Identification de déterminants génomiques impliqués dans la spécificité de fixation des facteurs de transcription

Raphaël Romero

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITE DE MONTPELLIER

En Biostatistiques ED I2S

École doctorale : Information, Structures, Systèmes

Unité de recherche IMAG & LIRMM

Identification de déterminants génomiques impliqués dans la spécificité de fixation des facteurs de transcription.

Présentée par Raphaël ROMERO

Le 14 Décembre 2021

Sous la direction de Jean-Michel MARIN,
Sophie LÈBRE,
Charles-Henri LECELLIER,
et Laurent BRÉHÉLIN

Devant le jury composé de

Marie-Laure MARTIN-MAGNIETTE, Directrice de Recherche INRAE, IPS2 et MIA Paris	Rapporteure
Étienne BIRMELÉ, Professeur Univ. Strasbourg, IRMA	Rapporteur
Anthony MATHELIER, Group Leader, NCMM, Oslo	Président du jury et Examinateur
Raphaël MOURAD, Maître de conférences Université Paul Sabatier, CBI	Examinateur
Jean-Michel MARIN, Professeur Univ. Montpellier, IMAG	Directeur de thèse
Sophie LÈBRE, Maître de conférences Univ. Paul-Valéry, IMAG	Co-encadrante
Charles-Henri LECELLIER, Directeur de Recherche CNRS, IGMM	Co-encadrant
Laurent BRÉHÉLIN, Chargé de Recherche CNRS, LIRMM	Co-encadrant

UNIVERSITÉ DE MONTPELLIER

Table des matières

Table des figures

Notations

Nous donnons ici les principales notations que nous utilisons dans cette thèse.

— Y : la variable à prédire,

— y_i : l'individu i de Y,

— X_j : une variable prédictive,

— $X = (X_1, \ldots, X_p)$: le vecteur des p variables prédictives,

— x_{ij} : l'individu i de la variable X_j,

— $||x||$: la norme de x,

— $|x|$: la valeur absolue de x,

— z : une séquence,

— Z : un ensemble de séquence,

— Z_i : une séquence dans Z,

— z_k : le k-ième symbole de z,

— Λ : l'ensemble des symboles possibles d'une séquence $\Lambda = \{A, C, G, T\}$,

— $|\Lambda|$: le cardinal de l'ensemble Λ.

Introduction

L'étude des mécanismes de régulation de l'expression des gènes est une des thématiques les plus anciennes en bioinformatique. Il est maintenant connu que ces mécanismes de régulation permettent d'assurer une grande diversité de types de cellulaires à partir d'un même ADN, ou encore qu'une dérégulation peut engendrer différentes pathologies, comme des cancers [1]. Cependant, le fonctionnement de ce système reste obscur et même en connaissant les acteurs de la régulation, il est difficile de décrire précisément ce processus. C'est pourquoi il est important de comprendre comment cette machinerie est orchestrée, d'identifier les acteurs impliqués dans cette régulation, et de quantifier leur activité.

Les régulations ont lieu à plusieurs niveaux, notamment au niveau transcriptionnel avec l'action des facteurs de transcription (TF). Ces protéines se lient à l'ADN au niveau de régions régulatrices plus ou moins éloignées du gène. Ils coopèrent entre eux, et viennent ainsi inhiber ou stimuler la transcription de leur gène cible, en se fixant sur des régions particulières du génome appelées sites de fixation. L'ensemble des sites de fixations possibles d'un TF donné est modélisé dans ce qui est communément appelé un motif de fixation. Grâce à un motif de fixation, il est possible d'identifier des milliers de sites de fixation potentiels sur le génome [2]. Cependant, l'analyse de données expérimentales (ChIP-seq) a pu montrer que seulement une infime fraction de ces sites sont réellement fixés par le facteur de transcription étudié [3, 4]. De plus, d'un type cellulaire à l'autre, les sites fixés diffèrent [3] afin d'assurer les fonctions propres à chaque cellule. Il est donc clair que le motif de fixation ne permet pas à lui seul d'expliquer la fixation d'un TF.

Les progrès techniques ont permis l'émergence d'énormément de données de séquençage et de méthodes pour expliquer le fonctionnement du vivant à l'échelle du génome. Cependant, comme Julia Zeitlinger l'a récemment écrit (en septembre 2020) dans un art-

cicle intitulé *Seven myths of how transcription factors read the cis-regulatory code* [5] : « Ironiquement, avec le développement de technologies génomiques et de méthodes de calcul de plus en plus puissantes au cours de la dernière décennie, les efforts pour déchiffrer le code cis-régulateur ont diminué au lieu d'augmenter. Plutôt que de se concentrer sur la relation entre la séquence et la régulation des gènes, les efforts de recherche sont de plus en plus portés sur les états de la chromatine, l'ARN et l'organisation 3D du noyau. Ainsi, les questions scientifiques ont évolué avec les nouvelles possibilités offertes par la technologie génomique et se sont détournées du problème fondamental du code cis-régulateur, qui a fini par être considéré soit comme résolu en principe, soit comme insoluble » (citation traduite depuis l'anglais). Dans cette thèse, au contraire, le code des séquences est au cœur de nos analyses. Nous ne cherchons pas directement son lien avec la régulation des gènes, mais étudions la fixation des facteurs de transcription qui est étroitement liée à cette régulation. Plus précisément, nous nous intéressons à la fixation des TF et aux spécificités de fixation dans des types cellulaires différents, au travers d'informations liées à la séquence. Pour cela, nous optimisons le motif de fixation et nous ajoutons différentes informations complémentaires portant sur l'environnement local des séquences. Ces informations sont liées aux fréquences de certains k-mers dans des régions particulières et à la position des facteurs de transcription coopérants avec le TF cible. Nous développons des méthodes interprétables d'apprentissage statistique utilisant ces différentes informations, afin d'identifier et quantifier celles qui semblent être les plus importantes pour expliquer les spécificités de fixation des TF dans différents types cellulaires.

Organisation du manuscrit

Ce manuscrit est séparé en deux parties, une partie « État de l'art » et une partie « Contributions ». La première partie se compose de trois chapitres. Nous présentons d'abord les mécanismes biologiques dans le premier chapitre, nous détaillons le fonctionnement de l'ADN, de la transcription et des facteurs de transcription ainsi que les méthodes usuelles d'analyses expérimentales associées. Dans le deuxième chapitre, nous présentons les méthodes d'apprentissage statistique qui seront utilisées par la suite. Le troisième chapitre présente, quant à lui, des méthodes de bio-informatique couramment utilisées pour modéliser et analyser la fixation des facteurs de transcription.

Dans la deuxième partie, nous voyons les différents travaux et les méthodes dévelop-

pées pour étudier les spécificités de fixation des facteurs de transcription dans différents types cellulaires humains. Dans le chapitre 4, nous décrivons un travail préliminaire publié en 2021, qui porte sur l'étude de facteurs de transcription d'une même famille, ciblant des sites de fixation très similaires. Ensuite, le chapitre 5 présente la problématique et les données que nous utilisons dans les approches détaillées aux chapitres suivants. Puis, dans le chapitre 6, nous décrivons le modèle développé pour étudier les spécificités de fixations propres au site de fixation. Le chapitre 7 détaille le modèle développé pour étudier l'influence des facteurs de transcription coopérants avec le TF cible dans les différents types cellulaires. Une fois que nous avons présenté les différentes informations considérées, nous proposons un modèle qui combine celles-ci, dans le chapitre 8. Enfin, nous concluons et discutons des différents résultats et des possibles perspectives que ce travail apporte dans le chapitre 9.

Première partie

État de l'art

Chapitre 1

Régulation de la transcription

1.1 ADN, gènes et transcription

Les organismes vivants sont composés de cellules biologiques à l'intérieur desquelles interagissent de nombreuses molécules leur permettant d'effectuer certaines tâches essentielles à leurs fonctions, leur survie et leur reproduction. Parmi ces molécules, on trouve l'ADN (Acide désoxyriboNucléique) qui est composé de deux brins complémentaires qui forme une structure en double hélice de nucléotides : Adénine, Cytosine, Guanine et Thymine notées A,C,G et T. Les nucléotides sont ainsi les unités de base de l'ADN reliées par des liaisons covalentes. La complémentarité est assurée par les appariements des bases A avec T et G avec C. Enfin, par convention, chaque brin d'ADN est orienté de son extrémité 5' phosphate vers l'extrémité 3' hydroxyle. Chez l'homme l'ADN compte 3,3 milliards de paires de bases, contenues dans quelques micromètres, il est donc replié sur lui même au sein de chaque cellule.

Le génome est l'ensemble du matériel génétique encrypté dans cet ADN qui conserve toute l'information utile au développement et au fonctionnement de la cellule. Chez les organismes eucaryotes le génome est contenu dans le noyau de la cellule. Chez l'homme, qui fait partie des eucaryotes, celui-ci est reparti sur un ensemble de 23 paires de chromosomes. Les chromosomes ont été décrits pour la première fois il y a presque 150 ans par Walther Flemming [6]. Dans chaque noyau de cellule l'ADN est compacté et enfermé par la chromatine, il est pourtant en interaction avec beaucoup de molécules et de protéines, dont les facteurs de transcription. La chromatine est une structure complexe qui subit des modifications chimiques permettant de moduler l'expression des gènes, sans modification

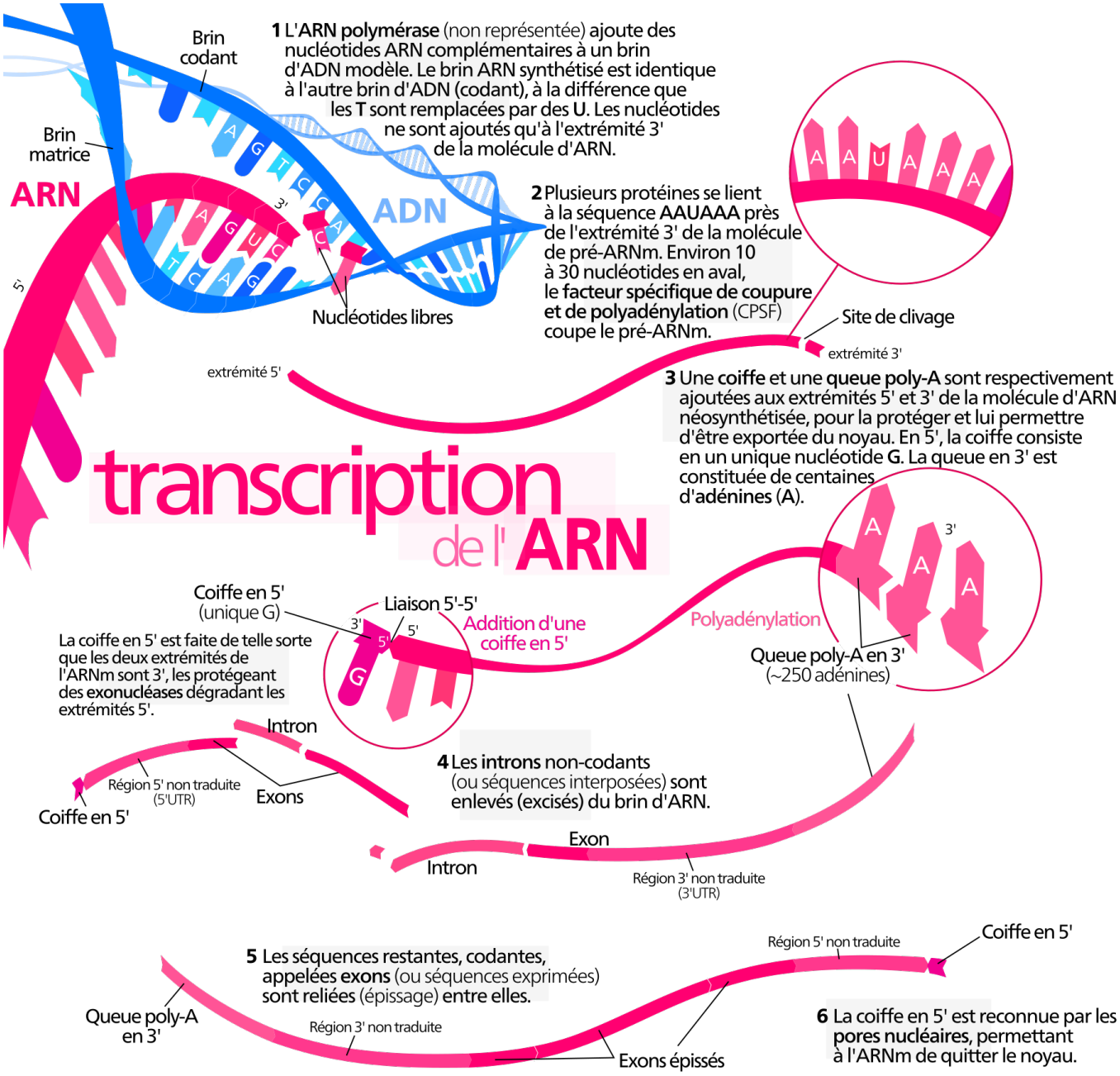

FIGURE 1.1 – *Schéma du fonctionnement de la transcription.*
Ce schéma décrit le fonctionnement de la transcription de l'ADN jusqu'à ARNm (ARN messager). Figure tirée de `https://commons.wikimedia.org/wiki/File:MRNA.svg`.

de la séquence ADN. L'unité de base d'organisation de la chromatine est le nucléosome, composé de huit protéines histones autour desquelles l'ADN est enroulé à la manière d'une bobine. Les nucléosomes sont directement impliqués dans la régulation de plusieurs processus liés au fonctionnement de l'ADN [7]. Le fonctionnement de la cellule est en effet régi à la fois par la séquence primaire de l'ADN (les gènes et les séquences régulatrices) et par des processus biochimiques qui conditionnent la lecture de cette séquence (mécanismes épigénétiques). Les nucléosomes sont par exemple impliqués dans la compaction de l'ADN et vont ouvrir ou fermer la chromatine régulant ainsi l'accès à l'ADN et la fixation des protéines régulatrices [7] (cf ci-après). Le développement de techniques microscopiques et moléculaires performantes permet aujourd'hui une étude poussée de l'organisation spatiale du génome dans le noyau des cellules [8].

Les gènes sont des portions de l'ADN qui vont être recopiées sous forme de molécules d'ARN (Acide RiboNucléique) qui sont éventuellement elles mêmes ensuite traduites en protéines (on parle alors d'ARN codants). La transcription de l'ADN en ARN est réalisée par des ARN polymérases. On en compte 3 principales chez les eucaryotes : ARN polymérases I, II et III [9]. Les ARN codants sont transcrits par l'ARN polymérase II. La transcription est contrôlée par la fixation de protéines sur la séquence d'ADN, appelées facteur de transcription. Au sein de la structure des gènes, on distingue deux types de séquences : les exons et les introns. Ces deux types de séquences sont retrouvés dans la première version de l'ARN transcrit (dit primaire) mais les introns seront ensuite excisés par un processus appelé épissage [10]. Seuls les exons seront contenus dans la séquence ARN mature. Par conséquent, dans le cas des ARN codants, seuls les exons sont traduits (les exons contiennent donc les séquences codantes). Notons cependant que les gènes ne codant pas de protéines (dit non codants), dont le nombre ne cesse de croître et dépasse aujourd'hui le nombre d'ARN codants, contiennent également des introns et leurs ARN sont également épissés. La taille des introns/exons est très variable et, si les exons codants des protéines ne représentent que 2% du génome, la somme des introns représente environ 50% du génome humain. La majeure partie du génome (98%) ne code donc pas de protéines et, il est aujourd'hui avéré que ces régions non codantes jouent des rôles clefs notamment dans les processus de régulations génomiques transcriptionnelles et/ou post-transcriptionnelles [11]. Bien que la séquence primaire d'ADN soit identique dans toutes les cellules d'un individu, la régulation des gènes varie d'un type cellulaire à l'autre afin d'assurer des fonctions différentes. Ce point est spécifiquement discuté en section 1.4.

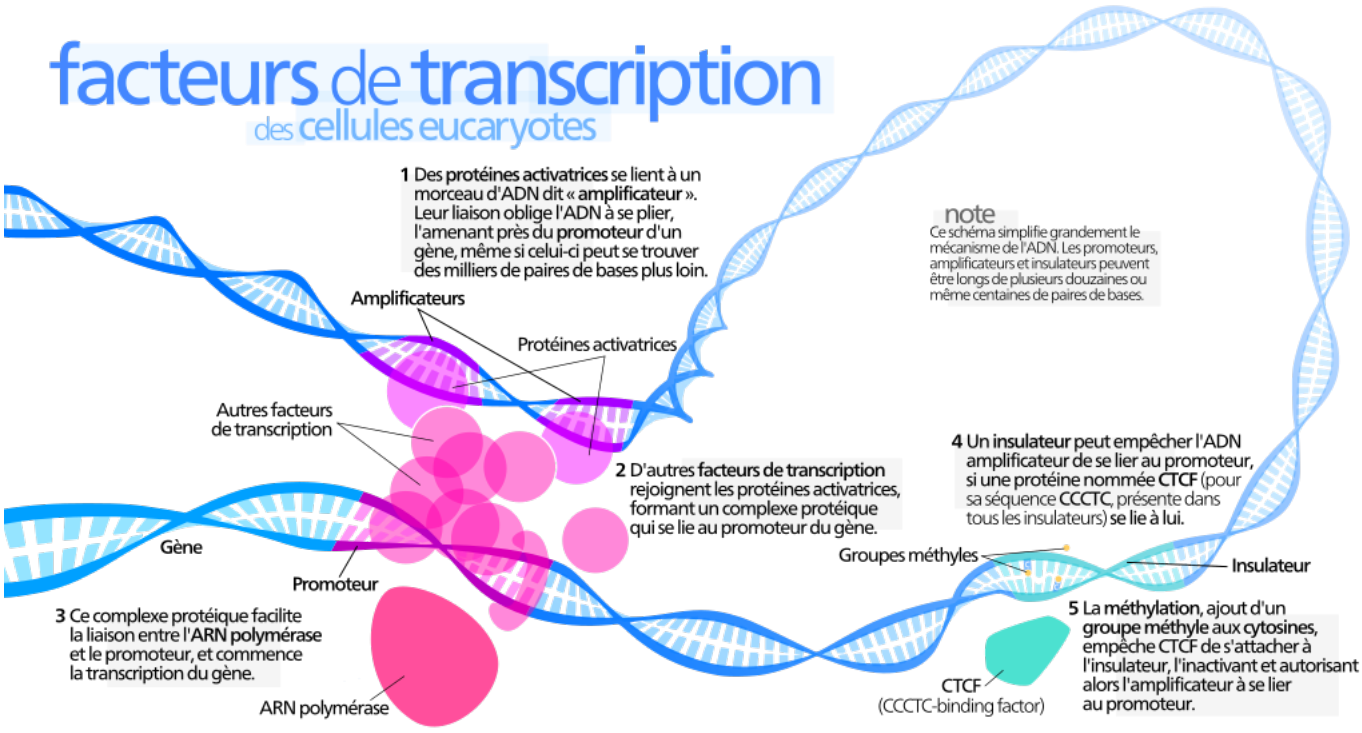

FIGURE 1.2 – *Schéma de la fixation des facteurs de transcriptions dans différentes régions de l'ADN dans les cellules eucaryotes.*
Ce schéma représente une molécule d'ADN et plus particulièrement la région génique, la région promotrice, les régions « enhancers » et « insulators » associées. Figure tirée de `https://commons.wikimedia.org/wiki/File:Transcription_Factors-fr.svg`.

1.2 Facteurs de transcription

Les facteurs de transcription sont des protéines nécessaires à l'initiation et à la régulation de la transcription des gènes [12]. Ces protéines se fixent directement sur le brin d'ADN sur des séquences appelées sites de fixation, mais peuvent aussi se fixer de manière indirecte en s'accrochant par exemple à un autre facteur de transcription qui peut être fixé directement ou non. Des facteurs de transcription vont « reconnaître » certains morceaux de séquences de taille K appelées k-mer. L'ensemble des k-mer reconnus par un facteur de transcription est appelé motif de fixation. Les motifs de fixation des facteurs de transcription sont généralement compris en 6 et 30 nucléotides, la modélisation des motifs est détaillée en section 3.1.1. Pour que la plupart des facteurs de transcription reconnaissant des motifs puissent accéder à leurs sites de fixation, l'ADN doit être décondensé (chromatine ouverte). Notons que d'autres mécanismes épigénétiques entrent en jeu

comme, par exemple, la méthylation de l'ADN qui peut inhiber ou activer la fixation des TF [13]. Il existe une catégorie de TF qui sont capables de se fixer sur de l'ADN condensé (chromatine fermée), ceux-ci sont appelés facteurs pionniers. Le modèle actuel propose que la fixation de ces facteurs pionniers permette le recrutement d'enzymes de modification d'histones [14] pouvant influer sur l'état de la chromatine et ainsi recruter d'autres facteurs de transcription. Les facteurs de transcription influencent donc l'expression des gènes dans les cellules, en régulant la transcription. Ils peuvent se fixer à proximité du gène qu'ils régulent, dans la région promotrice aussi bien qu'à des centaines de paires de bases du promoteur, dans les régions amplificatrices ou en anglais « enhancers ». Notons qu'à l'inverse, les TF peuvent également se fixer sur des séquences inhibitrices dîtes « silencers » à ce jour moins bien caractérisées que les enhancers. Les TF peuvent activer ou inhiber la transcription en respectivement, favorisant ou bloquant le recrutement de la machinerie de transcription [15]. Notons qu'un même TF peut être à la fois inhibiteur et activateur. Enfin, les facteurs de transcription peuvent être regroupés par famille qui comportent des structures similaires [16].

La Figure 1.3 est un schéma d'une molécule d'ADN, elle montre la fixation des TF dans la région génique, la région promotrice, les régions enhancers et les régions insulateurs associées.

1.2.1 Coopération des facteurs de transcription

Comme nous l'avons décrit les facteurs de transcription se fixent sur l'ADN et régulent la transcription. La transcription n'est pas régulée par un seul facteur de transcription, c'est une combinaison de TF qui recrute l'ARN polymérase et régule l'expression des gènes[17]. Plusieurs mécanismes peuvent conduire à cette coopération entre TF [18, 19]. La forme la plus simple de coopération est l'intéraction protéine-protéine entre TF, avant même la fixation sur l'ADN. En effet, plusieurs études ont montré que les interactions entre TF peuvent modifier leurs spécificités de fixation. Les différences entre les sites de fixation des facteurs de transcription pris individuellement et le motif de fixation des paires de TF ont été documentées pour de nombreuses paires de TF dans des études in vitro à grande échelle [20]. Ces effets sur le motif reconnu peuvent être obtenus par ces interactions protéine-protéine changeant directement la conformation de la protéine ou par des changements de forme de l'ADN induits par la liaison d'un facteur de transcription affectent la liaison d'autres TF [21]. La fixation de facteurs pionniers qui recrutent d'autres

facteurs de transcription est aussi une forme de coopération, en effet certains facteurs ne peuvent se fixer que si un facteur pionnier particulier s'est fixé pour agir sur l'ouverture de la chromatine [22]. Enfin la fixation d'un facteur de transcription peut altérer localement la forme de l'ADN pouvant ainsi augmenter l'affinité de fixation d'autres TF [18]. À l'inverse, les facteurs de transcription d'une même famille ont souvent des affinités de fixation sur des motifs très similaires, ils vont donc pouvoir rentrer en compétition sur les sites contenus dans les deux motifs [23]. Que les TF rentrent en compétition entre eux, ou participe au recrutement les uns des autres, il est clair qu'ils interagissent les uns par rapport aux autres et agissent ensemble sur la transcription. La coopération des facteurs de transcription est donc une information importante à étudier pour mieux comprendre les mécanismes de la transcription.

Plusieurs auteurs ont déjà étudié l'information de coopération des TF d'un point de vue statistique. Certains travaux ont par exemple étudié les paires de TF co-occurrents c'est à dire, les sites de fixation qui sont plus proches qu'attendus par hasard [24, 25, 26, 27]. TFcoop [28], montre que combiner l'information de plusieurs motifs de fixation permet de mieux prédire la fixation d'un TF donné (voir section 3.3.6).

1.2.2 Environnement proche du site de fixation

L'environnement nucléotidique proche d'un site de fixation potentiel peut aussi avoir un impact sur la fixation du facteur de transcription. Alors, deux facteurs de transcription peuvent avoir les mêmes affinités apparentes pour leur motif de fixation, mais différer dans leurs préférences pour l'ADN flanquant, ce qui leur permet de reconnaître différents sites génomiques. Par exemple, les deux facteurs bHLH, Cbf1 et Tye7 de la levure reconnaissent tous deux la même séquence (CACGTG), mais ont des préférences distinctes pour la forme de l'ADN des séquences flanquantes à ce motif, ce qui leur permet d'occuper des sites de fixation distincts [29]. Dans l'article de Dror *et al.*, 2015 « *A widespread role of the motif environment in transcription factor binding across diverse protein families* »[30], les auteurs effectuent une analyse de la séquence et de la forme de l'ADN entourant les sites de fixation de 239 TF extraits *in vivo* (HT-SELEX voir section 1.3.2) et les sites de fixation de 56 TF extraits de données de ChIP-seq *in vivo* (voir section 1.3.1). La comparaison du contenu nucléotidique dans les régions entourant les sites fixés par les TF par rapport aux régions non-fixées contenant les mêmes motifs de fixation a révélé des différences signifi-

catives qui s'étendent bien au-delà du site de liaison central. Les auteurs ont pu mettre en évidence des formes de l'ADN particulières entourant les sites de fixation. De plus, il ont pu trouver que les TF appartenant à une même famille ont présenté des caractéristiques similaires sur les régions entourant le site de fixation. Ils suggèrent que « ces caractéristiques uniques aident à guider les TF vers leurs sites de fixation correspondants ». Le contenu en GC de la séquence d'ADN flanquant peut également distinguer les sites *in vivo* liés à un TF donné, des occurrences de motifs non fonctionnels [31, 32]. La teneur en GC est associée aux caractéristiques de la forme de l'ADN et à la flexibilité de l'ADN, ce qui laisse penser que des mécanismes de reconnaissance de la forme peuvent expliquer ces préférences. Il est donc là aussi important de considérer l'environnement proche du site de fixation dans l'étude des interactions TF-ADN, ce que nous ferons en section 6.6.

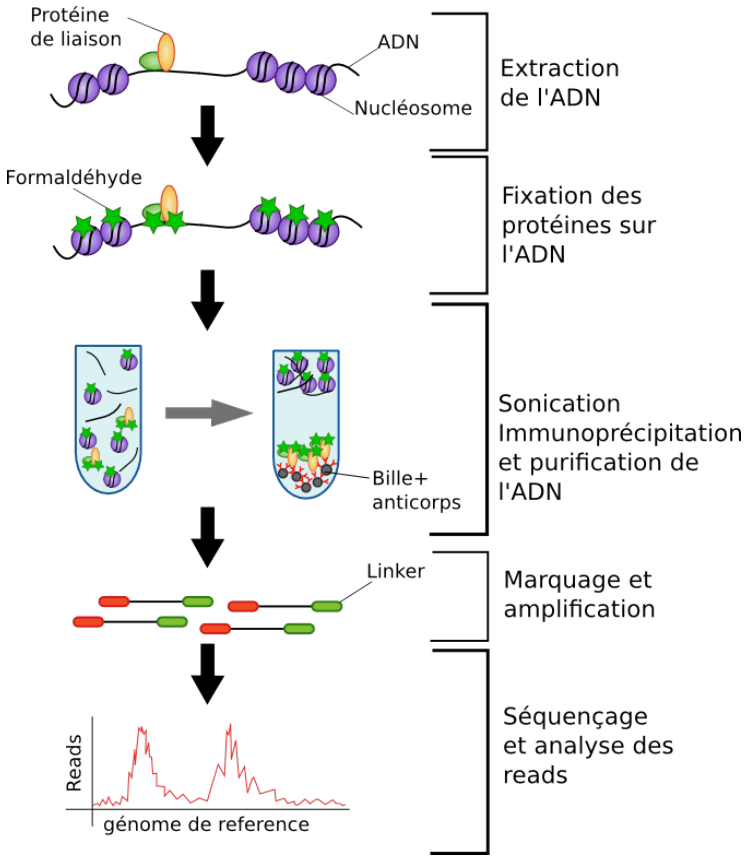

FIGURE 1.3 – *Schéma du fonctionnement d'une expérience de ChIP-seq.* Figure tirée de `https://bioinfo-fr.net/`.

1.3 Méthodes d'études de la fixation

1.3.1 ChIP-seq

Le ChIP-seq (Chromatin immunoprecipitation followed by sequencing) permet de déterminer l'ensemble des sites de fixation de protéines sur l'ensemble du génome. ChIP-seq permet donc de déterminer les sites de fixation des TF ou de localiser les nucléosomes/histones modifiés par certaines marques épigénétiques. Une expérience de ChIP-seq consiste à (1) figer les interactions ADN/protéines (au moyen du formaldéhyde par exemple), (ii) fragmenter l'ADN (par exemple par sonication), (iii) immunoprécipiter la protéine d'intérêt et (iv) purifier et séquencer les fragments d'ADN co-purifiés.

Une fois les fragments d'ADN séquencés, on procède à un alignement de ces fragments avec un génome de référence. Cela va permettre de retrouver où s'est fixé la protéine

d'intérêt sur le génome. En procédant de la sorte avec beaucoup de fragments d'ADN, on va pouvoir aligner quelques-uns de ces fragments aux mêmes endroits du génome. On peut donc étudier la distribution de ces fragments d'ADN sur le génome, et on peut ainsi observer des pics aux endroits où la protéine est vraisemblablement fixée. Ces pics sont appelés pics de ChIP-seq. Dans les bases de données de ChIP-seq, ce sont les positions de ces pics qui sont renseignées. Pour l'étude des préférences de fixation *in vivo* le ChIP-seq est la technique la plus répandue.

Les pics de ChIP-seq ne sont cependant pas toujours précis, plusieurs algorithmes optimisent la position des pics pour la découverte de sites de fixation de facteurs de transcription, comme MACS [33] ou WACS [34] par exemple. Il est aussi possible de capturer une fixation indirecte, c'est à dire que le TF n'est pas fixé sur le brin d'ADN, il peut par exemple s'être aggloméré à un autre TF. Dans ces cas il est donc possible d'observer des pics de ChIP-seq sans détecter de motifs de fixation pour le TF ciblé proche des pics. C'est notamment ce cas de figure que la base de données Unibind [35] écarte en plus de montrer l'imprécision des pics. La base de données Unibind est détaillée en section 3.4.2. Enfin, la technique ChIP-seq dépend de l'abondance de la protéine ciblée et de la spécificité de l'anticorps choisi, ce qui peut donc ajouter des biais supplémentaires.

1.3.2 Méthodes *in vitro*

La méthode SELEX (*Systematic Evolution of Ligands by EXponential enrichment*) est utilisée *in vitro*. Pour cela, les protéines d'intérêt sont incubées en présence d'un grand nombre d'oligonucléotides générés aléatoirement, elles viennent ainsi se fixer sur les motifs qu'elles reconnaissent. On extrait ensuite les complexes par immunoprécipitation et les séquences fixées par les protéines sont amplifiées. HT-SELEX (*High-Throughput Systematic Evolution of Ligands by EXponential enrichment*) est une extension haut débit de SELEX.

Une autre technique *in vitro* répandue est la PBM (*Protein Binding Microarray*) [36]. Dans cette technique une protéine d'intérêt est mise en contact avec une puce sur laquelle de nombreux oligonucléotides double brins sont fixés. Les fixations de facteur de transcription sont identifiées par immuno-détection en utilisant là encore un anticorps spécifique couplé à un fluorophore.

1.4 Spécificités de fixation

Comme nous l'avons déjà évoqué la régulation des gènes diffère d'une cellule à l'autre afin d'assurer les fonctions propres au tissu pour lequel elle a été spécialisée. La fixation des facteurs de transcription diffère alors entre les types cellulaires. Ces spécificités apparaissent de différentes façons et peuvent être observées sur des données de ChIP-seq.

En effet, la position des pics d'un facteur de transcription n'est pas identique dans des expériences de ChIP-seq issues de différents types cellulaires [3]. De même, elle diffère en fonction des traitements utilisés dans l'expérimentation. Ces différences observées entre types cellulaires sur les données de ChIP-seq peuvent en partie s'expliquer avec la fixation indirecte des TF (interactions protéine-protéine) qui peut survenir alors que le motif de fixation est absent. Ce n'est pas l'unique explication, on peut en effet observer des spécificités dans les fixations directes des facteurs de transcription [37]. Plusieurs effets peuvent expliquer ces spécificités de fixation, notamment à travers l'ouverture de la chromatine qui influe donc sur les positions des sites de fixation des facteurs de transcription [38]. Cependant l'ouverture de la chromatine n'est pas uniquement responsable, il a été montré que les facteurs de transcription influent sur les positions fixées par un autre TF dans les types cellulaires. En effet, le changement de partenaire de fixation entraîne des différences importantes dans les positions génomiques fixées par un facteur de transcription, entrainant une spécificité cellulaire [39]. C'est pourquoi l'étude de la fixation des cofacteurs semble importante pour étudier la spécificité de fixation entre types cellulaires, comme cela sera fait dans le chapitre 7.

Chapitre 2

Méthodes d'apprentissage

2.1 Apprentissage statistique

2.1.1 Principe

L'apprentissage statistique, ou apprentissage automatique ou encore apprentissage machine (de l'anglais « machine learning ») se base sur des approches mathématiques, statistiques et informatiques afin de créer différentes méthodes pour « apprendre » à résoudre certaines tâches spécifiques. On regroupe sous l'appellation apprentissage statistique un ensemble de méthodes adaptées à différents problèmes. Le premier enjeu quand on utilise une telle méthode est donc de bien étudier la nature du problème à résoudre ainsi que les données disponibles ou compilables. Une fois cela réalisé, on dispose de données et on souhaite modéliser un phénomène à l'aide de p variables explicatives notées $(X_j)_{j=1\dots p}$. On appliquera la ou les méthodes les plus appropriées en fonction des données et du problème. C'est à dire en fonction du type des variables explicatives (qualitatives, quantitatives, binaires etc...), du type de variable à expliquer et d'autres objectifs propres aux besoins du domaine dans lequel elle est utilisée (interprétation du modèle, sélection de variables etc...). De plus, on distingue les méthodes supervisées où les exemples sont annotés c'est à dire qu'on connaît la variable à expliquer, des non-supervisées où on « recherche » la variable à expliquer. De même qu'on distingue souvent les méthodes d'apprentissage profond qui modélisent avec un haut niveau d'abstraction, des méthodes avec un niveau d'abstraction plus bas. Dans cette thèse, nous considérons surtout des méthodes d'apprentissage supervisé (voir sections 2.2 et 2.3.1), dans lesquelles la variable à modéliser Y est utilisée pour apprendre le modèle, mais nous évoquerons tout de même

quelques méthodes non-supervisées en section 2.3.3.1 et 2.3.3.2. Plusieurs paramètres sont associés à l'apprentissage de toute méthode, le nombre de ces paramètres dépend de la méthode. Les valeurs de ces paramètres sont très importantes pour garantir une bonne adéquation du modèle aux données. L'apprentissage statistique est aujourd'hui très utilisé, il y a en effet à notre disposition des quantités de données qui ne sont pas humainement exploitables. Avec l'essor des méthodes de séquençage modernes la génomique jouit elle aussi d'une quantité de données considérable et l'apprentissage statistique peut permettre de mieux comprendre le fonctionnement biologique des séquences. Il est, par exemple, important d'identifier l'information contenue dans les séquences et de valider des informations biologiques. L'apprentissage statistique est donc un outil majeur dans ce domaine. Dans le cadre d'une méthode supervisée nous disposons de n observations de Y et des variables explicatives. Nous voulons ainsi modéliser Y en fonction des p variables explicatives $(X_j)_{j\in\{1,\dots,n\}}$, on cherche donc une fonction qui met en relation Y et $X = (X_{ij})_{i\in\{1,\dots,n\};j\in\{1,\dots,p\}}$. Soit β l'ensemble des paramètres et $\epsilon \sim \mathcal{N}(0,\sigma)$, on cherche $f(.,\beta)$ une fonction mesurable telle que

$$Y = f(X,\beta) + \epsilon. \tag{2.1}$$

Le choix de la relation que définit f pour modéliser Y par rapport aux $(X_j))_{j\in\{1,\dots,n\}}$ est donc primordiale. Cette relation peut être linéaire, polynomiale, exponentielle etc... Elle peut être choisie en avance en fonction des connaissances *a priori* du problème étudié, ou être choisie en essayant successivement plusieurs modélisations possibles afin d'en déterminer la meilleure. Cette fonction est liée à des paramètres qui devront être ajustés.

