'équivalent dans le domaine de la MS de la spectroscopie RMN 2D. Elle permet d'éviter la séparation chromatographique et les étapes d'isolement utilisées dans la technique classique de MS en tandem (Guan et Jones 1989).

Une séquence d'impulsions spécifique est utilisée et peut être optimisée pour obtenir une meilleure exactitude et précision des spectres (van Agthoven et al. 2014). Dans un premier temps, les ions précurseurs sont excités de manière cohérente par une première impulsion d'excitation, et ils commencent à tourner à leur propre fréquence cyclotron. Ensuite, une seconde impulsion excite ou désexcite de manière cohérente les ions précurseurs en fonction de la phase qu'ils ont accumulée pendant

la première période. Les ions précurseurs sont fragmentés dans la cellule ICR, en utilisant soit la méthode de dissociation induite par collision avec électrons (ECD) ou la méthode de dissociation induite par laser (IRMPD/UVPD), pour créer des ions fragments dont l'abondance dépend du rayon cyclotronique des précurseurs. Une troisième impulsion finale est appliquée à la fois sur le précurseur et le fragment, puis le signal est mesuré (van Agthoven et al. 2013).

Après une transformée de Fourier 2D et une calibration en masse, un seul spectre 2D est obtenu dans lequel apparaissent les masses des ions précurseurs et des fragments (Floris et al. 2016).

La 2D FT-ICR MS fournit donc plus d'informations sur les échantillons que les autres techniques utilisées aujourd'hui telles que la LC-MS/MS, le principal avantage de cette nouvelle méthode étant sa résolution.

Par ailleurs, cette méthode permet de se passer de l'étape de chromatographie permettant un gain en temps et en quantité d'échantillon.

De plus, en 2D FT-ICR MS, la génération des modèles de fragmentation ne dépend pas de l'abondance de la protéine. Ceci est principalement dû à l'approche d'acquisition indépendante des données, ou Data-Independent Acquisition en anglais (DIA), par rapport à l'approche classique d'acquisition dépendante des données, ou Data-Dependent Acquisition en anglais (DDA), en tandem MS. Dans une DIA, il n'est pas nécessaire de choisir les ions qui seront fragmentés. En effet, tous les parents qui sont détectés sont décomposés en fragments. Au contraire, avec une approche DDA, l'expérimentateur choisit les ions parents qui seront fragmentés et, la plupart du temps, les plus intenses sont choisis au détriment des ions moins abondants.

La FT-ICR MS permet également de ne pas perdre des familles moléculaires entières lorsqu'il y a des différences d'abondances importantes au sein de l'échantillon, lors de l'étape de LC (chromatographie liquide) par exemple.

La 2D FT-ICR MS est utilisable dans de nombreux domaines scientifiques et peut être optimisée pour presque tous les types d'études spécifiques. Néanmoins, les données produites par cette méthode sont très spécifiques et volumineuses et des développements sur les techniques d'analyse du spectre résultant sont nécessaires, avec pour objectif d'extraire le plus d'informations possible de ces données très complètes et difficilement analysables « à la main ».

I. Projet Européen Horizon 2020 « EU FTICR-MS »

Le projet

Le projet intitulé « EU FTICR-MS », débuté en Janvier 2018, soutenu et financé par le programme de recherche Européen Horizon 2020 (H2020) est un projet d'infrastructure qui vise à instaurer une communauté, un réseau et des outils autour de la technique spécifique de spectrométrie de masse par FT-ICR.

En effet, au vu des besoins de développements actuels dans le domaine, ce type de projet permet de rassembler des laboratoires et entreprises autour du sujet afin de combiner les efforts scientifiques et financiers afin d'obtenir des appareils de mesure et d'analyse innovants, de permettre le partage de données ainsi que d'initier des projets de recherche collaboratifs.

Plusieurs laboratoires et entreprises sont impliquées dans le projet à travers l'Europe, dont CASC4DE, qui est en charge de l'élaboration du plan de gestion des données mais également de la partie informatique de création d'une plateforme d'échange de données entre les différents partenaires. Une des tâches de CASC4DE est aussi le développement d'outils de traitements des données FT-ICR issues du projet incluant de l'analyse et visualisation « classiques » et de l'analyse globale par exploration des données.

A titre d'exemple, une communauté de ce type existe aux Etats-Unis et a permis la construction de deux spectromètres FT-ICR de 21 Tesla, qui sont à ce jour les appareils équipés des aimants les plus puissants au monde. Cela représente une avancée importante car la résolution et la sensibilité d'un spectromètre FT-ICR dépendent directement du champ magnétique dont il dispose (Smith et al. 2018). Ce type d'appareil permet d'assurer des résultats de très haute résolution et précision, afin d'identifier des composés au sein d'échantillons biologiques complexes.

Le plan de gestion des données, ou PGD, est un document texte fourni à tous les partenaires d'un projet décrivant les étapes concrètes qui seront suivies pour traiter toutes les données produites au cours du projet. Il permet de planifier comment, par qui, combien de temps et où les données seront gérées mais également qui peut y avoir accès, à quel moment ou encore à quelles données (Schiermeier 2018). Certaines informations sont obligatoires et doivent figurer dans le PGD - avec des spécificités selon le niveau du projet - comme fournir un résumé des données ou détailler les mesures de sécurité des données.

En réalité, dès qu'un projet traite des données - notamment du big data - un PGD est nécessaire (Editorial Nature 2018). D'ailleurs, pour de nombreux projets, y compris pour les projets européens, il s'agit d'une obligation.

La mise en place d'un PGD impliquait en particulier dans le cadre de ce projet la mise en place de mesures afin de produire des données dites « F.A.I.R. », pour « Findable, Accessible, Interoperable, Reusable », selon les lignes directrices des projets H2020. Des données F.A.I.R. sont nécessaires afin de pouvoir collaborer entre différents partenaires, cela est également indispensable dans l'idée de science ouverte et donc de données ouvertes.

Dans le cadre du projet, et après discussion avec les différents participants, il a été décidé de conserver les données brutes, les données « résultats », les métadonnées associées et les programmes utilisés pour le traitement des données brutes afin de conduire aux résultats. Ces informations sont considérées comme suffisantes afin d'être en mesure de pouvoir réutiliser les données acquises et reproduire les résultats obtenus ou bien de réaliser des traitements différents et complémentaires.

La mise en place de métadonnées structurées est donc un point central dans la rédaction d'un PGD et leur format doit être bien réfléchi afin d'être à la fois extensible et compréhensible. Idéalement on veut pouvoir y retrouver des informations systématiques ainsi que des informations plus ponctuelles et spécifiques à certains types de données par exemple, ne s'appliquant pas à toutes les acquisitions. Il faut également être en mesure d'automatiser leur génération basique, ainsi que de mettre en place la possibilité de les analyser de manière automatique.

Métadonnées

Les métadonnées sont des éléments spécifiques et très importants du PGD. Elles sont des données descriptives associées aux données brutes, ajoutant des informations supplémentaires sur les conditions de génération des données (acquisition et traitement). Elles peuvent contenir des informations sur le ou les échantillons, le ou les opérateurs, le type d'expérience, la configuration instrumentale, le logiciel de traitement, ainsi que toute autre information importante pour interpréter le contenu des données stockées.

Les métadonnées sont en effet des éléments essentiels dans un tel projet, mais elles doivent être élaborées en suivant au mieux les normes existantes dans des domaines scientifiques similaires. De nombreux formats de métadonnées standards sont définis et référencés dans la littérature, du fait qu'ils sont requis pour les exigences de certifications pour les normes de qualité dans les entreprises.
La structure des documents de métadonnées doit être décidée avant de les stocker afin d'améliorer la recherche par les métadonnées. Pour avoir des documents flexibles mais structurés, le mieux est d'utiliser une définition de type de document (DTD) ouverte qui comprend une structure de fichier où des champs peuvent être ajoutés à la demande de l'utilisateur.

L'utilisation de normes déjà déclarées permet d'obtenir des ensembles de données et des métadonnées associées qui peuvent être comparés aux données générées dans le cadre d'autres projets. Les mots utilisés pour décrire les données sont les mêmes quel que soit le laboratoire de production, tout utilisateur est donc en mesure de comprendre et d'extraire les informations nécessaires. En effet, cela améliore l'interopérabilité des données et rend possible l'exploration des données à travers de multiples ensembles de données stockés selon les mêmes normes.

Pour que les métadonnées soient entièrement consultables, elles doivent être complètes et basées uniquement sur les mots contenus dans une liste de mots-clés prédéfinie. Cette liste doit être construite méticuleusement car il ne sera pas possible de la modifier par la suite et elle doit contenir tous les mots nécessaires à la description de chaque condition expérimentale, dispositif, opérateur ou donnée.
Ce que l'on appelle liste de mots-clés représente en réalité une ontologie, c'est-à-dire un ensemble structuré, complet et non ambigu de termes et de concepts utilisés pour représenter et décrire un domaine entier (Figure 22). Souvent au format XML ou YAML, elle contient les définitions de chaque terme mais aussi les connexions entre les concepts et les termes (Hoehndorf, Schofield, et Gkoutos 2015).

A ce jour, il n'existe pas d'ontologie complète pour le champ spécifique de la FT-ICR MS, cependant pour en construire une, il est possible de s'appuyer sur d'autres ontologies qui sont actuellement largement utilisées dans des domaines plus larges comme la spectrométrie de masse (Mayer et al. 2013). Il n'est alors nécessaire que d'ajouter et spécifier les termes et les concepts utilisés dans le domaine FT-ICR.

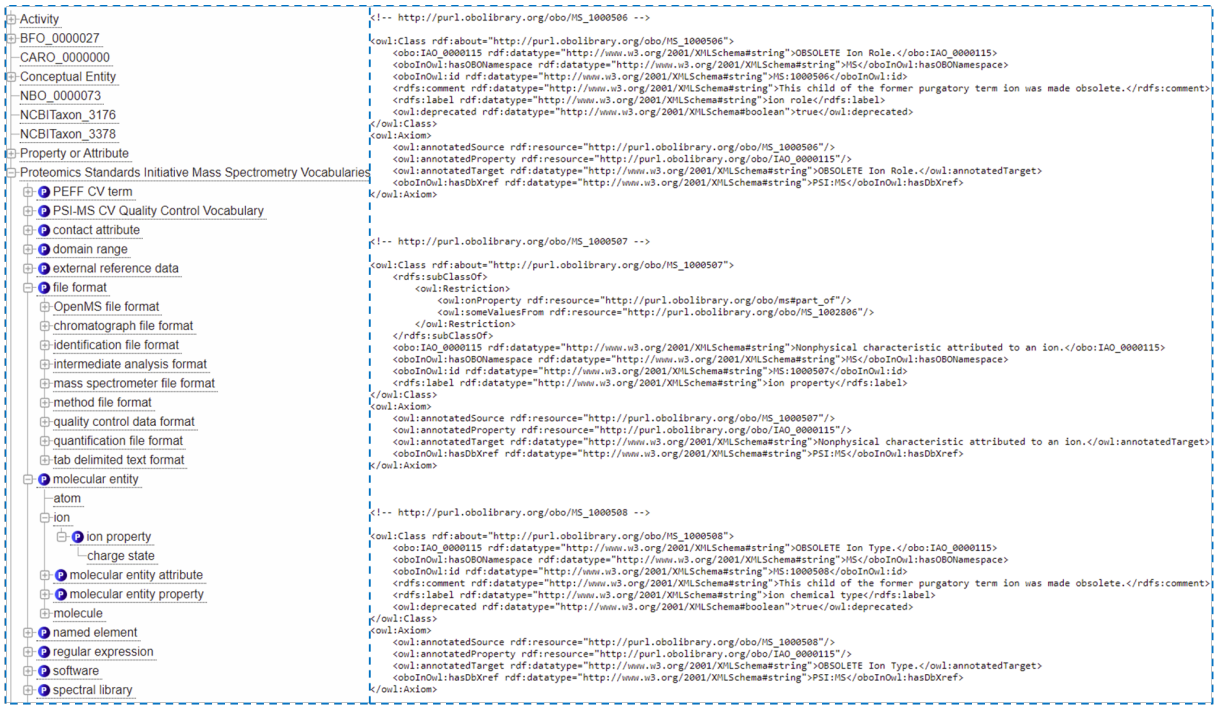


Figure 22 - Extrait de l'ontologie de spectrométrie de masse à droite dans un format web facilement lisible et accessible ici : https://www.ebi.ac.uk/ols/ontologies/ms. A gauche un extrait du format XML disponible entièrement à partir de cette adresse : https://raw.githubusercontent.com/HUPO-PSI/psi-ms-CV/master/psi-ms.owl.

Pour être le plus complet possible, il est important de demander à chaque laboratoire participant de fournir une liste de mots qu'ils utilisent pour décrire leurs ensembles de données et leurs dispositifs, afin de l'ajouter à l'ontologie. Cette approche permet d'obtenir des fichiers de métadonnées bien définis et structurés dans lesquels il sera possible d'effectuer des recherches, par exemple à la fin du projet.

L'implémentation des métadonnées réalisée dans le cadre du projet EU FT-ICR MS est donc conçue de manière à suivre au mieux les différents prérequis énoncés.

Un formulaire automatique de génération de métadonnées a en effet été mis en place (Figure 23). Les différents champs présents dans le formulaire ont été inspirés du formulaire requis pour la réalisation d'expériences au Synchrotron « Soleil ». Des champs complémentaires ont été ajoutés en fonction des besoins spécifiques des laboratoires participants du projet qui requéraient des mots-clés plus avancés afin de décrire leurs données au mieux.

L'utilisation du formulaire mis en place se fait de manière très simple. La première étape consiste à importer les expériences pour lesquelles des métadonnées doivent être générées.
Dans le cas où un fichier de métadonnées est déjà présent, son contenu sera utilisé pour préremplir le formulaire et il suffit alors de modifier les champs nécessaires avant la génération d'une version éditée.
Dans le cas contraire, certains champs « obligatoires » doivent être renseignés ainsi qu'un maximum d'autres champs possibles afin de décrire en détails les données de l'utilisateur.

Différentes sections sont présentes dans le formulaire afin d'y indiquer le maximum d'informations, en effet, la première section concerne les manipulations globales effectuées sur le ou les échantillons ainsi que les considérations de sécurité. Une section de description complémentaire de l'échantillon en fonction de sa nature est également disponible. Une description de l'expérience est ensuite demandée, en fonction du type d'expérience (LC/GC, Imagerie, etc…) des spécifications doivent être indiquées, suivies du type d'instruments utilisés (spectromètre). Cette section est complétée par des champs concernant l'acquisition expérimentale pour lesquels une méthode d'automatisation a été mise en place afin de la préremplir à partir des informations présentes dans le dossier d'expérience Bruker « .d » importé.

Le formulaire est enfin finalisé par deux champs de textes libres en vue d'indiquer dans le premier les informations concernant les prétraitements appliqués aux données brutes avant stockage s'il y en a et dans le second toute information complémentaire jugée utile.
Nous demandons aux utilisateurs d'être très précis, notamment dans ces champs de texte libre, afin que quiconque puisse comprendre, reproduire et réutiliser le jeu de données, en effet, les informations de ce formulaire ne sont pas seulement des notes pour un usage personnel mais elles visent à rendre les données F.A.I.R.

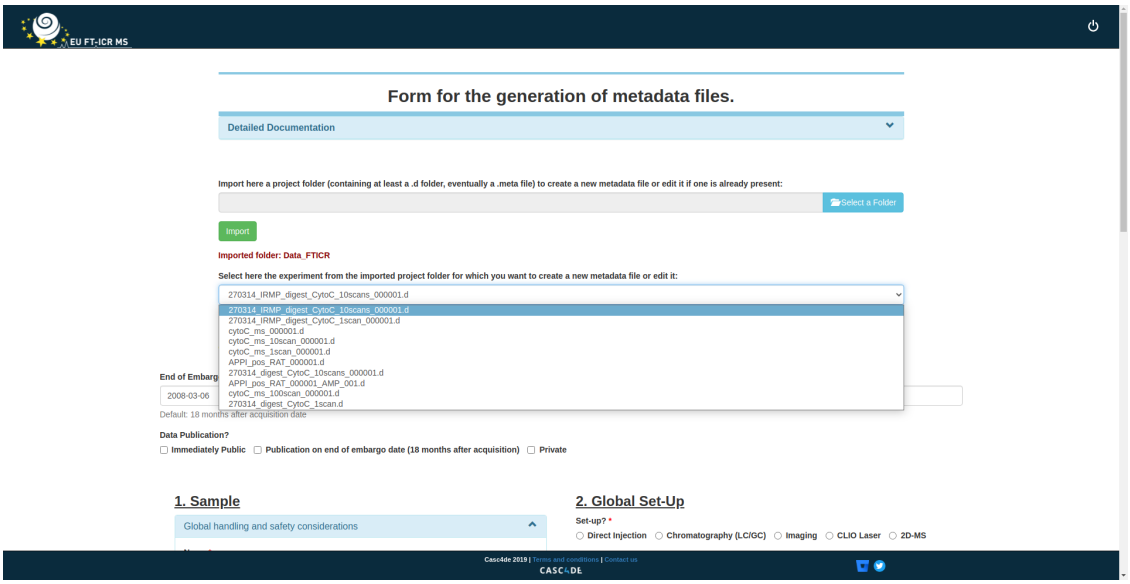


Figure 23 - Capture d'écran du formulaire pour la génération de métadonnées de FT-ICR mis en place dans le cadre du projet H2020.