Une fois le type de relation choisie, il faudra estimer les paramètres liés à la fonction f afin que celle-ci ajuste au mieux les X_j à Y par rapport à X en donnant par exemple différents poids à chaque variable X_j. Pour optimiser ces paramètres on cherche le jeu de paramètres β qui optimise un certain critère. Ce critère peut être l'erreur commise par le modèle. Celle-ci est modélisée par une fonction de risque $R(y_i,\hat{y}_i)$ où y_i est la valeur observée de Y_i et $\hat{y}_i$ est l'estimation de y_i faite par le modèle avec (2.1). Soient y et $\hat{y}$ les vecteurs des observations des y_i et $\hat{y}_i$ respectivement. On cherche donc β tel que $R(y,\hat{y})$ le risque global sur l'ensemble des n observations soit minimal. Par exemple, si $Y \in \mathbb{R}$ il est courant d'utiliser la fonction de risque quadratique, appelée erreur quadratique moyenne

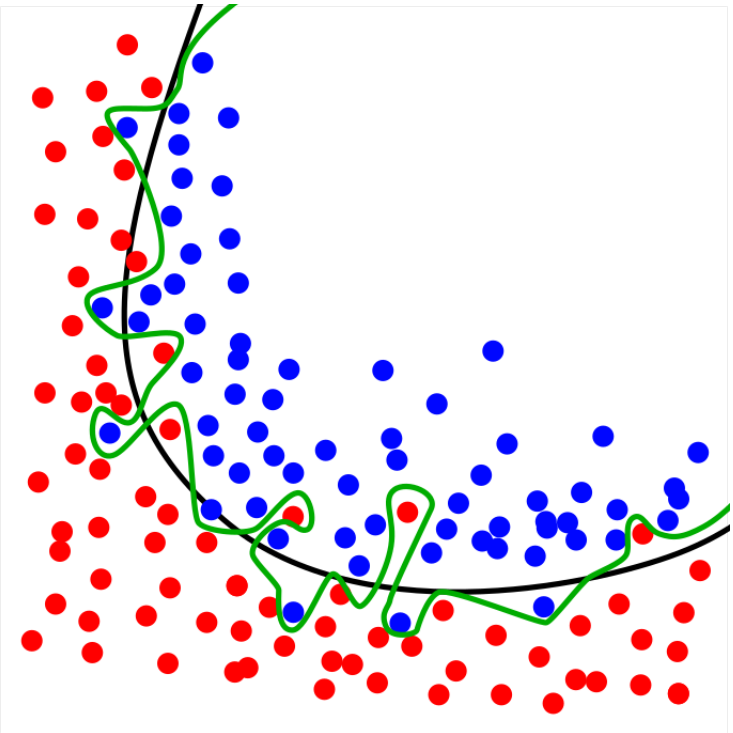

FIGURE 2.1 – *Illustration du surapprentissage.*
Exemple de surapprentissage sur un problème de partitionnement. La courbe verte correspond à un modèle surappris tandis que la courbe noire représente un modèle qui généralise mieux le partitionnement des données. Figure tirée de `https://commons.wikimedia.org/wiki/File:Overfitting.svg`.

ou MSE (de l'anglais « mean squared error ») comme défini dans l'équation (2.2).

$$R(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_j)^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_{ij}, \beta))^2 \tag{2.2}$$

L'estimation des β est donc le moyen d'optimiser le modèle choisi (défini par la fonction $f(., \beta)$). Cependant celle ci peut engendrer des problèmes tels que le surapprentissage.

2.1.2 Surapprentissage et validation

Le surapprentissage ou sur-ajustement (en anglais « overfitting ») est un problème qui survient dans les problématiques d'apprentissage statistique. En effet, lorsque l'on estime le vecteur des coefficients β de la fonction $f(., \beta)$ cette estimation peut être trop ajustée

aux données utilisées afin d'entraîner le modèle. À noter que ce sur-ajustement peut être important quand le nombre de variables p est élevé par rapport au nombre d'observations n. Ainsi, dans le cas du surapprentissage, le modèle correspond très fidèlement aux données mais ne représente qu'un échantillon des données existantes. De fait, ce modèle ne peut pas correspondre à des données supplémentaires du même type. On ne peut alors pas s'assurer que le modèle ait capturé une information traduisant un fait général et que l'on peut s'attendre à observer sur un nouveau jeu de données.

Pour pallier ce problème, on estime les coefficients sur un sous-ensemble des données appelé ensemble d'apprentissage et on teste les performances sur un sous-ensemble disjoint appelé ensemble de test. La majorité des données étant souvent utilisée comme ensemble d'apprentissage, l'équilibre entre la taille des deux ensembles varie avec le volume de données disponibles et les besoins de la méthode utilisée. Cette procédure de validation est donc essentielle pour prouver le bon fonctionnement de la méthode. Si on rencontre un problème de surapprentissage, les performances de la méthode en terme de prédiction seront donc bien plus mauvaises sur l'ensemble de test que sur l'ensemble d'apprentissage. Alors que des valeurs différentes des paramètres β auraient pu permettre d'obtenir des performances moindres sur l'ensemble d'apprentissage mais meilleures sur l'ensemble de test.

Une autre procédure de validation appelée validation croisée reprend cette idée de façon itérative. Dans une validation croisée on choisit N le nombre de validations. On sépare le jeu de données en N parties égales, puis chacune de ces parties seront utilisées comme ensemble de test l'une après l'autre, les $N-1$ autres parties seront successivement utilisées en tant qu'ensemble d'apprentissage. Cela permet d'utiliser toutes les données disponibles dans l'apprentissage de la méthode et permet de tester les performances sur toutes les données disponibles, en plus d'éviter les problématiques de surapprentissage. Attention, dans ce cas, l'évaluation des performances ne se fait pas sur un ensemble indépendant.

Il est possible de combiner ces deux approches de validation en choisissant un découpage apprentissage/test et de procéder à une validation croisée dans l'estimation des coefficients sur l'ensemble d'apprentissage. Cela permet d'utiliser l'ensemble des données d'apprentissage, tout en sélectionnant un modèle ayant de bonnes performances de pré-

diction sur un ensemble indépendant de données de test.

Sans considérer ces problématiques de surapprentissage, on pourrait s'attendre à ce que l'erreur commise par le modèle décroisse systématiquement quand on augmente le nombre de paramètres p du modèle. Dans les faits, ceci n'est attendu que sur l'ensemble d'apprentissage. En effet, sur l'ensemble de test, l'erreur du modèle va augmenter à nouveau quand le nombre de paramètre p atteint un certain seuil. Il est donc possible de trouver un p qui minimise l'erreur commise par le modèle, sur l'ensemble de test. Ceci constitue la vision la plus courante de la relation entre le nombre de paramètres et l'erreur commise par le modèle. Cependant, Belkin *et al.* montrent dans un article intitulé « *Reconciling modern machine-learning practice and the classical bias–variance trade-of* » et publié en 2019, que quand $n << p$, l'erreur sur l'ensemble de test va décroître quand p augmente. Ce phénomène est appelé « double descente » [40]. De plus, ils montrent que l'erreur maximale sur l'ensemble de test est atteinte quand p est proche du nombre d'observations n.

2.2 Régression linéaire multiple

On considère $Y = (y_i)_{i=1\dots n}$ une variable continue à prédire, et $X = (X_{i1}, \dots, X_{ip},)_{i\in\{1,\dots,n\}}$ des variables continues prédictives où pour $i \in [1, n]$, $X_{ij} = (x_{ij})_{j\in\{1,\dots,p\}}$. Le modèle s'écrit alors, $\forall i \in \{1, \dots, n\}$,

$$y_j = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_i x_{ij} + \cdots + \beta_p x_{pj} + \epsilon, \tag{2.3}$$

où $\epsilon \sim \mathcal{N}(0, \sigma)$.

Le modèle peut être écrit de façon matricielle, cette notation est plus simple à lire et à manipuler. Si on note de la manière suivante

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, X = \begin{pmatrix} 1 & x_{11} & x_{12} & \dots & x_{1p} \\ 1 & x_{11} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}, \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \epsilon = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$

où $\forall i \in \{1, \dots, n\}$, $\epsilon_i \sim \mathcal{N}(0, \sigma)$. Alors on peut écrire le modèle de régression linéaire

multiple avec l'équation,

$$Y = X\beta + \epsilon. \tag{2.4}$$

On cherche à minimiser l'erreur quadratique moyenne pour estimer les paramètres β. On cherche donc $\hat{\beta}$ qui minimise l'expression suivante :

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{n} \sum_{i=1}^{n} (y_j - (\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}))^2 \tag{2.5}$$

Minimiser l'erreur quadratique moyenne c'est chercher l'estimateur des moindres carrés, qui est aussi l'estimateur du maximum de vraisemblance dans le cas de la régression linéaire multiple. La vraisemblance d'un modèle décrit la plausibilité des valeurs des paramètres du modèle, sachant les observations des variables utilisées dans celui-ci. On cherche donc à maximiser la vraisemblance d'un modèle pour que celui-ci ait la meilleure adéquation possible entre Y et les $(X_j)_{j\in\{1,\dots,p\}}$.

L'application d'une régression linéaire multiple requière les hypothèses suivantes :
(i) $\forall i \in \{1, \dots, n\}$, les ϵ_i sont indépendants et identiquement distribués, $\epsilon_i \sim \mathcal{N}(0, \sigma)$.
(ii) Homoscédasticité : $\forall i \in \{1, \dots, n\}$, les ϵ_i ont la même variance σ^2
(iii) Pas (ou peu) de multicolinéarité entre les $(X_i)_{i\in\{1,\dots,p\}}$
(iv) $n >> p$
De plus, si $X^T X$ est inversible, il existe un unique $\hat{\beta}$ donné par l'expression suivante :

$$\hat{\beta} = (X^T X)^{-1} X^T Y \tag{2.6}$$

Dans un objectif explicatif, il n'est pas seulement important d'obtenir de bonnes performances avec un modèle linéaire. Il est aussi important d'expliquer ces performances en analysant le pouvoir prédictif des différentes variables utilisées par le modèle. Cependant si le nombre de variables p est grand, l'interprétation de ces variables est plus difficile, il est donc préférable de sélectionner un sous-ensemble de variables d'intérêt. Dans la littérature il existe différentes façons de faire, tels que les critères d'informations (AIC, BIC) ou les modèles linéaires pénalisés (voir section 2.4).

2.3 Classification

Dans une problématique de classification on cherche à affecter des exemples à plusieurs classes. Les modèles développés pour la classification sont appelés des classifieurs. Si on cherche à modéliser une variable Y non-quantitative, qu'elle soit binaire ou qualitative on se place donc dans un problème de classification supervisé. Par exemple, on peut citer Random Forest[41] et Support Vector Machine (SVM)[42] en classifieurs supervisés. Alors que l'algorithme K-means[43] est un classifieur non-supervisé.

2.3.1 Régression logistique

La régression logistique[44], est une méthode supervisée permettant de résoudre des problèmes de classification binaire. Contrairement à un cas de régression classique où la variable à expliquer est continue, nous avons ici besoin de mettre en relation une variable binaire Y avec des variables binaires ou continues. On appellera les éléments $Y = 1$ « positifs » ou appartenant à la classe positive, $Y = 0$ « négatifs » ou appartenant à la classe négative.

On considère Y la variable à expliquer et $X = (X_1, X_2, \ldots, X_p)$ les variables explicatives. La variable Y ne peut prendre ici que deux modalités $Y \in \{0, 1\}$. Les variables $(X_j)_{j\in\{1,\ldots,p\}}$ sont continues ($X_j \in \mathbb{R}$) ou binaires ($X_j \in \{0, 1\}$). On a n réalisations de Y et des X_j aussi appelées individus ou exemples. On note y_i et x_{ij} pour $i = 1 \ldots n$ et $j = 1 \ldots p$ les réalisations de Y et des X_j. On note $\mathbb{P}(Y = 1)$ la probabilité *a priori* que Y prenne la valeur 1 (loi de Bernoulli) et $\mathbb{P}(Y = 1|X)$ la probabilité a posteriori que Y prenne la valeur 1 en sachant la valeur de X. On définit la fonction $logit$:$]0; 1[\to \mathbb{R}$ telle que $logit(P) = ln(\frac{P}{1-P})$ où $P \in]0; 1[$ est une probabilité. Alors, on écrit un modèle de régression logistique de la manière suivante :

$$logit(\mathbb{P}(Y = 1|X)) = ln\frac{\mathbb{P}(Y = 1|X)}{1 - \mathbb{P}(Y = 1|X)} = f(X, \beta) \tag{2.7}$$

où f est une fonction mesurable et β représente les paramètres de f.

De cette manière on peut mettre en relation une variable binaire avec des variables continues ou binaires. On appelle ceci une régression logistique car la loi de probabilité est modélisée à partir d'une loi logistique dont la fonction répartition est la fonction appelée

sigmoïde. La fonction sigmoïde $S(x) = \frac{1}{1+e^{-x}}$ est l'inverse de la fonction *logit*, on peut donc écrire grâce à (3) : $\mathbb{P}(Y = 1|X) = S(f(X, \beta)) = \frac{1}{1+e^{-f(X,\beta)}}$. Cette écriture, en plus d'expliquer l'origine du terme, permet aussi de montrer plus clairement que l'on cherche ici à estimer la valeur de la probabilité $\mathbb{P}(Y = 1|X)$ afin de prédire $Y = 1$ ou $Y = 0$. On entraîne donc le modèle sur la valeur de la probabilité $\mathbb{P}(Y = 1|X)$ plutôt que directement sur la valeur de Y comme cela peut être fait quand Y est continu. Enfin il est possible à partir d'un seuil défini de réaliser une prédiction $\hat{Y}$.

Une fois cette prédiction acquise, $\forall j \in \{0, \ldots, n\}$, nous retrouvons quatre cas possibles :

— $\hat{y}_j = 1$ et $y_j = 1$ vrai positif (TP)

— $\hat{y}_j = 1$ et $y_j = 0$ faux positif (FP)

— $\hat{y}_j = 0$ et $y_j = 1$ faux négatif (FN)

— $\hat{y}_j = 0$ et $y_j = 0$ vrai négatif (TN)

Un bon classifieur maximisera donc le nombre de vrais positifs et de vrais négatifs, baissant ainsi le nombre de faux positifs et faux négatifs.

2.3.2 Performances d'une classification

Évaluer les performances d'une classification diffère d'un modèle de régression où Y est continu. Dans le cas où Y est continu, l'erreur quadratique moyenne ou le coefficient de corrélation entre le $\hat{Y}$ prédit par le modèle et le véritable Y peuvent être utilisés pour évaluer les performances d'un modèle et comparer les performances des modèles entre eux. Dans le cas d'une régression logistique cela n'est pas possible avec les mêmes méthodes. Il existe différentes façons d'étudier ces performances, la courbe ROC (« Receiver Operating Characteristic ») et la courbe PR (« Precision Recall ») en sont des exemples.

La courbe ROC (voir Figure 2.2) est une courbe de mesure des performances d'un classifieur binaire. Pour tracer une courbe ROC, on fait varier le seuil utilisé pour réaliser une prédiction, à chaque seuil prédit on a un vecteur $\hat{Y}$ pour les n individus/exemples et on place le point d'abscisse taux de faux positifs FPR (« False Positive Rate ») et d'ordonnée taux de vrais positifs TPR (« True Positive Rate »).

$$TPR = \frac{TP}{TP + FN} \text{ et } FPR = \frac{FP}{TN + FP} \tag{2.8}$$

où TP, FN, FP et TN sont les nombres de vrais positifs, faux négatifs, faux positifs et vrais négatifs respectivement, obtenus pour chaque seuil. TPR et FPR sont aussi appelé « spécificité » et « sensibilité ».

On cherchera donc à avoir $TPR >> FPR$, un classifieur « parfait » obtiendra donc $TPR = 1$ et $FPR = 0$. Notons que obtenir un FPR maximal (égal à 1) et un TPR nul revient à avoir un classifieur qui se « trompe » tout le temps, inverser les classes nous donne alors un classifieur parfait. Dans la pratique dès qu'un modèle est entraîné à prédire correctement, on aura souvent $TPR > FPR$ si on dispose de suffisamment d'exemples. Un seuil qui ne prédit que des 0 donnera $TPR = FPR = 0$ car il n'y a aucun positif prédit donc $TP = FP = 0$. De même, un seuil qui ne prédit que des 1 entraînera $TPR = FPR = 1$ car $FN = TN = 0$. Tout classifieur où $FPR = TPR$ pour toutes les valeurs de seuil est le moins bon possible, c'est un classifieur aléatoire, sa courbe ROC est la courbe bissectrice $x = y$. On cherche donc à obtenir un modèle de classification dont la courbe ROC « s'écarte » de la bissectrice avec $TPR > FPR$, c'est à dire que la courbe ROC est au dessus de la bissectrice. Plus celle-ci est au dessus, plus le modèle est un bon classifieur. L'avantage de la courbe ROC est qu'elle ne dépend pas de la distribution des classes positives et négatives, elle permet de mesurer les performances d'un modèle même si les classes sont très déséquilibrées.

Il est courant d'étudier l'aire sous la courbe ROC, l'AUC (Area Under Curve) ; ou ici AUROC (Area Under ROC), ainsi, un classifeur aléatoire obtient une AUROC égale $= 0,50$ alors qu'un classifieur parfait a une AUROC égale à 1 (ou 0 de manière équivalente). De cette manière on peut résumer la capacité de prédiction d'un classifieur en une mesure et ainsi comparer plus facilement des modèles. La Figure 2.2 montre la construction de courbes ROC en faisant varier le seuil successivement, avec un classifieur, un classifieur aléatoire et un classifieur parfait.

2.3.3 Classification non-supervisée

2.3.3.1 Regroupement hiérarchique

Le regroupement hiérarchique (Hierarchical Clustering) est une méthode de classification non-supervisée. Étant donné n individus x_i on cherche à affecter x_i dans un certain

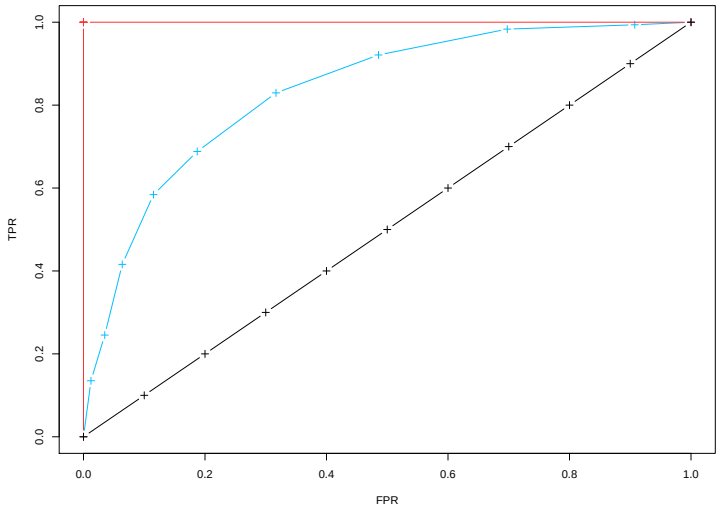

FIGURE 2.2 – *Construction d'une courbe ROC.*
Chaque point représente le TPR en fonction du FPR obtenus à chaque seuil. La courbe noire est associée à un classifieur aléatoire (AUROC=0,5), la courbe bleue est un exemple de classifieur entraîné (AUROC=0.835) et la courbe rouge représente un classifieur parfait (AUROC=1).

nombre de classe. Ces classes (clusters) sont agencées de façon hiérarchique, c'est à dire qu'une classe peut contenir plusieurs autres classes. Cette méthode a été originalement développée pour regrouper des espèces biologiques selon différents critères.

A l'état initial, on considère que chaque individu forme une classe, à chaque étape on regroupe deux classes, jusqu'à ce que tous les individus forment une seule classe. Pour regrouper deux classes a et b on utilise une mesure de dissimilarité $d(a, b)$. Cette mesure peut être par exemple une distance entre classes. Les classes ayant la dissimilarité la plus faible sont regroupées à chaque étape. Une fois tous les regroupements réalisés il est courant de représenter cette méthode selon un arbre binaire appelé dendrogramme. Un schéma de dendrogramme est représenté en Figure 2.3.

A noter que la procédure décrite est un regroupement hiérarchique ascendant, car elle part des feuilles de l'arbre vers la racine, elle est la plus couramment utilisée. Cette procédure existe aussi de façon descendante, où on part d'une seule classe puis on sépare la ou les classes en deux sous-classes suivant la mesure de dissimilarité. Il existe aussi une

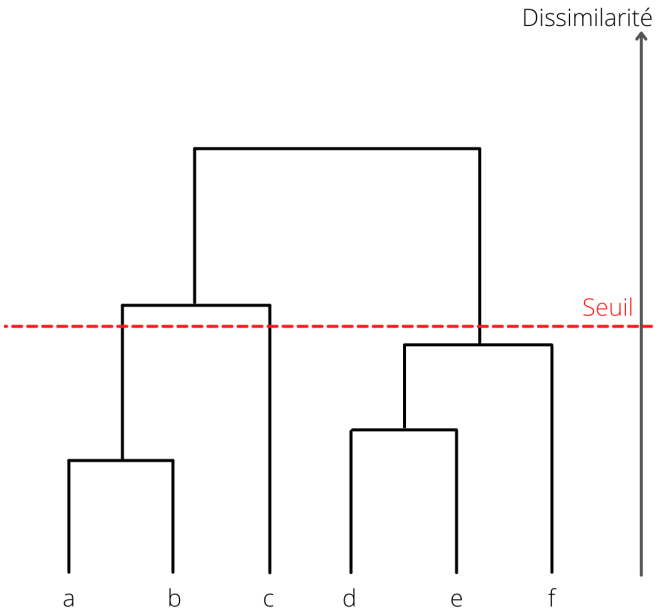

FIGURE 2.3 – *Schéma d'un dendrogramme.*
Regroupement hiérarchique sur 6 individus a, b, c, d, e, f. Un seuil est représenté en rouge, correspondant à une valeur de dissimilarité et une coupure en 3 classes.

procédure qui est à la fois ascendante et descendante appelée regroupement agglomératif hiérarchique (Hierachical Agglomerative Clustering [45]).

Une fois le regroupement hiérarchique effectué il est possible de définir un seuil permettant d'obtenir un nombre de classe final correspondant à une certaine dissimilarité entre classe. Le seuil peut être choisi en posant un nombre de classe ou en fixant une valeur de dissimilarité. Dans cette thèse nous utiliserons le regroupement hiérarchique en section 5.2.

2.3.3.2 K-means

Le K-means est un algorithme de classification non-supervisée aussi qualifié de partitionnement (clustering). Étant donné n points $x_i \in \mathbb{R}^p$ et un entier $K \in \mathbb{N}$ et $K < n$, l'algorithme K-means cherche à annoter les x_i dans K groupes $g = (g_1, g_2, \ldots, g_K)$, aussi appelés clusters. Chaque groupe est représenté par la moyenne m_{g_k} (où $k \in \{1, \ldots, K\}$) des points qui appartiennent à ce groupe. On cherche à affecter chaque point à un groupe de façon à minimiser la distance entre les points x_i et les moyennes m_{g_k}.

$$\arg\min_{\mathbf{g}} \sum_{k=1}^{K} \sum_{x_i \in g_k} ||x_i - m_{g_k}||^2 \tag{2.9}$$

L'algorithme K-means classique le plus utilisé consiste à choisir K points initiaux pour les moyennes $(m_{g_1}, m_{g_2}, \ldots, m_{g_K})$ représentant les barycentres des groupes, puis à affecter les points à chaque groupe en fonction de la distance aux moyennes et recalculer la moyenne des groupes en fonction des points attribués à ceux-ci, enfin répéter ces opérations jusqu'à convergence. Voici l'algorithme des K-means en pseudo code :

Algorithme 1 K-means

Tirer aléatoirement K points $(m_{g_1}^{(0)}, m_{g_2}^{(0)}, \ldots, m_{g_K}^{(0)})$
tant que $\forall k, m_{g_k}^{(t)} \neq m_{g_k}^{(t-1)}$ **faire**
$g_k^{(t)} = \left\{ x_i : \left\| x_i - m_{g_k}^{(t)} \right\| \leq \left\| x_i - m_{g_{k'}}^{(t)} \right\|, \forall\ k' \neq k \right\}$
$n_{g_k}^{(t)} = Card(g_k^{(t)})$
$m_{g_k}^{(t+1)} = \frac{1}{n_{g_k}^{(t)}} \sum_{x_i \in g_k^{(t)}} x_i$
fin tant que
return $\forall k, g_k$ la position des barycentres des groupes

L'algorithme ci-dessus est donc dépendant de l'initialisation, qui est ici aléatoire. Il n'est donc pas garanti de converger vers une solution optimale. Une autre initialisation aléatoire est la méthode Forgy qui choisit comme point de départ K points parmi les x_i. À noter qu'il existe aussi la méthode « K-means++ » [46] qui propose une initialisation améliorant les probabilités d'obtenir une solution optimale. Converger vers une solution optimale n'est pas garanti, cependant l'algorithme garanti la convergence en temps fini car la distance de chaque point à la moyenne des clusters diminue strictement à chaque itération. En considérant des $x_i \in \mathbb{R}^2$, on peut représenter les différentes étapes du K-means sur un plan. La Figure 2.4 représente les étapes du K-means jusqu'à convergence.

Le K-means est très utilisé pour réaliser des problématiques de partitionnement, cependant il reste nécessaire de connaître le nombre de groupes K à chercher. Ceci est rarement avantageux puisqu'il faut donc avoir un *a priori* assez fort sur le problème que nous essayons de résoudre. En petite dimension, il est possible de déterminer K en observant les données ou en regardant le comportement de la méthode à différentes valeurs de K. En revanche, quand la dimension dépasse 3, il devient difficile de fournir cet *a priori*

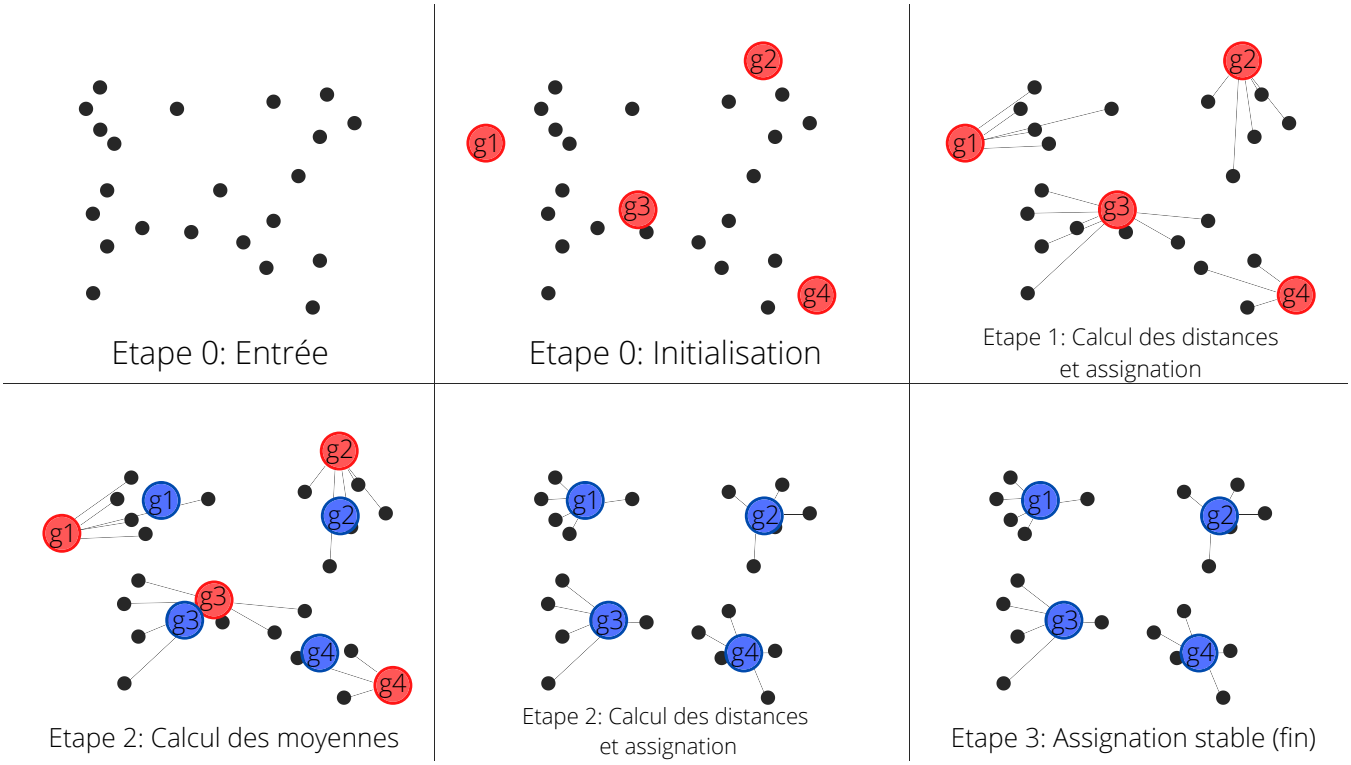

FIGURE 2.4 – *Schéma des différentes étapes de l'algorithme du K-means.*
Le K-means est réalisé sur un plan ($x_i \in \mathbb{R}^2$) avec $K = 4$. Les points noirs représentent les x_i et $g1, g2, g3, g4$ les barycentres des quatre groupes.

sans avoir d'idée sur les groupes recherchés. Nous utiliserons cette méthode en partie 8 pour analyser des résultats de différentes méthodes.

2.4 LASSO

2.4.1 Régression linéaire pénalisée

Un modèle à sélection de variable a été proposé par Tibshirani en 1996 appelé « Least Absolute Shrinkage and Selection Operator » (LASSO [47]). En plus de permettre la sélection de variable, ce modèle a été construit pour être efficace sur des problèmes de dimension c'est à dire quand $p >> n$. Le LASSO est une version pénalisée des moindres carrés. La solution $\hat{\beta}$ d'un LASSO est obtenue en minimisant l'erreur quadratique moyenne sous la contrainte $\sum_{j=1}^{p} |\beta_j| \leq t$ où t est un paramètre contrôlant le niveau de régularisation. Cette contrainte est incluse dans la fonction de risque en ajoutant la norme L1 des β_j pondérée par un réel λ, le paramètre de régularisation, comme dans l'équation suivante :

$$R(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \sum_{j=0}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=0}^{p} |\beta_j| \tag{2.10}$$

La contrainte en norme L1 permet de forcer à 0 les coefficients associés à des variables sans effet prédictif sur le modèle. Cette sélection de variable induite par la contrainte est un avantage majeur du LASSO par rapport à d'autres normes, comme par exemple, la contrainte en norme L2 qui ajoutée aux moindres carrés est appelée régression RIDGE. Cette dernière est utilisée pour optimiser les performances du modèle et permet de stabiliser les estimations des coefficients mais ne permet pas de sélection de variables. La Figure 2.5 représente l'estimation des coefficients sous contraintes avec LASSO et RIDGE, la forme de boule carrée de la norme L1 (à gauche) permet à l'intersection entre la contrainte et l'ellipse d'être sur l'axe β_2 entraînant $\hat{\beta_1} = 0$, grâce à la géométrie de la contrainte en norme L1. En effet, l'ellipse rencontre la boule carrée sur des angles entraînant certain coefficient à 0 (comme sur la figure 2.5 à gauche). À l'inverse, la forme circulaire de la norme L2, implique que l'ellipse rencontre la boule de la norme L2 sur un point quelconque de sa circonférence, réduisant la valeur et la variance des coefficients mais ne les mettant pas à 0.

Dans le contexte du LASSO, le choix du paramètre λ est crucial et dépend des données. Plus la valeur est grande, plus il y aura de coefficients à 0. Si $\lambda = 0$ la solution est la même qu'en utilisant l'erreur quadratique moyenne et si $\lambda \longrightarrow \infty$ tous les coefficients sont à 0. Pour estimer le modèle il faut donc choisir le meilleur λ. Dans les faits, plusieurs valeurs de λ sont testées en validation croisée, la valeur minimisant l'erreur commise est choisie. On appelle cette valeur λ_{min}. Cette valeur n'est pas la seule à être utilisée, λ_{1se}

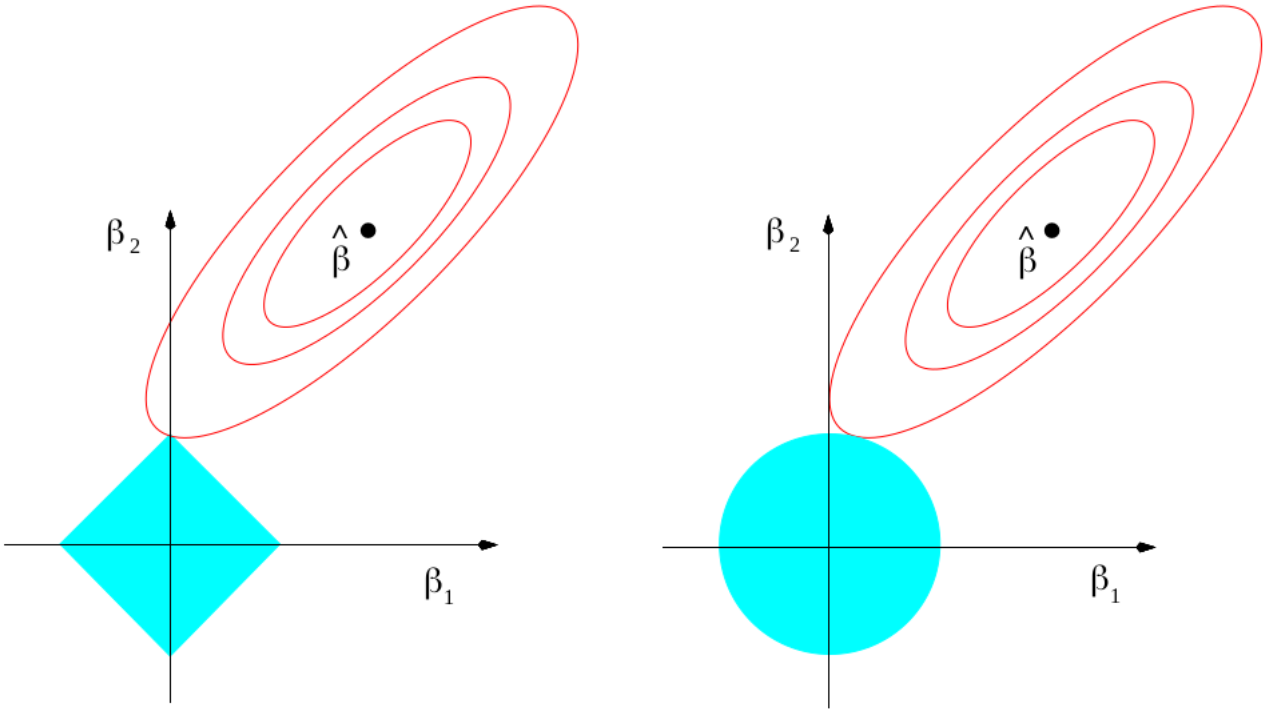

FIGURE 2.5 – *Estimation avec LASSO et RIDGE dans* $\mathbb{R}^2$.
Représentation de l'estimation des coefficients β_1 et β_2 du LASSO (à gauche) et de RIDGE (à droite). Les ellipses rouges représentent la somme résiduelle des carrés qui augmente du minimum $\hat{\beta}$. Les aires bleues correspondent aux contraintes des normes L1 (à gauche) et L2 (à droite). La solution de chaque optimisation du modèle est l'intersection entre l'ellipse et la contrainte. La géométrie de la pénalisation L1 permet de mettre des coefficients à 0 car, comme sur la figure de gauche, l'ellipse rencontre l'aire bleue sur un angle donc $\beta_1 = 0$. Alors qu'avec la pénalisation L2, l'ellipse rencontre la circonférence de l'aire bleue sur un point quelconque, ce qui implique que les coefficient β_1 et β_2 sont plus petits (et auront une variance plus faible), mais ne seront pas nuls. Figure adaptée de *« The elements of statistical learning : Data mining, inférence, and prediction Hastie, Tibshinari, Friedman 2009 »* [48].

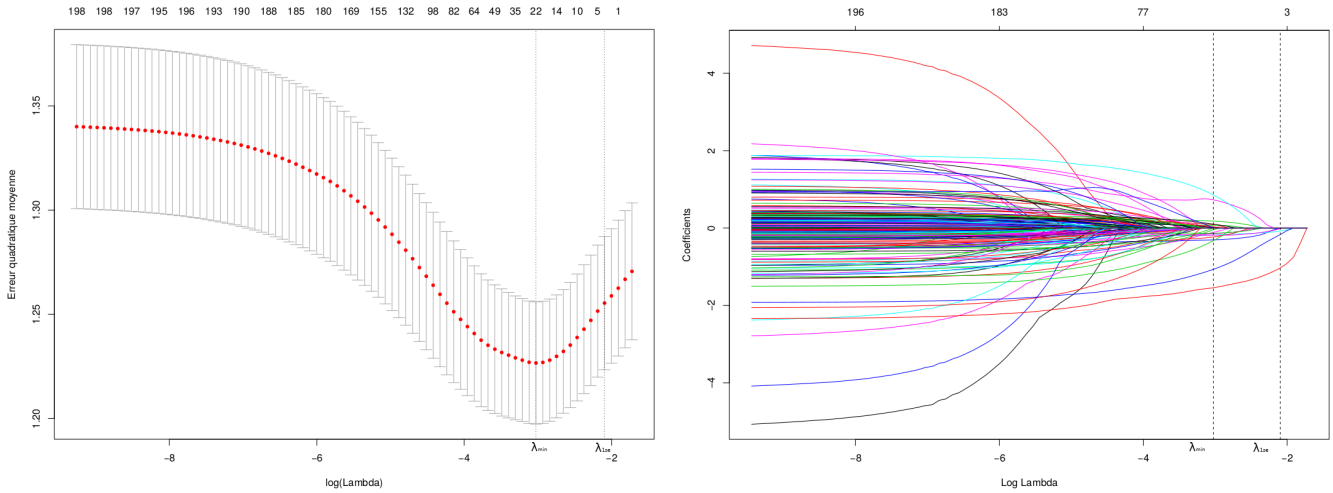

FIGURE 2.6 – *Représentation de la sortie d'un LASSO avec 200 variables.*
Le graphique de gauche représente $log(\lambda)$ en fonction de l'erreur moyenne quadratique. L'erreur quadratique moyenne calculée en validation croisée avec intervalle de confiance pour chaque valeur de lambda. Le graphique de droite représente $log(\lambda)$ en fonction des coefficients. Chaque courbe représente l'évolution d'un coefficient associé à une des 200 variables. Sur les deux graphiques les droites verticales en pointillés montrent les valeurs de λ_{min} et λ_{1se} et les valeurs en haut représentent le nombre de coefficients non nuls. Ces graphiques ont été obtenus avec le package *cv.glmnet* de R.

qui donne une erreur éloignée d'une erreur type de l'erreur minimum est commune aussi. λ_{min} implique de meilleures performances, alors que λ_{1se} permet de sélectionner moins de variables. Un exemple de régression pénalisée LASSO avec 200 variables pour prédire une variable continue est illustré en Figure 2.6. Le premier graphique (à gauche) montre la sortie de la validation croisée appliquée pour sélectionner la meilleure valeur de λ. Ce graphique exprime l'erreur quadratique moyenne en fonction de $log(\lambda)$. Le second graphique (à droite) illustre la valeur des coefficients du modèle en fonction de $log(\lambda)$, chacune des courbes correspond à la valeur d'un coefficient associé à une variable. Sur les deux graphiques, l'axe du haut correspond au nombre de coefficients non nuls pour chaque valeur de $log(\lambda)$. λ_{min} et λ_{1se} sont représentés en pointillés. Le modèle avec λ_{min} a 22 coefficients non-nuls pour une erreur quadratique moyenne égale 1.22 alors que λ_{1se} n'en a que 4 et une erreur quadratique moyenne de 1.25.

Pour estimer les coefficients de la régression linéaire pénalisée on utilisera un algorithme de descente de gradient. La descente de gradient est une méthode d'optimisation pour trouver un minimum (ou maximum en inversant le signe), elle est souvent utilisée quand les calculs des dérivées d'une fonction sont trop coûteux. Elle consiste à calculer la dérivée de la fonction en 1 point, puis à suivre le sens opposé de la pente avec un pas défini pour converger vers un minimum local. Comme $|\beta|$ n'est pas dérivable en 0, la fonction de perte du LASSO n'est pas dérivable en 0, alors on utilisera plutôt la descente de sous-gradient [49] pour optimiser les paramètres. Si la fonction est strictement convexe on est assuré de converger vers un minimum global. Si X^TX est semi-définie positive, alors la fonction de perte quadratique est convexe et comme $|\beta|$ est convexe, alors la fonction de perte du LASSO est convexe. Nous avons donc dans ce cas la garantie de converger vers un minimum global.

2.4.2 Régression linéaire logistique pénalisée

Nous avons vu la régression linéaire pénalisée en section 2.4.1 et la régression logistique en section 2.3.1. Il est possible de définir la régression linéaire logistique pénalisée, cette méthode diffère de la régression logistique par l'ajout de la pénalisation. Nous allons détailler ici plus particulièrement le fonctionnement de la régression logistique et les différences induites par la pénalité.

Dans ce cadre on écrit le modèle linéaire logistique de la façon suivante :

$$logit(\mathbb{P}(Y=1|X)) = ln\frac{\mathbb{P}(Y=1|X)}{1-\mathbb{P}(Y=1|X)} = \sum_{j=0}^{p}\beta_j X_j \tag{2.11}$$

où $X = (X_0, X_1, \ldots, X_p)$ avec $X_0 = (1, \ldots, 1)$ et $\beta = (\beta_0, \ldots, \beta_p)$ les coefficients de la régression linéaire. On va utiliser le maximum de vraisemblance pour estimer les coefficients de la régression linéaire. La vraisemblance s'écrit

$$L(\beta) = \prod_{i=1}^{n}\mathbb{P}(Y_i=1|X_i)^{Y_i}\mathbb{P}(Y_i=0|X_i)^{1-Y_i} = \prod_{i=1}^{n}\mathbb{P}(Y_i=1|X_i)^{Y_i}(1-\mathbb{P}(Y_i=1|X_i))^{1-Y_i}. \tag{2.12}$$

En appliquant la fonction logarithme sur cette expression on obtient

$$l(\beta) = \sum_{i=1}^{n} Y_i ln(\mathbb{P}(Y_i=1|X_i)) + (1-Y_i)ln(1-\mathbb{P}(Y_i=1|X_i)), \tag{2.13}$$

ce qui est équivalent à l'équation suivante

$$l(\beta) = \sum_{i=1}^{n} Y_i(\sum_{j=0}^{p}\beta_j X_j) - ln(1+e^{\sum_{j=0}^{p}\beta_j X_j}). \tag{2.14}$$

On ajoute à cette dernière équation la norme L1 de β pondérée par λ afin d'obtenir la pénalisation, et on cherche donc $\hat{\beta}_\lambda$ tel que,

$$\hat{\beta}_\lambda = \arg\max_{\beta}(\sum_{i=1}^{n} Y_i(\sum_{j=0}^{p}\beta_j X_j) - ln(1+e^{\sum_{j=0}^{p}\beta_j X_j}) + \lambda|\beta|). \text{ [50]} \tag{2.15}$$

Le paramètre λ est, là encore, optimisé en validation croisée. Ce type de régression est là encore appelée LASSO, car il repose là encore sur la pénalité L1. C'est celui-ci qui sera très utilisé dans nos analyses en Partie II.

2.4.3 Propriétés du LASSO

Le LASSO, qu'il soit utilisé pour un modèle linéaire ou linéaire logistique, comporte plusieurs avantages et inconvénients dont nous donnons un aperçu ici. L'avantage principal qui motive le choix de cette méthode est bien sûr la sélection de variables. Ceci permet au LASSO d'être efficace dans des problématiques où $n << p$ (avec n le nombre

d'observations et p le nombre de variables explicatives). Dans ce cas là, le LASSO pourra sélectionner au maximum n variables. Avec beaucoup de variables, il est fréquent d'avoir certaines variables qui sont fortement corrélées entre elles. Dans ce cas le LASSO ne sélectionnera généralement qu'une seule variable parmi le groupe de variables corrélées [51], considérant que les autres variables apportent la même information. Cependant le seuil de corrélation à partir duquel le LASSO ne sélectionne qu'une seule variable parmi deux très corrélées n'est pas fixe et on peut donc observer deux variables sélectionnées avec une corrélation relativement importante. Plusieurs extensions du LASSO existent, on peut tout d'abord citer Elastic Net [51] qui a été développé justement pour ces limitations de sélection des variables corrélées et de maximum de variables pouvant être sélectionnées quand $n << p$. L'idée dans Elastic Net est d'ajouter la pénalisation L2 (Ridge) au LASSO, c'est à dire d'utiliser les deux pénalisations simultanément. On peut aussi citer Group Lasso [52], qui permet de sélectionner des groupes de variables parmi des groupes prédéfinis. Ou enfin, Fused-Lasso [53] qui permet de réduire l'écart des coefficients associés à des variables proches.

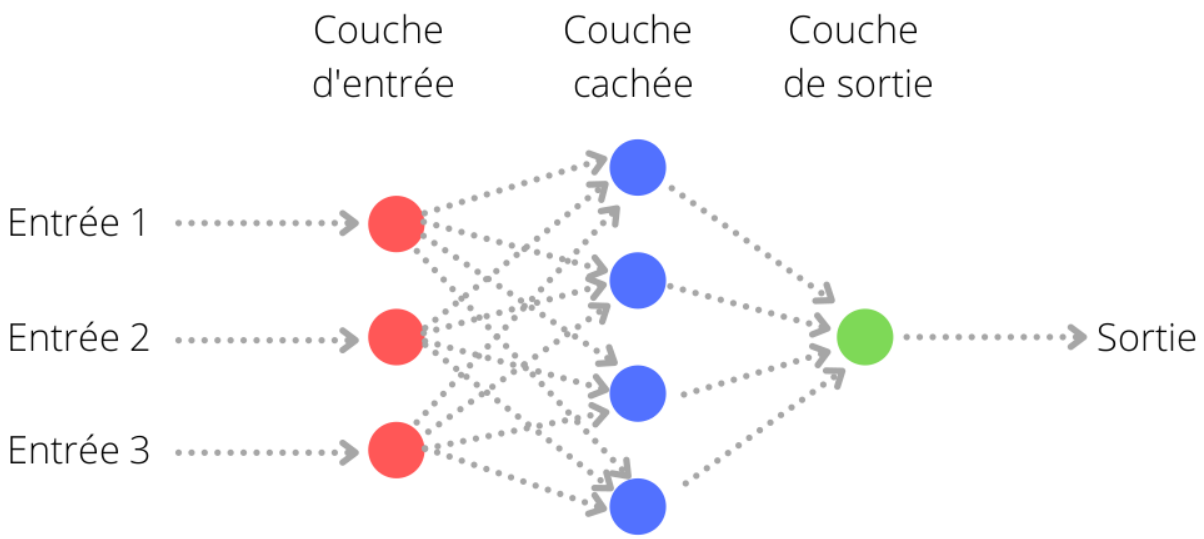

Figure 2.7 – *Schéma de l'architecture d'un réseau de neurones.*
Les disques correspondent à un neurone qui prend une ou plusieurs entrées pour fournir une sortie. Les flèches en pointillés indiquent les connexions entre les neurones.

2.5 Méthodes basées sur les réseaux de neurones

Un réseau de neurones est une méthode d'apprentissage statistique qui s'inspire directement du fonctionnement des neurones biologiques. Chaque neurone est une fonction mathématique qui prend en entrée une des données pour calculer une valeur de sortie. Les neurones sont généralement connectés entre eux sous forme de couches pour former un réseau de neurones (Figure 2.7), plus un réseau est profond, plus le nombre de couches est élevé. Le choix de l'architecture du réseau, les techniques et les méthodes d'apprentissage des paramètres du réseau est un domaine en plein essor qui ne sera pas détaillé ici. Cependant, il faut savoir que les réseaux de neurones sont souvent définis comme des « boites noires » et sont capables d'apprendre des relations complexes. Ces modèles peuvent être vus comme des modèles de régression linéaire, chaque fonction implémentée dans un neurone étant une fonction linéaire suivie d'une fonction de seuillage afin de s'affranchir des contraintes dûes à la linéarité. Les réseaux de neurones sont inférés sur la base d'algorithmes de maximisation de la vraisemblance [54]. Ils sont supposés pouvoir approximer n'importe quelle fonction et constituent donc une branche très intéressante de l'apprentissage statistique.

Ces algorithmes qualifiés de *deep learning* sont des outils très utilisés ces dernières années dans le domaine d'analyse de séquences biologiques comme par exemple DeepBind et DeepSEA (détaillés en sections 3.3.4 et 3.3.5) ou encore KEGRU [55] et Bichrom [56]. Ils peuvent s'avérer très utiles pour identifier des règles complexes. Ils se sont notamment illustrés en médecine, en analyse d'image et dans la détection d'éléments cellulaires [57]. Ils sont aujourd'hui beaucoup appliqués à la génomique dans différentes problématiques. Cependant leur fonctionnement « boite noire », permet rarement d'apporter une interprétation biologique pourtant primordiale dans le domaine, c'est pourquoi des recherches portent sur la façon de rendre interprétables ces modèles [58]. Quelques travaux ont proposé des pistes intéressantes pour pallier ce problème d'interprétabilité (DeepBlueR [59] et BPnet [60]). Pour plus de détails sur les réseaux de neurones et leur utilisation dans des domaines biologiques, nous proposons au lecteur de consulter la revue suivante [61].

Chapitre 3

Modélisation et apprentissage statistique pour le génome

3.1 Modélisation des sites de fixation

3.1.1 Matrices de position : PFM, PPM et PWM

Les matrices sont couramment utilisées pour décrire un modèle de séquences biologiques. Nous présenterons ici trois matrices : PFM (Position Frequency Matrix), PPM (Position Probability Matrix) et PWM (Position Weight Matrix). Ces modèles ont été introduits initialement par Gary Stormo comme alternative aux séquences consensus qui permettent de facilement décrire une collection de sites de fixation mais ne permettent pas de prédire l'apparition de nouveaux sites [62]. Gary Stormo a notamment utilisé ces modèles pour la modélisation de sites de fixation de l'ADN. Ces modèles se représentent par des matrices de taille $|\Lambda| \times K$ où $|\Lambda|$ est le nombre de symboles différents dans l'alphabet des séquences, et K le nombre de positions qui composent le motif.

À partir d'un ensemble Z de N séquences alignées de taille K, les éléments de la matrice M d'un PFM sont calculés de la manière suivante :

$$M_{i,k} = \sum_{\{z \in Z\}} I(z_k, \Lambda_i), \tag{3.1}$$

avec $i \in [1; |\Lambda|]$, $k \in [1; K]$, Λ_i le ième symbole de l'alphabet Λ, z_k le kième symbole de la séquence z et I une fonction indicatrice telle que $I(a, \Lambda_i) = 1$ si $a = \Lambda_i$ et $I(a, \Lambda_i) = 0$

sinon.

Une fois que nous avons créé le PFM, nous pouvons en déduire un PPM en divisant chaque valeur par le nombre total de séquences. Chaque colonne du PPM définit donc une distribution de probabilité sur Λ. L'expression de $M_{i,k}$ devient donc :

$$M_{i,k} = \frac{1}{N} \sum_{\{z \in Z\}} I(z_k, \Lambda_i), \tag{3.2}$$

Dans un PPM, chaque position est supposée indépendante : la probabilité d'apparition d'un élément à une position donnée ne dépend ni des positions suivantes, ni des positions précédentes. Nous pouvons donc calculer la probabilité $\mathbb{P}_{PPM}$ qu'une séquence z soit générée à partir du PPM de la manière suivante :

$$\mathbb{P}_{PPM}(z) = \prod_{k=1}^{K} M_{r(z_k),k}, \tag{3.3}$$

avec $r(z_k)$ l'indice du caractère z_k dans Λ.

Les PWM sont des matrices de poids représentant les affinités nucléotidiques associées à chaque position du motif. Les nucléotides favorables à la fixation du TF ont un poids positif, tandis que les nucléotides défavorables ont un poids négatif. Une manière de construire un PWM est de calculer l'odds ratio de chaque élément du PPM. Les éléments du PWM se calculent alors de la manière suivante :

$$W_{i,k} = log(\frac{\frac{1}{N} \sum_{\{z \in Z\}} I(z_k, \Lambda_i)}{b_i}) \tag{3.4}$$

où b_i est la probabilité *a priori* associée au symbole Λ_i dans l'organisme étudié.

De la même manière que dans un PPM, dans un PWM chaque position est supposée indépendante. Le score s d'une séquence z pour un PWM s'obtient de la manière suivante :

$$s(z) = \sum_{k=1}^{K} W_{r(z_k),k}. \tag{3.5}$$

Une séquence dont la probabilité est plus élevée dans le modèle nul aura donc un score négatif. Inversement, une séquence ayant une probabilité plus élevée dans le PPM aura

un score positif.

Afin d'éviter d'obtenir une probabilité nulle lorsque l'on utilise le PPM sur une nouvelle séquence (et donc un score de PWM de $-\infty$), il est souvent nécessaire d'appliquer une correction au préalable sur le PFM. Une correction simple est d'ajouter un pseudo-compte (ou estimateur de Laplace) à chaque case du PFM. Ajouter un pseudo-compte revient à ajouter de fausses séquences dans notre alignement qui, à partir de nos connaissances des séquences protéiques et nucléiques, permettent de modéliser simplement tous les événements qui pourraient se produire. Une approche alternative, plus sophistiquée mais que nous ne détaillerons pas ici, est basée sur l'utilisation de modèles de Dirichlet [63].

Pour déterminer si une nouvelle séquence appartient à la famille de séquences modélisées, il est nécessaire d'évaluer la significativité de son score. Pour cela, la méthode standard consiste à estimer la distribution de scores de séquences générées par le modèle nul, et d'en déduire une p-valeur correspondant à la probabilité d'obtenir un score aussi bon que le score observé avec une séquence aléatoire. Si la p-valeur est inférieure à un certain seuil, usuellement 1% ou 0.1%, nous pourrons en conclure qu'il est peu probable d'obtenir une séquence aussi bonne avec le modèle nul, et donc que, vraisemblablement, cette nouvelle séquence appartient à la famille en question.

3.1.2 Rechercher les occurrences d'un motif

En pratique, les PPM et PWM sont utilisées pour identifier les sites de fixations potentiels dans une séquence plus ou moins longue. Pour identifier des sites de fixation potentiels d'un facteur de transcription on va être amené à scanner et à scorer la séquence avec le motif. Une approche simple consiste à faire « glisser » la PWM le long de toute la séquence position par position. On peut ainsi obtenir le score du motif à chaque position. Cependant cette approche trouve rapidement ces limites quand on considère un grand nombre de séquences et/ou des séquences de grandes tailles. Une autre approche, utilisée dans beaucoup d'algorithmes de recherche d'occurrences actuels, permet d'augmenter considérablement la vitesse de traitement. L'idée est de commencer par énumérer l'ensemble des mots acceptés par le motif selon un seuil donné puis de rechercher ces mots dans les séquences. Ce seuil peut être fixé sur le score du motif ou sur la p-valeur. Pour cela, l'algorithme MotifSearch [64] va par exemple effectuer l'étape d'énumération en

utilisant un arbre où chaque étage correspond à une position du motif et où chaque nœud donne naissance à $|\Lambda|$ branches. Pour ce qui est de l'étape de recherche, le logiciel exploite une implémentation basée sur la transformée de Burrow-Wheeler [65] et du FM-index [66], les séquences à scanner sont ainsi indexées au début de la procédure de manière à pouvoir trouver rapidement si le mot est apparu dans une séquence et si oui, à quelle position.

Plusieurs outils de scan des séquences avec les PFM, PPM et PWM sont disponibles, tels que MotifSearch qui renvoie le score du motif et le score normalisé par rapport aux scores maximums et minimums, ou FIMO [67] qui renvoie le score, une p-valeur et une p-valeur corrigée. L'outil FIMO, qui fait partie de la suite MEME [68] est utilisé par la suite dans nos analyses. Nous paramétrons cet outil de façon à chercher les occurrences des motifs (p-valeur < 0.001) sur les deux brins de l'ADN.

Lorsque l'objectif est de scorer la séquence pour estimer si elle comporte au moins un site de fixation potentiel, l'approche classique consiste à utiliser le meilleur score identifié dans la séquence. Dans la suite de ce manuscrit on se référera à cette approche sous le terme d'« approche Best-hit ». Notons que lorsqu'aucune position d'une séquence n'obtient un score supérieur au seuil utilisé pour la recherche, on donne à la séquence un score proportionnel au score minimum trouvé dans l'ensemble des séquences.

3.1.3 Représenter un motif

Une manière classique de représenter une PPM est de calculer la quantité d'information associée à chaque position, et de faire varier la hauteur des lettres proportionnellement à la quantité d'information. La quantité d'information se calcule via l'entropie de Shanon :

$$I_k = 2 + \sum_{\Lambda=\{A,C,G,T\}} p_{\Lambda_i,k} log_2(p_{\Lambda_i,k}) \tag{3.6}$$

où $p_{\Lambda_i,k}$ est la probabilité de la lettre $\Lambda_i \in \Lambda$ à la position k de la PPM. On appelle cela une représentation logo. Un exemple de représentation logo de la PPM MA0112.3 ESR1 disponible dans la base de donnée JASPAR, est donné en Figure 3.1 (cf. section 3.4.1).

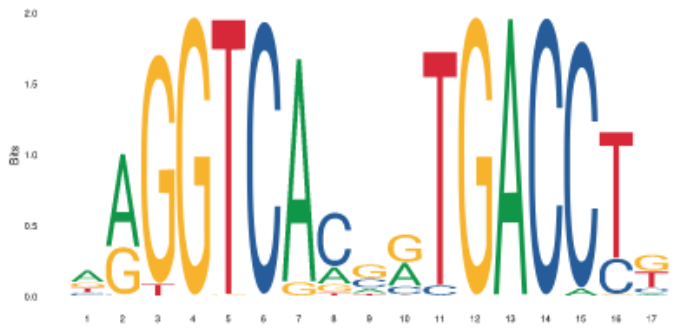

Figure 3.1 – *Représentation logo d'une PWM.*
PPM MA0112.3 ESR1 disponible dans JASPAR. Figure importée de `http://jaspar.genereg.net/`.

3.2 Apprentissage de PWM

En pratique, on ne dispose quasiment jamais de séquences alignées ayant strictement la taille du site de fixation. L'apprentissage des PWM ne peut donc se faire directement de la manière exposée ci-dessus à partir d'une simple matrice de fréquences. Plusieurs approches ont donc été proposées pour apprendre une PWM à partir de séquences longues et supposées contenir des sites de fixation du facteur cible. Deux grands types d'approches ont été proposés pour cela. Dans le premier type, l'apprentissage se fait uniquement sur la base d'un ensemble de séquences positives (c-à-d contenant le site de fixation recherché). Dans le second cas, on dispose d'un ensemble des séquences positives et négatives, et on cherche une PWM capable de discriminer au mieux ces deux ensembles. Les sections suivantes détaillent ces deux types d'approches.

3.2.1 Approches basées sur la vraisemblance

Dans ce type d'approche, on cherche un motif Θ de taille K que l'on veut estimer à partir d'un ensemble de séquences de taille T. Ces dernières sont supposées contenir chacune un site de fixation pour le facteur cible. À l'heure actuelle, ces séquences sont souvent issues de données de ChIP-seq ou SELEX/HT-SELEX, mais ce n'est évidemment pas toujours le cas, et la sélection des séquences données en entrée peut être faite de différentes manières. On a un PPM $\Theta = (\theta_1, \ldots, \theta_K)$ où θ_k est une distribution de probabilité sur {A,T,C,G}. On définit un modèle nul Θ^0 (background) exprimant les distributions *a priori* des éléments de l'alphabet. Un modèle simple est de considérer que tous les

symboles de Σ sont équiprobables, c'est à dire $\forall k \in [1, K]$, $\theta_k^0 = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$. Cependant, en pratique nous utiliserons plutôt les probabilités observées chez l'organisme étudié. Un critère souvent utilisé pour estimer les paramètres de Θ est de chercher dans chaque séquence Z_i une sous-séquence de taille K notée z_i, telle que la PPM reconstruite à partir de cet ensemble de sous-séquences maximise le rapport de log-vraisemblance :

$$\sum_{i=1}^{N} log(\frac{\mathbb{P}(z_i|\Theta)}{\mathbb{P}(z_i|\Theta^0)}) \tag{3.7}$$

où z_i représente la sous-séquence de taille K choisie dans Z_i et $\mathbb{P}(z_i|\Theta)$ est la probabilité de générer la séquence z_i avec le modèle Θ (voir équation 3.3).

Plusieurs méthodes peuvent être utilisées pour trouver Θ maximisant le rapport de log-vraisemblance, telles que le Gibbs sampler [69] ou les algorithmes EM [70].

3.2.1.1 Gibbs-sampler

Le Gibbs-sampler (Échantillonneur de Gibbs) est un algorithme de type MCMC (« Monte-Carlo Markov Chain ») de découverte de motifs [71, 72]. Pour cela, l'algorithme choisit aléatoirement N sous-séquences $z_i, i \in \{1, \ldots, N\}$ de longueur K (une par séquence), construit une PPM puis une PWM de taille K avec ces sous-séquences. Ensuite, chaque position de chaque séquence est scorée avec la PWM construite et on tire aléatoirement une nouvelle position pour chacune des séquences. Plus le score de la PWM est grand à une position plus la probabilité de tirer cette position est grande. Cet ensemble de positions définit un nouvel ensemble de sous-séquence que l'algorithme utilise pour construire une nouvelle PWM. Cette procédure est itérée un grand nombre de fois, jusqu'à stabilisation du log ratio (expression 3.7). Cet algorithme est stochastique, il est donc intéressant de le répéter plusieurs fois afin de trouver le motif optimal. Il est efficace en pratique, mais repose sur l'hypothèse que chaque séquence contient au moins une occurence du motif recherché. Le Gibbs-sampler a par la suite été adapté et amélioré dans diverses méthodes. On peut par exemple citer la version AlignACE [73] qui permet de découvrir plusieurs motifs en masquant les motifs déjà identifiés.

3.2.1.2 Espérance-Maximisation (EM)

L'algorithme Espérance-Maximisation [70] est un algorithme d'apprentissage qui cherche à maximiser la vraisemblance d'un modèle lorsqu'il y a des données cachées. Il a été adapté pour la recherche de motifs pour la première fois par Charles E. Lawrence and Andrew A. Reilly [74]. Bien qu'il cherche lui aussi à maximiser un log-ratio, la fonction à maximiser n'est pas exactement celle décrite par l'expression 3.7. En fait, les auteurs de [74] redéfinissent le problème en découpant l'ensemble des N séquences en un nouvel ensemble N' contenant toutes les sous-séquences de taille K présentes dans les séquences d'origine. Ils définissent ensuite un modèle de mélange à deux composantes : une composante correspondant à la PPM Θ recherchée, et une composante correspondant au modèle nul. L'algorithme EM est ensuite appliqué pour trouver les paramètres du modèle qui maximisent la probabilité de générer les N' sous-séquences de taille K. Cet algorithme peut être décrit en deux étapes, l'étape E où il calcule l'espérance de la vraisemblance et l'étape M où il estime le maximum de la vraisemblance trouvée à l'étape E. Ensuite on itère sur ces étapes jusqu'à convergence. L'avantage de cette méthode par rapport à l'approche avec le Gibbs-sampler est d'enlever la contrainte forçant une occurrence du motif recherché dans chaque séquence. Parmi les logiciels utilisant l'algorithme EM pour apprendre une PWM, MEME [75] (« Multiple EM for Motif Elicitation ») est l'un des plus connus. Depuis sa première version, MEME est désormais inclus dans une collection de méthodes appelée MEME-suite [68, 76].

3.2.2 Approches discriminantes

À coté des méthodes présentées ci-dessus, basées sur une maximisation de la vraisemblance calculée sur un ensemble de séquences positives, d'autres approches prenant en entrée deux ensembles de séquences (positives et négatives) ont également été proposées. Ces approches sont communément dénommées « approches discriminantes » car elles ont pour objectif la recherche d'un motif permettant de discriminer au mieux les deux classes de séquences. [77, 78]. On peut par exemple citer dans l'ordre de parution MDScan [79], DME [80], DIPS [81], CMF [82], DECOD [83], DREME [84], DiMO [85] ou STREME [86].

3.2.2.1 STREME

STREME [86] (*Simple, Thorough, Rapid, Enriched Motif Elicitation*) est un algorithme de découverte de motifs qui peut à la fois être utilisé de façon discriminante et *ab initio*. Il remplace depuis peu DREME dans MEME-suite. Il permet de trouver plusieurs motifs à la fois. Pour cela, il recherche des motifs en suivant différentes étapes. À l'initialisation, l'algorithme va « préparer » le jeu de donnée en remplaçant les caractères ambigus et en mélangeant aléatoirement l'ordre des séquences. Puis, si aucune classe négative n'a été fournie, l'algorithme va créer un jeu de séquences contrôles en mélangeant les séquences fournies en entrée. Ensuite, STREME va itérer sur 5 étapes jusqu'à atteindre le critère d'arrêt, choisi par l'utilisateur. Ce critère peut porter sur le nombre de motifs à conserver ou sur la significativité des motifs. La première des 5 étapes consiste à créer un arbre des suffixes. La seconde étape utilise cette arbre afin de compter le nombre de fois où un mot se trouve dans les séquences, pour tous les mots de taille comprise entre 3 et la largeur maximale du motif dans les séquences positives. Une p-valeur d'enrichissement est ensuite calculée pour chacun de ses mots. Les « meilleurs » mots au sens de cette p-valeur sont ensuite convertis en un motif (PWM) en étape 3. Dans cette étape 3, le motif va être raffiné de manière à ce qu'il discrimine au mieux les séquences positives et négatives (ou contrôles). L'étape 4 consiste à utiliser le meilleur site du motif raffiné de l'étape précédente dans chaque séquence pour faire une classification sur les séquences et calculer la significativité du motif. Enfin, la dernière étape consiste à masquer tous les sites qui sont des occurrences du motif raffiné à l'étape 3, en remplaçant ces sites par des caractères absents de l'alphabet des séquences. L'algorithme complet est alors relancé sur ces séquences afin d'identifier de nouveaux motifs. En utilisant différentes sources de données l'auteur montre que STREME est « plus précis, plus sensible et plus complet que plusieurs algorithmes largement utilisés (DREME [84], HOMER [87], MEME [75], Peak-motifs [88]). »

3.2.2.2 DiMO/DAMO

DiMO utilise un algorithme de classification supervisée, le perceptron [89], avec comme mesure de performance l'AUROC pour obtenir en sortie une PFM optimisée. Cette méthode a été créée pour discriminer des classes de séquences fixées contre des classes de séquences non-fixées.

DiMO utilise en entrée une matrice PFM qu'il va ensuite optimiser. Cette matrice est

transformée en une PWM notée W. À chaque étape, une nouvelle PWM notée $W^{updated}$ va être générée à partir de W, suivant l'équation,

$$W^{updated} = W + \alpha \times \delta, \tag{3.8}$$

où α est le taux d'apprentissage et δ représente la différence entre la valeur actuelle et la valeur ciblée de la variable à optimiser. Dans DiMO, δ est défini comme la différence entre deux matrices w^+ et w^- :

$$\delta = w^+ - w^-. \tag{3.9}$$

Ces deux matrices sont obtenues suivant la procédure suivante. Chaque séquence est scannée avec la matrice W, et le meilleur score obtenu à une des positions est mémorisé (méthode Best-hit, voir section 3.1.1). Ces scores sont ensuite ordonnés du plus haut au plus bas. Si W est un classifieur parfait, tous les scores du haut de la liste correspondent à des séquences positives, et tous ceux du bas de la liste à des séquences négatives. En pratique ce n'est pas le cas, et un certain nombres de séquences sont comprises entre le score le plus haut des séquences négatives, et le score le plus bas des séquences positives. Les occurrences des séquences positives comprises dans cette partie de la liste sont utilisées pour calculer w^+, tandis que celles des séquences négatives sont utilisées pour créer w^-.

À chaque étape, si $W^{updated}$ obtient une meilleure AUROC (en utilisant le Best-hit pour classer les séquences) que W, alors elle est acceptée et devient la nouvelle PWM W à l'étape suivante. Au commencement de la méthode le taux d'apprentissage α est égal à 1, puis décroit au fur et à mesure. La méthode s'arrête si aucune amélioration de l'AUROC n'est observée ou si le maximum d'itérations choisi par l'utilisateur est atteint. Enfin, DiMO convertit la PWM optimisée en PFM, et retourne cette PFM.

Cette méthode a ensuite été améliorée (méthode DAMO [90]) pour éviter les limitations dues aux modèles probabilistes (voir section 3.3) en retournant cette fois des PWM. DAMO est donc une méthode permettant d'obtenir un PWM discriminant. Elle comporte plusieurs options différentes comme par exemple l'ajout de dinucléotides adjacents. Les auteurs montrent que DAMO permet d'augmenter un peu les performances du modèle PFM + DNAshape auxquels ils se comparent. De plus, ils montrent que « les modèles PWM simples, lorsqu'ils sont optimisés pour un AUROC maximum, sont presque aussi

performants que des modèles non linéaires plus complexes » [90]. Enfin, ils montrent les avantages des PWM par rapport aux PFM en se comparant à DiMO, et que considérer des dinucléotides adjacents dans le modèle PWM additif peut améliorer ses performances sur au moins certains des ensembles de données.

3.2.3 Différences entre approches discriminantes et non-discriminantes

Comme on l'a vu les approches discriminantes cherchent à trouver le motif qui maximise la prédiction entre des classes de séquences. En fonction de la nature de ces classes de séquences un motif discriminant peut donc représenter différentes choses. Entraîner un motif discriminant sur des classes de séquences fixées/non-fixées décrira le motif de fixation du facteur de transcription cible. À l'inverse, si le facteur de transcription est vraisemblablement fixé dans les deux classes de séquence ou si la classe négative comporte un motif proche de celui du facteur cible, le motif discriminant pourra décrire les différences entre ces motifs plutôt que le motif de fixation en lui même. De plus, en optimisant directement une PWM le motif discriminant va pouvoir s'extraire de certaines limitations imposées par les PPM classiques. En effet, une PPM est une matrice de probabilités, elle a donc certaines contraintes : valeurs positives et chaque colonne somme à 1. Une PWM issue d'une PPM est toujours liée à ses contraintes, à l'inverse d'une PWM discriminante qui peut s'en soustraire (nous discuterons ce point plus longuement dans la section 6.2). Notons finalement qu'il n'est pas possible de représenter une PWM discriminante, de la façon présentée en section 3.1, puisque cela nécessite une PFM ou une PPM. Pour cette raison, nous proposerons en section 6.3 une façon de représenter ce logo, afin de la comparer visuellement à d'autres motifs.

3.3 Limites des PWM et approches alternatives

3.3.1 Limites

Les PFM/PPM/PWM sont de loin les modèles les plus classiques pour modéliser les sites de fixation des facteurs de transcription. Cette modélisation est simple, intuitive, facilement représentable et interprétable. Elle présente malgré tout des limites inhérentes qui l'empêchent de représenter avec précision les véritables probabilités de liaison. Une de ces limites se trouve dans le fait qu'en scorant une séquence on modélise la probabilité d'observer une séquence à partir d'une PPM $\mathbb{P}(Z_i|B)$ et non la probabilité qu'une séquence spécifique soit fixée $\mathbb{P}(B|Z_i)$ [91].

Une autre limite forte est l'indépendance des positions qui est imposée dans les matrices mais qui n'est peut-être pas une réalité pour certains TF. Une autre limitation des PWM est que ces modèles ne représentent que le motif associé au site de fixation du TF. Or d'autres informations telles que la structure de l'ADN autour du motif, ou la présence de sites de fixation de cofacteurs peuvent également avoir un impact fort pour la fixation effective du TF cible. Pour ces raisons, plusieurs alternatives aux PWM ont été proposées dans la littérature. Nous en détaillons quelques unes ci-dessous.

3.3.2 TFFM

Les TFFM [92] sont une autre façon de modéliser des motifs de fixation en utilisant des chaînes de Markov cachées, ou HMM (*Hidden Markov Model*). Une chaîne de Markov est un processus stochastique dans lequel un état dépend des n états précédents, où n est l'ordre de la chaîne de Markov. Contrairement aux chaînes de Markov, dans les HMM on observe la séquence générée par le modèle mais pas la séquence des états qui génèrent ces observations. Les TFFM sont des HMM qui modélisent les dépendances dinucléotidiques (la probabilité de générer un nucléotide à la position p dépend du nucléotide à la position $p-1$). De plus les TFFM incluent un état background qui représente les nucléotides qui entourent le site de fixation du facteur de transcription. Les paramètres du TFFM sont appris avec l'algorithme Baum-Welch. La Figure 3.2.A représente le HMM utilisé dans les TFFM. La Figure 3.2.B décompose chaque état du HMM, de manière à gérer les dépendances dinucléotidiques.