Le format de fichier « Json » a été choisi comme format de stockage des fichiers de métadonnées générés par le formulaire car il s'agit à la fois d'un fichier au format proche du texte, assez facile à lire, et à la fois d'un format facilement extensible, non fermé à des modifications ou à des évolutions comme des ajouts de champs de description. Par ailleurs, c'est un format qui permet le traitement automatique par des algorithmes pour de la fouille de données par exemple. Les mots-clés utilisés dans le fichier généré suivent, autant que possible, les usages du domaine et des différentes ontologies existantes. Un extrait du fichier généré est présenté en Figure 24.

Toutes les informations stockées au sein de ces métadonnées peuvent ainsi permettre, si elles sont assez nombreuses, des fouilles de données poussées, mais également leur réutilisation pour en tirer de nouvelles informations par exemple, puisque toutes les informations concernant les données associées sont renseignées.

La génération de métadonnées associées aux données du projet dans le cadre du PGD participe donc à rendre les données « F.A.I.R » selon les lignes directrices pour des données ouvertes proposées dans *Nature* en 2016 (Wilkinson et al. 2016).

```
{
  "MetaFileType": "EUFTICRMS v 1.0",
  "MetaFileVersion": "1.0.0",
  "MetaFileCreationDate": "2022-03-29T15:57:42.337152",
  "FileName": "544",
  "SpectrometerType": "Apex",
  "AcqDate": "2022-02-08",
  "EndEmbargo": "2023-08-08",
  "ExcHighMass": "1500.0",
  "ExcLowMass": "114.659424",
  "SpectralWidth": "1875000.0",
  "AcqSize": "4194304",
  "CalibrationA": "2.14987638694703 16E8",
  "CalibrationB": "10.57014825460323",
  "CalibrationC": "0.0",
  "PulseProgam": "basic",
  "MagneticB0": "14.0",
  "ExpName": "544 BTG-WS-Stock_R1.d",
[...]
  "SetUpMethod": "DirectInjection",
  "DI_Solvent": "",
  "ChromatoSys_Ref": "",
  "Column_Construct": "",
[...]
  "Spots_Dist": "",
  "Wavelength_Func": "",
  "CLIO_calib_file": "",
  "TwoD_Method": "",
[...]
  "Spectro_Source": "ESI",
  "ESIFlow": "",
  "ESISprayShield": "",
  "ESINeedleVoltage": "",
  "ESICapillaryEntrance": "",
  "ESICapillaryExit": "",
  "MALDIMatrix": "",
[...]
  "ExcSweepFirst": "",
  "ExcSweepLast": "",
  "RawPreprocess": "",
  "Comment": "",
  "MetaFileEditionDate": "2022-03-29T16:07:38.359255"
}
```


Figure 24 - Extrait de fichier « .meta » produit par le formulaire de génération de métadonnées mis en place.

Le code permettant d'utiliser le formulaire est disponible sur GitHub à l'adresse suivante : https://github.com/CASC4DE/EUFTICRMS_Metadata.

Ce programme est sujet à modifications en fonction des retours et des besoins possibles des utilisateurs ainsi que de l'évolution des méthodes expérimentales au cours du temps.

Le formulaire n'est pour l'instant pas déployé ni disponible en ligne, il ne l'est qu'au sein du réseau privé du projet H2020.

Cependant, il est facile d'installer l'outil au sein d'un laboratoire à partir du script disponible sur GitHub et d'utiliser le formulaire en local pour la génération de fichiers de métadonnées associés aux données de FT-ICR MS et compatibles avec les outils développés par CASC4DE pour leur analyse. En

particulier, un déploiement de ce type est en cours dans le laboratoire LCP-A2MC de l'Université de Lorraine possédant un spectromètre Solarix FT-ICR 7T Bruker (mode 2XR) et ayant pris contact avec nous afin de mettre en place la génération de métadonnées avec leur système. Quelques ajustements ont été réalisés spécifiquement suite aux retours utilisateurs afin d'adapter le formulaire au mieux et de le rendre utilisable notamment sous un environnement Windows.

II. Fouille des données du projet EU FT-ICR MS

Introduction

Le projet EU FT-ICR MS est à l'origine de la production d'un grand nombre de données. Une partie de ces données est produite par des utilisateurs externes de la plateforme, mais chaque laboratoire participant au projet a également été responsable de réaliser l'analyse par spectrométrie de masse FTICR d'un même échantillon afin d'évaluer la méthode, la répétabilité au sein du réseau ainsi que la qualité des différents laboratoires. Cette analyse dite en « round-robin » génère donc un jeu de données avec des paramètres variables pour une même molécule, ce qui est idéal pour l'application de méthode de fouilles de données.

Dans le cas présent, la molécule analysée par les différents laboratoires est le glutathion (« GSH ») de formule brute $C_{10}H_{17}N_3O_6S$ et de masse monoisotopique théorique de 307,0838 g/mol. Le glutathion, un tripeptide : Glutamate-Cystéine-Glycine, est naturellement présent dans de nombreuses plantes, cellules animales ou encore champignons.

Il est possible de simuler finement le spectre théorique de la molécule (Figure 25) afin de pouvoir le comparer aux résultats obtenus par les différents laboratoires. Il est ici étudié l'ion positif du glutathion ($GSH + H^+$), qui sera appelé dans cette étude « cation GSH », de masse monoisotopique 308,09 Da.
Sur le spectre théorique présenté en Figure 25, les massifs isotopiques jusqu'à GSH+3 sont simulés et affichés, des zooms sur les distributions fines de chacun des massifs ont été réalisés.
La structure fine, ou distribution isotopique, observée à l'intérieur de chacun des massifs isotopiques est due à la présence des isotopes principaux des atomes composants la molécule (^{13}C, ^{2}H, ^{15}N, ^{17}O, ^{18}O, ^{33}S, ^{34}S et ^{36}S) et aux combinaisons possibles de ceux-ci.

L'intérêt étant d'être capable de réaliser l'acquisition d'un spectre le plus proche possible du spectre simulé, avec une résolution suffisante des structures fines qui le compose.

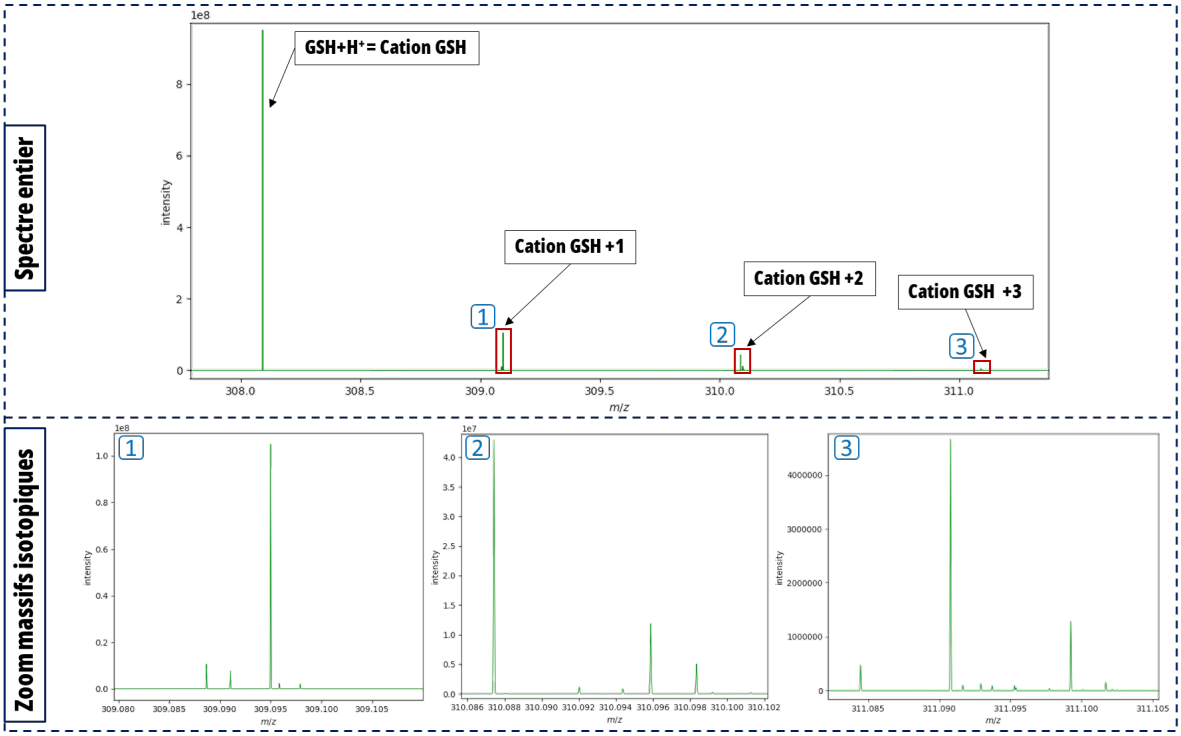


Figure 25 - Spectre théorique de l'ion positif du Glutathion (cation GSH) obtenu avec la bibliothèque de simulation NeutronStar et sur la ligne du bas des zooms sur les massifs isotopiques 1.Cation GSH +1, 2.Cation GSH +2 et 3.Cation GSH +3.

La fouille de données, aussi appelée « data-mining » est une méthode qui permet de manière générale d'analyser un jeu de données d'un point de vue plus global. Au-delà d'une simple analyse de spectre, la méthode consiste à mettre au point un processus de traitement automatique de la masse de données disponibles en vue de mettre en évidence une tendance au sein du jeu de données. Cela peut également permettre de détecter par exemple des biais systématiques de mesures ou de générer des modèles permettant par exemple de prédire des comportements futurs. Cette méthode peut donc permettre de fournir des réponses à des interrogations non résolues par des analyses isolées plus classiques.

Il a été choisi ici de mettre en place un traitement purement automatique de l'ensemble des données disponibles afin que la procédure soit généralisable et puisse s'adapter à un plus grand nombre de données au besoin.

Paramètres d'acquisition et de traitement des données

Ici, il est intéressant de suivre différents paramètres entre le spectre théorique et les spectres acquis par les laboratoires en fonction de variables d'acquisition et de traitement des données. Parmi les paramètres

suivis nous pouvons retrouver la position et l'intensité des pics, le rapport signal sur bruit ou encore les similarités générales entre les spectres pratiques et théorique. Concernant les variables qui s'appliquent aux différents spectres, il en existe de différentes sortes. Nous avons tout d'abord les variables intrinsèques à l'expérience, dont le champ magnétique (B_0) dépendant du spectromètre utilisé ainsi que la méthode d'acquisition narrowband ou broadband choisie. Nous avons ensuite les variables d'acquisition qui incluent la taille du fid ou encore le nombre de scans. Enfin, les variables de traitement des données dont l'apodisation choisie (type, valeur, combinaisons), le zero-filling (zf) ou la troncature du fid.

Le champ magnétique

Le champ magnétique utilisé lors d'une acquisition sur un spectromètre de masse FT-ICR a un impact sur les résultats obtenus. Théoriquement, plus le champ magnétique est élevé plus la puissance de résolution ainsi que la précision et la vitesse d'acquisition le sont (Karabacak et al. 2010).

Acquisition

La résolution et la précision du spectre obtenu sont également dépendantes du nombre de scans effectués et de la durée du fid.

En effet, plus le nombre de scans effectués pour augmenter le rapport signal sur bruit est important, plus il y a un risque de perdre en résolution par instabilité de mesure au fil des scans. Par ailleurs, si le fid est tronqué, il y a un risque important que la qualité de l'expérience s'en ressente.

La durée du fid (D) s'exprime comme suit : $D = N \times \Delta t$ avec N le nombre de points dans le fid et : $\Delta t = \frac{1}{2 \times SW}$, SW étant la largeur spectrale en Hz choisie par l'utilisateur lors de l'acquisition. On a donc : $D = \frac{N}{2 \times SW}$. Par ailleurs, la relation de Gabor indique que la durée d'observation (D) est inversement proportionnelle à la résolution, en fréquence, du spectre. D est un donc un paramètre observable important puisqu'il est directement lié à la qualité du spectre obtenu.

Type d'acquisition : NarrowBand et BroadBand

Les spectres MS dit en "narrowband" ou "broadband" sont issus de différents types d'acquisitions. En effet, les séquences d'impulsions utilisées vont être différentes et déterminées de manière à observer différentes spécificités (Wimperis 1994).

Les spectres broadband sont des spectres larges, qui vont balayer une grande largeur spectrale et obtenir des informations plus larges, avec le désavantage d'avoir souvent une résolution moins poussée. En effet, même si la résolution ne dépend théoriquement que de la durée du fid, pour avoir une haute résolution en broadband, donc sur une grande largeur spectrale, un grand nombre de points est nécessaire. Pour des raisons pratiques, le fid risque donc d'être tronqué et plus court que nécessaire pour avoir une très bonne résolution.

Un spectre en narrowband balaye une bande de fréquence plus réduite, en fonction de ce que l'on souhaite observer, on a des informations sur une zone très précise où l'on sait par exemple que l'on veut observer un signal particulier d'une molécule. Cette méthode présente l'avantage de la facilité de mesure puisqu'elle va contenir moins de points d'acquisition et va généralement demander une excitation plus simple. Cependant la dynamique de mesure est plus faible, on n'aura pas d'informations sur les signaux hors de la zone de mesure choisie.

Traitement

Lors de l'acquisition d'un spectre de masse, les ions vont avoir tendance à se déphaser au cours de l'expérience. Ce déphasage parmi les ions du signal observé va provoquer une perturbation lors de la transformée de Fourier (FT) du fid associé.

En effet, la FT utilise la moyenne du signal sur toute la durée de mesure et est donc impactée si des perturbations apparaissent et que le signal change au cours du temps.
L'application d'une apodisation avant d'effectuer la FT permet d'appliquer un poids variable aux différentes parties du fid, avec idéalement un poids plus faible pour les parties les plus sensibles aux perturbations. Ces parties « perturbées » du signal se trouvent en général au début du fid quand les ions ne sont pas encore totalement entrés en oscillation stable ainsi qu'à la fin du fid lorsque le « paquet » d'ion va commencer à se déphaser légèrement, suite à de petites collisions par exemple. Le début et la fin du fid sont également source de discontinuité du signal périodique qu'est le fid. Ces discontinuités sont sources de perturbations en $\frac{\sin(x)}{x}$ dans le signal. L'apodisation va donc permettre d'améliorer la forme des raies pour augmenter la résolution, cependant ce processus entraîne une perte

du rapport signal sur bruit puisqu'on coupe une partie du début du signal contenant beaucoup d'information.

Deux types d'apodisations combinées ont été testés dans ce travail, l'apodisation de Kaiser et l'apodisation en cloche sinusoïdale. Il s'agit de deux fonctions d'apodisation différentes, ces fonctions mettent en place une fenêtre où le signal va être conservé. Il s'agit généralement de fonction en forme de cloche, nulle en dehors du signal, qui vont avoir le maximum vers le milieu de l'intervalle (donc ici au centre du fid) et qui vont s'amincir sur l'un des bords ou les deux. Lors de la multiplication de ces fonctions avec le signal, l'information au centre va être conservée tandis que l'information sur les bords de la fenêtre d'apodisation va être éliminée.

Une apodisation de Kaiser dite « décalée » a donc été appliquée sur le signal avant FT.
L'apodisation de Kaiser classique, aussi connue sous le nom de Kaiser-Bessel et développée par James Kaiser pour les laboratoires Bell, est une famille de fonctions d'apodisation a un seul paramètre (que l'on appellera β ici) et est généralement utilisée dans l'analyse spectrale.
L'apodisation utilisée ici est une variante de l'apodisation de Kaiser et inclus un second paramètre « maxi » qui décale plus ou moins la position du sommet, rendant ainsi l'apodisation de Kaiser asymétrique. Comme la Figure 26 le montre, plus le paramètre β est grand plus la cloche est fine et donc plus le signal sur les bords du fid va être filtré pour ne conserver que le centre. L'apodisation de Kaiser permet d'englober différentes apodisations « classiques », en particulier celles de Hamming, Hanning ou encore Blackman, en fonction de la valeur de β utilisée, par exemple pour $\beta = 5$ on est en présence d'une apodisation de Hamming.

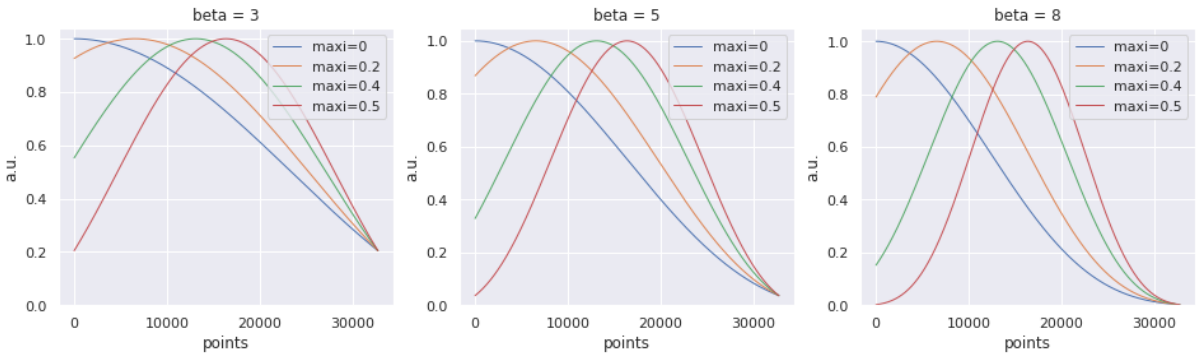


Figure 26 - Fonction d'apodisation de Kaiser dite « décalée » pour différentes valeurs de β et maxi.

Ce type d'apodisation est fréquemment utilisé en spectrométrie de masse pour l'analyse des spectres. La version alternative asymétrique de fonctions d'apodisations est présente dans certains travaux et permet l'analyse de spectres en évitant l'étape de correction de ligne de base (Kilgour et Van Orden 2015; Kilgour et al. 2015).

Cette fonction d'apodisation est assez générale et permet de représenter d'autres fenêtres d'apodisation en fonction des paramètres.