Les TFFM apportent une nouvelle façon de modéliser les sites de fixations, et une

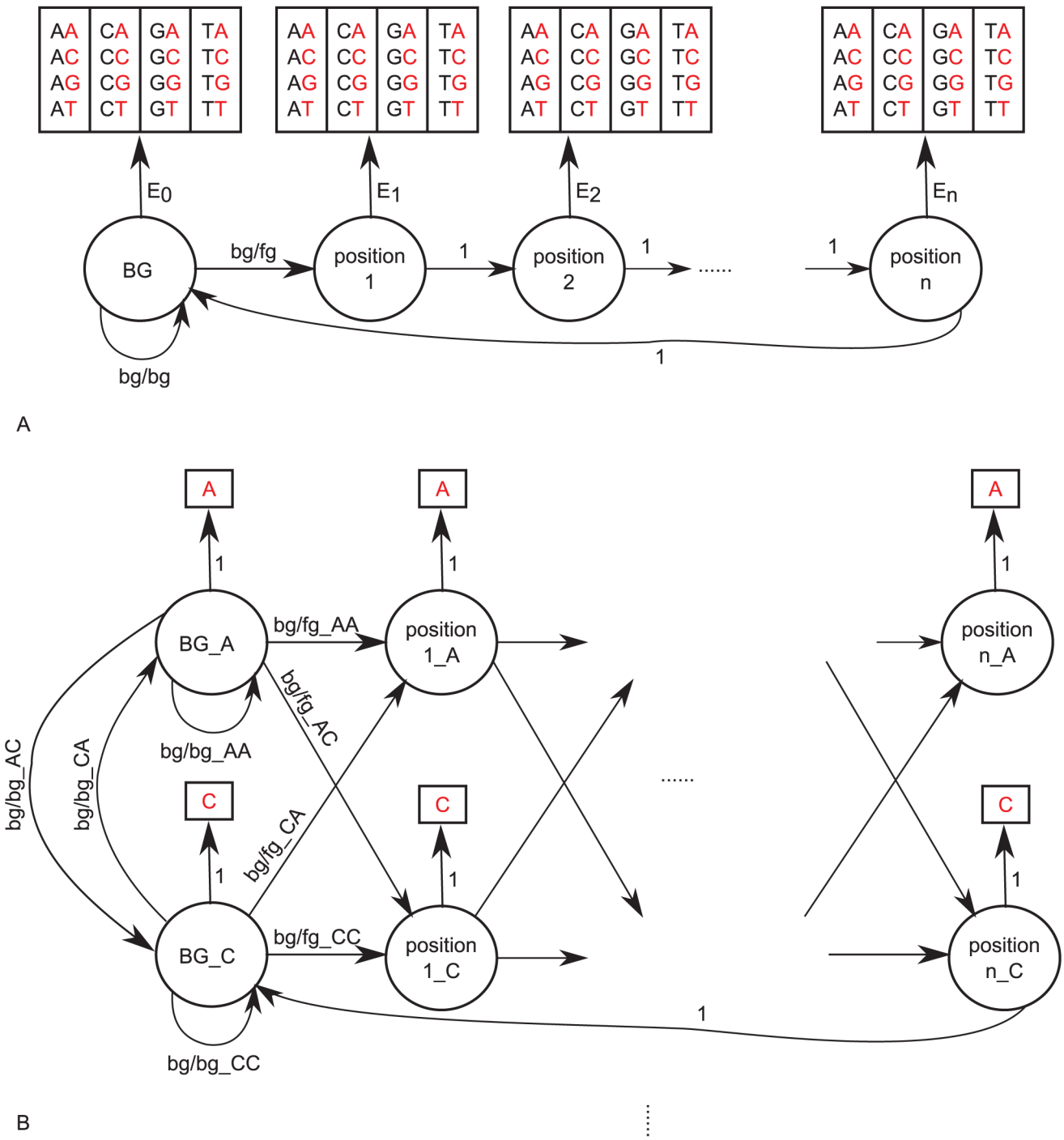

FIGURE 3.2 – *Schémas des HMM utilisés dans TFFM*
(A) Schéma HMM d'ordre 1 utilisé dans les TFFM où le premier état est l'état « background » et les états suivants les positions consécutives dans un site de fixation de TF. Chaque état émet un nucléotide avec une probabilité qui dépend du nucléotide émis précédemment. (B) Schéma détaillé du HMM utilisé dans les TFFM où chaque état dans le HMM est décomposé en quatre états (un par nucléotide). Les probabilités de transition reflètent les probabilités d'émission du HMM. Il permet le démarrage d'un site de fixation de TF en fonction du nucléotide émis par les états « background ». Figure tirée de A.Mathelier et W. Wasserman « *The Next Generation of Transcription Factor Binding Site Prediction* »,2013 [92].

représentation logo qui leur est propre permet de pouvoir les interpréter à la manière des motifs présentés précédemment. Ils permettent de mieux étudier les interactions entre nucléotides, d'étudier les positions flanquantes autour des sites de fixation et de considérer différentes tailles de motifs. En terme de performances, ils se sont montrés aussi performants que les modèles classiques pour la plupart des données considérées et ont obtenu de meilleures performances sur certains facteurs de transcription.

3.3.3 DNA-shape

La méthode DNAshape [93] fait la liaison entre la composition de la séquence ADN et sa structure fine (torsion de l'hélice, roulement...). Il a été montré qu'utiliser cette information de structure peut permettre de prédire la fixation des facteurs de transcription [94]. Pour cela, les auteurs utilisent à la fois les données DNA-shape et un motif de fixation (PWM ou TFFM). Une séquence de taille n considérée comme un site de fixation potentiel est alors représentée par $4n$ variables encodant l'information de la séquence et $8n$ variables qui décrivent la forme de l'ADN. Ils utilisent ces variables dans un classifieur basé sur du gradient boosting [95]. L'analyse de ces résultats a montré que certaines familles de facteurs de transcription sont mieux prédites lorsque l'information de DNA-shape est prise en compte. C'est le cas notamment des familles E2F et MADS-domain.

3.3.4 DeepBind

Alipanahi *et al.* (2015) ont proposé une architecture CNN peu profonde (avec une seule couche cachée) appelée DeepBind [96] qui prédit la liaison des facteurs de transcription à partir de courtes séquences d'ADN (101 paires de bases). Pour l'apprentissage, DeepBind encode les séquences de manière à avoir, pour chaque séquence de taille 101, une matrice 4×101 qui la décrit (encodage one-hot). Il procède ensuite à une convolution de la séquence avec p filtres de convolution, suivie d'une fonction ReLU (*Rectified Linear Unit*) et d'une opération de max-pooling tout au long de la séquence. Ainsi, DeepBind ajuste de nouveaux motifs afin de les utiliser pour scanner et scorer les séquences puis un réseau de neurones complètement connecté prédit la fixation du facteur de transcription à partir du scan. La Figure 3.3 représente l'architecture utilisée dans DeepBind.

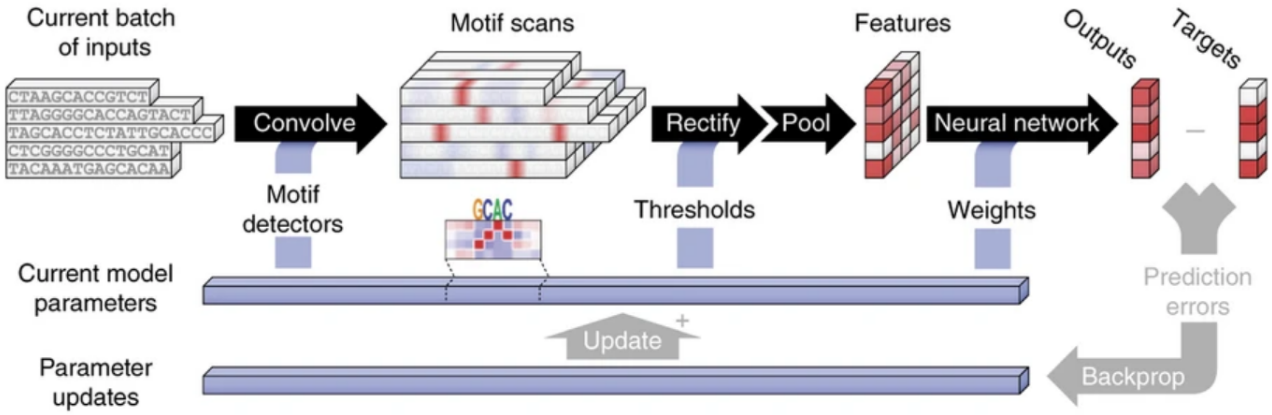

FIGURE 3.3 – *Architecture de DeepBind.*
Cinq séquences indépendantes traitées en parallèle par un seul modèle DeepBind. Les différentes étapes prédisent un score distinct pour chaque séquence en utilisant les paramètres actuels du modèle. Pendant la phase d'apprentissage, les étapes de « backprop » et « update » mettent à jour simultanément tous les motifs, les seuils et les poids du réseau du modèle pour améliorer la précision de la prédiction. Figure tirée de Alipanahi *et al.* (2015) « *Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning* ».

Une fois le modèle entraîné, les auteurs ont pu constater que DeepBind obtient de meilleures performances que la plupart des méthodes existantes en 2015, dans différents ensembles de données et sur différentes métriques d'évaluation des performances. Ils ont pu montrer que les modèles DeepBind formés *in vitro* fonctionnent bien pour évaluer les données *in vivo*, ce qui suggère une capacité à capturer les propriétés authentiques des interactions de liaison des acides nucléiques [96]. Enfin, DeepBind a pour avantage de fournir des représentations similaires aux PWM classiques ce qui permet ainsi de mieux interpréter les variables apprises par le modèle.

3.3.5 DeepSEA

DeepSEA [97] est une méthode d'apprentissage profond basée sur les réseaux de neurones. DeepSEA prend en entrée des séquences de taille 1000bp et prédit en sortie l'ouverture de la chromatine, ainsi que différentes marques d'histones et la fixation de plusieurs dizines de facteurs de transcription. Pour entraîner ce modèle, les auteurs ont compilé diverses données issues de ENCODE [98] et Roadmap [99], incluant des profils de chromatine, des motifs de fixation de TF, des marques d'histones et des DHS (*DNase I hyper-*

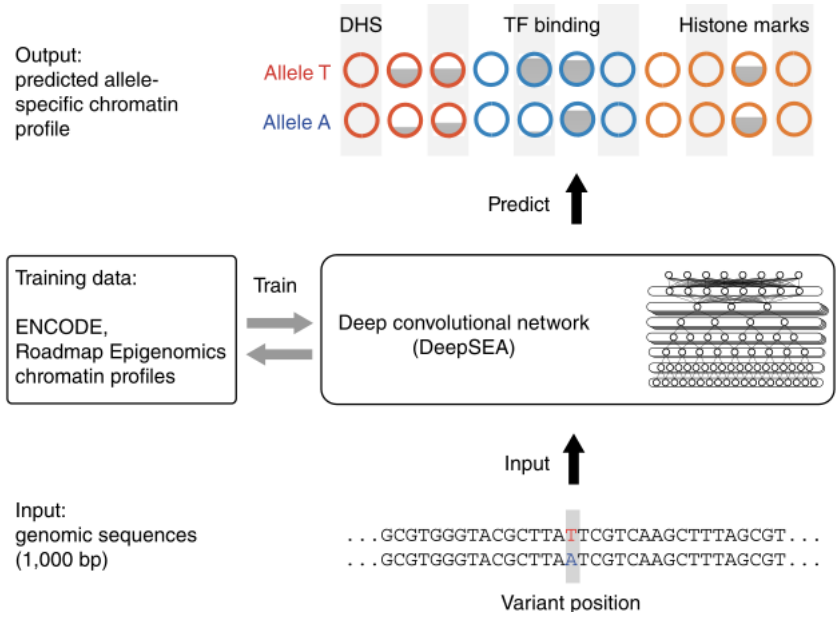

Figure 3.4 – *Schéma du fonctionnement de DeepSEA.*
Figure adaptée de Zhou et Troyanskaya (2015) « *Predicting effects of noncoding variants with deep learning–based sequence model* ».

sensitive site). La Figure 3.4 résume le fonctionnement de DeepSEA. La séquence est donc injectée en entrée d'un réseau de neurones convolutif qui permet d'extraire de l'information des séquences à différentes échelles spatiales. Chaque couche du réseau de neurones convolutif réalise une transformation linéaire de la sortie de la couche précédente en la multipliant à une matrice de poids, après cette transformation, une autre transformation non-linéaire est appliquée. La matrice de poids est ajustée durant l'apprentissage pour minimiser les erreurs de prédiction. Ce réseau est suivi par une couche « fully connected » qui intègre l'information des séquences entières. Enfin, une couche de sortie est utilisée pour prédire chacune des variables épigénétiques décrites ci-dessus.

DeepSEA obtient de bonnes performances dans la prédiction des variables épigénétiques considérées, notamment sur la fixation de facteur de transcriptions où il obtient une AUROC médiane de 0.958 sur leur données. Ils se comparent notamment à la méthode gkm-SVM [100] qui obtient sur ces mêmes données une AUROC médiane de 0.896. DeepSEA obtient aussi de bonnes performances sur les modifications d'histones (AUROC médiane = 0.856) et les DHS (AUROC médiane = 0.923). DeepSEA montre donc qu'il est possible de prédire des marques épigénétiques à partir de la séquence, comme d'autres

méthodes on pu le montrer telles que Epigram [101] et Basset [102].

3.3.6 TFcoop

La méthode TFcoop [28] a pour objectif d'étudier les combinaisons de motifs impliquées dans la fixation d'un TF cible. Elle se base pour cela sur des données de ChIP-seq ciblant un facteur de transcription particulier et se place dans une problématique de classification supervisée. Pour une expérience donnée, les N séquences présentant un pic de ChIP-seq seront utilisées comme classe positive ($Y = 1$). N autre séquences sont tirées aléatoirement parmi toutes les séquences qui n'intersectent pas avec les séquences positives pour former la classe négative. Dans les expériences, les séquences font 1000bp et sont centrées sur le TSS (site de démarrage de la transcription) ou sur les enhancers identifiés par le projet FANTOM5 [103].

Le modèle TFcoop utilise l'hypothèse de coopération des facteurs de transcription pour prédire la fixation. Comme on l'a vu au Chapitre 1, les facteurs de transcription agissent de manière combinée pour contrôler la transcription (voir section 1.2). Ces combinaisons mettent en œuvre plusieurs types de mécanismes : interactions physiques directes entre TF, interactions indirectes par le biais de la molécule d'ADN, concurrences pour l'occupation d'un site de fixation commun, etc... Afin de prédire la fixation du facteur de transcription cible, TFcoop utilise la totalité des PWM présentes dans la base de données JASPAR comme variables prédictives. Cela consiste à scanner toutes les séquences avec toutes les PWM disponibles et de retenir le score maximum dans chaque séquence pour chaque PWM.

En plus de ces variables le modèle TFcoop utilise également les taux des 16 dinucléotides dans chaque séquence. Ces variables sont intéressantes à considérer parce qu'il a été montré que la composition nucléotidique des séquences autour du site de fixation apporte une information importante sur la fixation des TF [104]. Il est donc intéressant d'inclure cette information pour améliorer la précision du modèle. De plus, du fait de son importance, il est nécessaire d'inclure cette information directement dans le modèle, pour éviter de la capturer de manière indirecte par le biais de PWM enrichies pour les nucléotides ou dinucléotides clefs. Par exemple, si les séquences fixées par le TF d'intérêt sont plutôt riches en CG, le modèle pourrait capturer cette information en combinant les

scores de toutes les PWM de la base JASPAR qui sont eux-mêmes riches en CG. Pour éviter cela, et s'assurer que les PWM sélectionnées dans le modèle reflètent réellement la présence de sites de fixation de cofacteurs du TF cible et ne sont pas là simplement pour capturer l'environnement nucléotidique large autour de ce TF, il est important de fournir directement au modèle les variables liées à cet environnement nucléotidique.

Ces deux types de variables sont intégrés dans un modèle de régression logistique pénalisé (LASSO, voir section 2.4.2) qui s'écrit de la façon suivante :

$$\ln \frac{\mathbb{P}(Y = 1 \mid X)}{1 - \mathbb{P}(Y = 1 \mid X)} = \beta_0 + \sum_{i=1}^{p} \beta_i X_i + \sum_{j=1}^{q} \gamma_j T_j, \tag{3.10}$$

avec $X = (X_1, \ldots, X_p, T_1, \ldots, T_q)$, X_i le score maximum du ième PWM dans la séquence, T_j le taux du jème dinucléotide dans la séquence et β_i et γ_j les coefficients de la régression linéaire.

La Figure 3.5 représente sous forme de violon plots les distributions des AUROC de différentes méthodes appliquées à 409 données de ChIP-seq. Sur cette figure on peut voir que TFcoop obtient de meilleures performances que la PWM de JASPAR seule, TRAP et DNA-shape pour prédire la fixation des facteurs de transcription. Les auteurs ont appliqué cette méthode sur des promoters et des enhancers et ont pu montrer que les règles gouvernant la fixation des facteurs de transcription étaient différentes suivant les types de séquences, comme cela a également été montré dans la référence [105]. Surtout, TFcoop démontre l'importance de l'information de coopération des facteurs de transcription pour expliquer leur fixation. Ces résultats nous montrent que la fixation ne dépend pas uniquement du motif du TF cible, mais également des autres cofacteurs présents, et de l'environnement nucléotidique de la séquence. En outre, ce modèle soulève d'autres questions, relatives notamment à l'organisation de la fixation des facteurs de transcription les uns par rapport aux autres c'est à dire leur ordre dans la séquence, la distance qui sépare chacun d'eux, etc... De même, dans leur étude, les auteurs de TFcoop ont montré que les règles de fixation n'étaient pas tout à fait les mêmes d'un type cellulaire à l'autre, mais ils n'ont pas étudié précisément ce qui différencie les sites de fixations de deux types cellulaires donnés. C'est sur ces deux thématiques que portent les contributions de cette thèse décrites dans la partie II du manuscrit.

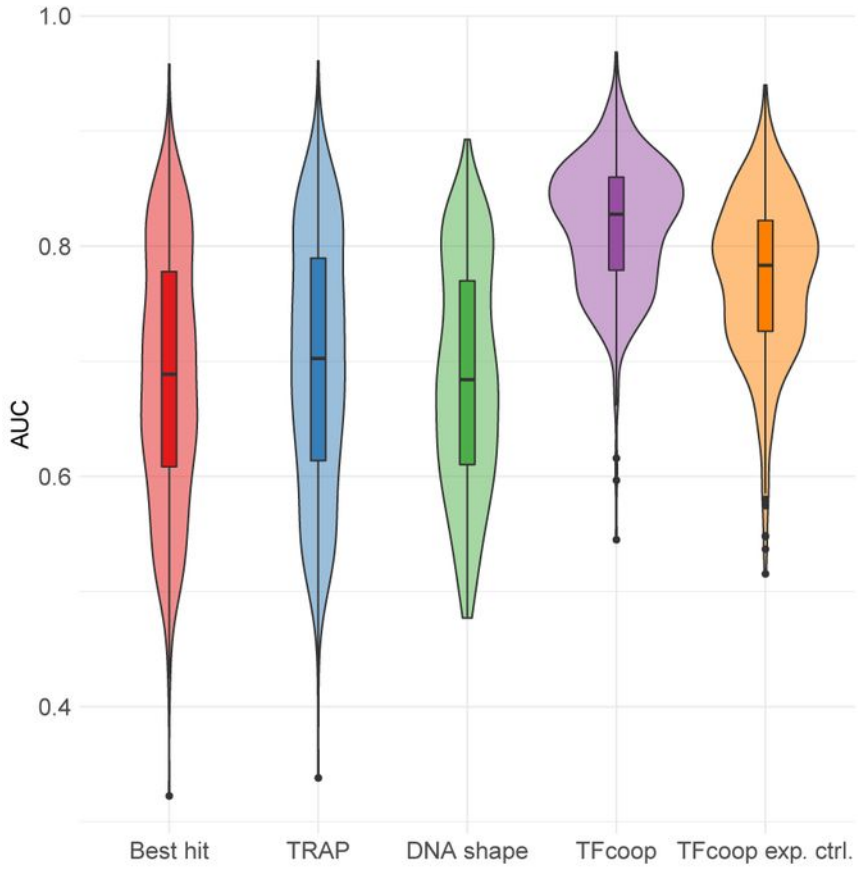

Figure 3.5 – *Distribution des AUROC de différentes méthodes appliquées à 409 données de ChIP-seq.*
Best-hit c'est à dire le modèle avec PWM de JASPAR (rouge), TRAP (bleu), DNAshape (vert), TFcoop (violet) et TFcoop appliqué à des gènes exprimés uniquement (orange). Adapté de *« Probing transcription factor combinatorics in different promoter classes and in enhancers »*, Vandel *et al.* BMC Genomics, 2019.

3.4 Données

3.4.1 Bases de données de motif

Il existe différentes bases de données de motif de fixation. On peut par exemple citer HOCOMOCO[106], CisBP[107], Factorbook[108], MethMotif[109] et JASPAR[110, 111].

La version v11 de HOCOMOCO intègre les modèles de fixation pour 680 TF chez l'homme et 453 chez la souris sous forme de PWM établies avec des données expérimentales issues de ChIP-seq et de HT-SELEX [112]. En plus des PWM classiques, HOCOMOCO contient aussi des diPWM.

MethMotif est une base de données de matrices de fixation, combinant à la fois les PPM, PWM et les données de méthylation CpG spécifiques aux types cellulaires. Les motifs de fixation proposés sont donc ici spécifiques aux types cellulaires (`https://methmotif.org/`).

Dans JASPAR, les motifs des sites de fixation des facteurs de transcription sont vérifiés manuellement. Différents types de modèles sont dispobibles (PFM, PPM, TFFM) dans 6 groupes taxonomiques différents, dont les vertébrés. Depuis ses débuts en 2004, JASPAR a été mis à jour à plusieures reprises. La dernière version de 2020 [111] comporte les PFM et PPM de 746 facteurs de transcription différents chez les vertébrés. Dans la partie contribution de cette thèse nous utiliserons la base de données JASPAR et plus particulièrement les PFM redondants des facteurs de transcription des vertébrés. Ce jeu de données correspond à la version 2020 de JASPAR et comporte 1011 motifs.

3.4.2 Unibind

UniBind [35] est une base de données d'interactions directes entre facteurs de transcription et l'ADN dans les génomes de neuf espèces différentes. Ces interactions sont déterminées à partir de jeux de données de ChIP-seq provenant de ReMap et GTRD. Les données de ChIP-seq sont ensuite analysées avec un algorithme basé sur l'entropie pour séparer les fixations directes des indirectes. Pour cela, les séquences +/-500bp autour des pics sont scannées avec la PWM associée au TF cible et optimisée avec DAMO [90] (voir section 3.2.2). Les scores maximums dans chaque séquence sont analysés avec un

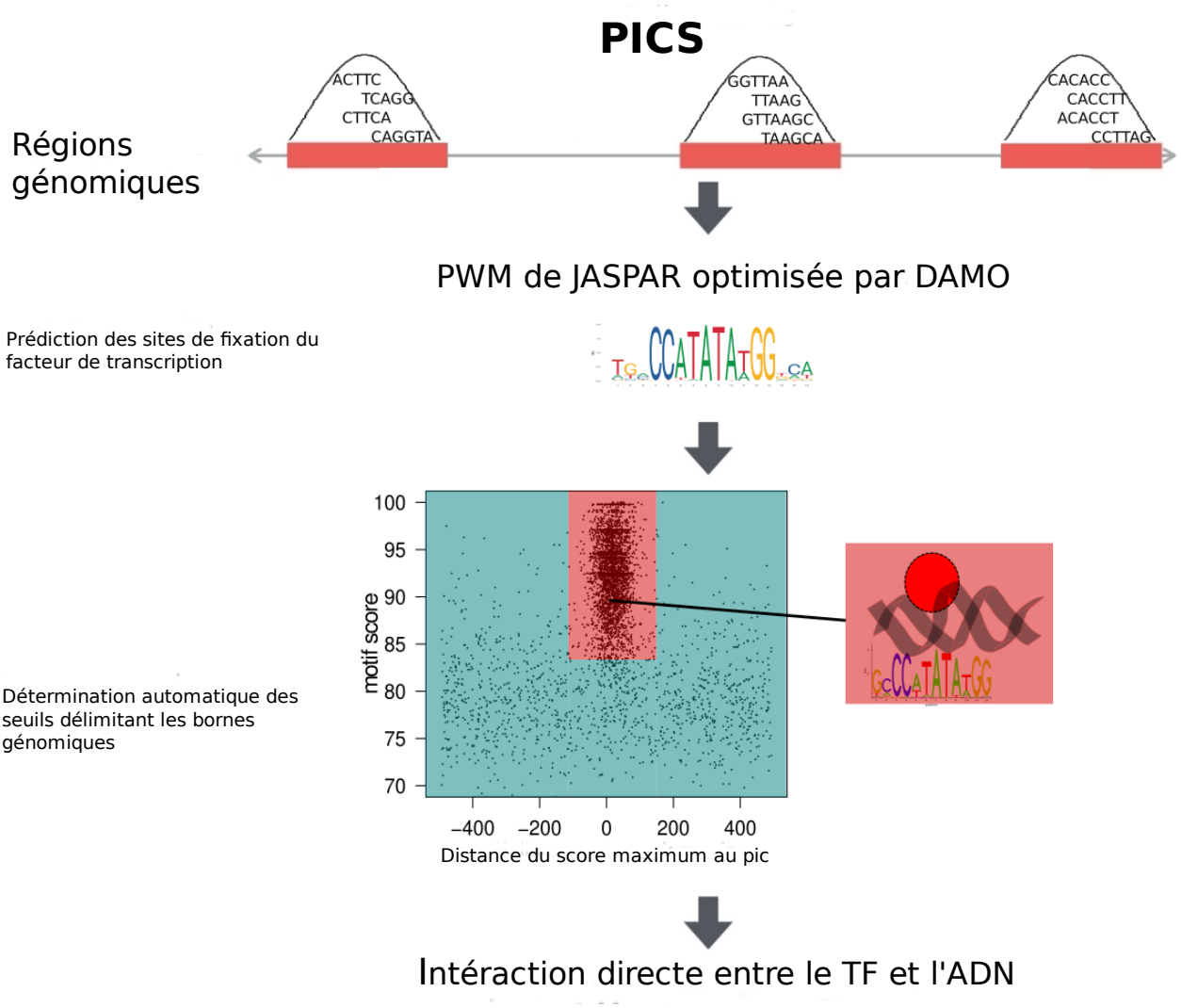

FIGURE 3.6 – *Procédure mise en place dans Unibind avec l'algorithme ChIP-eat pour déterminer les interactions directes entre le facteur de transcription et l'ADN.*
Figure adaptée de la documentation du site Unibind, `https://unibind.uio.no/docs/`.

algorithme basé sur l'entropie appelé ChIP-eat. Celui-ci permet de déterminer automatiquement des bornes génomiques au delà desquelles le site semble trop éloigné du pic pour être considéré comme un site fixé. Ces bornes génomiques définissent donc une zone d'enrichissement des interactions directes des facteurs de transcription avec l'ADN. Seuls les pics dont le score maximum est à l'intérieur de ces bornes sont donc conservés dans un premier temps. Dans un second temps, tous les pics dont le score maximum n'est pas dans la zone d'enrichissement sont réanalysés pour détecter si une sous-séquence avec un score non-maximum tombe dans la zone d'enrichissement. La Figure 3.6 est un schéma représentant la procédure de ChIP-eat décrite précédemment.

Dans Unibind, seules les données de ChIP-seq associées à des facteurs de transcriptions dont le motif est présent dans JASPAR sont conservées. Chaque donnée Unibind est donc une expérience de ChIP-seq restreinte à certains pics. Ces expériences sont associées à 2 seuils, un seuil de distance au pic et un seuil de score. Une p-valeur fournissant des informations sur la centralité des prédictions par rapport aux sommets des pics ChIP-seq est aussi associée à chaque expérience. Enfin dans Unibind une expérience est considérée comme robuste si (i) le motif optimisé par DAMO est suffisamment proche de la PWM originale présente dans JASPAR et si (ii) l'enrichissement des sites de fixation du facteur de transcription est centré autour des pics. Dans la suite on utilisera les données robustes et permissives et nous utiliserons la p-valeur associée pour ne conserver que des expériences dont la zone d'enrichissement est significative.

Utiliser les données Unibind permet de filtrer les expériences de ChIP-seq afin de ne conserver que des séquences vraisemblablement fixées directement, ce qui permet de s'assurer qu'un bon score du motif associé au facteur de transcription ciblé est une vraie occurrence et non un faux positif. Cette qualité sera primordiale dans la suite de notre analyse, où nous souhaitons notamment étudier finement les sites de fixation des facteurs de transcription ciblés.

3.5 DExTER

La méthode DExTER a pour objectif d'étudier les liens entre la composition du génome dans certaines parties des régions promotrices et le niveau d'expression des gènes. Plus précisément, DExTER[113] permet d'identifier des paires (k-mers, régions) dans lesquelles la fréquence du k-mer dans la région est corrélée à l'expression des gènes. La méthode se base sur une procédure itérative d'exploration de l'espace des paires (k-mers, régions) possibles. Les paires identifiées sont ensuite utilisées comme variables prédictives pour prédire l'expression des gènes.

La méthode DExTER prend en entrée un ensemble de séquences (une par gène) alignées sur un point d'intérêt, et un vecteur d'expression des gènes. Dans la plupart des expériences réalisées, le TSS est utilisé comme point d'alignement des séquences mais il est possible de considérer d'autres points d'alignement (début/fin du gène, bornes introns/exons etc...). La méthode présente deux grandes étapes dans la procédure. La première étape consiste à identifier les paires (k-mer, région) pour lesquelles la fréquence du k-mer dans la région est corrélée à l'expression. Pour cela, les auteurs ont développé une procédure d'exploration itérative basée sur l'optimisation de la corrélation. La seconde étape consiste à sélectionner les meilleures paires identifiées pour apprendre un modèle prédicteur de l'expression. Pour cela, les variables sont utilisées dans un modèle de régression linéaire pénalisé en norme L1 (LASSO, voir section 2.4).

3.5.1 Critère d'optimisation et procédure d'exploration

Dans DExTER, la corrélation de Pearson est utilisée comme critère d'optimisation. Le coefficient de corrélation ρ de Pearson est un critère souvent utilisé pour mesurer les performances d'un prédicteur [114] [115]. Le coefficient de Pearson permet de détecter des relations linéaires entre variables. Dans cette méthode, il est calculé de la manière suivante :

$$\rho(D_{k,r}, Y) = \frac{Cov(D_{k,r}, Y)}{\sigma_{D_{k,r}} \sigma_Y} \tag{3.11}$$

où Cov est la fonction de covariance, $D_{k,r}$ est un vecteur de fréquences du k-mer k dans la région r pour chaque gène, Y est le vecteur du niveau d'expression associé à chaque gène, et σ est l'écart type.

L'objectif de la méthode est alors d'explorer l'espace des k-mers ainsi que l'espace

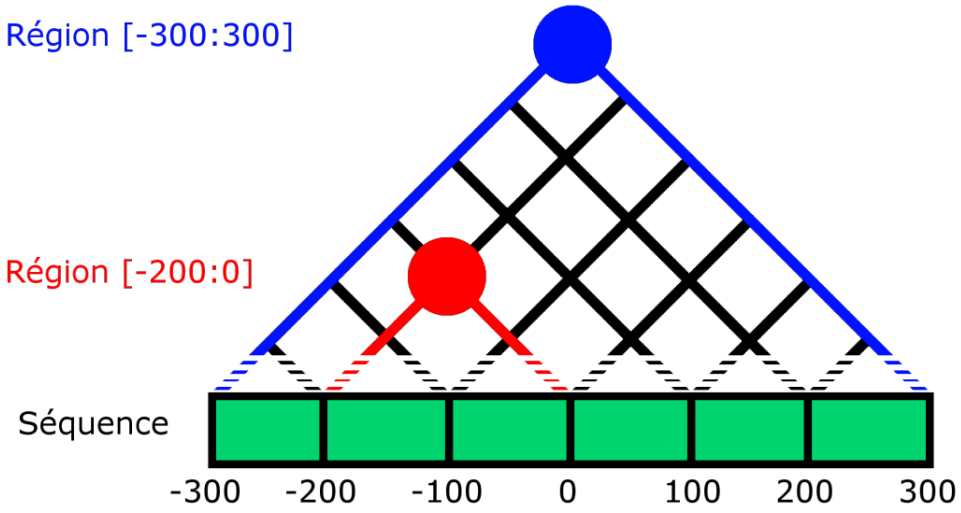

Figure 3.7 – *Exemple de demi-treillis.*
La séquence est découpée en sous-régions unitaires et on définit une structure de demi-treillis représentant les différentes régions considérées par DExTER. Chaque nœud du treillis correspond à une sous-région ou l'union de plusieurs sous-régions. Le sommet correspond à la séquence entière. Figure adaptée de la thèse de Christophe Menichelli *« Méthodes de découverte de nouveaux domaines dans les séquences biologiques : application à Plasmodium falciparum »* soutenue en 2019.

de toutes les régions possibles afin de maximiser le critère d'optimisation. Pour ne pas considérer naïvement tous les k-mers possibles dans toutes les régions possibles, DExTER utilise une heuristique. Celle-ci permet de réduire à la fois de nombre de sous-séquences et le nombre de k-mer à explorer. Premièrement, l'espace des régions possibles est réduit en procédant à un découpage des séquences en n régions unitaires disjointes. Les auteurs ont ensuite utilisé une structure en treillis où chaque nœud correspond à une région (voir Figure 3.7). En mathématiques, un treillis est une structure ordonnée dans laquelle chaque paire d'éléments a une borne supérieure et une borne inférieure. Ici les éléments sont des régions et la relation d'ordre considérée est l'inclusion. Formellement, on ne parle donc pas de treillis mais de demi-treillis, car chaque paire de région (R_i, R_j) est associée à une borne supérieure qui est la région minimale contenant R_i et R_j. La base (ou étage 0) du demi-treillis correspond aux n régions unitaires disjointes définies. Plus l'étage dans le demi-treillis est grand, plus les régions considérées sont grandes. Le sommet correspond donc à la séquence entière.

Une fois les séquences découpées en régions, la méthode se présente alors en l'alternance de deux phases : une phase de segmentation et une phase d'expansion. En prenant l'ensemble des dinucléotides comme point de départ, la méthode recherche les régions d'intérêt de chaque k-mer (phase de segmentation). Ensuite, sur chacune de ces régions d'intérêt, la méthode étend le k-mer considéré dans les huit (k+1)-mers possibles et elle évalue leurs performances (phase d'expansion). Les (k+1)-mers qui montrent de meilleures performances que le k-mer original sont sélectionnés pour continuer l'exploration, et les phases de segmentation et d'expansion sont appliquées récursivement. Cette façon de faire permet de focaliser la méthode sur l'exploration des k-mers qui semblent porter le plus d'information et d'ignorer les k-mers peu ou pas informatifs. Une fois que la phase d'expansion ne permet plus d'augmenter le critère d'optimisation, la procédure s'arrête. Enfin la méthode renvoie la liste des paires (k-mers, régions) retenues. Il est alors possible de construire une matrice de variables prédictives $N \times p$ contenant les taux de chaque k-mer identifié dans chaque région associée pour chaque gène (chaque séquence), avec N le nombre de gènes et p le nombres de paires (k-mers, régions) identifiées.

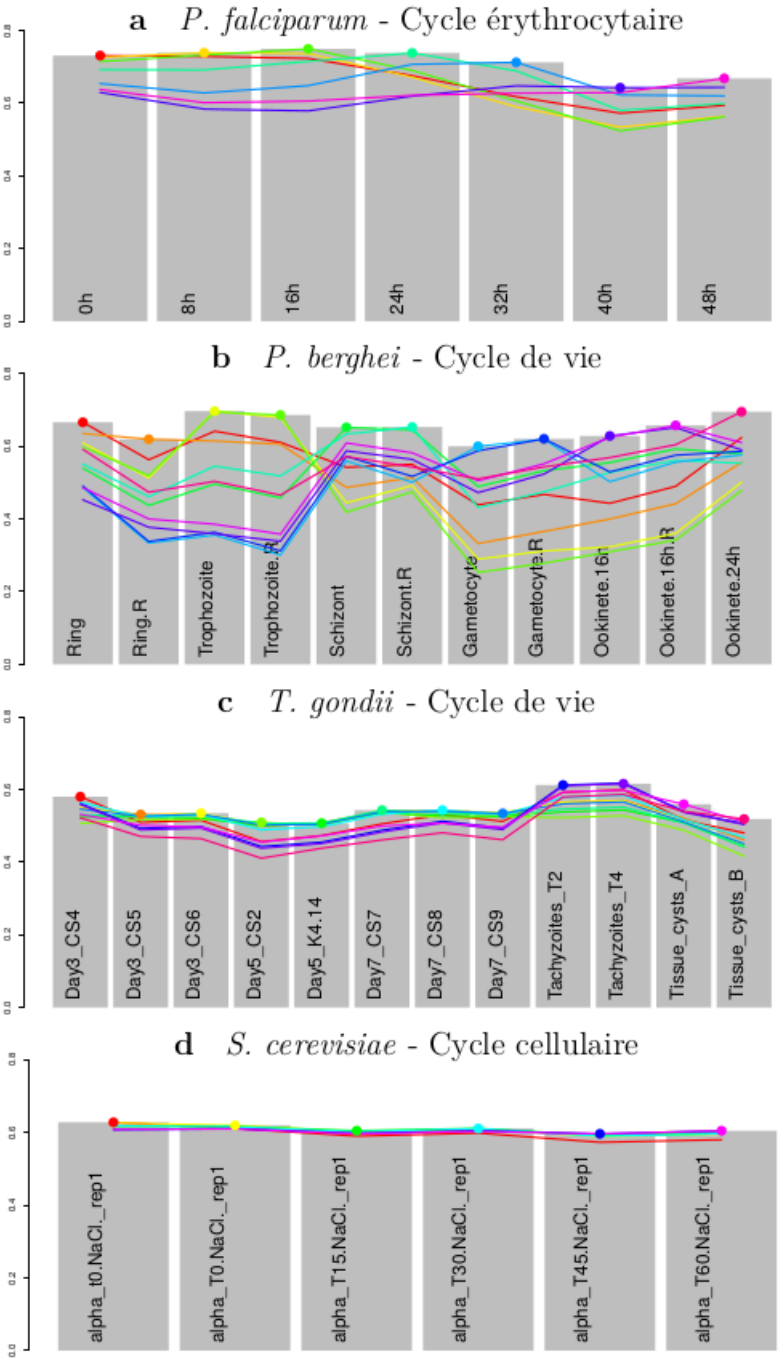

Figure 3.8 – *Exemple de résultats de DExTER pour prédire l'expression des gènes codants chez différentes espèces.*
Les barres grises représentent la corrélation entre l'expression prédite et l'expression observée des prédicteurs appris dans différentes conditions. Les courbes colorées représentent la précision obtenue lorsqu'on utilise un prédicteur appris sur une condition spécifique pour prédire les autres conditions de la même série. L'échelle des ordonnées est $[0; 0, 8]$. Figure adaptée de la thèse de Christophe Menichelli *« Méthodes de découverte de nouveaux domaines dans les séquences biologiques : application à Plasmodium falciparum »* soutenue en 2019.

3.5.2 Expériences

Les auteurs ont appliqué cette approche sur différents jeux de données ciblant différentes espèces. Ils observent de bons résultats, plus particulièrement sur deux espèces de *Plasmodium* et d'autres espèces eucaryotes. Suivant les espèces, la méthode identifie différentes grandes régions (plusieurs dizaines ou centaines de bp) dans lesquelles la fréquence de certains k-mers est très corrélée avec l'expression des gènes. Ils ont émis l'hypothèse que ces longues séquences biaisées pourraient constituer une nouvelle classe d'éléments régulateurs, qu'ils nomment domaines de régulation, et qui sont différents des sites de fixation classiques des facteurs de transcription. Les modèles appris permettent de prédire l'expression avec une précision comprise entre 50% et 60% suivant les espèces. Dans les deux espèces de Plasmodium considérées cette précision dépasse les 70%, ce qui semble indiquer que ces longs éléments régulateurs ont un rôle prédominant dans ces espèces. Certains résultats d'expériences, dont les Plasmodiums, sont présentés en Figure 3.8.

3.5.3 Classification avec DExTER

DExTER était jusqu'à présent adapté uniquement pour faire de la régression linéaire. Il était donc impossible d'étudier des problèmes de classifications avec cette approche. En travaillant avec Christophe Menichelli, qui a développé la méthode DExTER, nous proposons ici une version de DExTER applicable à des problèmes de classification. Au lieu d'utiliser la corrélation ρ de Pearson, cette version utilise l'AUROC comme critère. Cette version optimisée pour la classification identifie donc des paires (k-mers, régions) maximisant l'AUROC. Le but étant de pouvoir utiliser l'heuristique de DExTER dans différents types de problèmes.

Nous utilisons alors les variables dans une régression linéaire logistique pénalisée qui s'écrit :

$$\ln \frac{\mathbb{P}(Y = 1 \mid D)}{1 - \mathbb{P}(Y = 1 \mid D)} = \delta_0 + \sum_{j=1}^{p} \delta_j D_j, \tag{3.12}$$

avec δ_j les paramètres du modèle, $D = (D_1, \ldots, D_p)$ l'ensemble des variables identifiées par DExTER, et D_j la fréquence du j-ème k-mer dans la région qui lui est associée.

Deuxième partie

Contributions

Chapitre 4

Travail préliminaire

Dans ce chapitre nous allons détailler le travail préliminaire effectué conjointement avec une équipe de l'IGMM (Institut de génétique moléculaire de Montpellier) et décrit dans l'article Bejjani *et al.* « Fra-1 regulates its target genes via binding to remote enhancers without exerting major control on chromatin architecture in triple negative breast cancers » publié dans Nucleic Acid Research en 2021 [116].

Les facteurs de transcription Fra-1 (FOSL1) et Fra-2 (FOSL2) font partie de la famille de facteur AP-1 [117] (« Activator Protein 1 »). Cette famille peut être distinguée en deux sous-familles, JUN et FOS. Les membres de la sous-famille JUN peuvent s'hétérodimériser ou s'homodimériser entre eux alors que les FOS doivent s'hétérodimériser avec les JUN. Les membres de la famille AP-1 sont impliqués dans beaucoup de fonctions cellulaires et physiques. Par conséquent, leur dérégulation est impliquée dans certaines pathologies comme l'asthme [118], le psoriasis [119] ou certains cancers [120, 121, 122, 123]. Fra-1 et Fra-2, de la sous-famille FOS, se fixent sur le site de fixation très courant pour cette famille : TGANTCA. Toutes les versions de leurs PPM modélisant leur fixation nommées MA0477 FOSL1 et MA0478 FOSL2 dans JASPAR sont donc très similaires. Elles sont aussi très similaires à d'autres motifs modélisant des facteurs de transcription de la même famille tels que FOS, NFE2, JUN, BACH1 ou BACH2. La fixation de ces facteurs de transcription sur le génome est pourtant étroitement contrôlée pour permettre à chacun de remplir des fonctions différentes [116]. Nous avons donc voulu sonder ces préférences de fixation directement au niveau de la séquence.

Rang	Variables sélectionnées	Coefficients associés
1	MA0841.1 NFE2	-4,13
2	MA1135.1 FOSB : :JUNB	-10,70
3	MA0491.1 JUND	-0,08
4	GC	0,11
5	ApT	11,63
6	TpA	7,98
7	MA1130.1 FOSL2 : :JUN	-3,27
8	CpG	-0,07
9	MA0476.1 FOS	-1,85
10	MA0873.1 HOXD12	0,07

Table 4.1 – *Table des 10 premières variables sélectionnées par le LASSO dans la méthode TFcoop sur la classification Fra-1/Fra-2. Ces variables sont triées par ordre d'importance dans le modèle, la valeur des coefficients est obtenue dans un modèle LASSO avec λ choisi pour ne conserver que 10 coefficients non-nuls.*

4.1 Données et modèles

Nous disposons de données de ChIP-seq donnant la position des pics pour Fra-1 et Fra-2 dans un même type cellulaire (MDA-MB-231). Les expériences de ChIP-seq comportent 7 556 pics pour Fra-1 et 10 459 pics pour Fra-2. Nous restreignons l'analyse aux pics fixés uniquement par l'un des deux facteurs de transcription. Ce qui laisse 1 649 séquences fixées uniquement par Fra-1 et 4 557 séquences fixées uniquement par Fra-2. Enfin nous équilibrons les classes en tirant uniformément et sans remise 1649 séquences fixées par Fra-2. Nous étudions les séquences ($n = 3298$) de 1001bp centrées autour des sommets des pics fixés par Fra-1 ou Fra-2 strictement. Nous cherchons à distinguer ces pics grâce à la séquence en utilisant la méthode TFcoop détaillée en section 3.3.6. Cette méthode utilise les scores maximums dans les séquences de toutes les PWM disponibles dans JASPAR et le taux de dinucléotides dans les séquences dans un modèle de régression linéaire logistique pénalisé. Les séquences fixées par Fra-1 seront les séquences positives (Y=1) et celles fixées par Fra-2 les séquences négatives (Y=0).

Le modèle TFcoop obtient une bonne AUROC pour cette classification (AUROC = 0.80). Cependant l'analyse des variables utilisées par le modèle montre que ce n'est pas l'information des cofacteurs qui est utile pour distinguer les classes de séquences. Nous avons particulièrement étudié les 10 premières variables sélectionnées, ce qui correspond

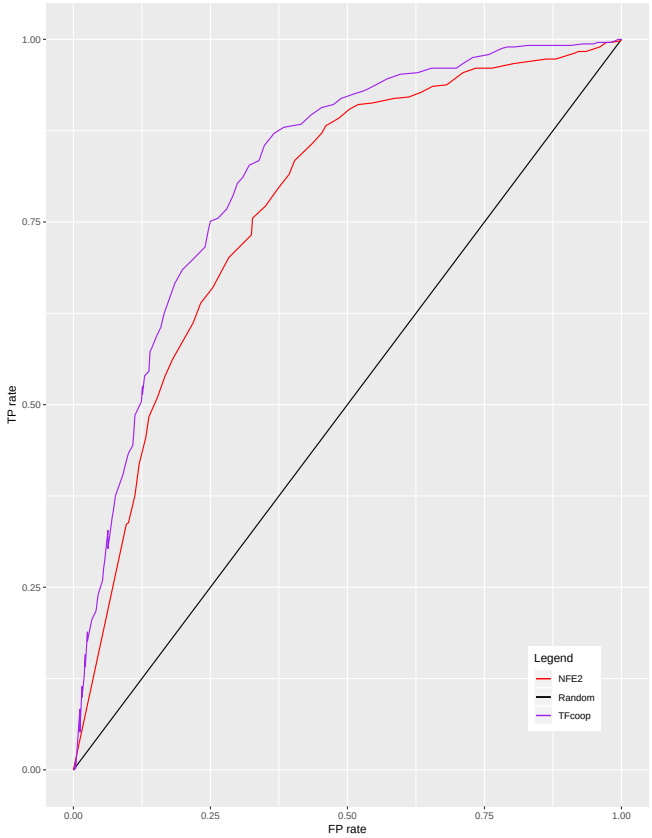

Figure 4.1 – *Courbes ROC.*
Performances du modèle TFcoop (en violet) et du modèle MA0841.1 NFE2 (en rouge) obtenu avec Best-hit.

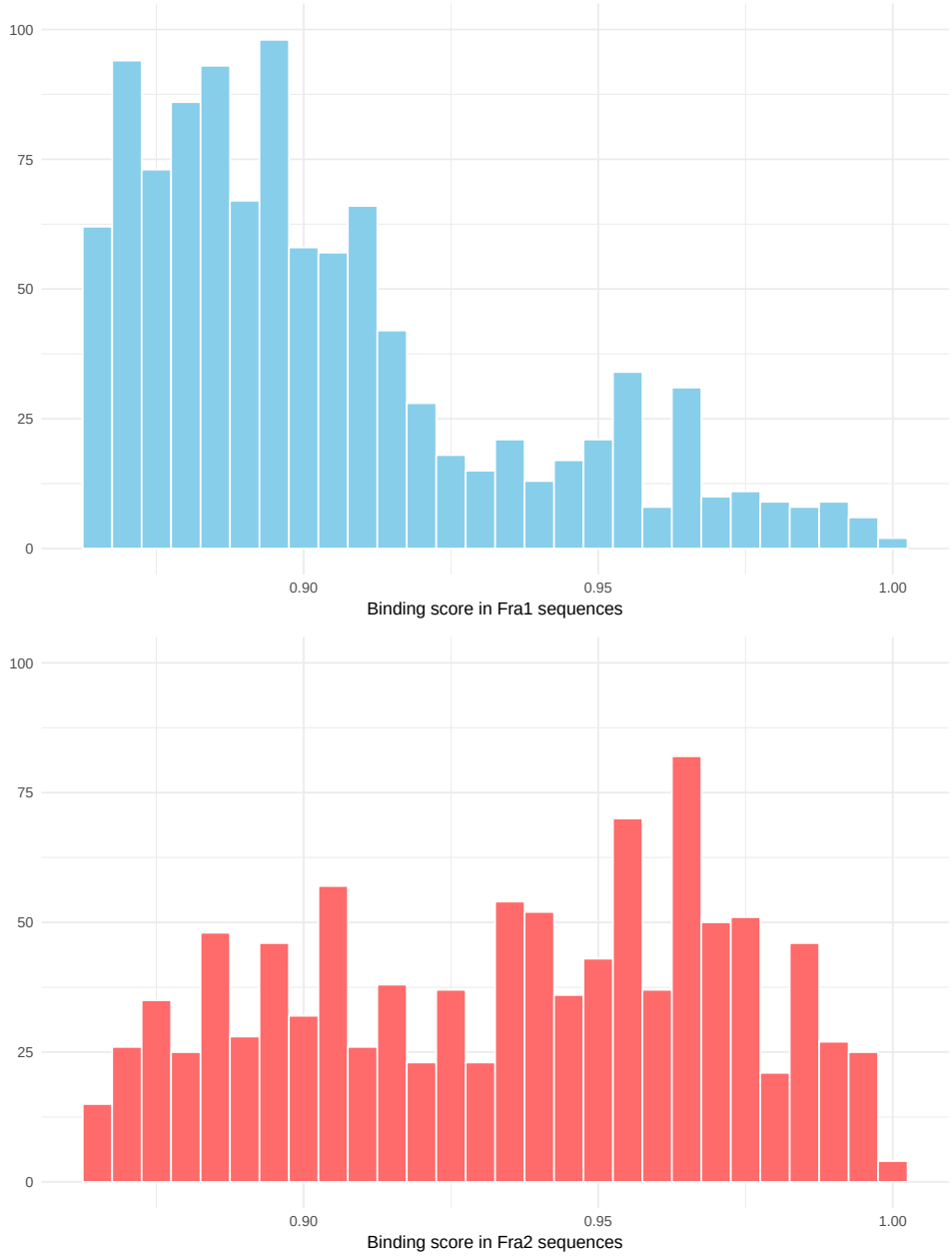

FIGURE 4.2 – *Histogrammes des distributions des scores maximums.* Scores obtenus avec MA0841.1 NFE2 dans les séquences fixées par Fra-1 (en bleu) et Fra-2 (en rouge).

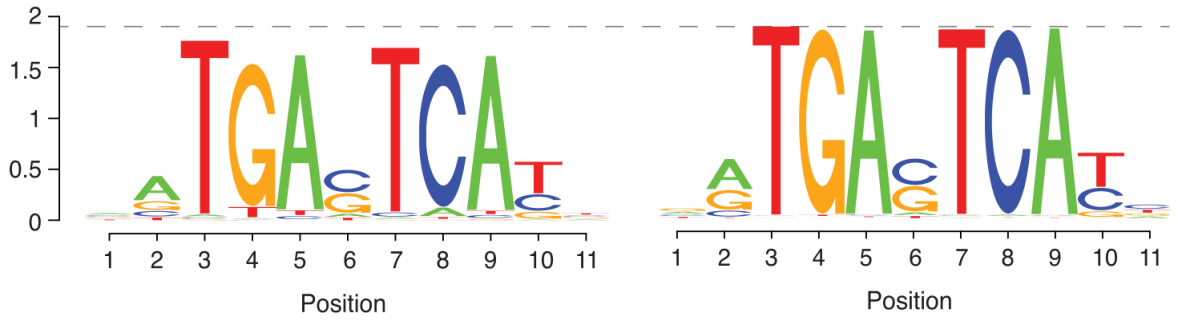

FIGURE 4.3 – *Représentations logo des PPM reconstruites à partir des données en utilisant le motif MA0841.1 NFE2.*
A gauche le logo de la PPM obtenue sur les séquences fixées par Fra-1 et à droite sur les séquences fixées par Fra-2.

aux variables obtenues avec un modèle LASSO dont λ est grand et ne permet que 10 coefficients non nuls. Ces variables sont les scores maximums dans les séquences obtenus avec des PWM de la famille AP-1 et des taux de dinucléotides. La Table 4.1 montre ces 10 variables sélectionnées ainsi que leurs coefficients dans le modèle LASSO. Pour obtenir le rang d'importance de ces variables nous retirons la variable du modèle (coefficient fixé à 0) sans ré-estimer les coefficients et calculons l'AUROC du modèle obtenu. Nous réalisons cela avec chacun des coefficients et classons les variables en fonction de la perte d'AUROC, le rang 1 est la variable traduisant la perte la plus importante. On peut voir sur cette table que MA0841.1 NFE2, le motif modélisant la fixation de NFE2, un facteur de transcription de la famille AP-1, est la variable la plus informative. De plus le coefficient de cette variable est négatif indiquant que ce motif est préférentiellement présent dans les séquences fixées par Fra-2. Cette variable seule permet de classer les pics Fra-1 et Fra-2 avec une AUROC égale à 0.75 (Figure 4.1) indiquant que ce motif est suffisant pour distinguer les deux classes de pics. Les modèles obtenus avec la méthode Best-hit utilisant uniquement soit la PWM MA0477.1 FOSL1 soit MA0478.1 FOSL2 ont eux aussi des AUROC proches de celle obtenue avec le motif de NFE2. Ce résultat suggère que le motif AP-1 (quelque soit sa forme/PWM) est suffisant pour distinguer les pics Fra-1 des pics Fra-2. Nous avons donc étudié plus en détail les distributions de scores des motifs AP-1 dans les deux classes (en utilisant la PWM MA0841.1 NFE2). Ces distributions sont représentées en Figure 4.2. Il semble en effet, que la classe Fra-2 comporte plus de bons scores que la classe Fra-1, expliquant donc les performances obtenues avec les motifs AP-1. Enfin, pour mettre en évidence les différences des motifs de fixation des deux fac-

teurs de transcription, nous avons reconstruit des PPM pour Fra-1 et Fra-2 à partir des données. Ces PPM sont reconstruites à partir du motif MA0841.1 NFE2 en utilisant les occurrences Best-hit de celle-ci dans chaque séquence. Les deux logos de ces PPM sont montrés en Figure 4.3. Nous pouvons voir sur ces logos que la quantité d'information aux positions caractéristiques des AP-1 (TGANTCA) est supérieure dans la PPM Fra-2. Cela implique que Fra-2 est plus « strict » quand il se fixe sur une occurrence, c'est à dire qu'il ne se fixe que sur des occurrences très similaires à son motif de fixation alors que Fra-1 est plus permissif.

4.2 Discussion

D'un point de vue statistique, comparer les séquences fixées par deux facteurs de transcription avec des motifs aussi similaires est proche d'une comparaison de séquences fixées par un même facteur de transcription. Il semblait donc difficile de pouvoir distinguer les classes de séquences fixées par Fra-1 et Fra-2 à partir de leur motif de fixation, car ces derniers sont très similaires. Nous avons cependant mis en évidence des préférences dans la fixation de ces deux facteurs de transcription.

Il est possible que les spécificités observées ne soient valables que pour un type cellulaire particulier. Cela pourrait expliquer pourquoi MA0477.1 FOSL1 et MA0478.1 FOSL2 ne reflètent pas cette spécificité car ces motifs ont été appris dans un autre type cellulaire. De plus, nous étudions ici les séquences fixées spécifiquement par l'un ou l'autre des facteurs de transcription alors que pour construire les PPM/PWM présentes dans la base de données JASPAR toutes les séquences de chaque ChIP-seq sont utilisées.

Cette idée de spécificités de fixation entre deux conditions *a priori* similaires ou proches, peut être étendue à la fixation d'un facteur de transcription entre deux types cellulaires, par exemple. De même qu'il serait possible d'observer des spécificités en fonction du type de traitement utilisé dans l'expérience de ChIP-seq. C'est ce à quoi nous nous intéresserons dans la suite de cette thèse, plus particulièrement sur la problématique de fixation entre types cellulaires.

En plus de montrer ces différences, ce travail préliminaire a permit de mettre en évi-

dence l'importance d'étudier particulièrement le motif du facteur de transcription d'intérêt dans ce type de problème. D'autant plus si celui-ci semble *a priori* se trouver de façon identique dans les deux classes de séquences.

Chapitre 5

Problématique

5.1 Spécificités de fixation entre types cellulaires

Dans la suite, nous allons nous intéresser à la fixation des facteurs de transcription dans différents types cellulaires chez l'homme. Nous avons vu au chapitre 1, que les sites de fixation associés à un TF donné varient d'un type cellulaire à l'autre. Nous allons donc, nous intéresser à ces différences afin de déterminer les spécificités génomiques pouvant expliquer les variabilités de fixation entre deux types cellulaires différents.

En nous basant sur les résultats du travail préliminaire présenté au chapitre 4, 3 types d'information génomique pouvant être impliqués dans ces différences ont été identifiés. Le premier type d'information concerne le motif de fixation lui même. Nous avons vu au chapitre 4 que ce motif semble être responsable de l'essentiel des différences de fixation observées entre les TF Fra1 et Fra2. Nous avons donc développé une méthode spécifiquement dédiée à l'identification des différences entre motifs de fixation. Cette méthode est présentée au chapitre 6. Le second type d'information concerne les co-facteurs du motif cible. Pour cela, nous avons proposé une extension de la méthode TFcoop, appelée Positional TFcoop, qui permet de prendre en compte la position des motifs de co-facteurs par rapport au motif cible. Cette méthode est présentée au chapitre 7. Enfin, le 3ème type d'information génomique, pouvant être impliqué dans la spécificité cellulaire de la fixation, que nous avons pris en compte concerne l'environnement nucléotidique large autour des sites de fixation. Ces spécificités nucléotidiques sont capturées grâce à la méthode DExTER présentée au chapitre 3. Ces trois types d'information sont intégrés dans un modèle global appelé TFscope et présenté au chapitre 8.

Dans la suite de ce chapitre, nous présentons les données utilisées pour notre étude, et nous définissons précisément le problème de classification qui nous servira de cadre dans les chapitres suivants.

5.2 Construction du jeu de données

5.2.1 Sélection des données

Afin d'augmenter la précision et de pallier les problèmes de fixation indirectes nous avons utilisé les données fournies dans Unibind (voir section 3.4.2). Les données Unibind sont composées de pics de ChIP-seq associés à une expérience visant un facteur de transcription dans un type cellulaire particulier, dans laquelle ont été utilisés différents traitements et conditions. Dans ce manuscrit nous nous intéressons plus particulièrement à la différence de fixation d'un facteur de transcription entre deux types cellulaires. Nous allons donc constituer des paires d'expériences ciblant un même facteur de transcription, avec un même traitement (mêmes conditions, ou mêmes traitements) mais dans deux types cellulaires différents.

5.2.1.1 Premiers filtres

Comme on l'a vu au chapitre 1, les données issues d'expériences de ChIP-seq ne sont pas toujours d'une grande précision. Pour cette raison, et comme expliqué en détail en section 3.4.2, les auteurs d'Unibind ont étudié la distance entre les pics de ChIP-seq et la position du site de fixation le plus probable (inféré avec la PWM associée au facteur de transcription étudié). Ils montrent que, parfois, le meilleur site de fixation est très éloigné du pic de ChIP-seq, et que ce pic est souvent un faux positif dans ce cas. En utilisant une méthode nommée ChIP-eat[35] les auteurs déterminent des bornes génomiques au delà desquelles les pics de ChIP-seq semblent être des faux positifs. En fonction du nombre de pics qui sont dans les bornes identifiées par ChIP-eat, Unibind associe une p-valeur reflétant la qualité de chaque expérience de ChIP-seq. Le premier filtre est basé sur cette p-valeur. Dans nos données, nous ne conservons que les expériences « robustes » et « permissives » (voir section 3.4.2) dont la p-valeur est inférieure à 1%, représentant ainsi 3600 expériences, dont 177 permissives.

De plus, parmi toutes les expériences disponibles dans la base de données Unibind, certaines ont été réalisées sur les mêmes TF, dans les mêmes types cellulaires, sous les mêmes conditions. Ces cas augmentent considérablement le nombre de paires possibles et rendent complexes les analyses biologiques. Nous souhaitons donc que chaque triplet (TF, type cellulaire, traitement) ne soit décrit que par une expérience de ChIP-seq. Nous avons

utilisé comme critère la p-valeur associée à l'expérience donnée par Unibind. Dans le cas de ces expériences « doublons », nous ne conservons que l'expérience dont la p-valeur est la plus petite, gardant ainsi la meilleure expérience au sens de cette mesure.

5.2.1.2 Sélection des paires d'expériences

À l'issu de ces premiers filtres, nous avons un total de 2 752 expériences de ChIP-seq représentant chacune un unique triplet (TF, type cellulaire, traitement). Nous voulons constituer des paires ayant le même couple (TF, condition) car nous souhaitons étudier les différences de fixation d'un TF entre types cellulaires en s'affranchissant des effets liés au traitement utilisé par l'expérimentateur. Cela représente un total de 25 573 paires possibles.

Toutes ces paires ne sont pas intéressantes à conserver dans l'analyse. Puisque l'on s'intéresse aux différences de fixation entre types cellulaires, nous allons sélectionner des paires montrant effectivement des pics de ChIP-seq différents. Nous utilisons pour cela une mesure de dissimilarité entre les expériences, basée sur la distance de Jaccard. L'indice de Jaccard est une mesure de similarité entre deux ensembles, définit comme le rapport entre le cardinal de l'intersection et le cardinal de l'union des ensembles considérés. Soit E et F deux ensembles, l'indice de Jaccard $J(E, F) \in [0; 1]$ est défini par

$$J(E, F) = \frac{|E \cap F|}{|E \cup F|}.$$

La distance de Jaccard est alors $1 - J(E, F)$.

L'intersection des pics est réalisée avec l'outil « window » de Bedtools [124], et l'union avec « merge », tous deux avec une fenêtre de 500bp en amont et en aval des pics. Grâce à cette mesure de distance nous réalisons un clustering hiérarchique de toutes les expériences partageant le même couple (TF, condition). Nous utilisons la méthode d'agrégation « lien complet », qui utilise la distance maximum entre deux groupes. Un exemple avec le couple (AR, sans traitement) est fourni en Figure 5.1.

Une fois les expériences regroupées par similarité, nous devons définir le seuil de dissimilarité au delà duquel nous considérons que deux expériences sont suffisamment diffé-

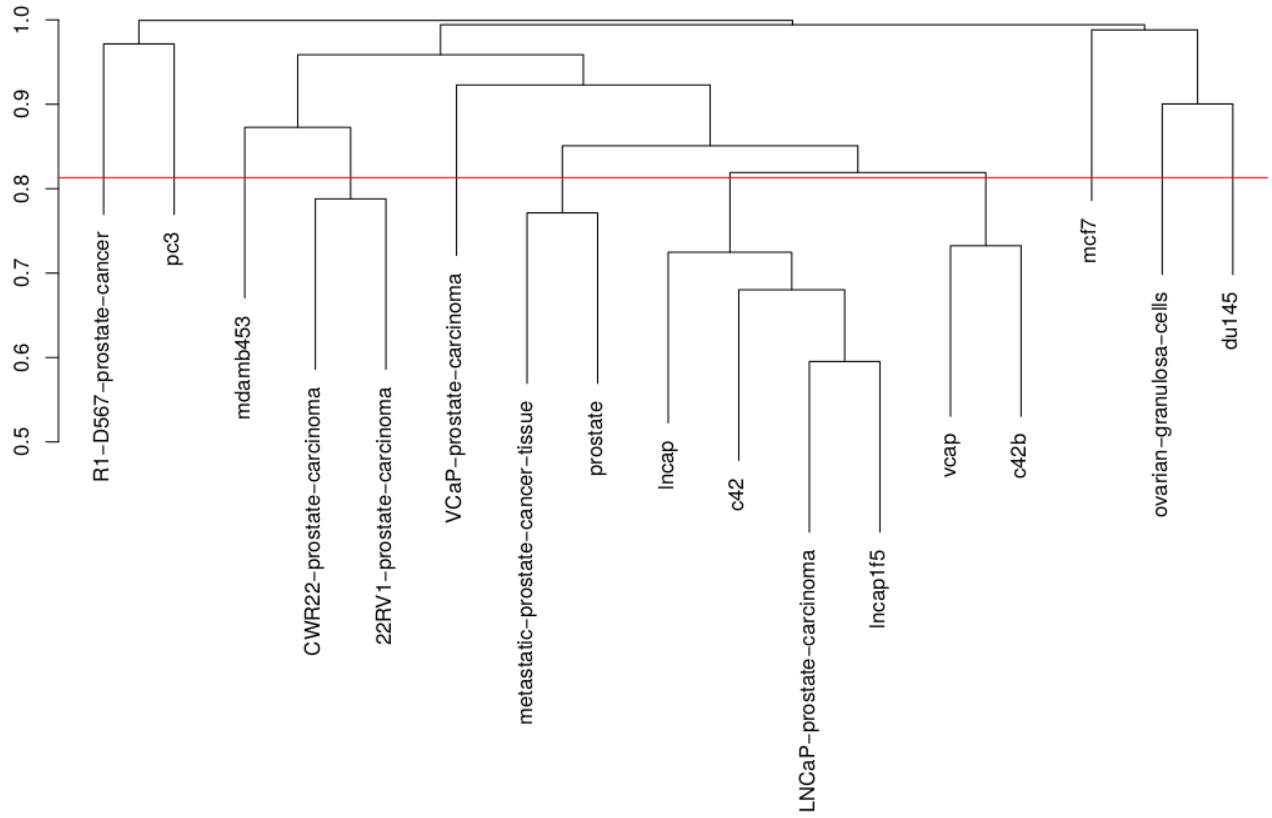

Figure 5.1 – *Exemple de dendrogramme regroupant par similarité les expériences.* Les expériences portent sur le facteur de transcription « AR » sans traitement.

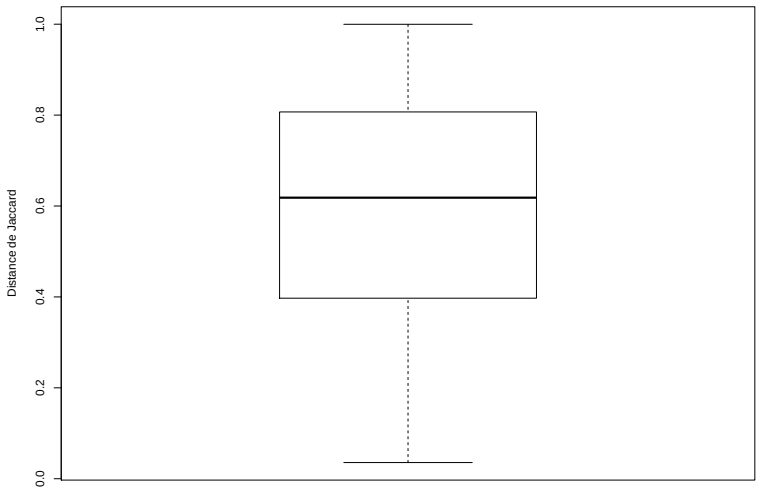

FIGURE 5.2 – *Boxplot de la distribution des distances de Jaccard*
Distances calculées entre les paires d'expériences associées au même triplet (TF, type cellulaire, traitement).

rentes pour être intéressantes à étudier dans le cadre de notre problématique. Pour cela nous avons calculé les distances de Jaccard correspondant aux paires d'expériences les plus similaires dont nous disposons *a priori*, c'est-à-dire les paires décrivant le même triplet (TF, type cellulaire, traitement) (voir section 5.2.1). La distribution de ces distances est représentée en Figure 5.2. Nous avons choisi le 3ème quartile (0.8131) de cette distribution comme seuil de dissimilarité. En utilisant ce seuil pour couper le dendrogramme des expériences associées à un couple (TF, traitement), on définit une partition de ces expériences. Par exemple, sur la Figure 5.1, le seuil choisi (en rouge) définit 8 clusters différents. Par définition, on sait que les paires d'expériences à l'intérieur d'un même cluster ont une distance inférieure au seuil sélectionné, et que deux expériences choisies dans deux clusters différents ont une distance supérieure au seuil.

Une fois tous les cluster identifiés nous sélectionnons une expérience représentative de chaque cluster. On utilise pour cela l'expérience comportant le plus de séquences, ce qui permet de conserver en priorité les données avec le plus d'exemples pour l'apprentissage des méthodes décrites dans les chapitres suivants. Après cette phase de clustering et le choix d'une expérience représentative de chaque cluster, le nombre de paires d'expériences

possibles est réduit à 1437. Pour réduire encore le nombre de paires à analyser, nous avons tiré parti de la structure du dendrogramme pour ne sélectionner que n paires différentes. On considère pour cela le dendrogramme obtenu en coupant le dendrogramme original au seuil sélectionné. Chaque feuille de ce dendrogramme correspond donc à un cluster d'expériences similaires, et chaque cluster est associé à une expérience représentative (celle qui comporte le plus de séquences). Chaque nœud interne de ce dendrogramme est utilisé pour définir une paire d'expériences. On considère pour cela le sous-arbre associé au nœud, et on sélectionne dans ce sous-arbre les deux expériences représentatives qui sont les plus dissimilaires. Cette façon de procéder permet de réduire grandement le nombre de paire à étudier tout en optimisant le nombre de séquence dans chaque jeu de données ainsi que la dissimilarité entre les expériences qui les constituent. À l'issue de cette phase, on dispose d'un total de 698 paires d'expériences.

Une fois les paires constituées, étant donné que l'on ne s'intéresse qu'aux différences entre types cellulaires et afin d'éviter d'avoir des séquences identiques entre les classes, nous nous limitons aux séquences uniquement fixées dans l'un ou l'autre des deux types cellulaires. On obtient ainsi deux classes de séquences : les séquences fixées seulement dans le type cellulaire 1, et les séquences fixées seulement dans le type cellulaire 2. Les séquences fixées dans les 2 types cellulaires sont écartées de l'analyse. Après cela, seules les paires d'expériences contenant plus de 1000 séquences par classe constitueront nos jeux de données dans les chapitres suivants, cela représente 502 paires d'expériences.

5.2.2 Alignement des séquences

À l'issue de l'étape de sélection décrite ci-dessus, on dispose donc de 2 classes de séquences issues de pics de ChIP-seq identifiés par Unibind comme étant vraisemblablement exempts de faux positifs. Il nous faut maintenant définir une règle d'alignement de ces séquences. Une façon simple de procéder serait de garder les séquences centrées sur le pic de ChIP-seq. Cependant, comme expliqué en sections 5.2.1.1 et 3.4.2, la précision de l'expérience de ChIP-seq est souvent faible. De plus, nous voulons pouvoir intégrer à nos analyses la position des co-facteurs du TF cible. Nous avons donc choisi d'aligner les séquences sur le site de fixation le plus probable du facteur de transcription ciblé identifié au voisinage du pic. En alignant toutes les séquences de la sorte, nous pouvons étudier plus facilement les positions relatives des facteurs de transcription coopérants avec le TF

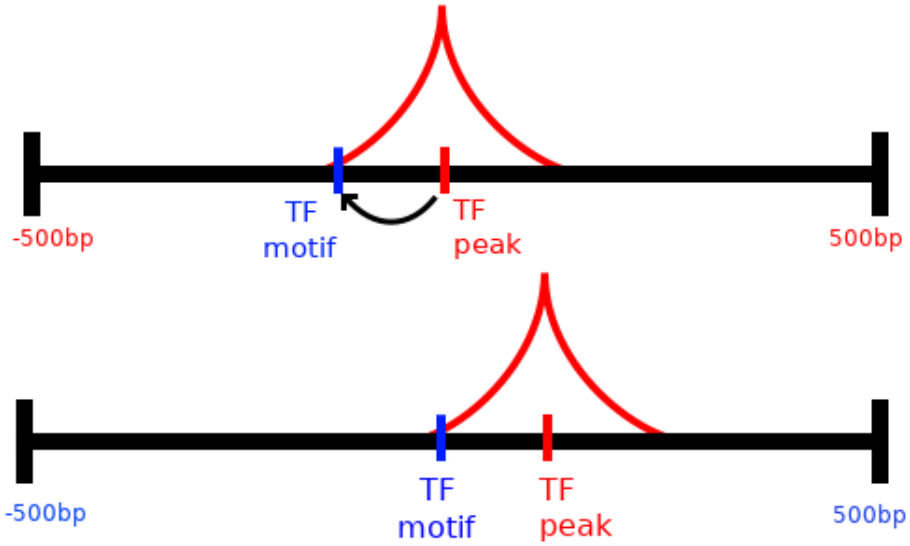

FIGURE 5.3 – *Schéma de réalignement des séquences.*
Le pic de ChIP-seq (en rouge) se trouve originellement au centre de la séquence de taille 1000bp. L'occurrence ayant le meilleur score de la PWM associée au TF devient le centre (en bleu).

ciblé.

En pratique, nous disposons de données BED indiquant la position du pic de ChIP-seq et nous savons qu'un bon score de PWM se trouve proche de celui-ci. Nous procédons alors à une recherche d'occurrences autour de ce pic, dans les bornes génomiques déterminées par ChIP-eat dans Unibind et en utilisant la PWM correspondant au TF ciblé par l'expérience et utilisée dans Unibind. Une fois cette recherche effectuée nous réalignons toutes les séquences sur l'occurrence ayant le score le plus haut dans cette région (Figure 5.3). L'Algorithme 2 détaille la procédure de réalignement.

5.2.3 Ensembles d'apprentissage, de test et équilibrage des classes

Pour chaque paire d'expériences sélectionnée, nous équilibrons le nombre de séquences dans chaque type cellulaire. Ceci permet d'éviter un déséquilibre trop important entre les classes, qui pourrait nuire au bon fonctionnement des algorithmes d'apprentissage. Pour ce faire, nous tirons uniformément et sans remise des séquences de la classe la plus grande

Algorithme 2 Procédure de réalignement

B_g = les bornes génomiques dans Unibind (en bp)
P_i = position du ième pic de ChIP-seq parmi les N présents dans les données
pour tout i de 1 à N **faire**
  $Z_i^0 = [P_i - B_g; P_i + B_g]$ séquence dans les bornes génomiques et centrée sur le pic
  S = scores des occurrences de la PWM associée à l'expérience dans Z_i^0
  $Smax = max(S)$ score maximum des occurrences de Z_i^0
  C_i = position de $Smax$, le nouveau centre de la séquence
  $Z_i^1 = [C_i - 500; C_i + 500]$ séquence finale
fin pour
return $Z = (Z_1^1, \ldots, Z_N^1)$

de façon à obtenir autant de séquences dans les deux classes.

Une fois les classes équilibrées les séquences sont partitionnées en 2 sous ensembles correspondant à l'ensemble de test (30% des séquences) et l'ensemble d'apprentissage (70% des séquences). Dans la suite, les différentes composantes de l'apprentissage (*i.e.* la sélection de variables, l'identification des régions, l'estimation des coefficients des modèles linéaires, etc.) sont réalisées sur l'ensemble d'apprentissage. L'ensemble de test est dédié uniquement au calcul des performances des modèles (AUROC, voir section 2.3.2).