Application de Zero-Filling

L'application d'un processus de Zero-Filling (ou zf) sur des données de MS consiste à ajouter des points vides (ajout de zéros) à la fin du FID avant d'effectuer le traitement par transformée de Fourier.

Comme indiqué, les points ajoutés sont des zéros, donc d'amplitude nulle, ce qui n'a donc théoriquement pas d'impact sur l'information de la donnée.

L'objectif de ce traitement est d'améliorer la qualité d'une donnée puisque le fait d'agrandir artificiellement le FID permet d'augmenter le nombre de points par ppm dans les données après traitement.

Matériel et Méthodes

Données

Les données utilisées pour cette analyse de data-mining sont donc, comme indiqué précédemment, celles résultant du round-robin appliqué à l'échantillon de Glutathionne réalisé dans le cadre du projet H2020 EU-FTICR-MS parmi les différents laboratoires impliqués.

En particulier nous avons intégré ici les données issues de 9 des laboratoires participants au projet EU FT-ICR MS :

- Laboratoire COBRA – UMR6014 CNRS/URN/INSA Rouen – France
- Laboratoire de Spectrométrie de Masse – Université de Liège – Belgique
- Institut Skoltech de Science et Technologie (Skolkovo) – Moscou – Russie
- Institut de microbiologie – Prague – République Tchèque
- Université de Rostock – Rostock – Allemagne

- Université Paris Sud – Orsay – France
- Université Lille – Lille – France
- Université de Finlande Est – Joensuu – Finlande
- Université de Lisbonne (Faculté des sciences) – Lisbonne – Portugal
- Université de Warwick – Warwick – Royaume-Uni

Chacun des laboratoires a réalisé une ou plusieurs acquisitions avec des paramètres spécifiques concernant par exemple le type d'acquisition (broadband/narrowband) ou encore la durée ou la taille du fid, et sur des sites d'analyses différents fournissant ainsi une vingtaine de données différentes pour l'analyse.

Algorithmes de traitement des données

Dans un premier temps le spectre théorique avec des massifs isotopiques fins a été produit.
Les listes de paramètres à explorer ont été mises en place, ainsi que la liste des jeux de données disponibles. Un itérateur a ensuite été mis en place pour générer un spectre et réaliser le peak-picking associé à chaque jeu de paramètres possibles afin de pouvoir les comparer au spectre théorique obtenu à l'aide de la librairie `NeutronStar` (Kreitzberg et al. 2020).

Les 20 données expérimentales présentent différents paramètres d'acquisition et permettent ainsi d'explorer en partie l'effet des paramètres expérimentaux utilisés. Une partie des expériences ont été réalisées en NarrowBand et l'autre part en BroadBand. Le champ magnétique utilisé lors des acquisitions est compris dans la liste suivante :

$$B_0\ (Tesla) = [7, 9, 12, 15]$$

Le nombre de scans (NS) et la durée du fid varient également entre les différentes données dans les listes suivantes :

$$NS = [4, 16, 24, 32, 50, 64, 100, 200, 400, 3000]$$
$$Durée\ du\ fid\ (\text{sec}) = [0.42, 0.7, 0.84, 1.21, 1.4, 1.68, 1.84, 3.36, 3.91, 4.82, 8.39, 11.38]$$

Par ailleurs, les paramètres liés au traitement des données ont également été variés. L'intensité du zero-filling appliqué aux données est compris dans la liste suivante (0 signifiant qu'aucun zero-filling n'est appliqué) :

$$Zero - Filling = [0, 2, 4, 8]$$

Concernant l'apodisation, l'apodisation de Kaiser décalée présentée a été testée en faisant varier les paramètres associés, Maxi ou β, dans les listes de valeurs suivantes :

$$\beta = [3, 5, 8]$$
$$Maxi = [0.2, 0.4, 0.5]$$

Toutes les combinaisons possibles entre les paramètres expérimentaux et les paramètres de traitement ont été réalisées, générant au total environ 1200 combinaisons différentes.

Pour des questions de lisibilité et à titre d'illustration pour les figures, les 20 noms de fichiers ont été remplacés par la suite par le nom du laboratoire suivi du numéro de donnée s'il y en a plusieurs pour un même laboratoire.

Pour le développement du programme, le langage `python` et différentes librairies associées ont été utilisées, en particulier les librairies `SPIKE` pour le traitement des données de spectrométrie de masse, et `NeutronStar`, programme permettant de simuler finement les massifs isotopiques.
La librairie `Pandas` a été utilisée pour la gestion des données après le traitement, et la librairie `Seaborn` est utilisée pour produire les graphes d'analyse.

Résultats

Une fois l'ensemble des données traitées en faisant fluctuer les paramètres présentés précédemment, les résultats sont stockés au format csv puis interprétés à l'aide de la librairie `pandas`. Pour la représentation des résultats de la fouille de données, nous utilisons des graphes « violons ».

Un graphe dit « violon » est un graphe hybride entre un diagramme en boîte et un diagramme de densité de noyau. Il est utilisé pour visualiser la distribution de données numériques. Contrairement à un diagramme en boîte qui ne montre que les statistiques, les graphes en « violon » représentent les statistique et la distribution entière des données, ils sont donc plus détaillés qu'un diagramme en boîte classique (Figure 27). La courbe du graphe violon étant ainsi un ajustement (ou fitting en anglais) de la distribution.

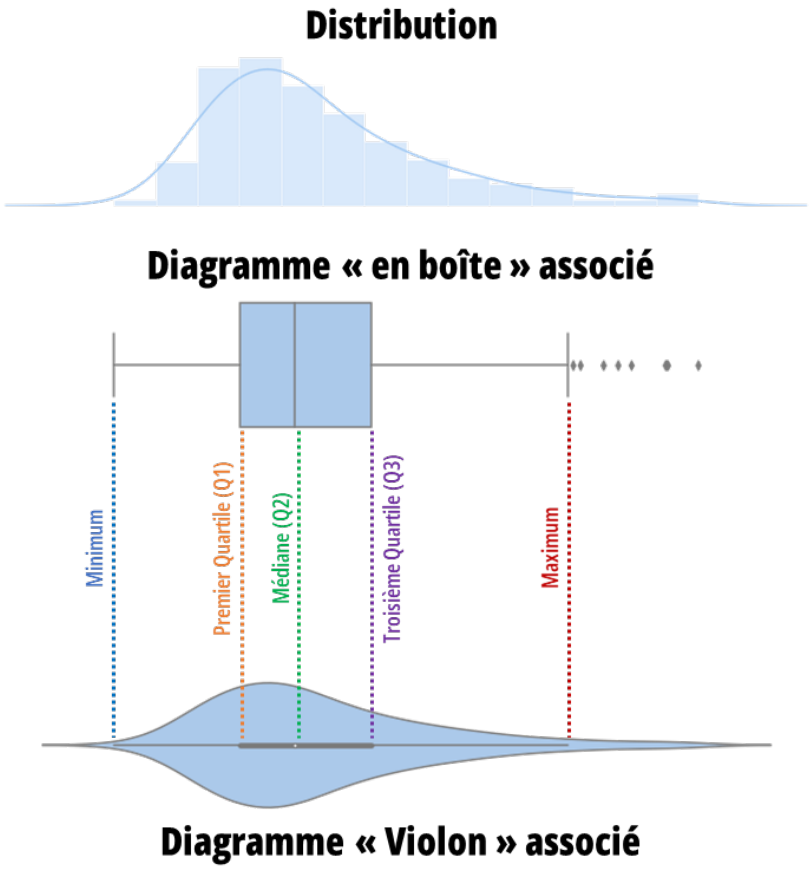


Figure 27 - Schéma de lecture d'un graphe violon en comparaison avec un diagramme en boîte. Sur chacun des diagrammes les informations de médiane et quartiles sont disponibles ainsi que le « minimum » et « maximum ». Les points aberrants sont également visibles à l'extérieur des points dits « minimum » et « maximum ». Généralement une valeur est considérée comme étant aberrante lorsqu'elle est en dehors de 1.5 fois l'intervalle interquartiles [Q1 – 1.5x(Q3-Q1) ; Q3 + 1.5x(Q3-Q1)].

Les effets des différents paramètres d'acquisition et de traitement des données ont été observés sur le rapport signal sur bruit (SNR) de la donnée acquise, sur les similarités de cosinus entre un spectre théorique et le spectre expérimental ainsi que sur les listes de pics en utilisant l'algorithme Hongrois pour l'association des listes de pics pratiques et théoriques.

Le rapport signal sur bruit (SNR) permet de comparer le niveau du signal souhaité au niveau du bruit de fond présent sur la mesure. En mesurant le SNR on décrit donc la différence de puissance entre le signal et le bruit. Le SNR peut donc être interprété comme une approximation de la qualité d'une mesure ou d'un spectre, en effet, plus le bruit est faible par rapport au signal désiré meilleure est la qualité de mesure. Ici le SNR, en décibels, est calculé comme suit :

$$SNR\ (dB) = 20 \times log_{10}(\frac{Max_{Signal}}{std_{Bruit}})$$

Avec Max_{Signal} le signal maximum observé dans le spectre et std_{Bruit} l'écart-type du bruit présent dans le spectre.

La similarité de cosinus permet, quant à elle, de décrire la similarité entre deux éléments en calculant le cosinus de l'angle entre deux vecteurs donnés. Il s'agit d'une méthode couramment utilisée en spectrométrie de masse pour la comparaison de spectre, notamment dans le cadre des réseaux moléculaires (GNPS).

Ici nous avons donc calculé les similarités de cosinus entre les spectres traités avec différents paramètres et le spectre théorique présenté en Figure 25. Afin de pouvoir appliquer le calcul, un prétraitement des données est appliqué en amont. Il faut en effet transformer les spectres en vecteurs. Chacun des deux vecteurs, le théorique et l'expérimental, sont calculés pour faire la même longueur, en ne conservant les informations qu'entre une valeur minimale et une valeur maximale de $^{m}/_{z}$ et uniquement sur les zones où des informations sont observées sur le spectre théorique, le reste des informations étant écartées. Les zones correspondant aux différents massifs isotopiques sur le spectre théorique et pour lesquelles les informations sont récupérées sont les suivantes : $[309.0877 : 309.0989]$, $[310.0847 : 310.1023]$, $[311.0834 : 311.1027]$, $[312.0856 : 312.0951]$. La zone associée au pic monoisotopique étant volontairement non-inclue car étant beaucoup plus intense elle risquerait de biaiser la similarité de cosinus et d'effacer des disparités plus faibles.

Les zones spécifiques du spectre sont alors parcourues par pas de 0.0005 Da et les intensités des données présentes dans chaque intervalle sont cumulées et transposées comme un point du vecteur final allant donc du minimum (ici 308 m/z) au maximum (ici 313 m/z) choisis, par pas de 0.0005 Da. La méthode

est appliquée au spectre théorique et au spectre pratique, donnant deux vecteurs V_{th} et V_{ex}. La similarité de cosinus est enfin calculée comme suit :

$$Similarité\ de\ cosinus = \frac{V_{th} \cdot V_{ex}}{\|V_{th}\|\|V_{ex}\|}$$

Avec V_{th} le vecteur du spectre théorique et V_{ex} le vecteur du spectre expérimental.
Le pas de 0.0005 Da utilisé ici correspond à 1.66 ppm, ce qui est plus large que la résolution des spectres de masse analysés ici. Cette méthode va donc permettre d'évaluer la qualité des intensités des différents massifs isotopiques, mais va effacer les fluctuations au niveau des positions.

Une troisième méthode d'évaluation de la qualité des spectres repose sur les listes de pics obtenues après peak-picking sur les données expérimentales. Ces listes de pics sont comparées à la liste de pics théoriques en calculant la distance cumulée entre ces listes après leur appariement à l'aide de l'algorithme Hongrois. L'algorithme Hongrois, aussi connu sous le nom d'algorithme de Kuhn–Munkres, est un algorithme classique utilisé pour la résolution de problèmes d'attributions (H. W. Kuhn 1955). Ici l'objectif est d'associer au mieux les listes de pics théoriques et pratiques en fonction de leur position et leur intensité. Il s'agit d'un problème d'attribution dite « équilibrée » car on souhaite attribuer un pic théorique à un pic pratique. L'algorithme va associer les pics de manière à minimiser le cout de la matrice de distance entre les listes. La distance cumulée après optimisation est ensuite calculée et utilisée pour évaluer la proximité entre le théorique et l'expérimental, plus la distance est faible, plus les spectres sont proches. L'implémentation de l'algorithme utilisée ici est celle proposée par la librairie python `scipy`.
Cette méthode est idéale pour mesurer la précision sur la position en masse des spectres acquis, la rendant très complémentaire à l'évaluation par similarité de cosinus.

L'application des différentes combinaisons de traitement sur la base de données disponibles génère environ 1200 jeux de données à analyser, chaque élément correspondant alors à une donnée issue d'un laboratoire associée à un jeu de paramètres de traitement (apodisation, zero-filling) spécifique. Pour chaque donnée un peak-picking est réalisé et les trois méthodes d'évaluation présentées ci-dessus sont appliquées systématiquement.

Le temps nécessaire pour le calcul sur l'ensemble de la base de données jusqu'à obtenir un tableau concaténant l'ensemble des résultats est de l'ordre de quelques heures sur un ordinateur équipé de 64Go de mémoire vive et d'un processeur Intel® Xeon® Silver 4114 à 20 cœurs cadencés à 2.20GHz.

Paramètres de traitement de données

Dans un premier temps, les variations du rapport signal sur bruit ainsi que de la similarité de cosinus entre les spectres expérimentaux et le spectre théorique ont été observées en fonction des paramètres utilisés pour le traitement de l'ensemble du jeu de données. Ici cela concerne donc en particulier les paramètres d'apodisation et de zero-filling.

La Figure 28 présente les paramètres β et maxi de l'apodisation de Kaiser décalée présentée plus haut. Ces deux graphes permettent d'observer l'effet des différentes combinaisons de paramètres sur le SNR moyen et la similarité de cosinus moyenne.

On observe, comme attendu, qu'une forte apodisation va avoir tendance à diminuer le rapport signal sur bruit et à augmenter la similarité de cosinus. La distance entre la liste de pics théorique et la liste de pics pratique, après association par l'algorithme hongrois des deux listes, diminue avec β et maxi qui augmentent.

Ainsi, plus l'apodisation appliquée est intense, plus les listes de pics sont proches et donc meilleure est la précision sur la position des spectres expérimentaux par rapport au spectre simulé.

Un jeu de paramètres optimaux peut ainsi être déterminé de façon à améliorer la similarité de cosinus et donc la proximité entre le spectre expérimental et théorique au niveau des intensités, sans pour autant trop perdre de signal, tout en minimisant la distance cumulée entre les listes de pics. Il semble que pour atteindre cet objectif, $\beta = 5$ et Maxi $= 0.4$ sont un bon compromis.

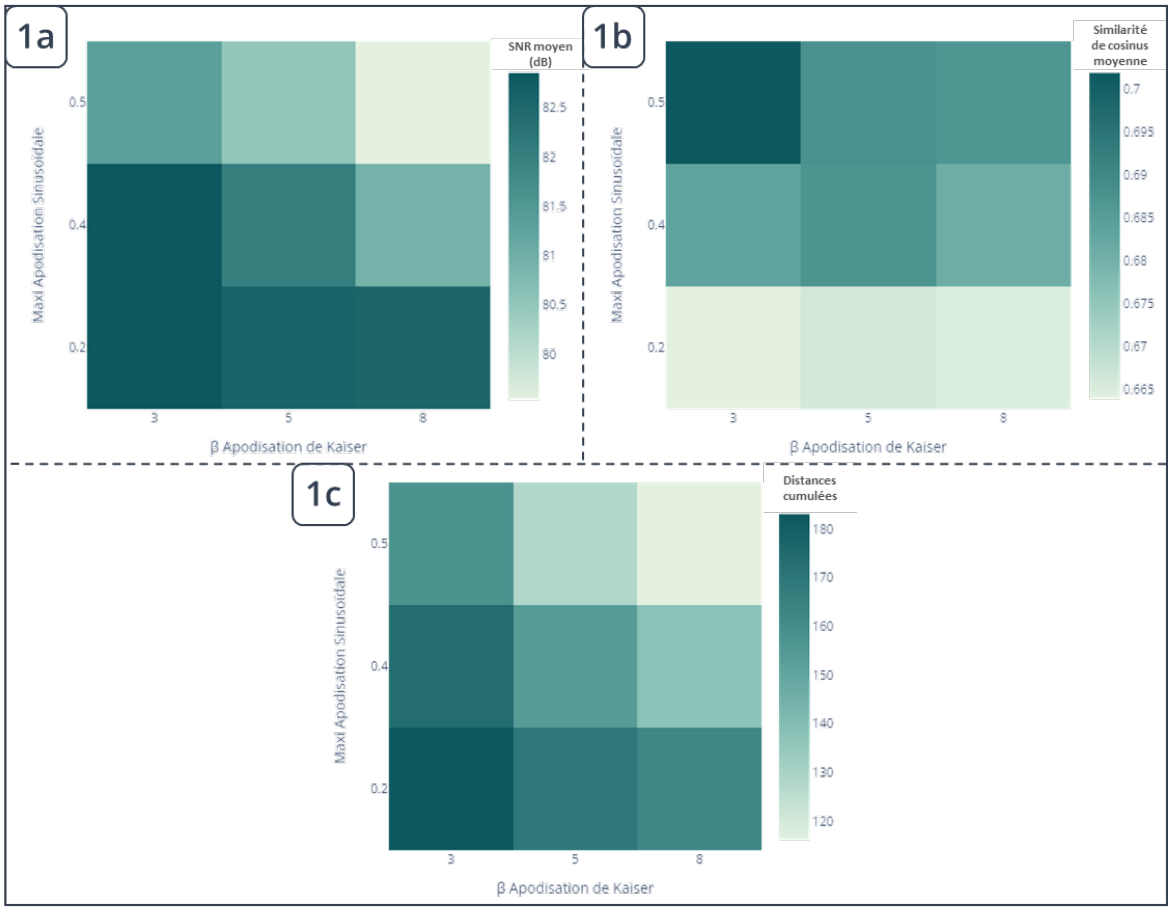


Figure 28 – Impact de la variation des paramètres Maxi et β associés aux apodisations sur (a) le SNR moyen, (b) la similarité de cosinus moyenne et (c) les distances cumulées entre les listes de pics après association par l'algorithme hongrois.

L'impact de l'application de zero-filling pendant le traitement des données a également été analysé. La Figure 29 montre les résultats obtenus sur la similarité de cosinus ainsi que sur le rapport signal sur bruit lors de l'application de différents degrés de zero-filling.

Ainsi, le zero-filling impacte la similarité de cosinus entre le spectre théorique et les spectres expérimentaux. En effet, avec application de zero-filling les données se rapprochent de la théorie puisque la similarité de cosinus tend vers 1. L'effet observé est plus intense entre pas de zero-filling et un zero-filling paramétré à 2.

L'intensification du zero-filling semble ne pas améliorer grandement la proximité des données expérimentales avec le théorique puisqu'un palier est observé entre 2, 4 et 8, bien que la distribution des données se resserre autour de la moyenne avec un zero-filling plus élevé.

Comme attendu, le rapport signal sur bruit ne semble pas être impacté par l'application du zero-filling, quel que soit l'intensité de ce dernier.

Il est donc envisageable d'utiliser un zero-filling à 2 ou 4 qui semblent des valeurs optimales pour l'ensemble du traitement des données, afin de faire tendre les données vers la théorie.

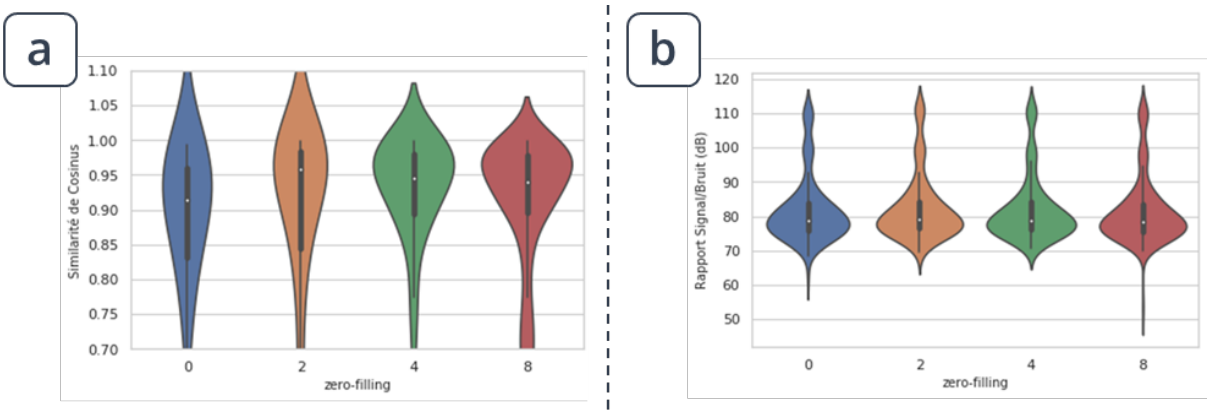


Figure 29 - Effet de l'application d'un zero-filling sur (a) la similarité de cosinus et (b) le rapport signal sur bruit (SNR).

Paramètres d'acquisition

Dans un second temps, les variations du rapport signal sur bruit ainsi que de la similarité de cosinus et de la distance entre les listes de pics entre les spectres expérimentaux et le spectre théorique ont été observées en fonction des paramètres d'acquisition, qui sont donc dépendants et choisis au moment de l'expérience.

Ces variations sont observées pour un traitement des données réalisé avec les paramètres optimaux évoqués précédemment pour l'apodisation soit $\beta = 5$ et Maxi $= 0.4$.

Tous les éléments ont été rassemblés sur la Figure 30, permettant ainsi de repérer les éléments qui vont impacter ou non les spectres expérimentaux. La figure expose, de gauche à droite et pour chaque paramètre expérimental étudié, les effets observés sur la similarité de cosinus, le rapport signal sur bruit et la distance entre les listes de pics théorique et pratique.

Dans le cadre de cette étude et avec les données dont nous disposons, il est donc observé à travers la Figure 30 que certains des paramètres d'acquisition ont un impact important sur la donnée obtenue après application d'une apodisation, d'un zero-filling et d'une transformée de Fourier. Il est à noter que

des points aberrants sont visibles sur ces analyses, les données concernées seront écartées du jeu de données pour les autres études.

En particulier, la qualité générale des spectres augmente avec le champ magnétique. En effet, la similarité de cosinus (et donc la précision de l'intensité des massifs isotopiques) et le rapport signal sur bruit augmentent légèrement.
La distance entre les listes de pics semble diminuer avec l'augmentation du champ magnétique ce qui est signe d'une meilleure qualité des positions des pics, avec une exception pour 12 Tesla. Cette exception peut être due à un trop faible nombre d'exemples pour cette valeur de B_0 ainsi qu'à la présence de spectres de moins bonne qualité parmi les exemples ayant un B_0 à 12 Tesla.
Les variations restent légères et peu significatives, laissant penser que l'influence de B_0 est assez faible sur la qualité du résultat.

La durée du fid a également un impact sur la qualité du spectre, plus particulièrement au niveau des intensité puisque l'on peut voir qu'une durée de fid d'au moins 1.4 secondes est nécessaire pour observer une similarité de cosinus supérieure à 0.8. Ce temps minimal, à partir duquel la similarité de cosinus plafonne, correspond à la durée de vie du paquet d'ions.
Ce paramètre semble moins impacter les autres variables d'évaluation observées. Néanmoins, on peut voir une corrélation entre le SNR et la distance entre les listes de pics, et ce même pour de grandes durées de fid. Cet effet peut être dû à un peak-picking avec un seuil trop bas, qui détecte donc des pics de bruits qui sont associés par l'algorithme Hongrois à des pics de la liste théorique.

Le nombre de scans effectués a peu d'influence sur la similarité de cosinus et le rapport signal sur bruit. On observe cependant que plus le nombre de scans est élevé, plus la distance entre la liste de pics théorique et pratique est importante. Ce résultat est surprenant, l'augmentation du nombre de scans ne devrait pas induire une différence aussi importante sur la précision de position des pics.

Enfin, le type d'acquisition n'impacte pas franchement la qualité des spectres obtenus et est donc relativement négligeable.

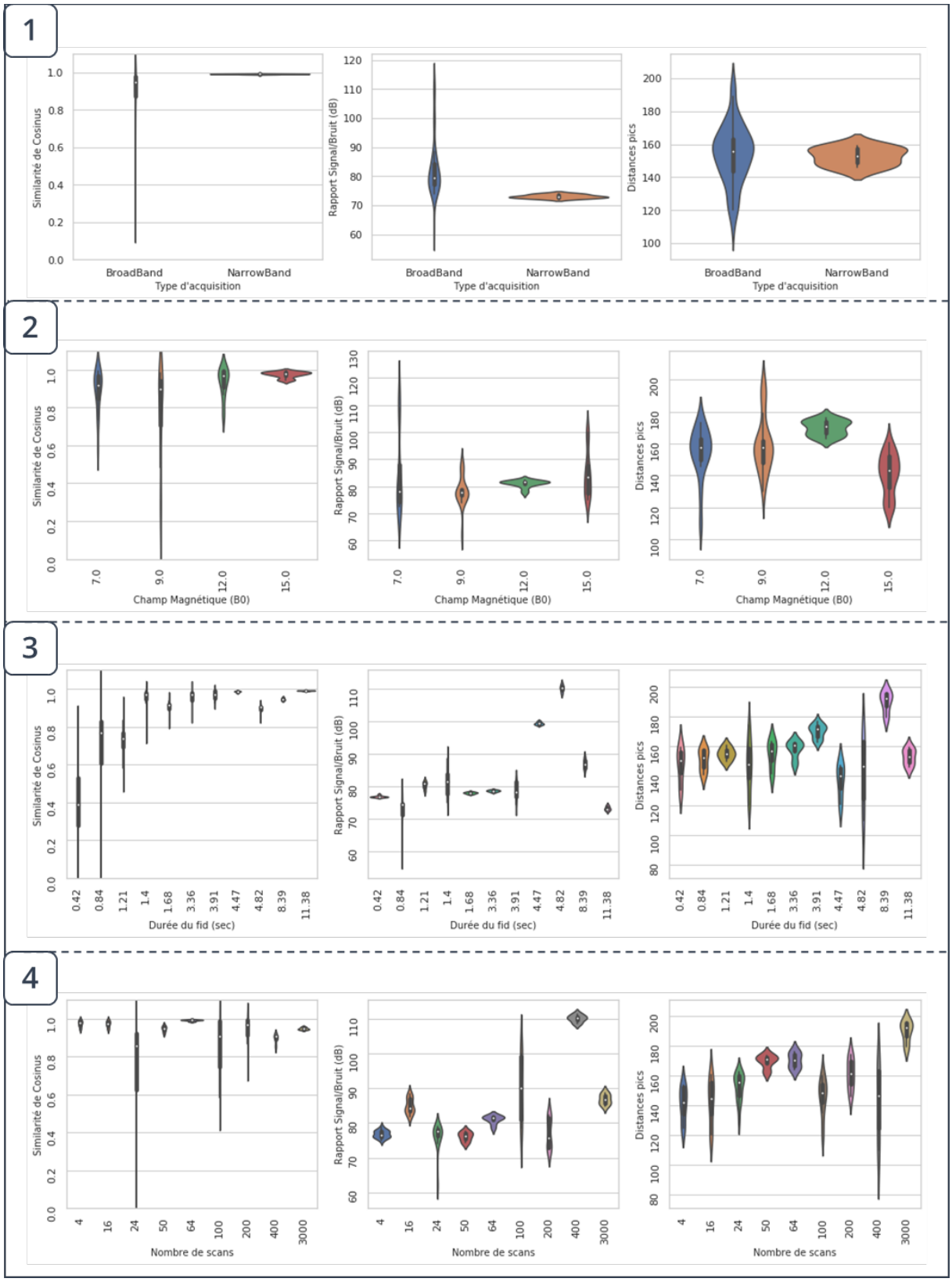


Figure 30 - Effets des différents paramètres intrinsèques à l'acquisition sur, de gauche à droite : la similarité de cosinus, le rapport signal sur bruit (SNR) et la distance entre les listes de pics, après un traitement de données incluant zero-filling et apodisation. 1. Le type d'acquisition (Narrow et Broad Band) 2. Le champ magnétique B_0, 3. La taille du fid (en points), 4. La durée du fid (en secondes), et 5. Le nombre de scans (NS).

Conclusions et Perspectives

Ce travail, mené sur des données issues de différents laboratoire, réalisées par différents expérimentateurs sur différents équipements, a donc permis de mettre en place une analyse commune de l'ensemble des données disponibles. Cette analyse incluant une mise au point d'un traitement post-acquisition ainsi que des comparaisons généralisées et globales des données entre elles ainsi qu'avec ce qui peut être attendu théoriquement.

L'évaluation de la reproductibilité d'analyse a pu être très largement poussée ici, puisqu'on réalise des mesures expérimentales dans des lieux différents, avec opérateurs et systèmes de mesure différents ainsi que dans des temporalités différentes sur un échantillon de glutathion similaire entre les différents laboratoires.

Dans le cadre de cette étude, de nombreux paramètres intervenant pendant l'acquisition ou lors du traitement qui suit ont été variés et leur impact a été évalué.
L'influence des paramètres de traitement de données sur les variables d'évaluation a pu être observée et ceci permet ainsi de mettre au point des paramètres optimaux pour le traitement global du jeu de données. Il a donc été possible de conclure qu'il est préférable, dans le cadre de cette étude et des données qui y sont associées, d'appliquer une apodisation de Kaiser dite « décalée » avec un Maxi = 0.4 et $\beta = 5$. Il a également été choisi d'appliquer un zero-filling, idéalement à 2 ou 4.
Ce traitement permettant d'obtenir les données les plus qualitatives possibles en regroupant toutes les données issues des différents laboratoires participants.
Il a par ailleurs été montré, pour un traitement optimal fixé, que certains paramètres expérimentaux ont également un impact relativement important sur la qualité des données pour les variables utilisées pour l'évaluation de la qualité des spectres, notamment la durée du fid ou le nombre de scans effectués. Une attention particulière doit cependant être portée sur le fait que pour certains paramètres testés, peu d'expériences étaient disponibles, des biais d'analyse sont donc susceptibles d'être induits et présents dans cette étude.

Pour aller plus loin et maintenant que les paramètres de traitement post-acquisition optimaux peuvent être déterminés, davantage de méthode de comparaison des spectres pour analyser leur qualité et leur

proximité avec le spectre attendu théoriquement pourraient être mise en place. Ceci est une étape qui sera plus efficacement mise en place par la collaboration avec les différents laboratoires qui seront à même d'identifier les points importants à observer lors de la comparaison, incluant entre autres l'exactitude de la position des pics ainsi que leur intensité. Les pistes de développement sont nombreuses car le domaine est très actif autour des méthodes de comparaison de spectres de masse avec, en particulier, le développement grandissant des réseaux moléculaires dont cette étape de comparaison est le point de départ. De plus, la méthode mise en place implémentant un traitement complètement automatisé, elle peut s'appliquer à un nombre plus important de données, issues de nombreux laboratoires pour une même molécule, afin d'élargir les aspects qui peuvent être analysés de manière globale.

III. Déconvolution MS algorithmique

Historique et objectifs

La spectrométrie de masse par résonance cyclotronique ionique à transformée de Fourier (FT-ICR MS) permet, comme indiqué précédemment, de mesurer avec précision le rapport masse sur charge des ions moléculaires en phase gazeuse. Elle offre une résolution et une précision de masse élevées, ce qui réduit l'ambiguïté des attributions, spécificité de la méthode très importante pour les applications MS.
Une procédure générale, l'algorithme « Primal-Dual Splitting », implémentant une approche par dictionnaire, a été récemment proposée pour effectuer une reconnaissance du modèle isotopique classique. La méthode se montre très efficace pour l'extraction de la masse monoisotopique et de l'état de charge pour des données en 1D de FT-ICR MS (Cherni, Chouzenoux, et Delsuc 2018).

L'objectif était donc d'explorer la possibilité d'étendre cette approche à la 2D FT-ICR MS qui a récemment connu de nombreux développements et pour laquelle un processus de déconvolution automatique serait d'autant plus utile en raison de la quantité d'informations présentes dans les jeux de données obtenus par cette technique. En effet, l'utilisation de l'approche en 2D permet une alternative à l'approche de caractérisation de protéine en « bottom-up ».
L'utilisation de la technique « bottom-up » consiste à analyser séquentiellement par tandem MS les peptides de la protéine à caractériser (Gillet, Leitner, et Aebersold 2016). Une étape de LC de haute résolution est ajoutée en amont pour séparer les différents peptides générés à partir de la protéine en raison de la complexité de l'échantillon.
L'approche 2D permet de réaliser un travail de caractérisation de protéine sans avoir besoin d'étape préliminaire de LC pour réaliser une séparation. Le résultat d'une expérience 2D est néanmoins très complexe et très informatif. Il s'avère donc très difficile de l'analyser « à la main » et d'en extraire la totalité des informations qu'il contient. L'objectif étant d'obtenir la position la plus exacte possible des pics monoisotopiques présents et d'en déterminer avec précision la masse et la charge.

En FT-ICR MS, la forme des signaux est très spécifique et s'explique principalement par deux phénomènes.

D'une part, les paramètres de la méthode expérimentale utilisée, en particulier la durée d'acquisition et la stabilité du nuage d'ions, vont influencer le signal observé en modifiant la forme, dont la largeur, du pic ou de la tâche.

Par ailleurs, lors de l'acquisition de spectres, les isotopes des noyaux sont présents et ont une distribution statistique connue qui est à l'origine de la génération de motifs isotopiques classiques. Ces motifs s'observent tant en 1D qu'en 2D et se caractérisent par la présence de multiples pics pour un seul signal dû à la différence en masse et en charge des isotopes.

En effet, il existe des isotopes naturels pour la plupart des éléments qui sont observés dans les molécules. Les motifs isotopiques sont donc le résultat des distributions de probabilités associées aux abondances relatives des isotopes des éléments qui composent la molécule.

En prenant l'exemple du carbone, il existe le ^{12}C et le ^{13}C avec des abondances respectives de 98.93% et 1.07%. Pour un seul carbone mesuré, deux pics sont donc observés en spectrométrie de masse. Si n carbones sont présents dans une molécule donnée, on a donc n remplacements possibles du ^{12}C par un ^{13}C soit 2n configurations et n+1 pics observés. Il en va de même pour les isotopes des autres éléments comme l'oxygène ou le soufre par exemple. La structure du motif isotopique, illustrée sur la Figure 31, va donc suivre une loi binomiale tendant vers une gaussienne si on ne considère que les isotopes du carbone et du soufre qui sont les plus répandus (Margrave et Polansky 1962; Yergey 1983).