Chapitre 6

Motif Discriminant

Le travail préliminaire détaillé en section 4 nous montre qu'une PWM peut permettre de distinguer avec une bonne précision (AUROC = 0,75) des classes de séquences fixées par deux facteurs de transcription dont les motifs sont très proches. Cependant, comme expliqué en partie 3.1 et 3.2.2, les PWM que l'on trouve dans les bases de données de motifs classiques sont apprises par maximum de vraisemblance, à partir d'un ensemble de sites de fixation donné. Elles ne sont pas créées pour discriminer deux ensembles de séquences, mais pour décrire au mieux les séquences correspondantes aux sites de fixation observés. Il semble alors possible d'obtenir de meilleurs résultats dans un cadre de classification comme le notre avec des motifs appris spécifiquement pour discriminer deux ensembles de séquences donnés. Dans cette partie nous détaillons l'approche que nous proposons pour l'apprentissage de motifs discriminants (appelée DM) et en quoi elle diffère des approches alternatives aux PWM existantes (présentées en section 3.3).

6.1 Présentation du modèle

D'après la procédure de réalignement de séquence détaillée en 5.2.2, nous savons que le meilleur score de la PWM associée au TF étudié se trouve au centre de chacune des séquences. Nous allons donc nous concentrer sur les k-mers situés à cette position. Nous obtenons ainsi deux ensembles de k-mers de taille K (K étant la taille du motif), et l'objectif est de construire un classifieur capable de discriminer ces deux classes simplement sur la base de leurs séquences de nucléotides.

Soit Z un de ces K-mers. On le représente par $4 \times K$ variables booléennes $Z_{k,j}$, avec $k \in \{1, \ldots, K\}$ et $j \in \{1 \ldots 4\}$. La variable j donne le rang des nucléotides $\{A, C, G, T\}$, avec, par convention, $j = 1 \Rightarrow A$, $j = 2 \Rightarrow C$, $j = 3 \Rightarrow G$ et $j = 4 \Rightarrow T$.

$$\text{On a alors } Z_{k,j} = \begin{cases} 1 & \text{si le nucléotide } j \text{ est observé en position } k \text{ dans le K-mer } Z \\ 0 & \text{sinon} \end{cases}$$

Notre approche de motif discriminant consiste à construire un modèle de régression logistique sur les variables $Z_{k,j}$. Ce modèle s'écrit classiquement

$$\ln \frac{\mathbb{P}(Y = 1 \mid Z)}{1 - \mathbb{P}(Y = 1 \mid Z)} = \alpha_0 + \sum_{\substack{k=1\ldots K \\ j=1\ldots 4}} \alpha_{kj} Z_{kj}, \tag{6.1}$$

avec α_0 et α_{kj} les paramètres du modèle. Ce modèle est de plus pénalisé en norme L1 (LASSO [47]) afin de réaliser une sélection de variable, et faciliter son interprétation.

6.2 Lien avec les PWM classiques

Nous avons décidé d'utiliser un modèle linéaire pour rester au plus proche du modèle proposé par une PWM classique. En effet avec cette dernière on peut calculer un score (voir section 3.1.1) qui est la somme des poids de chaque nucléotide à chaque position. Pour rappel, pour une PWM de taille K, le score d'un K-mer $z_1 z_2 \ldots z_K$ est calculé grâce à la formule :

$$s = \sum_{k=1}^{K} W_{r(z_k),k}, \tag{6.2}$$

avec $r(z_k) \in \Lambda$ l'indice du caractère z_k dans la PWM et $W_{r(z_k),k}$ le poids de z_k à la position k dans la PWM. Avec les variables booléennes Z_{kj}, cette expression peut se ré-écrire

$$s = \sum_{\substack{k=1\ldots K \\ j=1\ldots 4}} W_{kj} Z_{kj} \tag{6.3}$$

où $W_{i,k}$ est le poids du nucléotide i à la position k de la PWM.

En comparant (6.1) et (6.3) on voit que les deux formules s'écrivent comme des combinaisons linéaires de poids associés à des nucléotides et des positions. De fait, notre motif

discriminant est donc bien une PWM qui s'écrit comme une matrice $4 \times K$ dont les éléments sont les poids α_{kj} associés à chaque nucléotides et positions.

Bien que notre modèle définisse une PWM, il est important de noter que cette PWM présente, de par sa méthode d'estimation, des différences importantes avec les PWM classiques que l'on trouve dans les bases de données telles que JASPAR. De fait, nous utilisons pour notre modèle un apprentissage supervisé (la régression logistique) alors que les PWM de JASPAR sont issues d'un apprentissage non supervisé utilisé pour inférer les PPM. Alors que dans un apprentissage non supervisé, les coefficients W_{jk} sont estimés de manière indépendante pour les différentes positions k, dans un apprentissage supervisé ces coefficients sont estimés de manière conjointe, avec l'objectif de minimiser l'erreur du modèle. Une autre différence importante est l'absence de contrainte sur les valeurs des poids. En effet, dans le cadre de PWM issues de PPM, les éléments de cette dernière sont des probabilités, et sont donc soumis à certaines contraintes : somme égale à 1, toujours positif, etc... (voir section 3.1). Ici, nous n'avons plus ces contraintes, et nous pouvons donc par exemple avoir une colonne dont tous les poids sauf un sont égaux à zéro, ce qui est impossible avec les PWM issus de PPM.

Comme on l'a vu au chapitre 3.2.2, DM n'est pas la première méthode proposée pour l'apprentissage d'un motif discriminant. La méthode DAMO[90] par exemple, apprend elle aussi une PWM discriminante avec pour objectif la minimisation de l'erreur de classification. En revanche, à notre connaissance, DM est le premier à utiliser un modèle de régression logistique pour apprendre la PWM. Il bénéficie ainsi de toute l'algorithmique développée pour cette approche de classification, ainsi que de toutes les propriétés théoriques attachées à ce modèle. Une de ces propriétés est l'aspect convexe de la fonction à optimiser, qui garantit l'optimalité du modèle identifié par descente de gradient (cf. chapitre 2). De fait, et contrairement à l'ensemble des approches proposées précédemment, on a donc la garantie que la PWM identifiée par DM soit la PWM optimale pour le problème de classification qui nous intéresse. Il est essentiel de noter cependant que cette optimalité n'est rendue possible que par la spécificité de notre problème de classification, qui n'est pas identique au problème communément adressé par les méthodes concurrentes. Alors que notre problème se résume à distinguer deux ensembles de séquences ayant exactement la taille du motif, le problème adressé par les approches comme DAMO est de distinguer deux ensembles de séquences, généralement beaucoup plus longues que le mo-

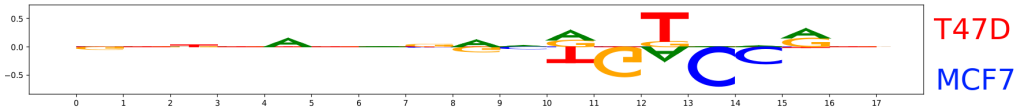

FIGURE 6.1 – *Représentation logo du modèle Discriminative Motif.*
Modèle appris pour distinguer la fixation du TF ESR1 entre les types cellulaires T47D (Y=1) et MCF7 (Y=0). L'ordonnée est la valeur des coefficients du modèle pour chaque variable. Ce logo est obtenu avec l'outil viz_sequence (adapté des outils de DeepLIFT[126]).

tif, sur la base de la présence/absence du motif recherché. Ainsi, dans notre cas on utilise le PWM discriminant uniquement pour distinguer les classes et non pour identifier les occurrences du motif (i.e. scanner les séquences). Nous avons donc toujours besoin de deux PWM : la PWM de JASPAR qui est utilisée pour scanner les séquences et identifier la meilleure occurrence du motif dans chaque séquence, et la PWM discriminante pour discriminer les deux classes. Au contraire, DAMO utilise la même PWM pour scanner et discriminer. Il adresse donc un problème de classification plus difficile, et connu pour être NP-difficile [125]. Cependant, cette problématique de classification dans laquelle les séquences ont exactement la taille du motif est tout à fait pertinente dans le cadre de notre étude des spécificités cellulaires liées à la fixation des TF, comme en attestent les expérimentations présentées dans la suite de ce manuscrit. De plus, il est intéressant de noter que cette même problématique a également été utilisée dans des études précédentes. Par exemple, Ruan et Stormo (2018) utilisent exactement cette problématique de classification dans leur étude sur l'importance du DNAshape pour distinguer les séquences fixées ou non par un TF donné. Notons enfin que l'utilisation d'une régression linéaire logistique nous permet d'y inclure une pénalisation L1 et donc de produire des PWM *a priori* plus interprétables que ceux des méthodes concurrentes, cet aspect sera évalué dans la section 6.5.

6.3 Représentation logo

Afin de visualiser et interpréter notre modèle nous proposons une représentation LOGO assez similaire à celles réalisées à partir de PPM classiques. La différence essentielle est que nous n'avons plus en ordonnée la quantité d'information de chaque nucléotide à chaque position, mais le coefficient estimé par notre modèle. De fait, ces deux 2 représenta-

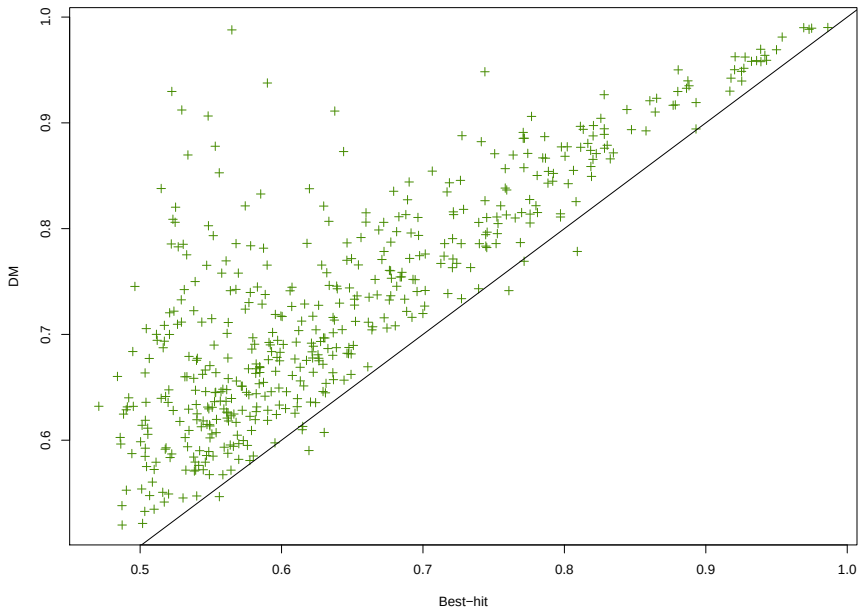

Figure 6.2 – *Comparaisons des AUROC de la PWM originale (en abscisse) et DM (en ordonnée).*
Réalisé sur 502 expériences.

tions ne sont pas directement comparables, mais elles permettent toutes deux d'identifier les nucléotides et les positions les plus importantes. Une différence essentielle par rapport aux logos classiques est le fait que les coefficients peuvent prendre des valeurs négatives. Ceci permet d'identifier facilement la classe (type cellulaire) dans laquelle un nucléotide est le plus courant. La figure 6.1 montre un exemple de logo pour le facteur de transcription ESR1. On voit par exemple que le modèle a affecté des coefficients négatifs aux variables associées aux nucléotides T, G, A, C et C respectivement aux positions 10, 11, 12, 13 et 14. Il semble donc que ces nucléotides sont importants pour distinguer les classes de séquences quand ils sont observés dans le type cellulaire MCF-7. Alors que la variable T à la position 12 a un coefficient positif, elle semble donc importante pour distinguer les types cellulaires quand on l'observe dans T47D.

6.4 Comparaison avec les PWM de JASPAR

Nous souhaitons mesurer l'apport de ce modèle par rapport à l'utilisation de la PWM originale seule. Pour une paire d'expérience donnée, nous disposons de deux ensembles de séquences. Nous scannons les séquences avec la PWM originale. Dans chacune des séquences on extrait un k-mer de taille K (score max dans la région unibind) et on souhaite classer les séquences à partir de ce k-mer. Dans le premier cas, nous utilisons le score de la PWM originale pour ce k-mer comme variable prédictive. Dans le second cas, nous entraînons un DM (70% de séquences en apprentissage, 30% pour le test) et nous scorons le k-mer avec le modèle appris.

La Figure 6.2 présente un nuage de point comparant les performances en terme de d'AUROC de la PWM de JASPAR (en abscisse) et de DM (en ordonnée). On peut voir sur cette figure que DM obtient globalement de meilleures performances pour discriminer les types cellulaires. L'AUROC de notre motif discriminant est strictement supérieure à celle de la PWM originale dans 98,3% des cas. Cela montre, qu'il est en effet possible d'optimiser le modèle de fixation pour un problème particulier. Surtout, à la vue de l'écart de performance entre le modèle original (qui décrit le motif de fixation général du facteur de transcription) et notre modèle (qui reflète les différences de la fixation entre types cellulaires), cette expérience tend à montrer que le motif de fixation diffère parfois suivant les types cellulaires.

La figure 6.3 compare les logos obtenus dans une expérience portant sur la fixation du TF USF2 entre les types cellulaires HepG2 (hepatoblastoma, classe Y=1) et GM12878 (female B cells lymphoblastoid cell line, classe Y=0). DM (AUROC = 0.929) obtient une bien meilleure AUROC que la PWM originale (AUROC= 0.522). La figure 6.3 - A est le logo de la PWM MA0526.3 USF2 présente dans JASPAR tandis que la 6.3 - B représente les coefficients du modèle appris pour distinguer la fixation du TF USF2 entre les types cellulaires HepG2 et GM12878. On observe par exemple aux positions 2, 3 et 6 que les nucléotides G, T et C présents dans la PWM originale sont associés à des coefficients positifs et donc plus présents dans HepG2. Au contraire, aux positions 6, 10 et 11 les nucléotides T, A et C ont des poids négatifs et sont donc plutôt associés à GM12878. Ceci explique en partie pourquoi le motif original ne parvient pas à discriminer les deux types cellulaires, alors que notre motif discriminant y arrive très bien. Enfin on remarque que certaines positions importantes de la PWM de JASPAR n'apparaissent pas dans DM

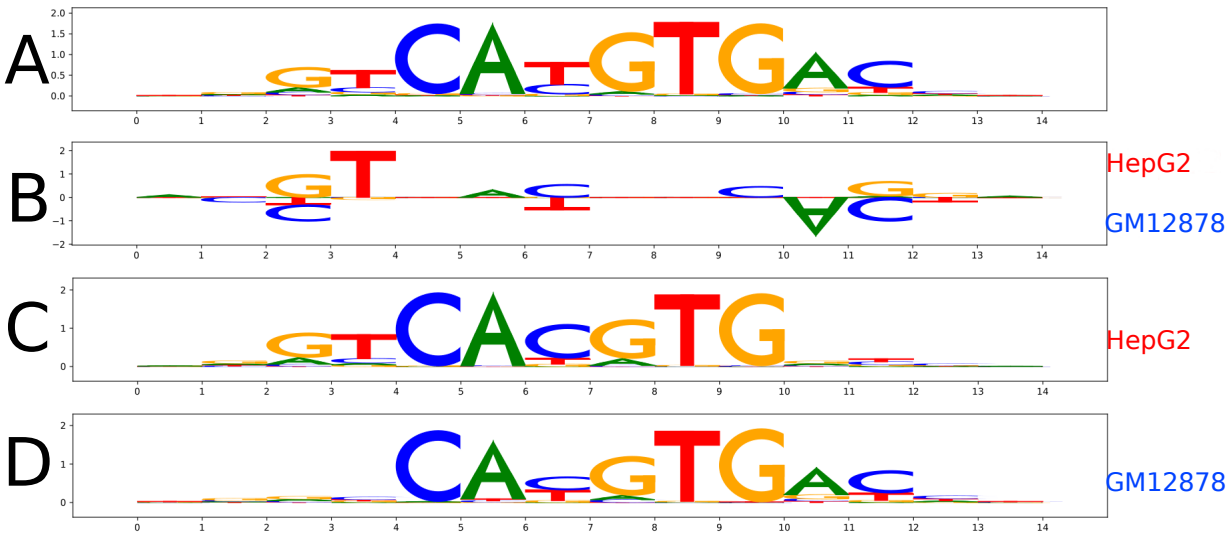

Figure 6.3 – *Représentation logo de différents motifs.*
A : PWM MA0526.3 USF2 présente dans JASPAR. **B** : les coefficients du modèle DM appris pour distinguer la fixation du TF USF2 entre les types cellulaires HepG2 (Y=1) et GM12878 (Y=0). **C** : motif de fixation de USF2 spécifique au tissu HepG2. **D** : motif de fixation de USF2 spécifique au tissu GM12878.

parce qu'elles sont communes aux deux classes, comme par exemple le C en position 4 ou le T en position 8.

Nous avons ensuite cherché à confirmer ces observations. Pour cela, nous avons construit des PPM tissus spécifiques en utilisant les séquences de taille K sur lesquelles DM est appris et testé. Les deux motifs spécifiques aux tissus sont présentés en Figures 6.3 - C et 6.3 - D. Sur ces deux motifs, on retrouve en partie les différences mises en exergue par le modèle DM. On retrouve plus particulièrement sur les positions 3 et 4 du motif construit sur HepG2 les nucléotides G et T d'une part et sur les positions 11 et 12 du motif construit sur GM12878 les nucléotides A et C d'autre part.

Nous avons ensuite croisé ces résultats avec les données présentes dans la base Meth-Motif (voir section 3.4.1) qui contient des motifs dépendants du type cellulaire auxquels sont ajoutés des scores de méthylation à chaque position. Dans cette base de donnée le facteur de transcription USF2 montre bien des différences de fixation entre les types cellulaires. Ces différences sont systématiquement liées aux GT en 5' et AC en 3'. Dans les types cellulaires HepG2 et GM12878, on retrouve un enrichissement de AC en 3' alors

qu'on observe un enrichissement en GT en 5' dans HepG2 dans notre analyse. Une raison possible à cette différence est sans doute liée au jeu de séquences utilisé pour l'apprentissage des modèles. En effet, Methmotif utilise tous les pics de ChIP-seq dans chaque type cellulaire pour construire ces motifs alors que dans notre analyse nous utilisons seulement les pics présents strictement dans un ou l'autre des deux types cellulaires. En effet, sur 7 536 pics présents dans HepG2 5 188 sont communs au type cellulaire GM12878 qui comporte 36 541 pics.

6.5 Comparaison avec DAMO

La méthode DAMO[90] décrite en section 3.2.2 optimise aussi le modèle PWM original afin de discriminer des classes de séquences. Bien que cette méthode soit originellement créée pour distinguer un ensemble de séquences fixées d'un ensemble de séquences non-fixées par un facteur de transcription, il est intéressant de comparer DM à cette méthode car elles ont toutes deux pour but de créer un motif discriminant.

Il est important de rappeler ici que DM et DAMO diffèrent fondamentalement sur leur critère d'optimisation puisque la PWM de DAMO peut être apprise sur la base de séquences plus longues que celle de la PWM. De fait, la matrice de poids en sortie de DAMO peut être utilisée pour scanner des séquences sans recourir à la PWM originale, alors que DM doit systématiquement identifier la meilleure occurrence du motif avec la PWM originale avant de procéder à la classification de la séquence.

Nous avons appliqué la méthode DAMO sur les mêmes données que DM. Ici, DAMO utilise donc des séquences qui font exactement la taille de la PWM de JASPAR (taille $= K$) pour estimer ses paramètres. De plus, entraînements et tests sont réalisés sur les mêmes ensembles d'apprentissages et de tests que ceux utilisés pour DM (pour rappel 70% des séquences pour l'apprentissage des deux modèles et 30% pour l'ensemble de test). La Figure 6.4 est un nuage de point ayant en abscisse l'AUROC obtenue par DAMO et en ordonnée l'AUROC obtenue par la méthode DM pour chacune des 502 expériences. Sur cette figure on voit que DM obtient de meilleures performances dans la majorité des cas (72% des cas) et que les écarts substantiels sont invariablement en faveur de DM.

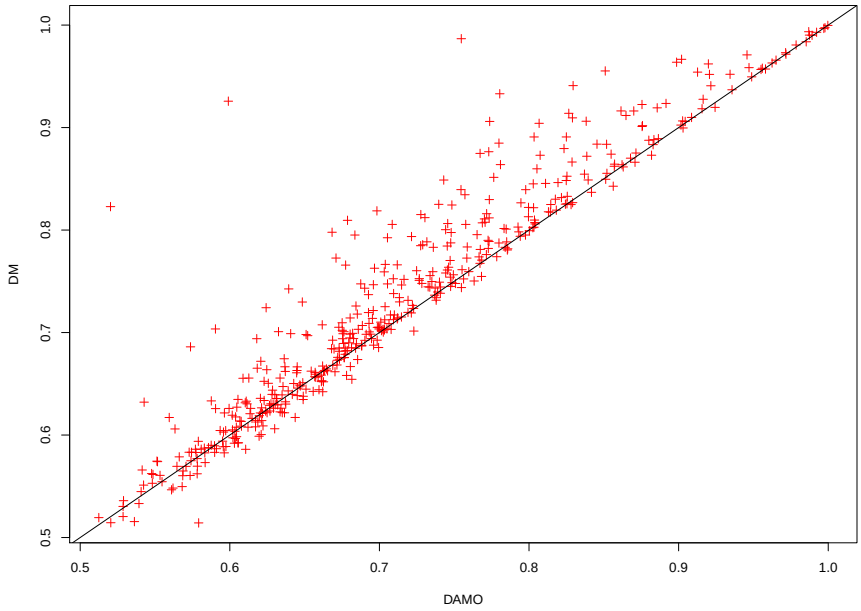

Figure 6.4 – *Comparaisons des AUROC de DAMO et DM.*
DAMO (en abscisse) et DM (en ordonnée), réalisé sur 502 expériences.

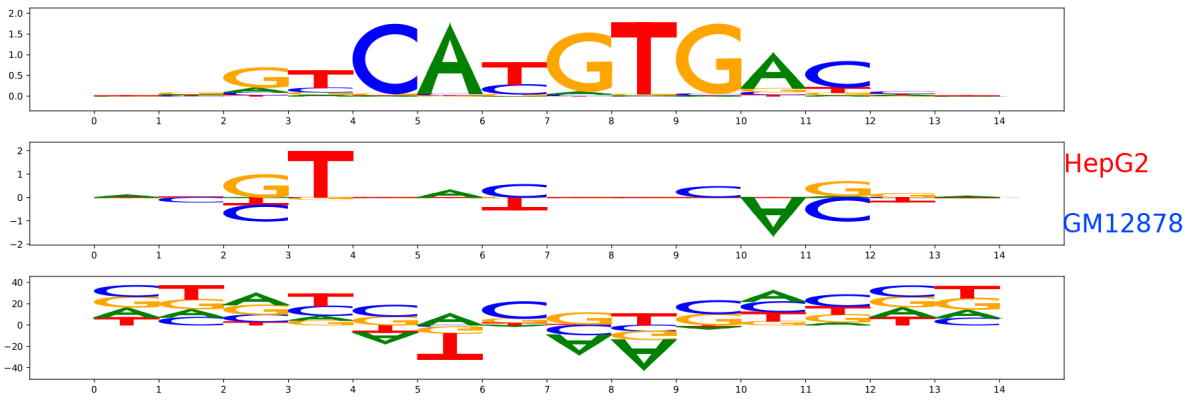

Figure 6.5 – *Représentation logo de différents motifs.*
En haut, la PWM MA0526.3 USF2 présente dans JASPAR. Au centre : les coefficients du modèle DM appris pour distinguer la fixation du TF USF2 entre les types cellulaires HepG2 (Y=1) et GM12878 (Y=0). En bas : les poids présents dans la PWM apprise par DAMO pour discriminer les même classes.

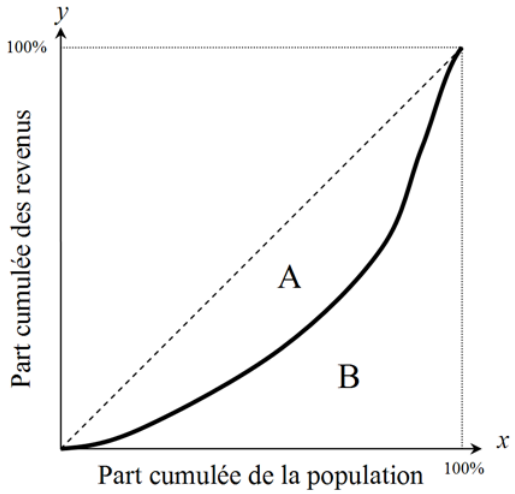

FIGURE 6.6 – *Exemple d'une courbe de Lorenz.*
La surface A est définie entre la courbe de Lorenz et la droite d'égalité parfaite de répartition. La surface B est définie entre la courbe de Lorenz et la courbe d'inégalité parfaite de répartition.

Cependant, en dehors des performances de prédiction qui sont bien sûr importantes, l'interprétation des modèles est également une question majeure pour notre problématique. Comme on peut le voir sur la Figure 6.5, qui représente 3 motifs obtenus sur la même expérience, il semble que le modèle DM (au centre) soit plus simple à interpréter que DAMO (en bas). Dans l'article « Towards A Rigorous Science of Interpretable Machine Learning »[127] les auteurs définissent l'interprétabilité comme la capacité d'expliquer ou de présenter un modèle en termes compréhensibles à un humain. Suivant cette définition, nous considérons ici qu'un modèle avec une bonne interprétabilité est un modèle qui contient peu de variables, permettant de résumer l'information de manière plus efficace et rendant donc plus compréhensible la lecture. Afin de comparer l'interprétabilité des modèles DM et DAMO nous avons utilisé un critère qui mesure si l'information du motif est contenue dans un petit nombre de variables ou si elle est partagée entre beaucoup de variables. Classiquement, la quantité d'information contenue dans un motif est calculée par une mesure d'entropie issue de la théorie de Shannon (voir section 3.1). Cette mesure ne s'applique cependant que dans le cadre de modèles probabilistes et donc que pour les PPM. Pour les PWM, *a fortiori* lorsqu'elles ne sont pas issues de PPM, une telle mesure ne peut s'appliquer. C'est pourquoi, nous avons utilisé le coefficient de Gini.

La courbe de Lorenz est une représentation graphique de l'inégalité de répartition d'une donnée entre individus. Elle est particulièrement utilisée en économie pour représenter l'écart de répartition des richesses dans une population. Elle s'obtient en ordonnant les individus dans l'ordre de leurs revenus, et en calculant la part cumulée des revenus en fonction de la part cumulée des individus. Dans le cas d'une égalité parfaite des revenus entre tous les individus, la courbe suit la droite $y = x$. Dans le cas contraire, elle se trouve sous cette droite. Un exemple de courbe de Lorenz est représenté en Figure 6.6. La surface A est l'aire entre la courbe de Lorenz et la droite d'égalité parfaite de répartition. La surface B est l'aire entre la courbe de Lorenz et la courbe d'inégalité parfaite (tous les revenus appartiennent à un seul individu). Le coefficient de Gini $C_G \in [0; 1]$ est défini comme :

$$C_G = \frac{A}{A + B} \tag{6.4}$$

Ce coefficient est donc égal à 1 si tous les revenus appartiennent à un seul individu et égal à 0 si toute la population se partage les richesses de façon équilibrée. Dans le cas d'une PWM, nous utilisons le coefficient de Gini sur l'ensemble des poids de la matrice. Plus précisément, on collecte les $4 \times K$ poids de la matrice, toutes positions confondues, on les ordonne par ordre croissant en valeur absolue, et on calcule la courbe de Lorenz et le coefficient de Gini associés à cet ensemble de poids. Un coefficient de Gini petit traduit donc une égalité de répartition des coefficients, c'est à dire que l'information est dispersée sur beaucoup d'éléments de la matrice. Au contraire, un coefficient de Gini grand (proche de 1) indique que certains éléments de la matrice résument toute l'information et que beaucoup d'éléments de la PWM sont à 0 ou proches de 0. On peut donc le voir comme une mesure de l'interprétabilité des modèles, où un modèle ayant un coefficient de Gini grand sera plus interprétable qu'un modèle avec un coefficient de Gini proche de 0.

La Figure 6.7 représente les distributions des coefficients de Gini pour les 2×502 PWM inférées par DAMO et DM. Sur cette figure, on peut voir l'écart des distributions des deux modèles, montrant que, du point de vue de cette mesure, le modèle DM est plus interprétable que le modèle DAMO.

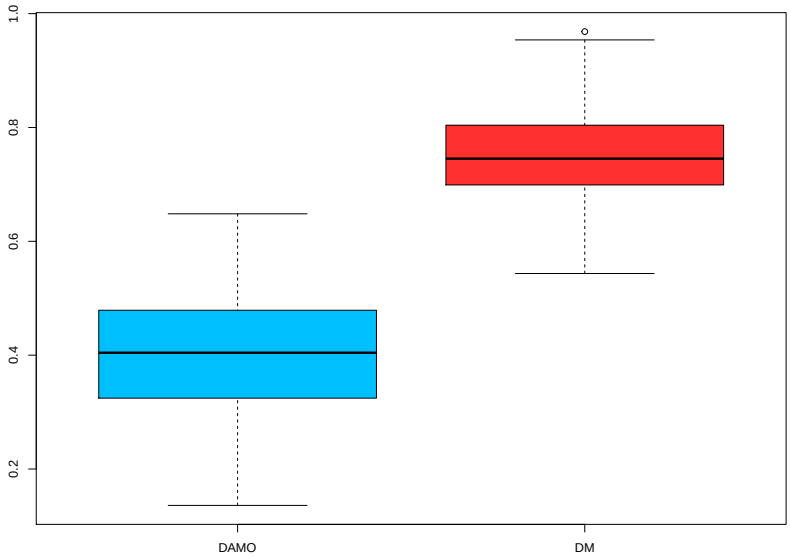

FIGURE 6.7 – *Boites à moustaches des distribution des coefficients de Gini pour les modèles DAMO et DM.*
Les deux distributions sont obtenues sur 502 modèles.

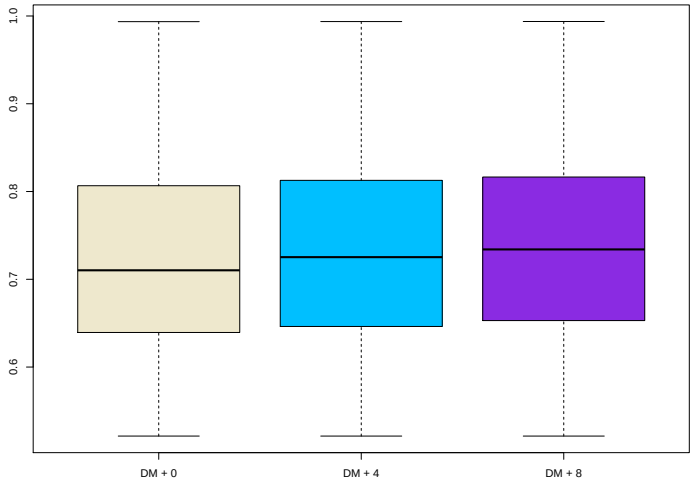

FIGURE 6.8 – *Distribution des AUROC des trois modèles « DM+0 »,« DM+4 »,« DM+8 ».*
Comparaisons des effets des bases proches du motif de fixation. Coefficients optimisés sur un ensemble d'apprentissage représentant 70% du nombre de séquences total, les AUROC sont calculées sur un ensemble de test indépendant représentant 30% du nombre de séquences total.

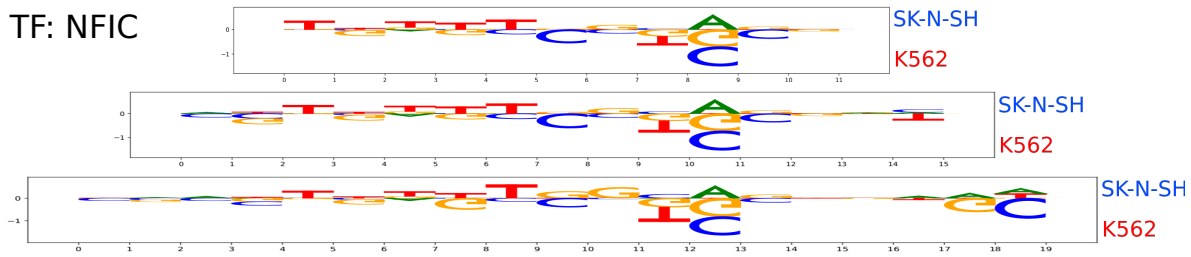

Figure 6.9 – *Représentations logo des trois modèles « DM+0 »,« DM+4 »,« DM+8 ».* Modèles appris pour distinguer la fixation du facteur de transcription NFIC entre les types cellulaires K562 (myelogenous leukemia) (Y=0) et SK-N-SH (neuroblastoma)(Y=1). L'ordonnée est la valeur des coefficients du modèle pour chaque variable.

6.6 Environnement nucléotidique proche du motif

Une question ouverte de la littérature concerne l'importance des quelques bases qui sont immédiatement adjacentes au motif pour la fixation du facteur de transcription. Pour notre problème, il est donc possible que ces bases portent une partie de l'information responsable de la différence de fixation entre deux expériences. Afin de vérifier cette hypothèse, nous avons agrandi notre motif discriminant de quelques bases de chaque coté. Notons cependant que nous ne pouvons pas trop augmenter le nombre de positions étudiées, car chaque position supplémentaire ajoute 4 variables au modèle, ce qui pourrait rendre plus difficile l'estimation de coefficients et pourrait nuire à l'interprétation du modèle. Nous avons donc choisi d'ajouter 2 et 4 bases de chaque coté du motif et de comparer les résultats obtenus. Nous avons alors trois motifs discriminants à étudier. Pour une PWM originelle de taille K, nous avons le premier modèle, « DM+0 », comportant $4 \times K$ variables modélisant un motif de taille K, le second, « DM+4 », comportant $4 \times (K + 4)$ variables modélisant un motif de taille $K + 4$, et le dernier, « DM+8 », comportant $4 \times (K + 8)$ variables et modélisant un motif de taille $K + 8$.

Les boites à moustaches de la Figure 6.8 montrent les distributions des AUROC des trois modèles. Nous pouvons voir sur cette figure une très légère augmentation de l'AUROC à chaque fois que l'on rajoute des bases au motif (l'AUROC moyen des méthodes est 0.7299, 0.7371 et 0.7413 respectivement).

Bien que l'impact des bases adjacentes semble en moyenne assez faible, pour certaines expériences l'effet est plus fort. Par exemple, pour la fixation du facteur de transcrip-

tion NFIC entre les types cellulaires K562 (myelogenous leukemia) et SK-N-SH (neuroblastoma), les bases proches du motif semblent importantes (AUROC : 0.68, 0.70, 0.76 respectivement). Les logos des trois modèles sont représentés en Figure 6.9. L'ajout des variables aux positions 17,18 et 19 du modèle « DM+8 » semble être important et traduire l'augmentation de 8% d'AUROC par rapport au modèle « DM+0 ». Notons cependant que dans ce cas, il est assez difficile de dire si l'augmentation d'AUROC est liée à la prise en compte des bases adjacentes au motif, ou à la modélisation d'une partie d'un second motif sur lequel viendrait se fixer un cofacteur quelques bases à côté du TF cible. Dans la suite de nos analyses on conservera ses bases adjacentes dans DM car elles semblent apporter une information supplémentaire dans certains cas.

Chapitre 7

Utilisation de la position des cofacteurs (Positional TFcoop)

Dans ce chapitre nous détaillons la méthode « Positional TFcoop » qui utilise les informations de présence et de position des cofacteurs. Elle est basée comme la méthode DM sur une régression logistique, et nous analysons l'apport des positions relatives des facteurs de transcription par rapport à la méthode originale intitulée TFcoop [28].

Nous avons vu en section 3.3.6 ce qu'une méthode comme TFcoop peut apporter pour l'étude des déterminants génomiques impliqués dans la fixation d'un facteur de transcription (séquences fixées versus un background). En particulier, l'étude des facteurs de transcription coopérants (appelés cofacteurs dans la suite) semble être une information importante pour expliquer la fixation d'un facteur de transcription donné. Cette même information pourrait donc s'avérer importante pour notre problème également. Cependant, jusqu'à présent l'information relative aux cofacteurs n'est modélisée dans TFcoop qu'avec le score maximum de chaque PWM dans chaque séquence. Cela n'informe donc que sur la présence potentielle d'un TF dans la région +/-500bp autour du site de fixation du facteur de transcription étudié. Afin d'affiner cette information et de pouvoir améliorer l'interprétation dans la fixation des cofacteurs, il nous semble important de pouvoir prendre en compte leurs positions par rapport au facteur cible. Par exemple, savoir qu'un TF A est situé systématiquement à +100bp du TF cible dans un type cellulaire et pas dans l'autre serait une information forte à ajouter au modèle.

Comme on l'a vu en section 5.3, nos séquences sont centrées sur le site de fixation le plus

probable du TF étudié dans l'expérience. De fait, en étudiant la position des cofacteurs, nous étudions la position entre ces TF et le TF ciblé par l'expérience. Nous proposons donc un nouveau modèle, dans lequel les variables ne sont plus les scores maximums observés pour chaque séquence mais les scores maximums observés dans des sous-régions particulières de la séquence. Toute la difficulté est alors de sélectionner, pour chaque cofacteur la région la plus informative du point de vue de la classification.