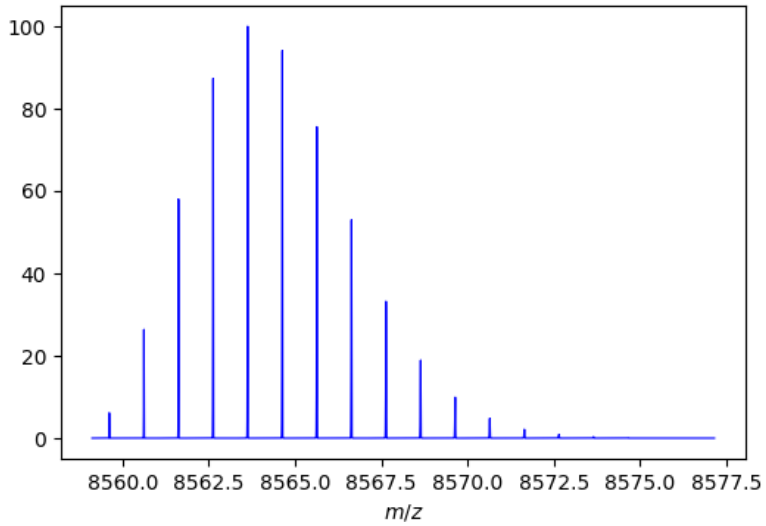


Figure 31 - Illustration de la distribution d'un motif isotopique pour l'exemple du peptide de l'ubiquitine "MQIFVKTLTGKTITLEVEPSDTIENVKAKIQDKEGIPPDQQRLIFAGKQLEDGRTLSDYNIQKESTLHLVLRLRGG" et dont la formule chimique associée est $C_{378}H_{629}N_{105}O_{118}S$.

Pour ce travail, un modèle de motif isotopique diagonal a été utilisé pour la 2D avec une intensité des pics dépendante du fragment et du précurseur.

La Figure 32 présente un zoom sur un motif isotopique typique issu de l'acquisition 2D FT-ICR MS sur l'extrait de levure complet. Le pic monoisotopique étant le signal le plus en bas à gauche du motif isotopique. On peut extraire la masse de l'ion, ainsi que la charge du précurseur z_1 et celle du fragment z_2.

Le modèle utilisé est valable lorsque la taille du fragment est à peu près équivalente à la taille du précurseur.

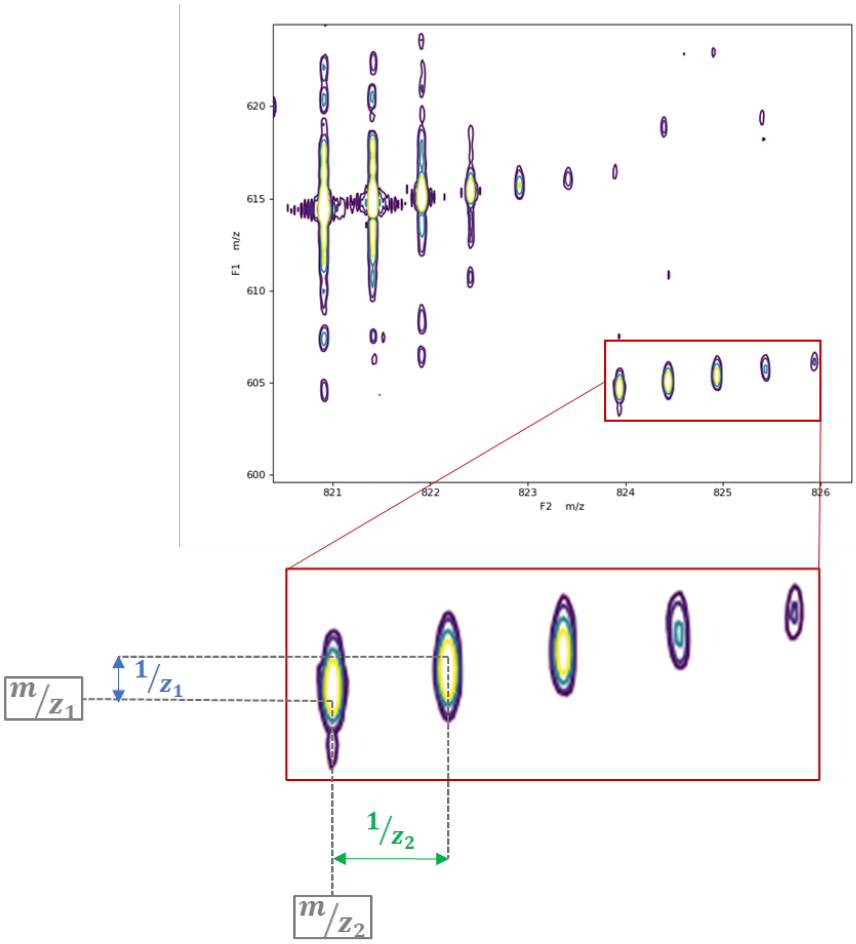


Figure 32 - Zoom sur un motif isotopique observé en 2D FTICR-MS, issu ici du jeu de données d'extrait de levure complet. Les tâches (ou pics) sont séparées le long de chaque dimension par l'inverse de l'état de charge du précurseur z1 et du fragment z2. Le pic monoisotopique est la tâche en bas à gauche du motif.

Néanmoins, nous verrons plus tard qu'il s'agit d'un modèle simplifié qui n'englobe pas toute la complexité du problème. Il faudrait en effet étendre le modèle indiqué en 1D pour la 2D en implémentant

la loi binomiale complète pour les précurseurs et les fragments. Un calcul pour les intensités relatives de la distribution isotopique théorique d'un motif isotopique en 2D a été proposé dans (Halper et al. 2020), et indique que :

« Pour un isotopologue précurseur N contenant n atomes ^{13}C, chaque ^{13}C peut se retrouver soit dans le fragment ou dans son complément. Si le précurseur N se dissocie en un fragment Q contenant q<n atomes ^{13}C, alors son complément contient n-q atomes ^{13}C. La probabilité p(Q, N) d'obtenir le fragment Q à partir du précurseur N se calcule alors comme suit : $p(Q,N) = \frac{P'(Q)P''(N-Q)}{P(N)}$ où p'(Q) est la probabilité d'avoir q ^{13}C dans le fragment, p''(N-Q) la probabilité d'avoir n-q ^{13}C dans son complément et p(N) la probabilité de trouver n ^{13}C dans l'ion précurseur. »

Il est à noter que, comme indiqué pour le motif isotopique 1D, p(N) suit la loi binomiale :

$$\mathrm{p(n, N, 0.013)} = \frac{N!}{n!\,(N-n)!} 0.013^n (1-0.013)^{N-n}$$

Avec N le nombre d'atomes de carbone dans le précurseur, n le nombre de ^{13}C et 0.013 la proportion $^{13}C/^{12}C$.

En spectrométrie de masse, le principe de déconvolution vise à extraire les informations utiles des spectres réalisés. Ici, l'intérêt est donc de réussir à récupérer les informations de masse monoisotopique et de charge pour chaque massif isotopique. Ce processus s'accompagne de différentes problématiques, la principale étant la superposition des signaux sur les spectres tant en 1D qu'en 2D. Les spectres sont d'autant plus difficiles à déconvoluer qu'ils contiennent d'informations, ce qui nécessite donc des techniques automatisées.

En protéomique, de manière générale, des méthodes basées sur l'entropie maximale sont utilisées pour la décomposition en masse ou en charge des spectres acquis. Les premiers algorithmes de déconvolution ont vu le jour dans les années 1990, se basant principalement sur le principe d'entropie (Reinhold et Reinhold 1992).

Le principe d'entropie maximale consiste à approximer une fonction par une distribution de probabilités en identifiant les contraintes auxquelles elle doit répondre. Parmi toutes les distributions qui peuvent répondre aux contraintes évoquées, celle ayant la plus grande entropie est conservée. Ce choix est réalisé

car il s'agit alors de la distribution contenant le moins d'informations et étant par conséquent la plus neutre. Une combinaison d'un algorithme qui effectue l'une puis l'autre décomposition, en masse et en charge, permet d'avoir une déconvolution complète des spectres de masse.

MaxEnt, un algorithme développé par Cambridge Software, est l'un des algorithmes les plus anciens et classiques pour la déconvolution en charge des spectres de masse. Basé sur l'entropie maximum et développé dans un premier temps pour la déconvolution d'image, il a vite été adapté à la RMN et aux spectres Raman puis à la spectrométrie de masse. L'algorithme MaxEnt présente par ailleurs divers avantages qui le rendent performant pour l'analyse dans le domaine, avec entre autres l'amélioration du rapport signal sur bruit ou encore la résolution des spectres (Ferrige et al. 1991). Cet algorithme permet principalement la déconvolution en charge des spectres de masse.

Par ailleurs, l'algorithme « THRASH » pour « Thorough High Resolution Analysis of Spectra by Horn » est très complet et souvent utilisé pour la déconvolution en masse. Cet algorithme se base sur une méthode des moindres carrés pour s'approcher au mieux de la distribution isotopique suivie généralement d'un algorithme de déconvolution de charge. « TRASH » est implémenté de manière open-source par le logiciel « Decon2LS » (Jaitly et al. 2009).

Néanmoins, la majorité des algorithmes utilisés couramment pour la déconvolution ne sont pas vraiment disponibles publiquement car ils sont pour la plupart sous la propriété des constructeurs qui les proposent.

C'est ce qui est l'origine du développement de l'algorithme « Primal-Dual Splitting » présenté plus haut et étendu à la déconvolution en 2D dans ce travail. Il est, comme ceux utilisés classiquement, basé sur une recherche de solution d'entropie maximale avec une méthode des moindres carrés pour s'approcher au mieux des données et sur des régulations d'entropie et de sparsité. Contrairement aux algorithmes constructeurs, l'algorithme « Primal-Dual Splitting » est disponible en open-source.

Données et matériel

Le développement du programme et la réalisation des calculs de déconvolution ont été réalisé sur un ordinateur intégrant un processeur à 4 cœurs, fréquencés à 3.50 GHz, Intel Xeon® CPU E3-1240 v5. L'ordinateur comporte également 16Go de mémoire vive et une carte graphique NVIDIA Quadro® K1200 avec 4Go de mémoire dédiée.

Un jeu de données de FT-ICR 2D, portant sur un extrait de levure digéré par trypsine, a été réalisé à Warwick par l'équipe de P. O'Connor et nous a été transmis dans le cadre de ce travail.

L'acquisition a été réalisé sur un spectromètre FTICR Solarix de Bruker équipé d'un aimant de 12 Tesla. Le spectre associé à ces données est présenté sur la Figure 33 ainsi qu'un zoom sur la partie d'intérêt. Cette représentation de la matrice permet de réaliser la quantité d'informations disponibles dans ce type de données et appuie d'autant plus la nécessité d'avoir des outils automatisés pour l'analyse des cartes 2D FTICR.

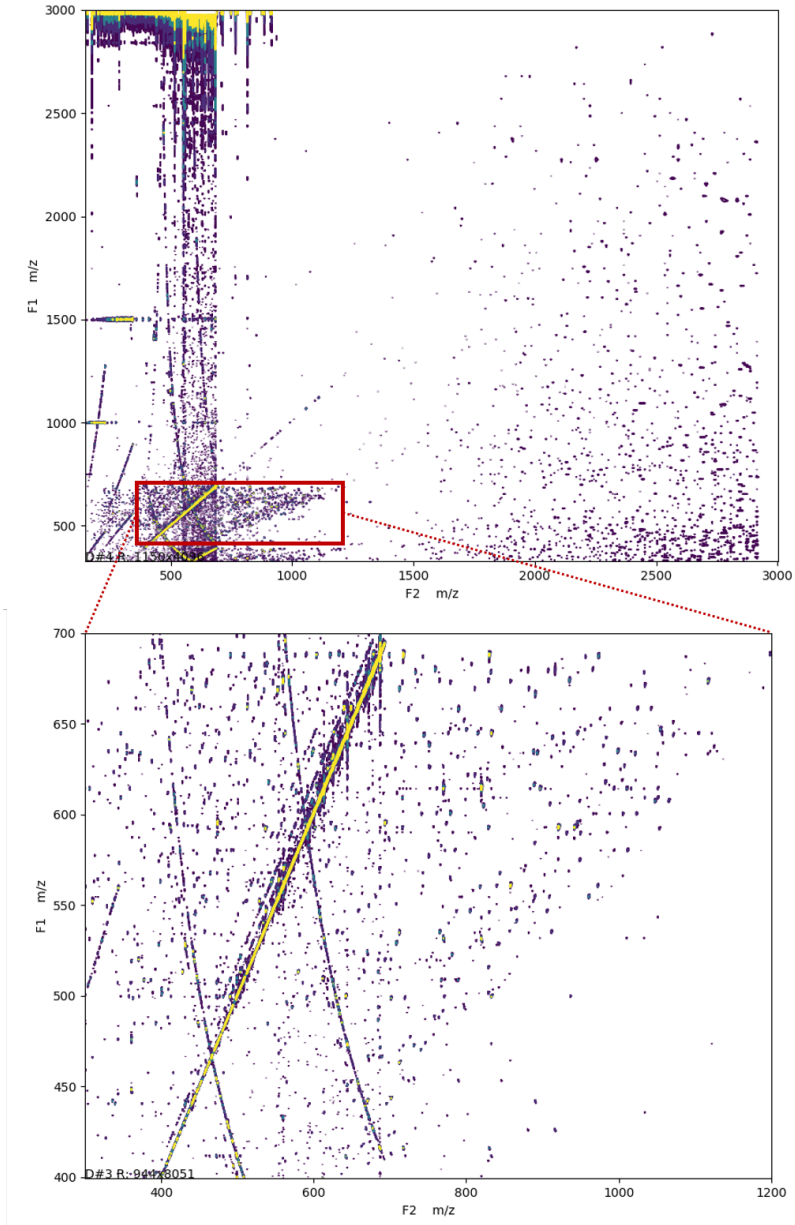


Figure 33 - Donnée de 2D FTICR sur un extrait de levure après digestion trypsique obtenu après transformée de Fourier et conversion en m/z. En bas, un zoom sur la zone d'intérêt porteuse d'information et sur laquelle l'analyse va se porter.

Les données de 2D FTICR sont des cartes en 2D stockées en tant que matrices, il ne s'agit donc pas d'images bien qu'il s'agisse d'éléments 2D et leur analyse ainsi que les procédures de traitements automatiques associées doivent être spécifiquement adaptées. Une méthode de stockage hiérarchique structuré nommé HDF5 a été implémentée et permet de stocker et de manipuler ces données massives de manière plus aisée. En particulier, la matrice d'origine est également stockée sous différentes versions sous-échantillonnées permettant d'accéder rapidement à la donnée entière au besoin.
Pour le jeu de données portant sur l'extrait de levure utilisé dans le cadre de ce développement, la matrice d'origine est de 4k x 256k points. Quatre matrices sous-échantillonnées sont également stockées, représentant au total 1 milliard d'entrées et 9.3 Go d'espace disque. Ceci démontre bien l'ampleur des données de type 2D FTICR qui sont bien un exemple de « big data ».

Méthodes

L'algorithme développé pour la déconvolution en 1D se base sur une approche par dictionnaire. Ce programme était entièrement fonctionnel pour l'application à la 1D FTICR, il a donc fallu adapter les fonctions existantes afin de pouvoir les appliquer en multi-dimensions, cela impliquant d'ajouter par exemple des boucles afin d'inclure plus d'une dimension dans les différents objets et fonctions.

Par ailleurs, comme indiqué précédemment, les données de 2D FTICR sont de grosses données que l'on ne peut pas charger intégralement d'une fois et sur lesquelles il est donc impossible de travailler en un bloc. Il a donc fallu réaliser une fonction permettant de diviser le spectre en plusieurs morceaux qui peuvent être chargés indépendamment les uns des autres et sur lesquels il va être possible d'appliquer les outils de déconvolution adaptés à la 2D. Le chargement de parties du spectre indépendamment les unes des autres est permis grâce à la technologie HDF5 qui permet, à partir d'un objet compressé, de ne le charger qu'en partie sans réaliser une décompression totale de la donnée.

La séparation par morceaux du spectre d'origine est une étape importante. En effet, l'échantillonnage doit être adapté à la zone du spectre et est non régulier car dépendant de $\frac{1}{m}$. Ce type d'échantillonnage permet d'échantillonner plus finement dans les zones de $\frac{m}{z}$ plus faibles qui contiennent plus de données.

La taille des morceaux de spectres analysés est donc une variable importante et de manière générale plus la taille des morceaux est grande, plus le calcul va être long mais meilleur sera le résultat puisque l'on va intégrer une plus grande zone en une fois (Figure 34).

Une fenêtre glissante a également été intégrée avec des « marges » autour des morceaux qui permettent de couvrir l'équivalent d'un motif isotopique (une largeur de 2 Thomson, unité représentant le rapport masse sur charge, a été choisie) afin d'éviter de perdre de l'information entre deux morceaux. Ceci implique par ailleurs un risque d'avoir des détections dédoublées si un motif se retrouve détecté dans deux morceaux. Cela renforce l'idée que plus on arrive à intégrer de gros morceaux en une seule fois, moins le risque sera important d'observer des zones plusieurs fois déconvoluées.

Les marges, comme les tailles de morceaux, sont aussi calculées en fonction de la position dans le spectre pour les mêmes raisons qu'évoqué précédemment, dans une optique d'adapter les valeurs de « découpe » aux données pour le $\frac{m}{z}$ spécifique où l'on se positionne.

Le nombre de points dans chacun des morceaux est cependant calculé pour rester le même, la taille des morceaux va donc varier en $\frac{m}{z}$.