Pour rappel, TFcoop utilise les taux de dinucléotides des séquences en plus des scores maximums des PWM. Dans ce chapitre, nous nous concentrerons essentiellement sur les scores de PWM et non les taux de dinucléotides, afin de mesurer seulement l'apport de l'information de position relative. On utilisera par la suite (voir Chapitre 8) l'information relative à l'environnement nucléotidique en ajoutant les variables de DExTER (voir section 3.5) aux variables de PosTFcoop. Les variables de DExTER remplaceront ainsi les taux de dinucléotides en y ajoutant une information positionnelle.

Abstraction faite des variables relatives aux taux de dinucléotides, notre nouveau modèle s'écrit donc comme le modèle utilisé dans TFcoop (Expression 3.10) :

$$\ln \frac{\mathbb{P}(Y = 1 \mid R)}{1 - \mathbb{P}(Y = 1 \mid R)} = \gamma_0 + \sum_{j=1}^{p} \gamma_j R_j, \tag{7.1}$$

Les variables R_j remplacent les variables X_j de l'expression 3.10 et désignent le meilleur score identifié pour la PWM j dans la sous-région associée à ce PWM. Nous utiliserons comme pour TFcoop la base de données JASPAR, et nous aurons donc $p = 1011$ variables.

Tout le travail de PosTFcoop est donc d'identifier, pour chaque PWM, la région pour laquelle son score maximum est le plus informatif pour discriminer les deux classes de séquences. Pour cela nous avons développé une méthode de segmentation qui est détaillée ci-après.

7.1 Segmentation

Notre objectif est de déterminer une région associée à chaque PWM, qui maximise l'AUROC que l'on obtient lorsqu'on utilise le score maximum trouvé dans cette région

pour discriminer les séquences. Explorer l'ensemble des régions possibles d'une séquence de taille $L = 1000$ reviendrait à étudier L sous-séquences de taille 1, plus $L-1$ sous-séquences de taille 2, plus *etc...* On a donc un total de $\sum_{i=1}^{L} i = \frac{L(L+1)}{2} = 500\ 500$ sous-séquences à étudier. Si on considère un jeu de données avec n séquences, il faudrait donc « parser » $n(L(L+1)/2)$ sous séquences au total. Il est donc nécessaire de réduire le nombre de régions à considérer.

Nous partons pour cela d'une segmentation de la séquence originale en N sous-séquences élémentaires de même tailles. Plus N est grand, plus la résolution sera importante, mais plus le coût en terme de calcul sera élevé. Si le reste de la division entière de la taille de la séquence (=1000) par N est non nul, alors les sous-régions aux extrémités de la séquence se verront ajouter chacune la moitié des bases supplémentaires. On considère ensuite une structure en treillis où chaque nœud correspond à une région. Formellement, un treillis est un ensemble partiellement ordonné dans lequel chaque paire d'éléments admet une borne supérieure et une borne inférieure. Ici, les éléments sont des régions, et la relation d'ordre est l'inclusion. Nous ne considérons donc pas des treillis mais des demi-treillis : chaque paire de régions A et B est associée à une borne supérieure qui est la région minimale qui contient A et B. En revanche ces paires n'ont pas de borne inférieure. Les nœuds à la base du demi-treillis correspondent aux N sous-séquences élémentaires (voir Figure 7.1). Les nœuds des étages supérieurs s'obtiennent par concaténation des sous-régions inférieures. Plus on monte dans le demi-treillis, plus les régions considérées sont grandes. Le sommet correspond à la séquence entière. De cette manière le nombre de régions considérées dans notre exploration est égal à $\frac{N(N+1)}{2}$.

Une fois la structure du treillis construite, elle est utilisée pour calculer le score maximum associé à chaque région de chaque séquence pour une PWM donnée. Pour cela, la séquence est scannée avec la PWM à l'aide de l'outil FIMO [67], et le score maximum obtenu dans chaque région élémentaire est stockée dans les nœuds correspondants à la base du treillis. Ensuite, il est facile de compléter les étages supérieurs en utilisant l'opération $max(,)$: si S_A et S_B sont les scores maximums dans les régions A et B et $C = A \cup B$ alors $S_C = \max(S_A, S_B)$ est le score maximum dans la région C. De cette manière, on calcule les scores associés à chaque région considérée en ne scannant qu'une seule fois la séquence originale, et en propageant ensuite les scores maximums dans le reste du treillis. Avec N sous-séquences élémentaires, cela se fait donc en $\mathcal{O}(N^2)$ opérations. Ce calcul est répété

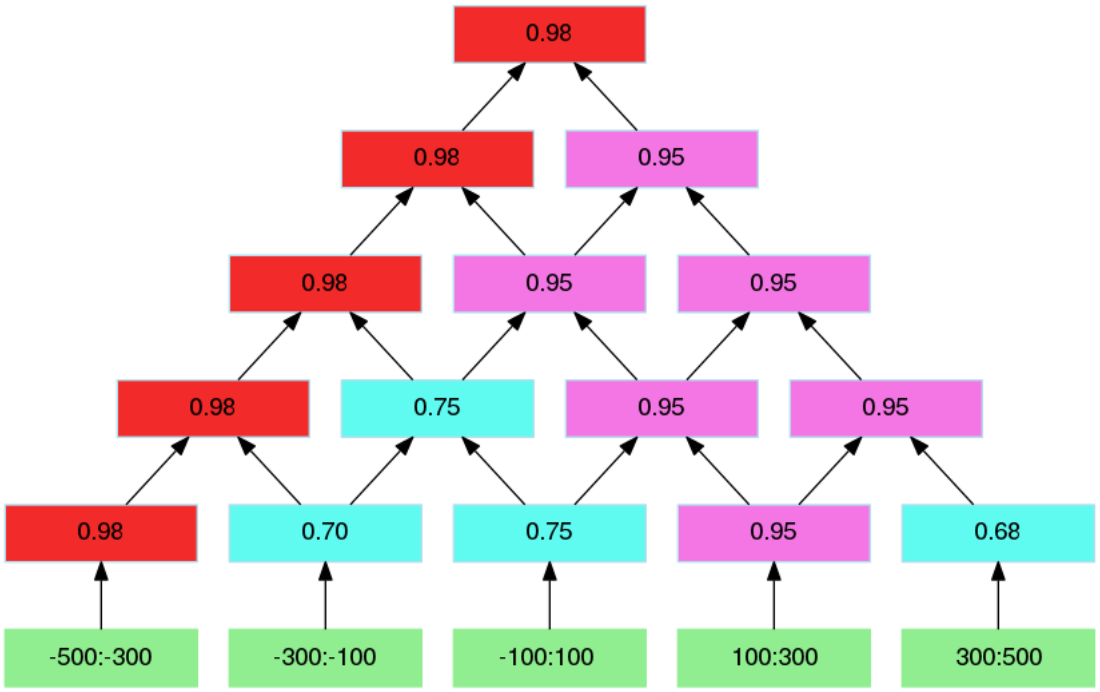

FIGURE 7.1 – *Exemple de demi-treillis de score.*
La séquence de taille 1000 est segmentée en 5 régions de taille 200. Chaque cellule correspond au score maximum dans la région qu'elle couvre. Les scores sont exprimés en ratio min-max.

pour chaque séquence. Un exemple de treillis avec les scores associés à une séquence particulière est représenté en Figure 7.1. L'algorithme 3 donne le pseudo code de la procédure de calcul des scores pour chaque PWM et chaque séquence. Dans cet algorithme, on note $E_i^{(k)}$ le $i$ème élément de l'étage k du treillis.

7.2 Sélection de la région

Dans cette section nous cherchons à sélectionner pour chaque PWM la meilleure région permettant de discriminer les deux classes de séquences. Cette sélection est réalisée en utilisant uniquement l'ensemble des séquences d'apprentissage. Nous avons, pour une PWM donnée, un ensemble de demi-treillis, chaque séquence d'apprentissage étant associée à un demi-treillis. On peut, dans cet ensemble, distinguer deux sous-ensembles suivant que les séquences d'apprentissage appartiennent à une classe ou à l'autre. Chaque nœud d'un

Algorithme 3 Remplir un demi-treillis de score

Entrées :
-une PWM
-une séquence z
-N, le nombre de sous-régions unitaires
Scanner z avec la PWM
Segmenter z en N régions R_i
pour i de 1 à N **faire**
  Remplir les éléments i de l'étage 0 : $E_i^{(0)}$
fin pour
pour k de 1 à $N-1$ **faire**
  pour i de 1 à $N-k$ **faire**
    $E_i^{(k)} = max(E_i^{(k-1)}, E_{i+1}^{(k-1)})$
  fin pour
fin pour
return $\forall k, i$, $E_i^{(k)}$ l'ensemble des scores associés aux $N(N+1)/2$ sous-régions possibles

	A	non A	Total
B	a	b	a+b
non B	c	d	c+d
Total	a+c	b+d	n=a+b+c+d

Table 7.1 – *Table de contingence entre les variables A et B*

treillis est associé à une sous-région de la séquence et à un score correspondant au score max du PWM étudié dans cette sous-région (voir section 7.1). L'objectif est d'identifier la sous-région qui permet de discriminer le mieux les deux classes de séquences en fonction des scores calculés dans cette sous-région. Pour un nœud i, on note $S_0(i)$ et $S_1(i)$ l'ensemble des scores calculés dans la région associée au nœud i chez les séquences de la classe 0 et de la classe 1, respectivement. Deux critères ont été évalués pour mesurer le pouvoir discriminant du nœud i :

— un test statistique de comparaison des scores entre les deux classes,

— une mesure de performance basée sur l'AUROC.

Ces deux critères comportent différents avantages et défauts, qui sont détaillés ci-après.

	Y=1	Y=0	Total
Score $\geq$ seuil	1256	312	1568
Score $<$ seuil	8071	9015	17086
Total	9327	9327	18654

Table 7.2 – *Table de contingence des scores dans les classes 0 et 1.*

7.2.1 Critère de sélection basé sur un test exact de Fisher

Le premier critère que nous avons étudié est basé sur le test exact de Fisher. Ce test statistique créé par Ronald Fisher est utilisé dans l'analyse de tables de contingence. Soit la table de contingence 7.1 entre deux variables qualitatives A et B. Si A et B ne sont pas indépendants on aura une sur-représentation d'individus dans une ou plusieurs cases du tableau. H_0 est l'hypothèse d'indépendance des variables A et B. Sous H_0, la probabilité p d'observer ce tableau est :

$$p = \frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{n}{a+c}} = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{a!b!c!d!n!} \tag{7.2}$$

Pour calculer la p-valeur, on somme les probabilités de toutes les tables qui sont plus ou autant déséquilibrées que la table observée. Si cette p-valeur est faible on rejettera donc H_0.

Ici, nous voulons tester la dépendance du score par rapport à la classe des séquences sur toutes les régions. A correspond à la classe $Y = 1$ et $nonA$ à la classe $Y = 0$. B correspond à un score « haut » et $nonB$ à un score « bas ». Il nous faut donc utiliser un seuil de score t qui délimite un score « haut » d'un score « bas ». Une fois t défini, nous comptons combien de séquences ont un score associé au nœud i supérieur à t. Nous pouvons alors construire la table de contingence donnée en Table 7.2 et tester si une des deux classes a un nombre de séquences avec un haut score significativement plus grand que celui de l'autre classe. Si la p-valeur est faible on pourra conclure que cette région comporte des différences de score significatives entre les classes. En réalisant ce test sur toutes les régions des treillis associés à une PWM on obtient une p-valeur associée à chaque région. On compare ensuite ces probabilités et on sélectionne la région avec la p-valeur la plus faible. À noter que nous n'utilisons pas de seuil de significativité sur la p-valeur (on choisit systématiquement la région de p-valeur minimale), il n'est donc pas nécessaire d'appliquer

une correction pour les tests multiples.

Ce critère de sélection a pour avantage d'être assez peu gourmand en temps de calcul, mais il a comme inconvénient la nécessité de définir un seuil de score pour les PWM. Ce seuil est dépendant des PWM et reste donc difficile à optimiser pour l'ensemble des PWM. Un autre inconvénient de cette méthode est qu'elle ne se base que sur le score de la PWM pour identifier la meilleure région et ne peut pas intégrer l'effet des variables liées aux taux des différents k-mers qui seront dans un second temps identifiés via la procédure DExTER et intégrés au modèle global. En d'autres termes, la sélection se fait de manière complètement indépendante des autres variables, ce qui n'est *a priori* pas l'idéal pour un classifieur construit sur un nombre potentiellement important de variables.

7.2.2 Critère de sélection basé sur l'AUROC

Étant donné un nœud i, le premier critère que nous avons proposé évalue le pouvoir discriminant de la région associée au nœud en comparant le nombre de scores de $S_0(i)$ et $S_1(i)$ qui sont supérieurs à un certain seuil. Notre second critère calcule directement le pouvoir discriminant des scores associés au nœud i sous la forme d'une AUROC. Plus exactement, nous construisons un modèle de régression logistique intégrant ces scores ainsi que les variables identifiées par l'approche DExTER (voir section 3.5 et nous entraînons ce modèle pour discriminer les deux classes de séquences. Le pouvoir discriminant de la région associée au nœud i pour la PWM étudiée est alors évaluée par l'AUROC obtenu par ce classifieur.

L'intérêt de faire rentrer l'effet de l'environnement nucléotidique dans la sélection de la meilleure région associée à un PWM donné est justement de s'extraire de cet effet, et de se concentrer uniquement sur ce qu'apporte le PWM en terme de pouvoir discriminant. Cependant, les variables renvoyées par DExTER peuvent être en nombre important (plusieurs dizaines de variables usuellement). Comme nous entraînons un modèle logistique pour chaque nœud du treillis, l'ajout de toutes ces variables peut être coûteux en temps. Pour réduire le temps de calcul nous effectuons au préalable une régression logistique avec seulement les variables de DExTER. Puis, nous utilisons le vecteur de prédiction donné par ce modèle comme variable supplémentaire pour les régressions effectuées à chaque nœud du treillis. À chaque nœud nous ajustons donc un modèle de régression logistique

à deux variables, où seule la variable x associée au score maximum dans la région change entre les modèles. Chacun des $= N(N+1)/2$ modèles s'écrit :

$$ln\frac{\mathbb{P}(Y=1|(v,x))}{1-\mathbb{P}(Y=1|(v,x))} = \beta_0 + \beta_1 v + \beta_2 x \tag{7.3}$$

avec v le vecteur de prédiction obtenu par le modèle linéaire logistique entraîné sur les variables de DExTER, et x le vecteur de scores associé à la région d'intérêt.

Afin d'obtenir des estimations d'AUROC non biaisées, il est nécessaire d'estimer cette statistique sur un ensemble de validation indépendant de l'ensemble d'apprentissage utilisé pour entraîner le classifieur. Une procédure de validation croisée (fold=10) est donc implémentée sur tous les modèles réalisés sur les treillis. La méthode associée à ce critère est donc beaucoup plus coûteuse en temps que le test exact de Fisher présenté avant, mais elle a l'avantage de ne nécessiter aucun seuil et elle permet de contrôler à la fois le surapprentissage et le lien entre environnement nucléotidique et motif de fixation.

7.2.3 Comparaison des deux critères de sélection

La Figure 7.2 représente les distributions d'AUROC de la méthode PosTFcoop suivant les deux critères de sélection proposés pour choisir une région associée à un PWM. Ces expérimentations ont été réalisées sur les 100 expériences dans lesquelles PosTFcoop (critère de sélection AUROC) obtient les meilleures AUROC parmi les 502 expériences de notre jeu de données. Dans chaque modèle nous utilisons les variables de DExTER en plus des variables PosTFcoop, afin de minimiser l'effet de sélection de l'information d'environnement nucléotidique par des variables issues de PWM. Comme on l'a vu, le critère de sélection basé sur le test de Fisher nécessite un seuil utilisé pour compter le nombre de scores considérés comme hauts. Plusieurs seuils ont été utilisés dans ces expériences : -5,0,5,10,15. Les performances obtenues par les 3 meilleurs seuils (0,5 et 10) sont représentées en Figure 7.2. Rappelons qu'un seuil de score égal à 0 correspond à la valeur de score au delà de laquelle l'occurrence a plus de probabilité d'avoir été générée par la PWM que par le modèle nul (voir section 3.1.1). Comme on peut le voir sur la Figure 7.2, le critère de sélection avec le test exact de Fisher est très dépendant du seuil initialement choisi. Améliorer les performances de ce critère semble donc possible, par exemple en déterminant des seuils optimums particuliers pour chaque PWM à la place d'un seuil

global comme cela a été fait dans nos expériences. Néanmoins, les difficultés posées par la définition de ce seuil, et le fait de pouvoir intégrer au critère de sélection les informations relatives à l'environnement nucléotidique (ce qui sera d'une grande importance lorsque l'on cherchera à évaluer l'apport de chaque type d'information, voir Chapitre 8) nous ont conduit à utiliser le critère basé sur l'AUROC dans la suite de ce manuscrit.

7.3 Évaluation de l'apport de l'information positionnelle

Dans cette section nous évaluons l'apport de l'information de position relative des cofacteurs en comparant la méthode Positional TFcoop à TFcoop [28] (voir section 3.3.6).

Comme nous l'avons précisé en introduction, ici ce que nous appelons TFcoop est en fait un modèle qui n'utilise que les variables de TFcoop relatives au cofacteur (les taux de dinucléotides ne sont pas intégrés au modèle). L'information nucléotidique dans les séquences sera par la suite contenue dans les variables de DExTER et intégrées à un modèle global (voir chapitre 8).

Les comparaisons sont réalisées sur 502 expériences. Les modèles sont appris sur un même ensemble d'apprentissage représentant 70% des séquences de chaque expérience, et les AUROC sont calculées sur un ensemble de test indépendant (30%). La Figure 7.3 présente un nuage de point avec en abscisse les AUROC de TFcoop et en ordonnée celles de PosTFcoop. On peut voir sur cette figure que PosTFcoop obtient de meilleures performances que TFcoop : les AUROC sont supérieures dans 94% des cas et les écarts substantiels sont quasi systématiquement observés en faveur de cette approche. L'information de position des cofacteurs semble donc bien être une information importante à prendre en compte lorsque l'on étudie les différences génomiques pouvant expliquer les différences cellulaires observées dans la fixation des TF.

7.4 Analyse d'un modèle et identification des variables les plus importantes

Dans cette section nous présentons deux études de cas sur les résultats de la méthode PosTFcoop (segmentation en $N = 13$ régions). On s'intéresse tout d'abord à une expérience qui se focalise sur la fixation du facteur de transcription SP2 entre les types cellulaires K562 ($Y = 1$) et HEK293 ($Y = 0$). La méthode PosTFcoop obtient une AUROC de 0.85 sur ce problème de classification. La Figure 7.4 montre les régions associées aux variables les plus informatives du modèle. La procédure *ad hoc* que nous utilisons pour sélectionner ces variables consiste à lancer un apprentissage en spécifiant le paramètre « $max\_features = 15$ » du package « *glmnet* » de python. Avec ce paramètre, la fonction *glmnet* choisit un λ tel que $|\beta \neq 0|$ soit le plus proche possible de 15. Les segments sur la Figure 7.4 indiquent les régions sélectionnées, tandis que la couleur de ces segments renseigne sur le signe du coefficient associé à cette variable dans le modèle. De plus, ces variables sont triées par ordre d'importance dans le modèle. Cette mesure d'importance est évaluée via la procédure suivante. Les variables R_j sont considérées les unes après les autres et leur coefficient β_j est mis à 0 dans le modèle. Ce nouveau modèle est alors utilisé pour prédire la classe des séquences de test, et l'écart entre l'AUROC de ce modèle et l'AUROC du modèle original est utilisé comme mesure d'importance de la variable pour le modèle. Sur la Figure 7.4 les variables les plus importantes sont en tête ou en queue de liste, suivant que leur coefficient est positif ou négatif. On voit sur cette figure que deux PWM correspondantes à des facteurs de transcription issus de la famille NFY semblent se fixer à une position proche de celle du facteur cible dans les séquences du type cellulaire K562. On observe également plusieurs PWM sélectionnées dans une région en intersection avec la région centrale avec un coefficient négatif, dont deux très proches du motif SP2 (MA0079.3 SP1, MA0516.1 SP2). Enfin, on peut observer la PWM associée au facteur de transcription Sox11 dans la région $[691, 842]$, c'est-à-dire +200bp en aval du site de fixation de SP2 dans le type cellulaire K562, ainsi que la PWM associée à GATA1 sur une large région en amont de SP2.

De même, nous pouvons étudier l'expérience de fixation du facteur de transcription JUNB entre les types cellulaires A549 et K562. Cette expérience obtient une AUROC de 0.91 avec la méthode PosTFcoop. La Figure 7.5, représente les variables les plus importantes ainsi que les régions sélectionnées. Sur cette figure, on peut remarquer que JUNB

a tendance à se fixer avec des cofacteurs de la famille FOX dans le type cellulaire A549, alors que dans le type cellulaire K562 les cofacteurs semblent être liés à GATA de la famille des zinc-fingers, et aux cofacteurs de la famille AP1.

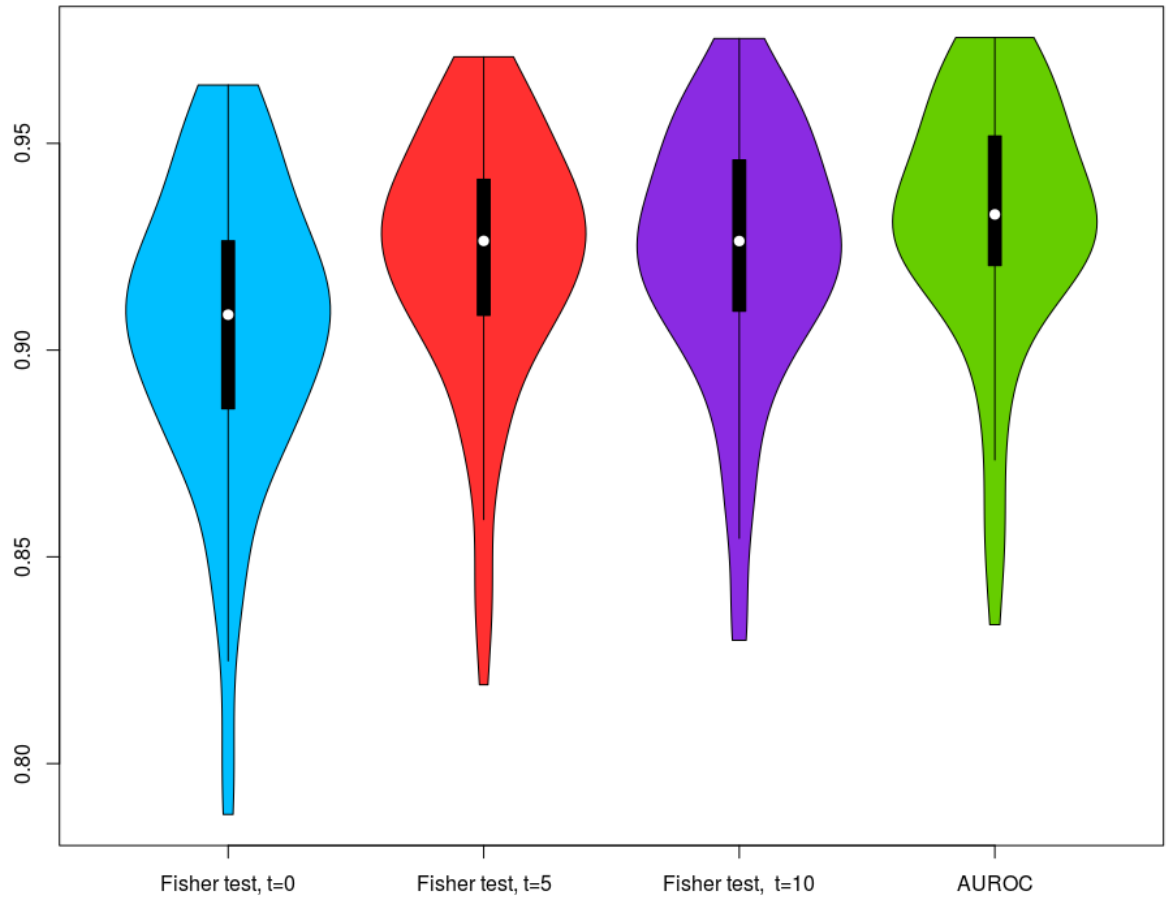

Figure 7.2 – *Représentation en violon des distributions d'AUROC mesurant les performances de modèles LASSO entraînés avec les variables de DExTER et de PosTFcoop sur 100 expériences.* Les variables de PosTFcoop sont sélectionnées avec différents critères de sélection. Les bleu, rouge et violet correspondent au critère du test exact de Fisher en utilisant respectivement un seuil de 0, 5 et 10 pour identifier les scores hauts. Le « violin plot » vert est obtenu avec le critère basé sur l'AUROC.

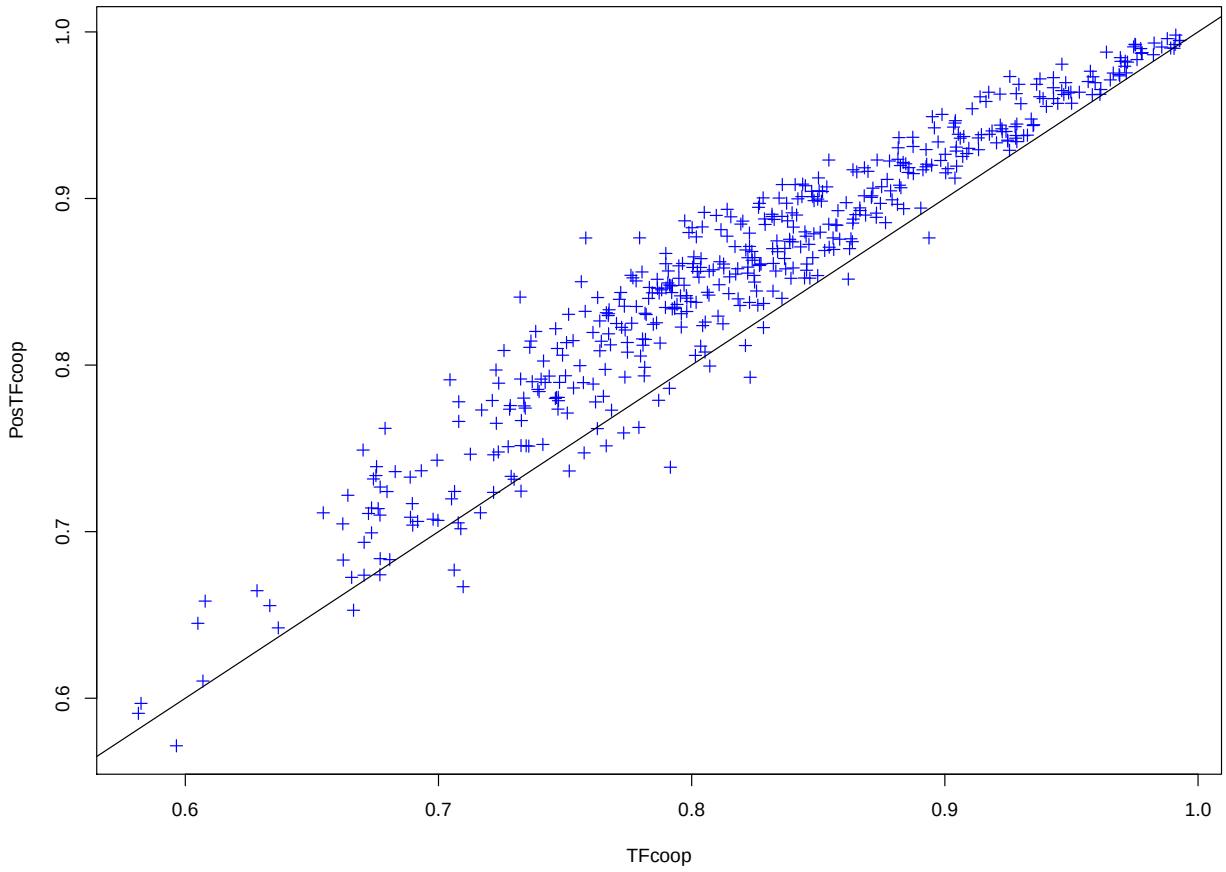

Figure 7.3 – *Comparaisons des AUROC de TFcoop (en abscisse) et PosTFcoop (en ordonnée) réalisé sur 502 expériences.*

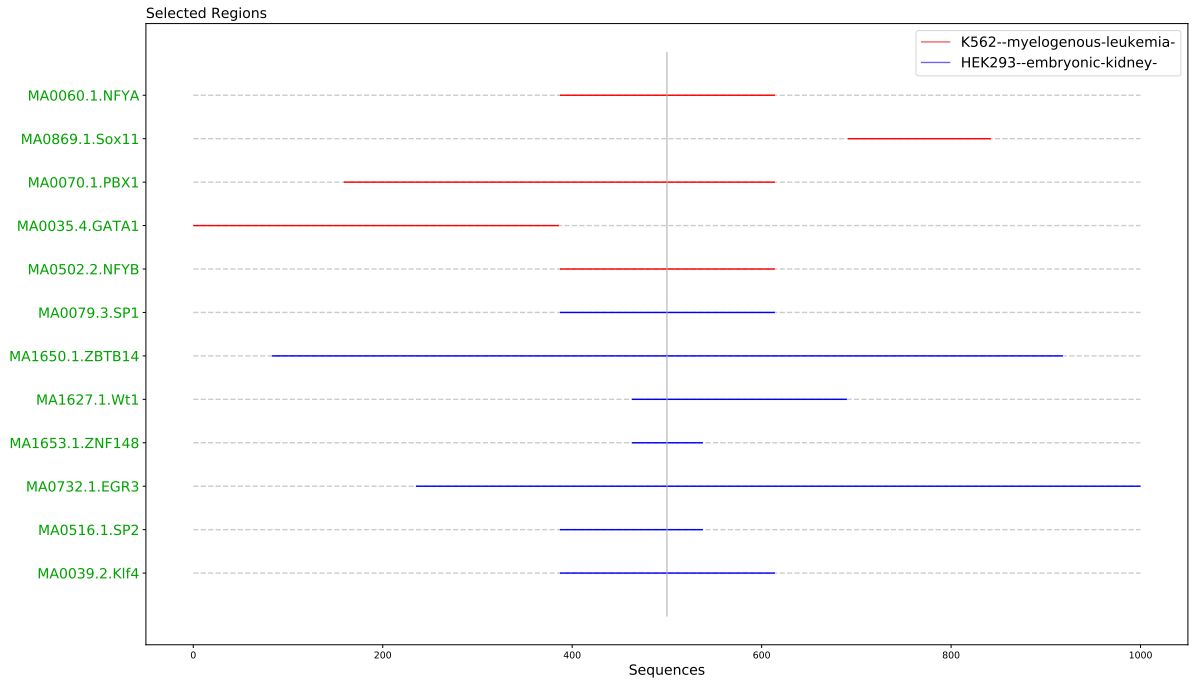

FIGURE 7.4 – *Représentation des meilleures variables sélectionnées pour distinguer la fixation du facteur de transcription SP2 dans les types cellulaires K562 ($Y = 1$ en rouge) et HEK293 ($Y = 0$ en bleu).*
Les traits rouges et bleus indiquent quelle région de la séquence a été sélectionnée : rouge si le coefficient associé est positif et bleu s'il est négatif. Les variables sont triées par ordre d'importance, avec en haut les variables les plus importantes avec un coefficient positif, en bas les variables les plus importantes avec un coefficient négatif.

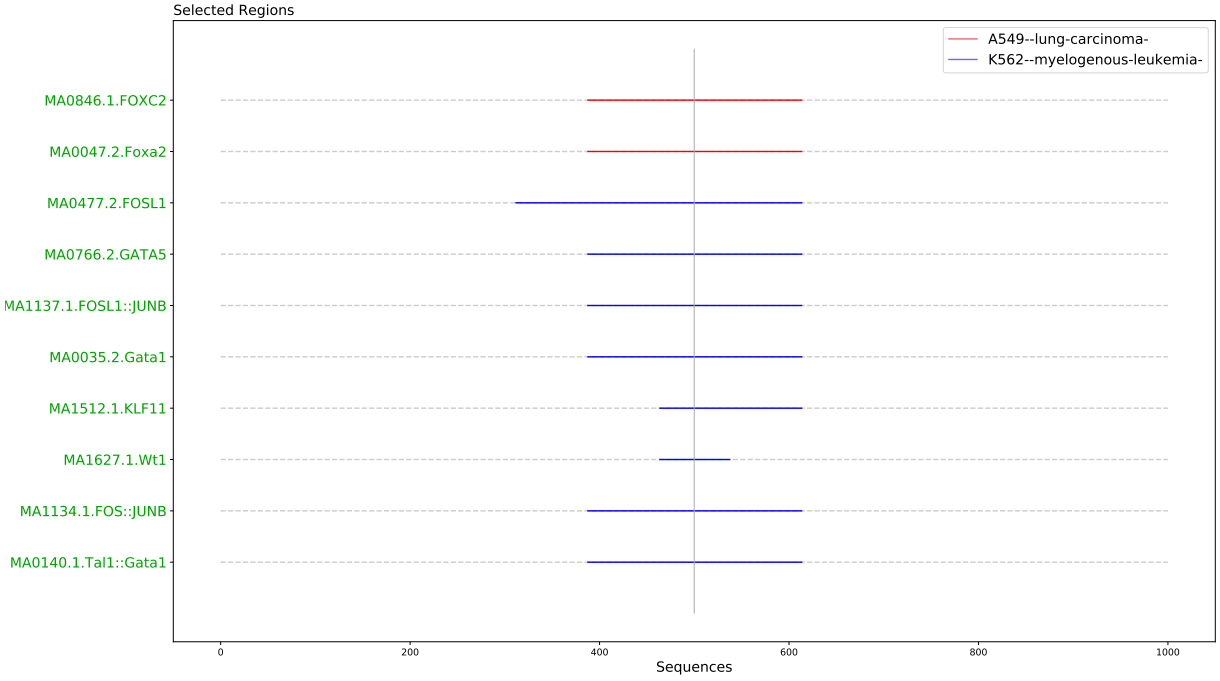

Figure 7.5 – *Représentation des meilleures variables sélectionnées pour distinguer la fixation du facteur de transcription JUNB dans les types cellulaires A549 ($Y = 1$ en rouge) et K562 ($Y = 0$ en bleu).*
Les traits rouges et bleus indiquent quelle région de la séquence a été sélectionnée : rouge si le coefficient associé est positif et bleu s'il est négatif. Les variables sont triées par ordre d'importance, avec en haut les variables les plus importantes avec un coefficient positif, en bas les variables les plus importantes avec un coefficient négatif.

Chapitre 8

Combinaison des différents types d'information (TFscope)

Dans ce chapitre nous présentons le modèle appelé TFscope qui combine les trois types d'information décrits dans les chapitres précédents pour discriminer les séquences :

— Les spécificités nucléotidique du site de fixation, apprises avec l'approche Motif Discriminant (DM) (voir 6.1),

— L'environnement nucléotidique des séquences fixées, capturé par la méthode DExTER (voir 3.5),

— La présence et la position des facteurs de transcription coopérants avec le TF cible, identifiés grâce à la méthode PosTFcoop (voir 7).

Les variables issues de ces trois méthodes sont jointes dans un modèle linéaire logistique pénalisé, qui s'écrit :

$$\ln \frac{\mathbb{P}(Y = 1 \mid X)}{1 - \mathbb{P}(Y = 1 \mid X)} = \beta_0 + \beta_1 S + \sum_{k=1}^{d} \delta_k D_k + \sum_{j=1}^{r} \gamma_j R_j, \tag{8.1}$$

avec $\beta_0, \beta_1, \delta_1, \ldots, \delta_d, \gamma_1, \ldots, \gamma_r$ les paramètres du modèle, $X = (S, D_1, \ldots, D_d, R_1, \ldots, R_r)$, r le nombre de PWM considérées dans la base de données JASPAR (en pratique $r = 1011$) et d le nombre de variables retenus pas DExTER. Les R_j représentent les variables de PosTFcoop, D_k les variables de DExTER, et S résume l'information contenue dans DM. Pour cette dernière variable, on apprend un modèle DM séparément puis on utilise le vecteur de prédiction de modèle. S peut donc être vu comme le score du motif discriminant dans chaque séquence.

À noter que dans toute cette section, nous utilisons des segmentations en 13 séquences élémentaires pour PosTFcoop et DExTER, de manière à ce que ces deux méthodes étudient exactement les mêmes sous-séquences. Les séquences originales faisant 1000bp, découper en 13 séquences élémentaires revient à former des sous-séquences élémentaires de taille 77. Cela permet un compromis intéressant entre temps de calcul et précision. Tous les modèles sont ajustés sur un ensemble d'apprentissage représentant 70% des séquences disponibles dans chaque expérience et testés sur 30%. Lors de la phase d'apprentissage, le calcul du λ_{min} (voir section 2.4) est fait sur la base d'une validation croisée à 10 ensembles et l'AUROC est utilisée comme fonction de perte pour déterminer la valeur de λ_{min}.