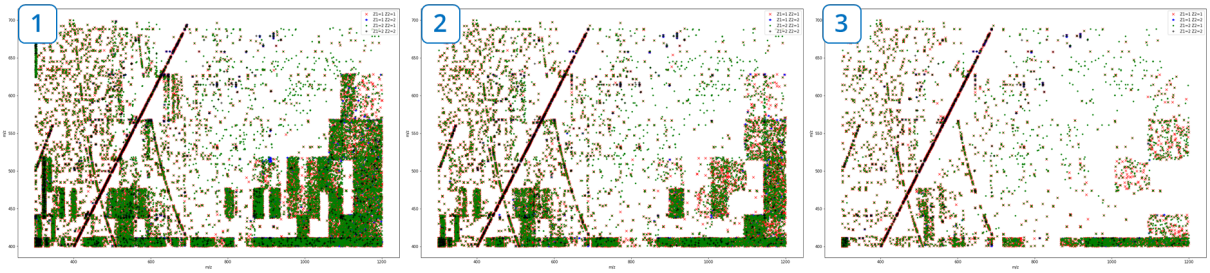


Figure 34 - Illustration de l'impact de la taille des morceaux d'analyse. En (1), chacun des morceaux fait 1024x256 points, en (2) les morceaux font 1536x256 points et en (3) 3072x256 points.

Une optimisation de certains paramètres en fonction des morceaux a également été nécessaire, notamment pour la largeur des pics. En effet, cet élément est important afin de pouvoir réaliser une déconvolution précise.

Il a été observé que la largeur des pics est relativement constante en termes de fréquences. Or $f_c \cong \frac{z\,B_0}{m}$ où B_0 est le champ magnétique, z la charge de l'ion, m sa masse et f_c la fréquence de résonance

cyclotronique. Ceci implique que la largeur des pics Δm est proportionnelle à $\left(\frac{m}{z}\right)^2$. Par ailleurs, la résolution R en spectrométrie de masse s'exprime comme suit $R = \frac{m}{\Delta m}$, ce qui implique que R est proportionnel à $\frac{1}{m/z}$ et il est alors possible d'en déduire que $R\,{}^{m}/_{z}$ est constant sur l'expérience.

La valeur de cette constante a donc été estimée et optimisée en faisant varier la valeur de la largeur des pics sur certaines zones avant d'être appliquée pour calculer la valeur de la largeur des pics pour le reste de l'analyse. Cette optimisation est illustrée sur la Figure 35 à travers l'image (c) où « Width1 » et « Width2 » correspondent aux valeurs de largeur de pics testées sur les axes 1 et 2 (puisque nous travaillons ici en 2D).

Il a été utilisé ici la valeur normalisée de χ^2 après la convergence de l'algorithme de déconvolution pour choisir un optimum, plus cette valeur est faible (vers les tons bleu foncé ici) meilleure est la combinaison de valeurs « Width1 » et « Width2 », puisque la convergence de l'algorithme est meilleure.

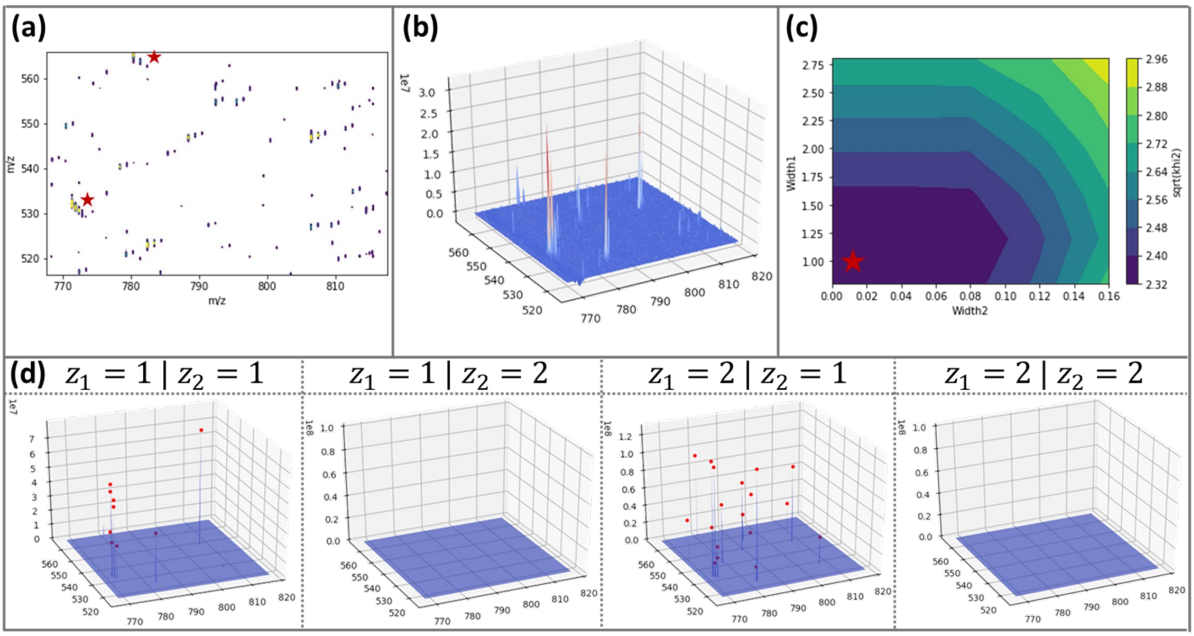


Figure 35 - Processus de déconvolution. a) une région zoomée issue de la donnée analysée, les étoiles indiquent les signaux des harmoniques repliées. b) la même région que dans a) représentée en 3D. c) la recherche des valeurs optimales en faisant varier les paramètres de largeur le long des deux axes, l'étoile indique la valeur utilisée pour l'ensemble de l'analyse ; d) le résultat de la déconvolution optimisée sur la même région pour les différentes paires (z1 ; z2).

L'algorithme de déconvolution peut donc ensuite être appliqué à chacun des morceaux. Dans le cadre de ce travail, seules des valeurs de z_1 et z_2 égales à 1 ou 2 ont été traitées totalisant donc 4 paires (z_1, z_2) présentée en (d) de la Figure 35 pour une région du spectre déconvolué.

Il est à noter que cette manière de faire est très adaptée à la possibilité de paralléliser les calculs puisque la déconvolution de chaque morceau sera indépendante du morceau suivant. Ceci permettant alors de gagner du temps sur l'analyse de l'entièreté de la donnée.
Pour reconstruire la donnée déconvoluée au complet il suffit alors de regrouper les résultats obtenus pour chaque morceau.

L'étape finale de déconvolution consiste à supprimer les combinaisons impossibles avec en particulier, physiquement, seules des paires ($z_1 = 2$, $z_2 = 1$) devraient être détectées sous la diagonale.
La réalisation d'un peak-picking sur la donnée déconvoluée permet d'avoir une liste de pics utilisables.
Il est aussi possible d'ajouter une étape de suppression des pics « doublons » qui peuvent être observés de par la fenêtre glissante utilisée pour la découpe de la donnée en différents morceaux.

Résultats

Comme indiqué plus tôt, il est indispensable dans le domaine de la spectrométrie de masse d'être capable de récupérer le plus précisément possible la position du pic monoisotopique et donc la masse monoisotopique ainsi que la charge pour chacun des motifs isotopiques. Il s'agit d'une tâche compliquée tant en 1D qu'en 2D de par la quantité d'informations présentes dans les données de MS.

L'objectif ici est donc d'appliquer la méthode développée afin de récupérer une liste de pics déconvolués qui puisse être utilisée pour déterminer les composés du mélange étudié.

Le panneau (d) de la Figure 35 montre les résultats de la méthode de déconvolution obtenus pour une région spécifique de la donnée. Il peut être observé que les différents motifs ont été déconvolués (pics bleus), avec une séparation des charges. Les points rouges correspondent au résultat du peak-picking

après déconvolution. Il est à noter ici que la séparation des charges n'est pas parfaite pour cette partie de la donnée.

La Figure 36 présente le résultat complet de la déconvolution par la méthode développée. Le spectre (a) est le spectre original de la donnée analysée sur la zone étudiée, restreinte entre 400 et 700 $^{m}/_{z}$ en ordonnée et 400 à 1200 $^{m}/_{z}$ en abscisse. La donnée d'origine comporte 4096x262144 soit 1 073 741 824 points.

Un peak picking classique a été réalisé, avec un seuil de détection fixé à une intensité de 2E6, et a permis de détecter 10 633 pics.
Après l'application de l'algorithme de déconvolution mis en place on arrive à 6180 pics détectés et 5203 pics après nettoyage des combinaisons impossibles présentes dans les résultats de la déconvolution. La déconvolution permet par ailleurs de séparer les différents pics en fonction des charges (z1 ; z2), ce qui n'est pas possible avec une détection de pics classique.

Cet algorithme permet donc d'une part de diviser par deux le nombre de pics superposés détectés par rapport au peak picking, et donc de réduire la quantité d'informations, notamment redondantes, récupérées. Par ailleurs, les séparations en charges permettent d'augmenter les informations disponibles pour une analyse et de filtrer les informations qui ne seraient pas cohérentes physiquement.

Les temps de calculs nécessaires pour l'application du processus de traitement de données restent assez raisonnables mais sont assez dépendants des paramètres choisis, en particulier de la taille choisie pour les morceaux de la donnée d'origine à analyser.
En effet, comme présenté dans le Tableau 12 pour des morceaux de 3072 x 256 points un morceau est déconvolué en 10 min 30 sec et la donnée complète l'est en moins de 24 h sur un ordinateur sans parallèlisation.

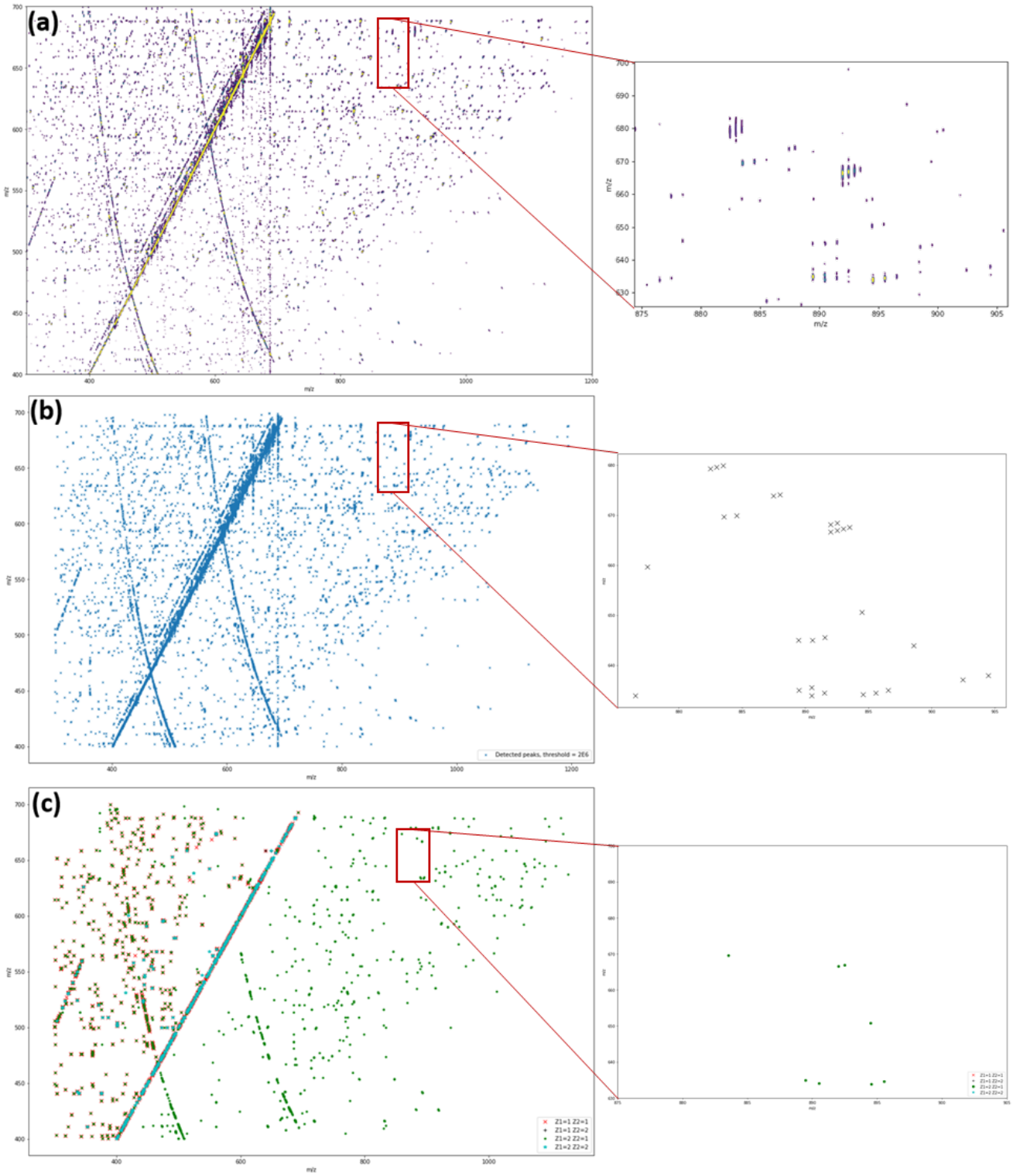


Figure 36 - Résultat de la déconvolution complète après regroupement des différents morceaux pour la donnée 2D FTICR d'extrait de levure. a) Zoom sur la région d'intérêt de la donnée d'origine. b) Résultat d'un simple peak-picking sur la donnée d'origine avec un seuil de détection à 2E6 d'intensité. c) Résultat de la déconvolution après application de la procédure détaillée. Les différentes paires $z_1 - z_2$ sont superposées et différenciées par des couleurs différentes, en rouge : $(z_1 = 1, z_2 = 1)$, en noir : $(z_1 = 1, z_2 = 2)$, en vert : $(z_1 = 2, z_2 = 1)$ et en bleu : $(z_1 = 2, z_2 = 2)$.

Des tailles de morceaux plus importantes ont été testées mais n'ont pas été conservées car les temps de calculs devenaient plus importants pour un gain en efficacité trop faible.

La taille 3072 x 256 points permet donc une obtention de résultats assez rapide ainsi qu'assez peu d'artéfacts dû à la procédure d'analyse comme cela était montré sur la Figure 34.

Tableau 12 - Temps de calculs observés et estimés pour différentes tailles de morceaux choisis pour la découpe de la donnée de 2D FTICR MS.

Taille des morceaux	Temps de traitement observé pour 1 morceau	Temps de traitement estimé pour l'expérience complète	Temps de traitement observé pour l'expérience complète
1024 x 256 points	1 min 15 sec	18.3 h	12.6 h
1536 x 256 points	2 min 30 sec	24.5 h	14.8 h
3072 x 256 points	10 min 30 sec	51.5 h	20.4 h

Conclusions et Perspectives

La méthode mise en place fonctionne et permet d'obtenir des résultats satisfaisants de déconvolution en 2D comme cela était possible en 1D suite à la publication de l'algorithme « Primal-Dual » basé sur une approche par dictionnaire.

Les temps de calculs sont raisonnables et une parallèlisation est possible ce qui permet une application facilitée de l'algorithme.

Le code associé à ce travail est public et disponible sur GitHub : https://github.com/LauraDuciel/MSDeconv.

Par ailleurs, les résultats obtenus étant des listes de pics peuvent être utilisés comme entrées dans les bases de données, comme Mascot, afin d'identifier les peptides présents (Perkins et al. 1999).

Néanmoins, suite et grâce à la publication d'un pré-print en archive ouverte de ce travail, il a été réalisé que l'approximation utilisée pour le modèle du motif isotopique 2D ne permettait pas de représenter

toute la complexité du problème (Duciel, Cherni, et Delsuc 2019). Cela peut expliquer les problèmes persistants, comme la mauvaise séparation des charges, malgré les tentatives d'optimisations.

En effet, comme le montre la Figure 37 qui est un extrait d'une figure issue du papier de (Delsuc, Breuker, et van Agthoven 2021), publié après le papier associé au travail présenté ici, les motifs isotopiques des spectres de 2D FT-ICR MS ne suivent pas un modèle de motif diagonal et on retrouve le modèle de la distribution binomiale complète en 2D. Le motif se rapproche du modèle diagonal uniquement lorsque les tailles des fragments et précurseurs sont similaires.

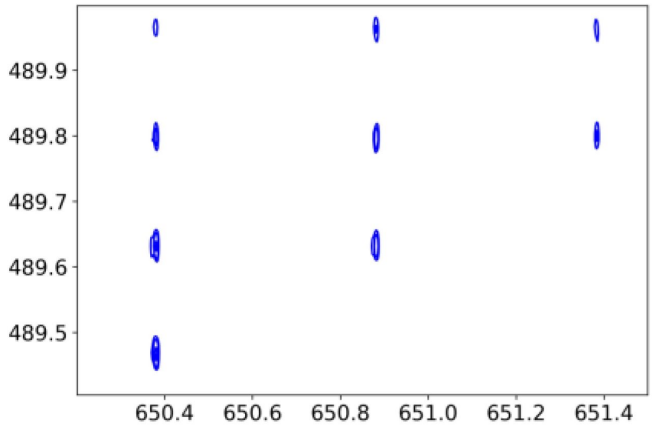


Figure 37 - Zoom sur un motif isotopique 2D complexe, qui ne correspond pas à un modèle de motif diagonal. Image issue de: "Delsuc, M.-A.; Breuker, K.; van Agthoven, M.A. Phase Correction for Absorption Mode Two-Dimensional Mass Spectrometry. Molecules 2021, 26, 3388".

Ceci montre l'intérêt de la publication en archives ouvertes. En effet, d'une part cela participe au concept de science ouverte et permet un partage plus rapide des résultats pouvant être obtenus.

Les publications en archives ouvertes permettent également de diffuser plus aisément des résultats négatifs par exemples qui, bien souvent, sont omis des publications dans les journaux scientifiques, bien que cela commence à apparaitre. Les résultats négatifs ont un apport considérable sur les temps de recherche, puisque cela évite de reproduire et de perdre du temps sur des essais non concluants que d'autres équipes de recherche ont potentiellement déjà réalisés.

D'autre part, la publication en archives ouvertes permet une révision par les pairs (peer-reviewing) plus importante et plus rapide. En effet, toute personne portant un intérêt au sujet abordé aura tendance à lire la publication et à y apporter des remarques si nécessaire. Cela permet de détecter des erreurs ou

encore d'apporter des conseils pour l'amélioration de projets par un regard externe au travail mis en œuvre.

Enfin, un article disponible en archive ouverte sera accessible par tous dans cette version, là où les publications peuvent être d'accès restreint en fonction des journaux dans lesquels elles sont soumises, participant donc au concept de science ouverte

Il serait intéressant dans le futur de mettre en place une méthode de déconvolution par Deep Learning pour la spectrométrie de masse. En effet, cela permettrait d'avoir une alternative à l'élaboration d'un modèle mathématique compliqué pour représenter le modèle complexe du motif isotopique 2D obtenu par cette technique d'analyse.

Un article paru cette année présente un algorithme de DL permettant d'obtenir des images de microscopie similaires à ce qui serait réalisable par microscopie confocale à partir d'images issues d'un microscope à champ large classique (Li et al. 2022). L'algorithme présenté permet donc de déconvoluer l'image afin de détecter les cellules ainsi que les éléments qui la compose pour reconstituer une image complexe, le problème résolu est donc assez similaire au problème rencontré en FT-ICR. La structure du réseau de neurones utilisée et présentée dans cet article pourrait donc permettre, avec des adaptations et les données d'entraînement adéquates, de réaliser la déconvolution des spectres de masses FT-ICR qui nous intéresse.

CONCLUSIONS GENERALES

Ce projet de doctorat a été réalisé en collaboration entre l'entreprise CASC4DE et l'IGBMC (Institut de génétique et de biologie moléculaire et cellulaire) dans le cadre d'une thèse CIFRE. L'objectif du projet était de développer des solutions basées sur des algorithmes d'apprentissage automatique, en Machine Learning (ML) et Deep Learning (DL), pour résoudre des problèmes d'analyse de données de biophysique dus à la taille des données ou bien à leur haute résolution, faisant d'elles des « Big Data ». Le projet s'est en particulier orienté sur des données issues de deux techniques d'analyse courantes en biophysique : la Résonance Magnétique Nucléaire (RMN) et la Spectrométrie de Masse (MS).

La première partie du manuscrit présente les différents projets axés sur les données issues de la RMN.

Le premier projet, intitulé « Plasmodesma », a permis de mettre en place une analyse globale d'une série de données de RMN d'un échantillon bioactif fractionné en vue d'obtenir l'empreinte spectrale du composé responsable de la bioactivité. Ce travail met en place d'une part une technique combinée d'enrichissement et de réduction de dimension des données, nommée « bucketing », en les réduisant à des valeurs statistiques porteuses d'informations. Une analyse par algorithme de ML qui consiste de manière générale à éliminer les données jusqu'à ne conserver que celles étant en corrélation avec la bioactivité permet ensuite de récupérer l'empreinte spectrale. L'analyse est implémentée et réalisable via un serveur web en ligne accessible à tous.

« Fluovial » est un second projet, utilisant une base de données de spectres de ^{19}F RMN avec pour objectif la mise en place d'un algorithme pour la reconnaissance de molécules fluorées. Un algorithme de Random Forest (RF) a été optimisé est mis en place après un traitement des données incluant entre autres une réorganisation et une réduction de dimension des données. Ce projet est un premier pas vers l'industrialisation de la technique pour l'analyse de mélanges de composés fluorés issus d'échantillons de sols pollués par des mousses extinctrices de feux d'hydrocarbures contenant des cocktails de PFAS ayant des effets néfastes sur l'environnement et la santé.

Le projet « RESCUE 3 » est quant à lui le renouvellement d'un algorithme du même nom permettant l'attribution spectrale automatique des protéines à partir de déplacements chimiques

obtenus par RMN. L'idée étant qu'avec les avancées technologiques et l'augmentation des données disponibles au sein de la BMRB, l'efficacité de l'algorithme étant toujours utilisé aujourd'hui pouvait être augmentée. Un réseau de neurone a été mis en place et permet en quelques minutes d'obtenir des prédictions d'acides aminés fiables à partir des valeurs de déplacements chimiques. Un filtrage des données a été mis en place par un mécanisme de scénarios, permettant d'adapter facilement l'algorithme aux données disponibles à l'acquisition en fonction des expériences RMN réalisées.

La seconde partie du manuscrit expose quant à elle les projets portants sur des données de FT-ICR MS.

Dans un premier temps le projet H2020 EU FT-ICR MS est présenté, ainsi que les enjeux qu'il présente et les problématiques associées. Les données générées au sein de ce projet devant être collectées et l'objectif étant de pouvoir les analyser globalement, un système de génération de métadonnées associées aux données a été mis en place. Le format des métadonnées générées a été étudié de manière à englober le plus d'éléments descriptifs des données possibles et le système de formulaire est proposé de façon à pouvoir l'installer et l'utiliser facilement dans un laboratoire en possession d'un spectromètre FT-ICR.

Dans un second temps, les données acquises lors d'un test round-robin sur une même molécule entre les différents laboratoires participants ont été analysées par méthode de fouille de données. L'objectif était de mettre en place un processus d'analyse commun aux différentes données issues des multiples laboratoires, de comparer l'impact des paramètres d'acquisition ainsi que de détecter les paramètres de traitement idéaux, donnant les meilleurs résultats pour l'ensemble des données.

Enfin, l'extension d'une méthode algorithmique de déconvolution des spectres FT-ICR MS a été réalisée pour les données en 2D. Ces données étant très volumineuses et ayant une architecture spécifique, les techniques d'analyses ont dues être adaptées pour permettre une déconvolution par zones sans perte d'informations avec la meilleure efficacité possible. La déconvolution réalisée sur les données de test permet d'obtenir des résultats satisfaisants puisque certaines superpositions vues par peak-picking ne sont plus présentes après l'application de l'algorithme mis en place. Néanmoins, les erreurs de déconvolution, notamment sur les charges, sont toujours présentes. Après analyses par les pairs suite à une publication en archive ouverte, il a été réalisé que le modèle utilisé pour le motif isotopique à déconvoluer était trop simplifié, induisant de potentielles erreurs. L'idée sera donc de mettre en place

une analyse par Deep Learning pour résoudre le problème et éviter la mise en place d'un modèle mathématique très sophistiqué qui représenterait le modèle général exact et complexe du motif isotopique en 2D. En effet, un réseau de neurone adapté pourra apprendre des données et généraliser un modèle permettant la déconvolution plus précise.

L'ensemble des projets présentés et réalisés au cours de ce travail de thèse présentent une problématique liée à la mise en place d'une analyse efficace des données disponibles et obtenues par une méthode biophysique. D'un sujet à l'autre, bien que le problème soit similaire, les données sont différentes et nécessitent des traitements différents et des algorithmes d'apprentissage automatiques spécifiques et adaptés aux besoins. Les possibilités sont multiples pour les choix des algorithmes à utiliser et la recherche évolue vite dans le domaine de la science des données, les solutions apportées évoluent donc également et s'adaptent de mieux en mieux. Ainsi, les choix réalisés et les solutions mises en place dans ce travail pourraient être amenées à être adaptées sur le long terme pour des algorithmes plus efficaces. De manière générale, le domaine en pleine expansion, très riche et intéressant, et la diversité des projets abordés au cours de la thèse ont rendu le travail et l'apprentissage effectués au cours de ces 4 années de doctorat passionnants.

REFERENCES BIBLIOGRAPHIQUES

Aebersold, Ruedi, et Matthias Mann. 2003. « Mass Spectrometry-Based Proteomics ». *Nature* 422 (6928): 198-207. https://doi.org/10.1038/nature01511.

Agthoven, Maria A. van, Lionel Chiron, Marie-Aude Coutouly, Akansha Ashvani Sehgal, Philippe Pelupessy, Marc-André Delsuc, et Christian Rolando. 2014. « Optimization of the Discrete Pulse Sequence for Two-Dimensional FT-ICR Mass Spectrometry Using Infrared Multiphoton Dissociation ». *International Journal of Mass Spectrometry* 370 (septembre): 114-24. https://doi.org/10.1016/j.ijms.2014.06.019.

Agthoven, Maria A. van, Marc-André Delsuc, Geoffrey Bodenhausen, et Christian Rolando. 2013. « Towards Analytically Useful Two-Dimensional Fourier Transform Ion Cyclotron Resonance Mass Spectrometry ». *Analytical and Bioanalytical Chemistry* 405 (1): 51-61. https://doi.org/10.1007/s00216-012-6422-8.

AlQuraishi, Mohammed, et Peter K. Sorger. 2021. « Differentiable Biology: Using Deep Learning for Biophysics-Based and Data-Driven Modeling of Molecular Mechanisms ». *Nature Methods* 18 (10): 1169-80. https://doi.org/10.1038/s41592-021-01283-4.

Bax, Ad, et Mitsuhiko Ikura. 1991. « An Efficient 3D NMR Technique for Correlating the Proton And15N Backbone Amide Resonances with the α-Carbon of the Preceding Residue in Uniformly15N/13C Enriched Proteins ». *Journal of Biomolecular NMR* 1 (1): 99-104. https://doi.org/10.1007/BF01874573.

Beresniewicz, Aleksander. 1986. Tertiary perfluoroalkoxides as surfactants in PTFE dispersion polymerization. United States US4564661A, filed 12 mars 1985, et issued 14 janvier 1986. https://patents.google.com/patent/US4564661A/en.

Bergstra, James, et Yoshua Bengio. 2012. « Random search for hyper-parameter optimization ». *The Journal of Machine Learning Research* 13 (10): 281-305.

Bergstra, James, Dan Yamins, et David D. Cox. 2013. « Hyperopt: A Python Library for Optimizing the Hyperparameters of Machine Learning Algorithms ». *Proceedings of the 12th Python in Science Conference*, 13-19. https://doi.org/10.25080/Majora-8b375195-003.

Chang, Ellen T., Hans-Olov Adami, Paolo Boffetta, Philip Cole, Thomas B. Starr, et Jack S. Mandel. 2014. « A critical review of perfluorooctanoate and perfluorooctanesulfonate exposure and cancer risk in humans ». *Critical Reviews in Toxicology* 44 (sup1): 1-81. https://doi.org/10.3109/10408444.2014.905767.

Charniak, Eugene. 2021. *Introduction au Deep Learning*. Dunod.

Cherni, Afef, Émilie Chouzenoux, et Marc-André Delsuc. 2018. « Fast Dictionary-Based Approach for Mass Spectrometry Data Analysis ». In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 816-20. https://doi.org/10.1109/ICASSP.2018.8461720.

Cheung, Ming-Sin, Mahon L. Maguire, Tim J. Stevens, et R. William Broadhurst. 2010. « DANGLE: A Bayesian Inferential Method for Predicting Protein Backbone Dihedral Angles and Secondary Structure ». *Journal of Magnetic Resonance* 202 (2): 223-33. https://doi.org/10.1016/j.jmr.2009.11.008.

Chiron, Lionel, Marie-Aude Coutouly, Jean-Philippe Starck, Christian Rolando, et Marc-André Delsuc. 2016. « SPIKE a Processing Software dedicated to Fourier Spectroscopies ». *arXiv:1608.06777 [physics]*, août. http://arxiv.org/abs/1608.06777.

Chollet, François. 2018. *Deep Learning with Python*.

Comisarow, Melvin B., et Alan G. Marshall. 1974. « Fourier Transform Ion Cyclotron Resonance Spectroscopy ». *Chemical Physics Letters* 25 (2): 282-83. https://doi.org/10.1016/0009-2614(74)89137-2.

Coote, Paul, Wolfgang Bermel, et Haribabu Arthanari. 2021. « Optimization of phase dispersion enables broadband excitation without homonuclear coupling artifacts ». *Journal of magnetic resonance (San Diego, Calif. : 1997)* 325 (avril): 106928. https://doi.org/10.1016/j.jmr.2021.106928.

Delsuc, Marc-André, Kathrin Breuker, et Maria A. van Agthoven. 2021. « Phase Correction for Absorption Mode Two-Dimensional Mass Spectrometry ». *Molecules* 26 (11): 3388. https://doi.org/10.3390/molecules26113388.

Domon, Bruno, et Ruedi Aebersold. 2006. « Mass Spectrometry and Protein Analysis ». *Science* 312 (5771): 212-17. https://doi.org/10.1126/science.1124619.

Duciel, Laura, Afef Cherni, et Marc-André Delsuc. 2019. « Deconvolution of isotopic pattern in 2D-FTICR Mass Spectrometry of peptides and proteins ». arXiv. https://doi.org/10.48550/arXiv.1906.06218.

Editorial Nature. 2018. « Everyone Needs a Data-Management Plan ». *Nature* 555 (7696): 286-286. https://doi.org/10.1038/d41586-018-03065-z.

Fan, Jerome, Suneel Upadhye, et Andrew Worster. 2006. « Understanding Receiver Operating Characteristic (ROC) Curves ». *Canadian Journal of Emergency Medicine* 8 (1): 19-20. https://doi.org/10.1017/S1481803500013336.

Ferrige, A. G., M. J. Seddon, S. Jarvis, John Skilling, et Robert Aplin. 1991. « Maximum Entropy Deconvolution in Electrospray Mass Spectrometry ». *Rapid Communications in Mass Spectrometry* 5 (8): 374-77. https://doi.org/10.1002/rcm.1290050810.

Floris, Federico, Maria van Agthoven, Lionel Chiron, Andrew J. Soulby, Christopher A. Wootton, Yuko P. Y. Lam, Mark P. Barrow, Marc-André Delsuc, et Peter B. O'Connor. 2016. « 2D FT-ICR MS of Calmodulin: A Top-Down and Bottom-Up Approach ». *Journal of the American Society for Mass Spectrometry* 27 (9): 1531-38. https://doi.org/10.1007/s13361-016-1431-z.

Giesy, John P., et Kurunthachalam Kannan. 2001. « Global Distribution of Perfluorooctane Sulfonate in Wildlife ». *Environmental Science & Technology* 35 (7): 1339-42. https://doi.org/10.1021/es001834k.

Gillet, Ludovic C., Alexander Leitner, et Ruedi Aebersold. 2016. « Mass Spectrometry Applied to Bottom-Up Proteomics: Entering the High-Throughput Era for Hypothesis Testing ». *Annual*

Review of Analytical Chemistry 9 (1): 449-72. https://doi.org/10.1146/annurev-anchem-071015-041535.

Grigorescu, Sorin, Bogdan Trasnea, Tiberiu Cocias, et Gigel Macesanu. 2020. « A Survey of Deep Learning Techniques for Autonomous Driving ». *Journal of Field Robotics* 37 (3): 362-86. https://doi.org/10.1002/rob.21918.

Grinberg, Miguel. 2018. *Flask Web Development: Developing Web Applications with Python*. O'Reilly Media, Inc.

Grinsztajn, Léo, Edouard Oyallon, et Gaël Varoquaux. 2022. « Why do tree-based models still outperform deep learning on tabular data? » https://hal.archives-ouvertes.fr/hal-03723551.

Grzesiek, Stephan, et Ad Bax. 1993. « Amino Acid Type Determination in the Sequential Assignment Procedure of Uniformly 13C/15N-Enriched Proteins ». *Journal of Biomolecular NMR* 3 (2): 185-204. https://doi.org/10.1007/BF00178261.

Guan, Shenheng, et Patrick R. Jones. 1989. « A theory for two-dimensional Fourier-transform ion cyclotron resonance mass spectrometry ». *The Journal of Chemical Physics* 91 (9): 5291-95. https://doi.org/10.1063/1.457575.

Hafsa, Noor E., et David S. Wishart. 2014. « CSI 2.0: A Significantly Improved Version of the Chemical Shift Index ». *Journal of Biomolecular NMR* 60 (2): 131-46. https://doi.org/10.1007/s10858-014-9863-x.

Halper, Matthias, Marc-André Delsuc, Kathrin Breuker, et Maria A. van Agthoven. 2020. « Narrowband Modulation Two-Dimensional Mass Spectrometry and Label-Free Relative Quantification of Histone Peptides ». *Analytical Chemistry* 92 (20): 13945-52. https://doi.org/10.1021/acs.analchem.0c02843.

Ho, Tin Kam. 1995. « Random decision forests ». In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 1:278-82 vol.1. https://doi.org/10.1109/ICDAR.1995.598994.

Hoehndorf, Robert, Paul N. Schofield, et Georgios V. Gkoutos. 2015. « The Role of Ontologies in Biological and Biomedical Research: A Functional Perspective ». *Briefings in Bioinformatics* 16 (6): 1069-80. https://doi.org/10.1093/bib/bbv011.

Jaitly, Navdeep, Anoop Mayampurath, Kyle Littlefield, Joshua N. Adkins, Gordon A. Anderson, et Richard D. Smith. 2009. « Decon2LS: An open-source software package for automated processing and visualization of high resolution mass spectrometry data ». *BMC Bioinformatics* 10 (1): 87. https://doi.org/10.1186/1471-2105-10-87.

Jumper, John, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, et al. 2021. « Highly Accurate Protein Structure Prediction with AlphaFold ». *Nature* 596 (7873): 583-89. https://doi.org/10.1038/s41586-021-03819-2.

Karabacak, N. Murat, Michael L. Easterling, Nathalie Y. R. Agar, et Jeffrey N. Agar. 2010. « Transformative Effects of Higher Magnetic Field in Fourier Transform Ion Cyclotron Resonance Mass Spectrometry ». *Journal of the American Society for Mass Spectrometry* 21 (7): 1218-22. https://doi.org/10.1016/j.jasms.2010.03.033.

Kay, Lewis E, Mitsuhiko Ikura, Rolf Tschudin, et Ad Bax. 1990. « Three-Dimensional Triple-Resonance NMR Spectroscopy of Isotopically Enriched Proteins ». *Journal of Magnetic Resonance (1969)* 89 (3): 496-514. https://doi.org/10.1016/0022-2364(90)90333-5.

Kilgour, David P. A., et Steven L. Van Orden. 2015. « Absorption Mode Fourier Transform Mass Spectrometry with No Baseline Correction Using a Novel Asymmetric Apodization Function ». *Rapid Communications in Mass Spectrometry* 29 (11): 1009-18. https://doi.org/10.1002/rcm.7190.

Kilgour, David P. A., Steven L. Van Orden, Bao Quoc Tran, Young Ah Goo, et David R. Goodlett. 2015. « Producing Isotopic Distribution Models for Fully Apodized Absorption Mode FT-MS ». *Analytical Chemistry* 87 (11): 5797-5801. https://doi.org/10.1021/acs.analchem.5b01032.

Klukowski, Piotr, Roland Riek, et Peter Güntert. 2022. « Rapid Protein Assignments and Structures from Raw NMR Spectra with the Deep Learning Technique ARTINA ». *Nature Communications* 13 (1): 6151. https://doi.org/10.1038/s41467-022-33879-5.

Kramer, Oliver. 2016. « Scikit-Learn ». In *Machine Learning for Evolution Strategies*, édité par Oliver Kramer, 45-53. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-33383-0_5.

Kreitzberg, Patrick, Jake Pennington, Kyle Lucke, et Oliver Serang. 2020. « Fast Exact Computation of the k Most Abundant Isotope Peaks with Layer-Ordered Heaps ». *Analytical Chemistry* 92 (15): 10613-19. https://doi.org/10.1021/acs.analchem.0c01670.

Kuhn, H. W. 1955. « The Hungarian Method for the Assignment Problem ». *Naval Research Logistics Quarterly* 2 (1-2): 83-97. https://doi.org/10.1002/nav.3800020109.

Kuhn, Max, et Kjell Johnson. 2013. « Over-Fitting and Model Tuning ». In *Applied Predictive Modeling*, édité par Max Kuhn et Kjell Johnson, 61-92. New York, NY: Springer. https://doi.org/10.1007/978-1-4614-6849-3_4.

Labudde, D., D. Leitner, M. Krüger, et H. Oschkinat. 2003. « Prediction Algorithm for Amino Acid Types with Their Secondary Structure in Proteins (PLATON) Using Chemical Shifts ». *Journal of Biomolecular NMR* 25 (1): 41-53. https://doi.org/10.1023/A:1021952400388.

Langtangen, Hans Petter. 2009. « A Primer on Scientific Programming with Python ». In . https://doi.org/10.1007/978-3-642-54959-5.

Lee, Young Jin, David C. Perdian, Zhihong Song, Edward S. Yeung, et Basil J. Nikolau. 2012. « Use of Mass Spectrometry for Imaging Metabolites in Plants ». *The Plant Journal* 70 (1): 81-95. https://doi.org/10.1111/j.1365-313X.2012.04899.x.

Li, Bowen, Shiyu Tan, Jiuyang Dong, Xiaocong Lian, Yongbing Zhang, Xiangyang Ji, Xiangyang Ji, Ashok Veeraraghavan, et Ashok Veeraraghavan. 2022. « Deep-3D Microscope: 3D Volumetric Microscopy of Thick Scattering Samples Using a Wide-Field Microscope and Machine Learning ». *Biomedical Optics Express* 13 (1): 284-99. https://doi.org/10.1364/BOE.444488.

Maaten, Laurens van der, et Geoffrey Hinton. 2008. « Visualizing Data using t-SNE ». *Journal of Machine Learning Research* 9 (86): 2579-2605.

Mahesh, Batta. 2019. *Machine Learning Algorithms -A Review*. https://doi.org/10.21275/ART20203995.

Margrave, J. L., et R. B. Polansky. 1962. « Relative abundance calculations for isotopic molecular species ». *Journal of Chemical Education* 39 (7): 335. https://doi.org/10.1021/ed039p335.

Margueritte, Laure, Petar Markov, Lionel Chiron, Jean-Philippe Starck, Catherine Vonthron-Sénécheau, Mélanie Bourjot, et Marc-André Delsuc. 2018. « Automatic Differential Analysis of NMR Experiments in Complex Samples ». *Magnetic Resonance in Chemistry* 56 (6): 469-79. https://doi.org/10.1002/mrc.4683.

Marin, Antoine, Thérèse E. Malliavin, Pierre Nicolas, et Marc-André Delsuc. 2004. « From NMR Chemical Shifts to Amino Acid Types: Investigation of the Predictive Power Carried by Nuclei ». *Journal of Biomolecular NMR* 30 (1): 47-60. https://doi.org/10.1023/B:JNMR.0000042948.12381.88.

Marx, Vivien. 2013. « The Big Challenges of Big Data ». *Nature* 498 (7453): 255-60. https://doi.org/10.1038/498255a.

Mathur, A., et G. M. Foody. 2008. « Multiclass and Binary SVM Classification: Implications for Training and Classification Users ». *IEEE Geoscience and Remote Sensing Letters* 5 (2): 241-45. https://doi.org/10.1109/LGRS.2008.915597.

Mayer, Gerhard, Luisa Montecchi-Palazzi, David Ovelleiro, Andrew R. Jones, Pierre-Alain Binz, Eric W. Deutsch, Matthew Chambers, et al. 2013. « The HUPO Proteomics Standards Initiative-Mass Spectrometry Controlled Vocabulary ». *Database: The Journal of Biological Databases and Curation* 2013: bat009. https://doi.org/10.1093/database/bat009.

McKinney, Wes. 2011. « Pandas: A Foundational Python Library for Data Analysis and Statistics », 9.

Moon, Kevin R., David van Dijk, Zheng Wang, Scott Gigante, Daniel B. Burkhardt, William S. Chen, Kristina Yim, et al. 2019. « Visualizing Structure and Transitions in High-Dimensional Biological Data ». *Nature Biotechnology* 37 (12): 1482-92. https://doi.org/10.1038/s41587-019-0336-3.

Nasrudin, Mohd Wafi, Nizam Shahrul Yaakob, Nazren Amir Abdul Rahim, Mohd Zamri Zahir Ahmad, Nuraminah Ramli, et Mohd Shaiful Aziz Rashid. 2021. « Moment Invariants Technique for Image Analysis and Its Applications: A Review ». *Journal of Physics: Conference Series* 1962 (1): 012028. https://doi.org/10.1088/1742-6596/1962/1/012028.

Natekin, Alexey, et Alois Knoll. 2013. « Gradient boosting machines, a tutorial ». *Frontiers in Neurorobotics* 7. https://www.frontiersin.org/articles/10.3389/fnbot.2013.00021.

National Academies of Sciences, Engineering and Medicine. 2018. *Open Science by Design: Realizing a Vision for 21st Century Research*. National Academies Press. https://doi.org/10.17226/25116.

Newman, David J., et Gordon M. Cragg. 2016. « Natural Products as Sources of New Drugs from 1981 to 2014 ». *Journal of Natural Products* 79 (3): 629-61. https://doi.org/10.1021/acs.jnatprod.5b01055.

Papadakis, Georgios Z., Apostolos H. Karantanas, Manolis Tsiknakis, Aristidis Tsatsakis, Demetrios A. Spandidos, et Kostas Marias. 2019. « Deep learning opens new horizons in personalized medicine (Review) ». *Biomedical Reports* 10 (4): 215-17. https://doi.org/10.3892/br.2019.1199.

Perkins, David N., Darryl J. C. Pappin, David M. Creasy, et John S. Cottrell. 1999. « Probability-Based Protein Identification by Searching Sequence Databases Using Mass Spectrometry Data ». *ELECTROPHORESIS* 20 (18): 3551-67. https://doi.org/10.1002/(SICI)1522-2683(19991201)20:18<3551::AID-ELPS3551>3.0.CO;2-2.

Pfändler, Peter, Geoffrey Bodenhausen, Jacques Rapin, Raymond Houriet, et Tino Gäumann. 1987. « Two-Dimensional Fourier Transform Ion Cyclotron Resonance Mass Spectrometry ». *Chemical Physics Letters* 138 (2): 195-200. https://doi.org/10.1016/0009-2614(87)80367-6.

Pons, J. L., et M. A. Delsuc. 1999. « RESCUE: An Artificial Neural Network Tool for the NMR Spectral Assignment of Proteins ». *Journal of Biomolecular NMR* 15 (1): 15-26. https://doi.org/10.1023/a:1008338605320.

Probst, Philipp, Anne-Laure Boulesteix, et Bernd Bischl. 2019. « Tunability: importance of hyperparameters of machine learning algorithms ». *The Journal of Machine Learning Research* 20 (1): 1934-65.

Ramírez, J., J. M. Górriz, A. Ortiz, F. J. Martínez-Murcia, F. Segovia, D. Salas-Gonzalez, D. Castillo-Barnes, I. A. Illán, et C. G. Puntonet. 2018. « Ensemble of Random Forests One vs. Rest Classifiers for MCI and AD Prediction Using ANOVA Cortical and Subcortical Feature Selection and Partial Least Squares ». *Journal of Neuroscience Methods*, A machine learning neuroimaging challenge for automated diagnosis of Alzheimer's disease, 302 (mai): 47-57. https://doi.org/10.1016/j.jneumeth.2017.12.005.

Reddy, G. Thippa, M. Praveen Kumar Reddy, Kuruva Lakshmanna, Rajesh Kaluri, Dharmendra Singh Rajput, Gautam Srivastava, et Thar Baker. 2020. « Analysis of Dimensionality Reduction Techniques on Big Data ». *IEEE Access* 8: 54776-88. https://doi.org/10.1109/ACCESS.2020.2980942.

Refaeilzadeh, Payam, Lei Tang, et Huan Liu. 2016. « Cross-Validation ». In *Encyclopedia of Database Systems*, édité par Ling Liu et M. Tamer Özsu, 1-7. New York, NY: Springer. https://doi.org/10.1007/978-1-4899-7993-3_565-2.

Reinhold, B. B., et V. N. Reinhold. 1992. « Electrospray Ionization Mass Spectrometry: Deconvolution by an Entropy-Based Algorithm ». *Journal of the American Society for Mass Spectrometry* 3 (3): 207-15. https://doi.org/10.1016/1044-0305(92)87004-I.

Ripley, Brian D. 2007. *Pattern Recognition and Neural Networks*. Cambridge University Press.

Rougier, Nicolas P., Konrad Hinsen, Frédéric Alexandre, Thomas Arildsen, Lorena A. Barba, Fabien C. Y. Benureau, C. Titus Brown, et al. 2017. « Sustainable Computational Science: The ReScience Initiative ». *PeerJ Computer Science* 3 (décembre): e142. https://doi.org/10.7717/peerj-cs.142.

Salloum, Salman, Ruslan Dautov, Xiaojun Chen, Patrick Xiaogang Peng, et Joshua Zhexue Huang. 2016. « Big Data Analytics on Apache Spark ». *International Journal of Data Science and Analytics* 1 (3): 145-64. https://doi.org/10.1007/s41060-016-0027-9.

Schiermeier, Quirin. 2018. « Data Management Made Simple ». *Nature* 555 (7696): 403-5. https://doi.org/10.1038/d41586-018-03071-1.

Shen, Yang, et Ad Bax. 2013. « Protein Backbone and Sidechain Torsion Angles Predicted from NMR Chemical Shifts Using Artificial Neural Networks ». *Journal of Biomolecular NMR* 56 (3): 227-41. https://doi.org/10.1007/s10858-013-9741-y.

Smith, Donald F., David C. Podgorski, Ryan P. Rodgers, Greg T. Blakney, et Christopher L. Hendrickson. 2018. « 21 Tesla FT-ICR Mass Spectrometer for Ultrahigh-Resolution Analysis of Complex Organic Mixtures ». *Analytical Chemistry* 90 (3): 2041-47. https://doi.org/10.1021/acs.analchem.7b04159.

Sorzano, C. O. S., J. Vargas, et A. Pascual Montano. 2014. « A survey of dimensionality reduction techniques ». arXiv. https://doi.org/10.48550/arXiv.1403.2877.

Tolles, Juliana, et William J. Meurer. 2016. « Logistic Regression: Relating Patient Characteristics to Outcomes ». *JAMA* 316 (5): 533-34. https://doi.org/10.1001/jama.2016.7653.

Truong, Anh, Austin Walters, Jeremy Goodsitt, Keegan Hines, C. Bayan Bruss, et Reza Farivar. 2019. « Towards Automated Machine Learning: Evaluation and Comparison of AutoML Approaches and Tools ». In *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, 1471-79. https://doi.org/10.1109/ICTAI.2019.00209.

Ubel, F. A., S. D. Sorenson, et D. E. Roach. 1980. « Health status of plant workers exposed to fluorochemicals - a preliminary report ». *American Industrial Hygiene Association Journal* 41 (8): 584-89. https://doi.org/10.1080/15298668091425310.

Uhrín, Dušan, Stanislava Uhrínová, Claire Leadbeater, Jacqueline Nairn, Nicholas C Price, et Paul N Barlow. 2000. « 3D HCCH3-TOCSY for Resonance Assignment of Methyl-Containing Side Chains in 13C-Labeled Proteins ». *Journal of Magnetic Resonance* 142 (2): 288-93. https://doi.org/10.1006/jmre.1999.1951.

Wang, Mingxun, Jeremy J. Carver, Vanessa V. Phelan, Laura M. Sanchez, Neha Garg, Yao Peng, Don Duy Nguyen, et al. 2016. « Sharing and Community Curation of Mass Spectrometry Data with Global Natural Products Social Molecular Networking ». *Nature Biotechnology* 34 (8): 828-37. https://doi.org/10.1038/nbt.3597.

Ward, Jonathan Stuart, et Adam Barker. 2013. « Undefined By Data: A Survey of Big Data Definitions ». arXiv. https://doi.org/10.48550/arXiv.1309.5821.

Wilkinson, Mark D., Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, et al. 2016. « The FAIR Guiding Principles for Scientific Data Management and Stewardship ». Comments and Opinion. Scientific Data. 15 mars 2016. https://doi.org/10.1038/sdata.2016.18.

Wimperis, S. 1994. « Broadband, Narrowband, and Passband Composite Pulses for Use in Advanced NMR Experiments ». *Journal of Magnetic Resonance, Series A* 109 (2): 221-31. https://doi.org/10.1006/jmra.1994.1159.

Yergey, James A. 1983. « A General Approach to Calculating Isotopic Distributions for Mass Spectrometry ». *International Journal of Mass Spectrometry and Ion Physics* 52 (2): 337-49. https://doi.org/10.1016/0020-7381(83)85053-0.

Ying, Xue. 2019. « An Overview of Overfitting and Its Solutions ». *Journal of Physics: Conference Series* 1168 (février): 022022. https://doi.org/10.1088/1742-6596/1168/2/022022.

Zhang, Min-Ling, et Zhi-Hua Zhou. 2007. « ML-KNN: A Lazy Learning Approach to Multi-Label Learning ». *Pattern Recognition* 40 (7): 2038-48. https://doi.org/10.1016/j.patcog.2006.12.019.

Zhang, Shuai, Lina Yao, Aixin Sun, et Yi Tay. 2019. « Deep Learning Based Recommender System: A Survey and New Perspectives ». *ACM Computing Surveys* 52 (1): 5:1-5:38. https://doi.org/10.1145/3285029.

Université de Strasbourg

École Doctorale des Sciences de la Vie et de la Santé STRASBOURG

Laura DUCIEL

Développements d'algorithmes d'apprentissage machine pour l'analyse automatique de données biophysiques de haute résolution.

RESUME

La spectrométrie de masse (MS) et la résonance magnétique nucléaire (RMN) sont deux techniques courantes en biologie avec de nombreuses applications pharmaceutiques, environnementales ou moléculaires. La taille des données ne cesse d'augmenter et il devient difficile de les analyser manuellement. De nouveaux outils pour automatiser le traitement et l'analyse sont nécessaires. Le Machine et le Deep Learning (ML/DL) sont les principaux outils disponibles pour l'automatisation dans ce domaine. Le ML comprend différents algorithmes de régression, clustering, classification ou réduction de la dimensionnalité. Ces algorithmes construisant un modèle afin de prédire à partir de données sont principalement utilisés pour l'analyse de big data. Les algorithmes DL sont un type spécifique d'algorithmes ML dans lesquels l'extraction des caractéristiques à analyser est automatisée. Ces outils sont ici appliqués aux domaines de la RMN et de la MS à travers différents projets de la thèse de doctorat.

Mots-clés : Spectrométrie de masse, résonance magnétique nucléaire, apprentissage automatique, Big Data, automatisation

RÉSUMÉ EN ANGLAIS

Mass Spectrometry (MS) and Nuclear Magnetic Resonance (NMR) are two techniques used in biology with many pharmaceutical, environmental or molecular applications. Datasets sizes are continuously increasing and become hard to manually analyze. New tools to automatize the big data processing and analysis are thus required. Machine Learning (ML) and Deep Learning (DL) are the main tools available to create automatization in that domain. ML includes different algorithms for regression, clustering, classification or dimensionality reduction. These algorithms, building a model function and predicting from data, are mainly used for the analysis of large datasets. DL algorithms are a specific kind of ML algorithms in which feature extraction is also automated. These tools are applied during this work into NMR and MS domains through different projects realized during this PhD Thesis.

Keywords: Mass Spectrometry, Nuclear Magnetic Resonance, Machine Learning, Big Data, automatization

igbmc institut de génétique et de biologie moléculaire et cellulaire

CASC4DE