8.1 Représentation des contributions de chaque type d'information

Le modèle TFscope permet d'intégrer les 3 types d'information considérés en un même modèle. Outre l'amélioration des capacités prédictives du modèle, son objectif premier est de pouvoir quantifier l'apport de chaque type d'information afin de déterminer les plus importantes pour expliquer les différences de fixation entre types cellulaires. Pour cela, les performances de ce modèle en terme d'AUROC sont comparées à celles de plusieurs autres modèles : le modèle DM+8 tout seul, qui évalue l'importance du motif discriminant seul (avec 8 positions flanquantes), le modèle DExTER seul, qui évalue l'importance de l'environnement nucléotidique, et 3 autres modèles TFscope incomplets. Ces trois modèles sont obtenus à partir du modèle TFscope complet en mettant à zéro, soit le coefficient associé à la variable DM+8 (TFscope w/o DM), soit les coefficients associés aux variables de DExTER (TFscope w/o Nucl. env.), soit les coefficients associés aux variables de PosTFcoop (TFscope w/o Cofactors). Ces trois derniers modèles permettent d'évaluer ce qui, dans le modèle TFscope complet, semble être le plus important pour discriminer les séquences. À titre d'information, on ajoute également à toutes ces comparaisons les performances de la PWM originale ainsi que celles du modèle DM+0 (c-à-d sans les positions flanquantes). Toutes ces comparaisons sont résumées dans un graphique de type radar, ce qui permet de comparer facilement l'apport de chaque type d'information en fonction des expériences.

Un premier exemple de radar est en Figure 8.1. Il mesure les performances prédictives

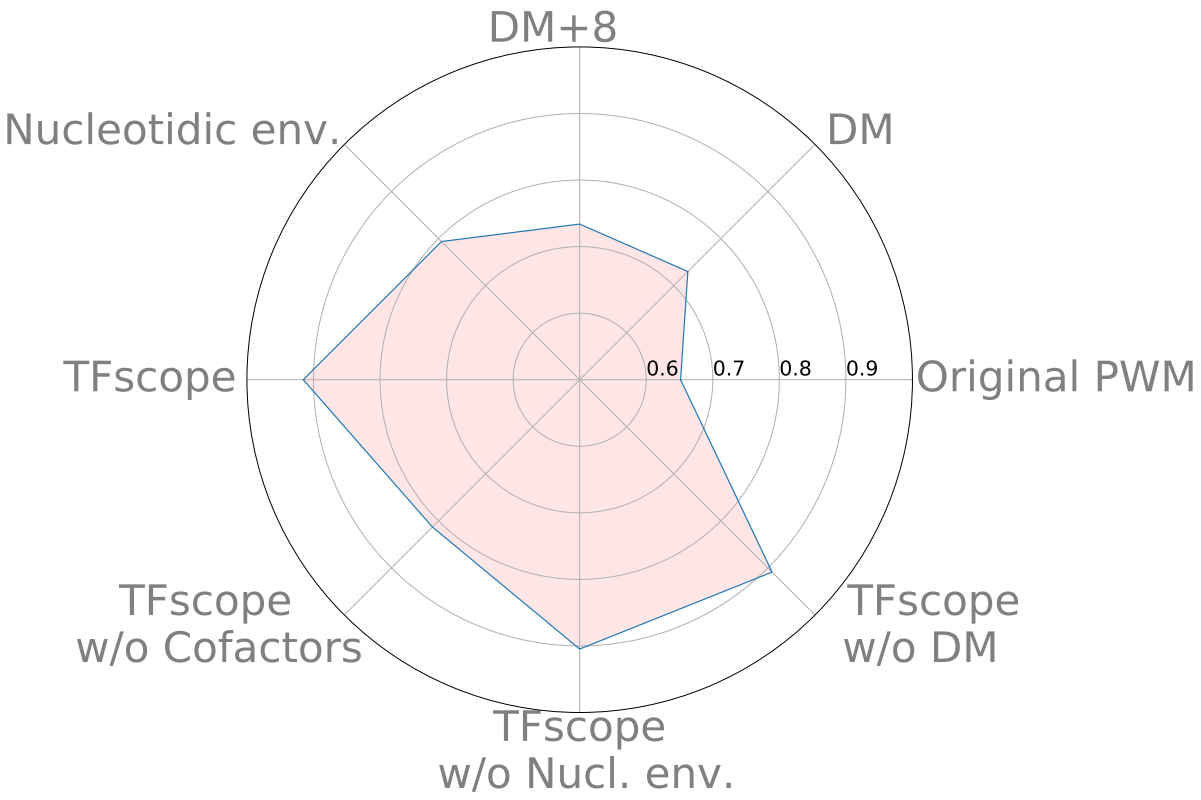

Figure 8.1 – *Graphique radar mesurant les AUROC des différentes approches et leurs combinaisons.*
Expérience : facteur de transcription ESR1 entre les types cellulaires T47D ($Y = 1$) et MCF7 ($Y = 0$). Les AUROC sont obtenues sur un même ensemble de test représentant 30% des séquences. « Original PWM » correspond à l'utilisation de la PWM présente dans JASPAR et utilisée dans Unibind (MA0112.3 ESR1). « DM » et « DM+8 » sont les modèles « motif discriminant » avec respectivement 0 et 4 positions flanquantes de chaque coté. « Nucleotidic env. » correspond aux variables de DExTER et « Cofactors » aux variables de PosTFcoop.

sur l'expérience de fixation du facteur de transcription ESR1 dans les types cellulaires T47D ($Y = 1$) et MCF7 ($Y = 0$) (ces deux expériences ont été traitées avec progestérone). Sur ce radar on peut observer que TFscope obtient une AUROC de 0.91. Il semble que l'information la plus importante soit apportée par les cofacteurs du TF cible, puisque la plus grande chute d'AUROC est obtenue lorsque l'on enlève les variables de PosTFcoop. De plus, on peut également observer que l'information d'environnement nucléotidique (variables de DExTER) est plus discriminante que l'information de spécificités de fixation apportée par DM (cf. logo en Figure 6.1).

La Figure 8.2 montre les résultats obtenus sur 8 autres expériences décrites ci-dessous. Les radars A, B, C et D correspondent à des expériences déjà décrites dans le chapitre Motif Discriminant pour A et B (sections 6.4 et 6.6) et dans le chapitre portant sur PosTFcoop pour C et D (7.4). En annexe se trouvent les représentations logo du modèle DM ajusté sur chacune de ces 8 expériences, ainsi que les graphiques représentant les meilleures variables sélectionnées, et leur région associée. La sélection des meilleures variables est réalisée en utilisant la procédure décrite à la section 7.4.

A : USF2, HepG2/GM12878. Comme nous l'avons expliqué en section 6.4, le modèle DM obtient une AUROC très supérieure à celle de la PWM présente dans JASPAR. L'environnement nucléotidique semble être l'information la moins importante pour expliquer les différences de fixation entre les deux types cellulaires, tandis que le motif cible semble contenir l'information la plus importante.

B : NFIC, SK-N-SH/K562. Nous nous étions intéressé à cette expérience dans l'étude des positions flanquantes dans DM : en effet l'écart d'AUROC entre « DM » et « DM+8 » est grand (0.68 et 0.76 respectivement).

C : SP2, K562/HEK293. Les informations apportées par les différentes méthodes semblent équivalentes, avec des AUROC qui avoisinent les 0.80 pour presque chaque méthode, et peu de pertes, quelle que soit l'information qui est retirée à TFscope.

D : JUNB, A549/K562. On obtient ici quelque chose d'assez similaire au radar A : beaucoup d'apport avec DM par rapport à la PWM originale, mais ici l'environnement nucléotidique contient un peu plus d'information que pour le radar A.

E : SOX2, TT/HNSC. Dans cette expérience, seuls les cofacteurs du TF cible apportent de l'information. Ce modèle obtient une AUROC de 0.90 alors que DM n'apporte pratiquement aucune information, et l'environnement nucléotidique atteint pratiquement 0.70 d'AUROC.

F : FOXA1, MCF-7/zr75-1. Ici, l'information est partagée entre DM et les cofacteurs (AUROC >0.80). L'environnement nucléotidique n'apporte que peu d'information (AUROC < 0.70) en comparaison.

G : CTCF, HUES64/HAP1. Contrairement au radar F, c'est l'environnement nucléotidique qui apporte le plus d'informations sur cette expérience. Bien que les performances du modèle TFscope global soient assez moyennes (AUROC = 0.77) comparées à d'autres expériences.

H : AR, VCaP/LNCaP. Sur ce dernier radar on peut observer que seul le motif discriminant apporte de l'information au modèle (AUROC = 0.79). Les autres types de variables ne semblent pas être informatives pour la classification, puisque le modèle TFscope, en sélectionnant des variables supplémentaires, perd en performance. On est donc vraisemblablement dans un cas de surapprentissage pour TFscope.

	DM+8	DExTER	PosTFcoop
DM+8	1	-0.85	-0.37
DExTER		1	0.09
PosTFcoop			1

Table 8.1 – *Matrice de corrélation des AUROC normalisées des trois méthodes « DM+8 », « DExTER » et « PosTFcoop ».*

DExTER et PosTFcoop sont très peu corrélées, alors qu'on observe une forte anti-corrélation entre DM+8 et DExTER ainsi qu'une anti-corrélation plus faible entre DM+8 et PosTFcoop.

8.2 Analyse des résultats sur les 502 expériences

8.2.1 Performances

La Figure 8.3 représente les distributions des AUROC des différentes méthodes présentées et leurs combinaisons. Ces distributions, obtenues sur 502 expériences de classification, montrent plus globalement certaines observations qui avaient déjà été faites dans les chapitres précédents. En dehors du fait que le motif discriminant obtient de meilleures performances que la PWM originale (ce que l'on avait déjà observé sur la Figure 6.2), on voit par exemple que le motif discriminant (DM) et l'environnement nucléotidique apportent des performances équivalentes en moyenne, mais que leur combinaison permet de meilleures prédictions. Au total, l'information la plus importante en terme d'AUROC semble être liée à l'information de position des cofacteurs. Enfin, avec la distribution associée à TFscope, on observe également que la combinaison des trois types d'information permet d'atteindre de meilleures performances.

Une autre manière d'étudier les liens existants entre les différents types d'information est de calculer des mesures de corrélation entre les AUROC des différents modèles. La Table 8.1 est une matrice de corrélation des AUROC de « DM+8 », « DExTER » et « PosTFcoop ». Ces deux dernières sont très peu corrélées, mais on observe en revanche une forte anti-corrélation entre « DM+8 » et « DExTER » ainsi qu'une anti-corrélation plus faible entre « DM+8 » et « PosTFcoop ». Ce point est intéressant car il semble indiquer qu'il existe en fait deux catégories d'expériences de classification entre types cellulaires : celles que l'on peut discriminer sur la base du motif cible, et celles que l'on peut discriminer sur la base de l'environnement nucléotidique du motif.

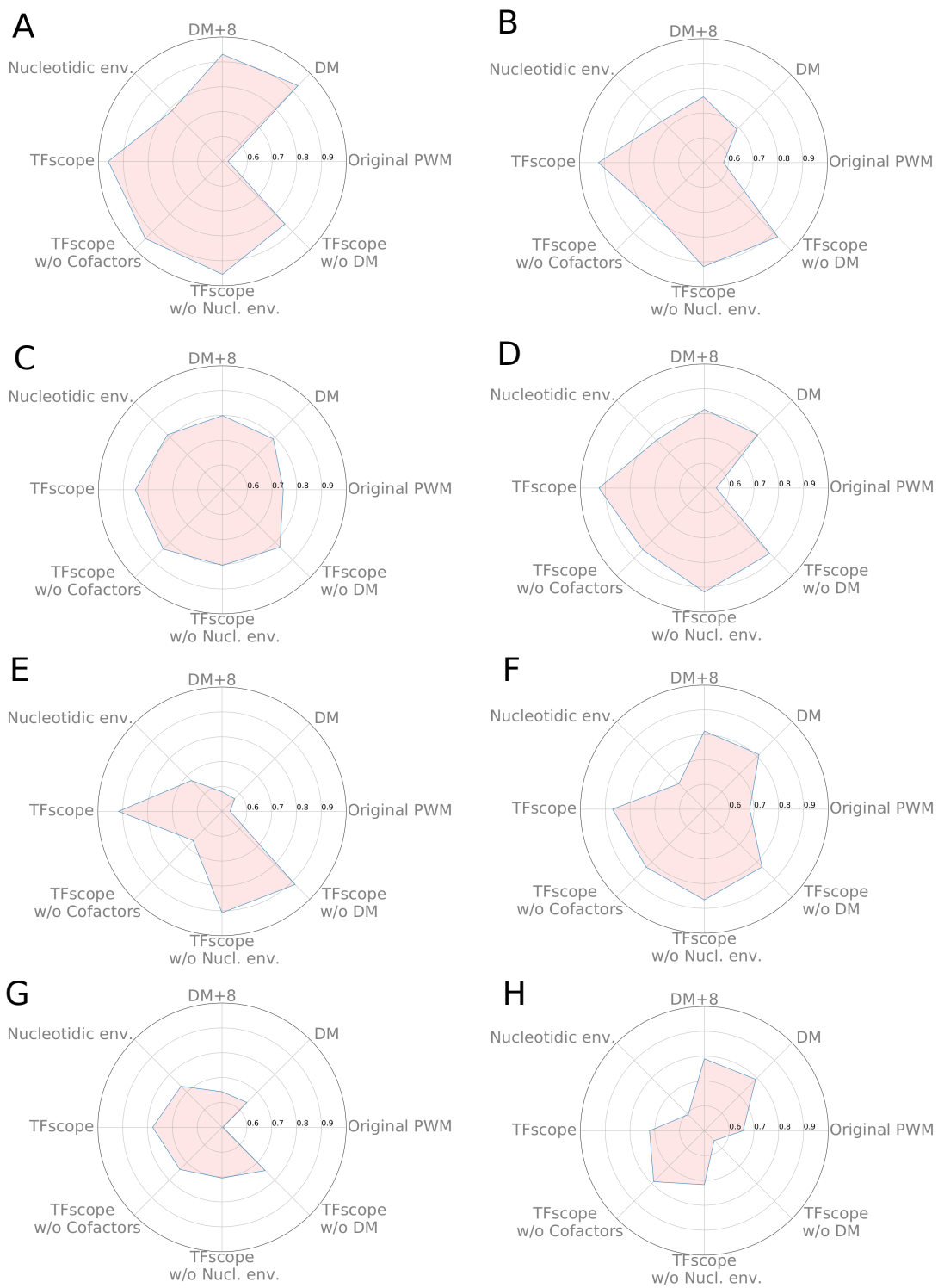

Figure 8.2 – *Graphiques radars de différentes expériences.*
A : USF2, HepG2/GM12878. **B** : NFIC, SK-N-SH/K562. **C** : SP2, K562/HEK293. **D** : JUNB, A549/K562. **E** : SOX2, TT/HNSC. **F** : FOXA1, MCF-7/zr75-1. **G** : CTCF, HUES64/HAP1. **H** : AR, VCaP/LNCaP.

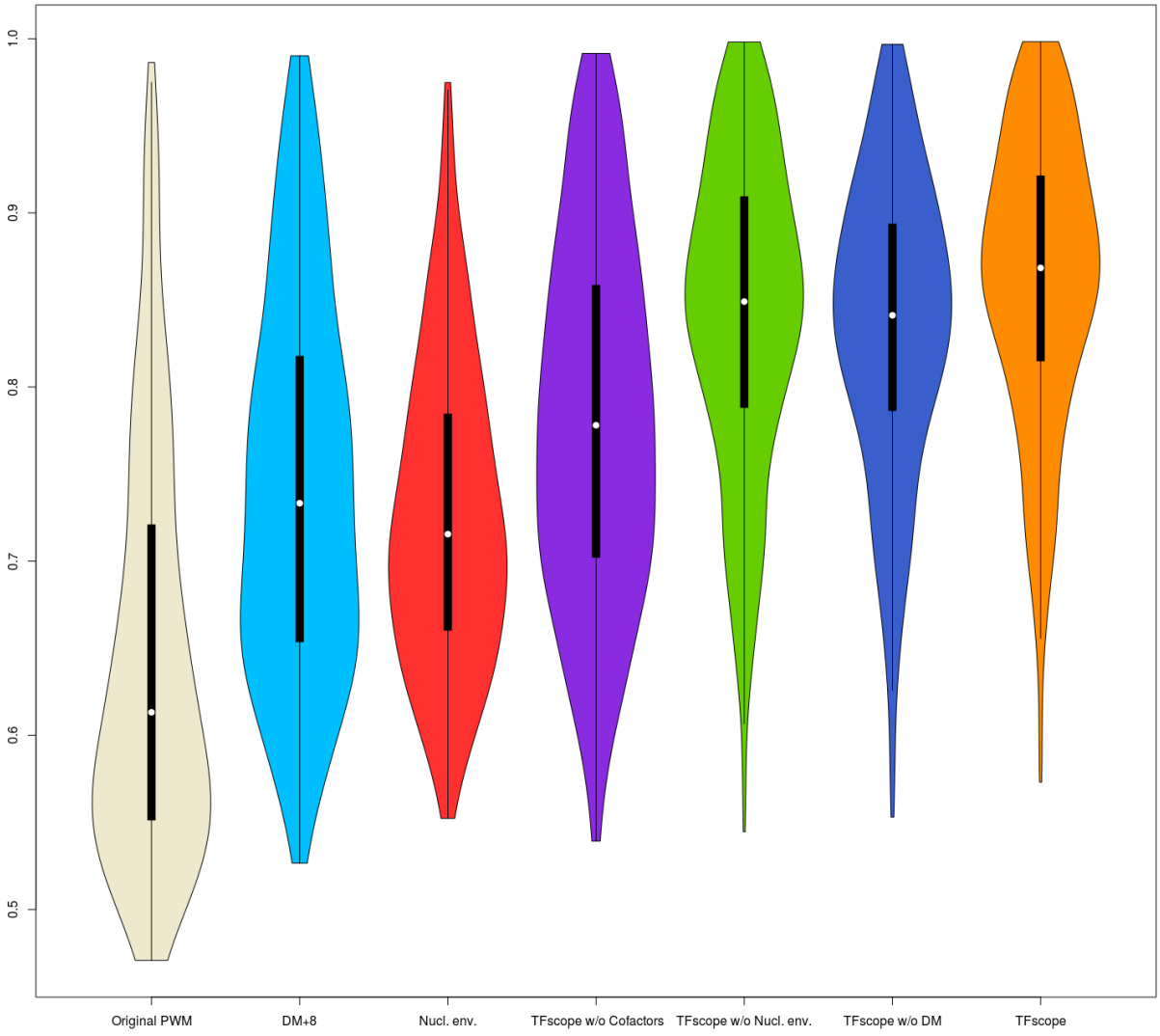

Figure 8.3 – *Représentation en violon des distributions des AUROC de différentes méthodes et leurs combinaisons réalisées sur 502 expériences.*
Les AUROC sont calculées sur un même ensemble de test indépendant représentant 30% des séquences.

8.2.2 Apport des différents types d'information

Afin d'aller plus loin dans l'exploration des résultats obtenus par notre méthode, et de mieux comprendre les spécificités génomiques qui expliquent la différence de fixation entre types cellulaires, nous avons essayé de classer chaque expérience en fonction du type d'information qui semble être le plus important pour discriminer les deux classes de séquences.

Pour cela nous avons utilisé un K-means (voir section 2.3.3.2) pour regrouper les expériences en fonction du type d'information qui semble le plus important. Pour chaque expérience, nous prenons les AUROC de « TFscope w/o cofactors », « TFscope w/o Nucleotidic. env. » et « TFscope w/o DM » et nous les soustrayons une à une à l'AUROC de TFscope. Ainsi, la valeur obtenue par le calcul de $AUROC_{\text{TFscope}} - AUROC_{\text{TFscope w/o cofactors}}$ mesure le poids des variables des cofacteurs dans le modèle TFscope. Nous avons alors pour chaque expérience trois valeurs de pertes associées aux trois types d'information, et nous avons lancé un K-means sur ces données. La valeur de K (le nombre de classes), a été choisie expérimentalement de la manière suivante. On attend au maximum 7 classes : 3 classes avec un seul type d'information importante, plus 3 classes avec deux informations importantes, plus une classe avec 3 informations importantes. Nous avons donc lancé un premier K-means avec K=7. En étudiant les centres des groupes (centroïdes) déterminés par l'algorithme, seuls 5 groupes différents semblaient pourtant apparaître : la classe des trois informations conjointes et la classe combinant DM et l'environnement nucléotidique étaient manquantes. Le fait que cette dernière classe ne ressorte pas du K-means va dans le sens de l'anti-corrélation observée entre les AUROC de ces méthodes en Table 8.1. Nous avons donc lancé un second K-means avec $K = 5$. La Figure 8.4 montre le nombre d'expériences dans chacune des 5 classes. On peut observer que plus de la moitié des expériences sont annotées dans la classe « Cofactors » (= 266) et que cette information semble donc la plus importante pour la majorité des expériences.

8.2.3 Préférence de fixation des facteurs de transcription

Nous avons ensuite voulu identifier s'il ressortait des préférences de fixation propres à chaque facteur de transcription. Certains TF n'étant décrits que dans peu d'expériences, nous nous sommes restreints aux TF présents dans au moins 5 expériences. Cela représente 27 facteurs de transcription d'intérêt sur les 134 à l'origine dans 241 expériences de

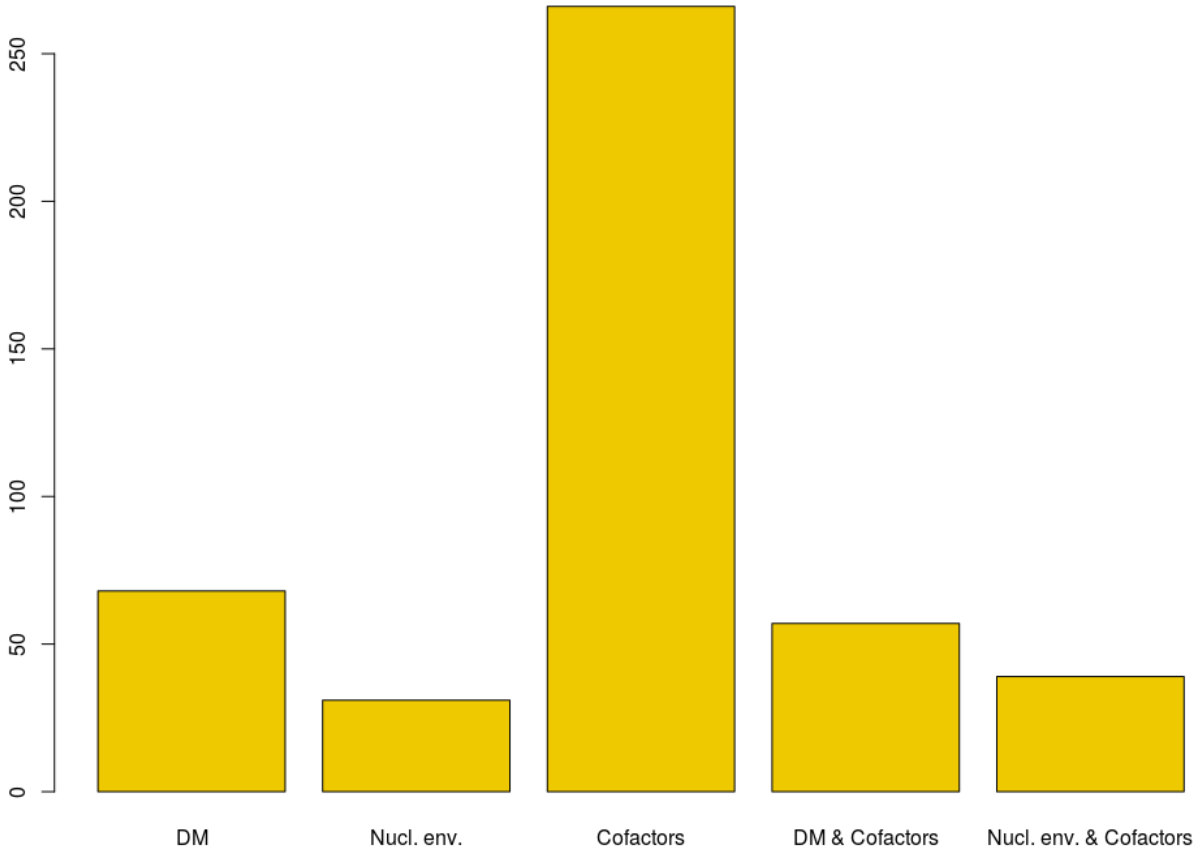

FIGURE 8.4 – *Graphique en barre du nombre d'expériences dans chacun des groupes.* Chaque barre renseigne le nombre d'expériences dans chacun des groupes déterminés par K-means pour $K = 5$.

	DM	Nucl. env.	Cofactors	DM & Cofactors	Nucl. env. & Cofactors	Total	p.valeur
AR	6	3	5	6	2	22	5.57×10^{-1}
CEBPA	0	1	2	4	1	8	2.18×10^{-1}
CTCF	0	**5**	0	0	1	6	3.50×10^{-3}
ERG	0	1	3	0	1	5	1.99×10^{-1}
ESR1*	2	1	**18**	2	3	26	4.83×10^{-8}
FLI1	0	2	3	0	1	6	2.25×10^{-1}
FOXA1*	5	2	**17**	3	2	29	1.20×10^{-5}
FOXA2	3	0	2	0	2	7	2.72×10^{-1}
GATA2	1	0	3	3	1	8	3.42×10^{-1}
GATA3	0	0	**5**	0	**3**	8	1.01×10^{-2}
MAX	2	1	4	2	0	9	2.98×10^{-1}
MYCN	**4**	1	0	0	0	5	1.73×10^{-2}
MYC	2	0	6	1	0	9	8.03×10^{-3}
NR2F2*	0	0	**5**	0	0	5	4.99×10^{-4}
NR3C1	0	1	4	3	0	8	8.28×10^{-2}
RELA	1	0	**5**	0	1	7	1.53×10^{-2}
REST	0	0	3	1	1	5	1.99×10^{-1}
RUNX1*	0	1	**9**	1	0	11	2.25×10^{-5}
SOX2	1	0	3	0	1	5	1.99×10^{-1}
SPI1	2	2	1	1	1	7	9.30×10^{-1}
SRF	2	0	5	0	0	7	8.26×10^{-3}
STAT3	0	0	3	2	1	6	2.25×10^{-1}
TCF12	1	0	**4**	0	0	5	1.73×10^{-2}
TCF7L2	0	0	**5**	0	1	6	3.50×10^{-3}
TEAD4	1	0	6	2	0	9	8.03×10^{-3}
TP53	4	0	2	1	0	7	9.15×10^{-2}
USF2	**4**	0	1	0	0	5	1.73×10^{-2}

Table 8.2 – *Table du nombre d'expériences représentant chaque facteur de transcription dans les 5 groupes. La dernière colonne est la p-valeur obtenue avec un test du χ^2 sur chaque ligne, avec comme hypothèse nulle une distribution uniforme dans les classes. Les astérisques représentent les TF pour lesquels la p-valeur est inférieure au seuil de 5% corrigé avec Bonferonni. En gras, les TF pour lesquels une ou deux classes semblent majoritaires*

classification.

La Table 8.2 montre le nombre d'expériences dans chaque classe pour chaque facteur de transcription. Certains TF (en gras) semblent plus souvent associés à une des classes qu'à une autre. C'est le cas par exemple de CTCF, pour lequel, dans 5 des 6 expériences où il intervient, l'environnement nucléotidique ressort comme information la plus importante. Malgré le faible nombre de réplicats, certains TF obtiennent même des p-valeurs significatives pour un test du χ^2 (noté d'une astérisque) avec une correction de test multiples de Bonferroni [128]. C'est le cas des 4 facteurs de transcription ESR1, FOXA1, NR2F2 et RUNX1, qui sont tous associés au groupe « Cofactors ».

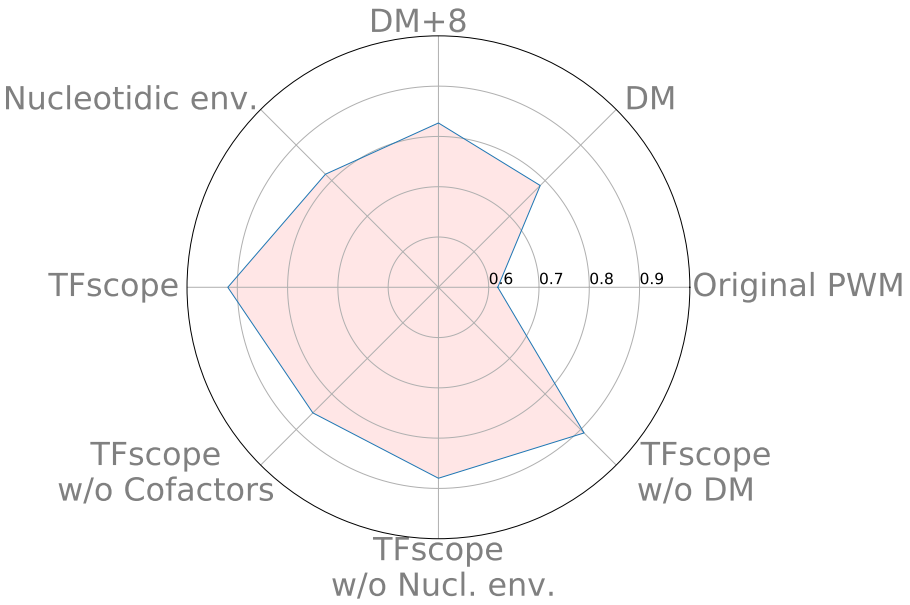

Figure 8.5 – *Graphique radar des AUROC des différentes approches ajustées sur l'expérience : CEBPG HepG2/K562.*

8.3 Comparaison avec l'information d'ouverture de la chromatine

Afin d'apporter une interprétation biologique supplémentaire, nous allons ici comparer notre modèle TFscope avec l'information d'ouverture de la chromatine. Les données DNase permettent d'obtenir cette information tout au long de la séquence [129]. Nous avons choisi une expérience parmi notre jeu de données de 502 expériences de classification entre types cellulaires. L'expérience étudiée ici correspond à la fixation du facteur de transcription CEBPG entre les types cellulaires HepG2 ($Y = 1$) et K562 ($Y = 0$), les performances obtenues avec nos différentes approches sont disponibles en Figure 8.5. Nous avons utilisé les données DNAse issues de « Roadmap epigenomics project » [99] associées d'une part au type cellulaire HepG2 et d'autre part à K562. Les données de DNAse le long des séquences ont été obtenues avec le fichier BED de CEBPG :HepG2/K562 en utilisant la commande « bwtool extract » [130]. Une fois ces données récupérées tout au long de la séquence, nous avons calculé la moyenne des données de DNase par séquence afin d'obtenir une valeur pour chaque séquence et chaque type cellulaire. On obtient ainsi deux variables qui correspondent à la moyenne des valeurs de DNase le long des séquences,

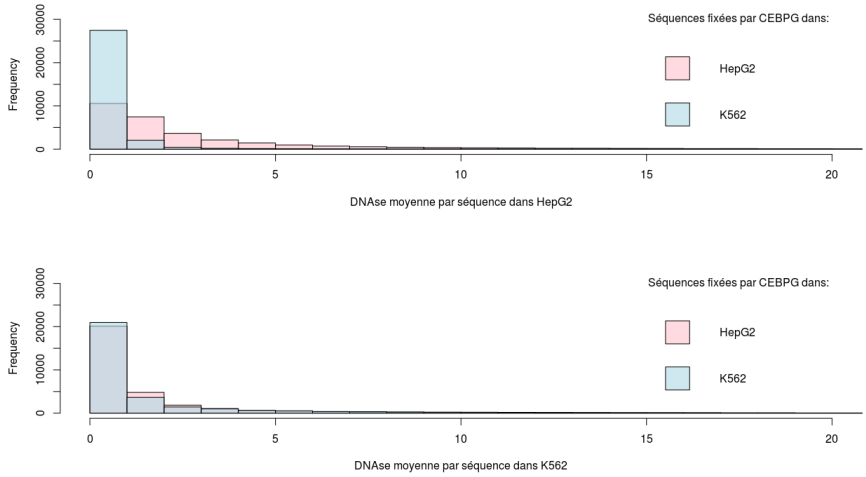

Figure 8.6 – *Histogramme des distributions de DNAse dans les séquences fixées par CEBPG.* Les histogrammes représentent les distributions de DNAse (moyenne sur chaque séquence) sur les séquences fixées par CEBPG dans HepG2 (en rouge) et dans K562 (en bleu). La figure du haut correspond aux données de DNAse dans HepG2 et la figure du bas aux données de DNAse dans K562.

l'une observée dans HepG2 et l'autre dans K562. La Figure 8.6 représente les distributions de ces deux variables dans chaque classe. On peut voir sur ces figures des différences dans les distributions, notamment la DNAse de HepG2 qui prend des valeurs plus hautes dans les séquences fixées par CEBPG dans HepG2 par rapport à celles fixées dans K562. Alors que les distributions de DNAse de K562 semblent similaires entre les deux classes de séquences. Nous avons ajusté deux modèles linéaires logistiques avec ces deux variables. Le modèle avec la variable de DNase dans HepG2 obtient une AUROC de 0.846, ce qui est au dessus des performances obtenues par DM+8 (AUROC=0.827) et DExTER (AUROC=0.816). En revanche, le modèle avec la variable de DNase dans K562 obtient une AUROC de 0,518, ce qui s'explique par les histogrammes de la Figure 8.6. L'ouverture de la chromatine dans K562 n'est alors pas informative pour distinguer les classes. Il semble donc que l'information d'ouverture de la chromatine puisse distinguer correctement les classes dans certains cas, mais que cela n'est pas systématique.

Sur cette même expérience, TFscope obtient une AUROC de 0.918, il semble donc que celui-ci contient de l'information qui n'est pas présente dans les données de DNase. Nous avons ensuite lancé un second modèle, cette fois-ci avec les deux variables DNase

et l'ensemble des variables présentes dans TFscope. L'ajout des variables DNase permet d'augmenter les performances de prédiction et d'obtenir une AUROC de 0.935. Ces résultats suggèrent donc que l'information contenue dans TFscope diffère de l'information de DNase dans les séquences, a minima pour cette expérience en particulier. Il sera bien-sur nécessaire de répéter cette comparaison sur plus d'expériences afin de montrer avec plus de certitudes que ces deux informations sont distinctes et de pouvoir comparer ces informations de manière globale.

8.4 Étude de cas extrêmes

Dans ce chapitre, nous avons présenté TFscope, un modèle qui combine trois types d'information afin de distinguer des classes de séquences et de pouvoir identifier quelle est l'information génomique qui semble la plus importante dans chaque cas. La méthode présente des résultats souvent bons en terme d'AUROC, avec parfois même des valeurs très proches de 1 (Figure 8.3). On notera que dans ces cas, même le modèle le plus simple constitué de la PWM présente dans JASPAR obtient des AUROC élevées. Cela concerne 20 expériences avec une AUROC supérieure à 0.90, dont 5 supérieures à 0.95. Ces performances extrêmes sont assez surprenantes et on peut légitimement se demander si elles ne sont pas le reflet d'un problème expérimental ou d'un biais technologique dans une ou deux des expériences de ChIP-seq.

Par exemple intéressons nous à l'expérience portant sur la fixation de FOS entre les types cellulaires GM12878 ($Y = 1$) et MCF7 ($Y = 0$), qui est l'expérience où le modèle avec la PWM de JASPAR obtient la meilleure AUROC (0.98), TFscope atteignant une AUROC de 0.99. Il est clair que le motif seul permet de discriminer presque parfaitement les séquences, ce qui est confirmé par l'étude du motif discriminant, où l'on retrouve le motif original (MA0476.1 FOS, voir Figure 8.7.A), associé à des coefficients négatifs (voir Figure 8.7.B). Cela traduit le fait que ce motif (TGANTCA) se trouve en grande majorité dans le type cellulaire MCF7 et non dans GM12878. Comparer les ChIP-seq d'où est issue cette classification n'apporte pas d'informations supplémentaires : Unibind renseigne des p-valeurs inférieures à 10^{-30} dans les deux ChIP-seq pour mesurer la significativité des bornes génomiques apprises par ChIP-eat. De plus, les bornes génomiques relatives au score dans ces expériences de ChIP-seq sont proches et les expériences sont classifiées comme robustes. Il faut noter cependant que les motifs utilisés dans Unibind ont été préalablement optimisés avec la méthode DAMO. Un motif optimisé par DAMO trop éloigné du motif original présent dans JASPAR pour l'expérience dans le type cellulaire GM12878, pourrait donc expliquer ces valeurs extrêmes. Cependant, l'expérience étant classifiée comme robuste, Unibind considère que le motif optimisé par DAMO est suffisamment proche du motif original. Afin de vérifier ce point, nous avons analysé les PWM optimisées avec DAMO dans Unibind (voir Figures 8.7.C et 8.7.D). Celles-ci montrent que l'on retrouve le motif canonique de FOS seulement dans les séquences de MCF7. Il semble donc bien que les séquences spécifiques à GM12878 soient dépourvues de ce motif, bien qu'elles obtiennent un score suffisamment bon pour être considérées comme de vrais

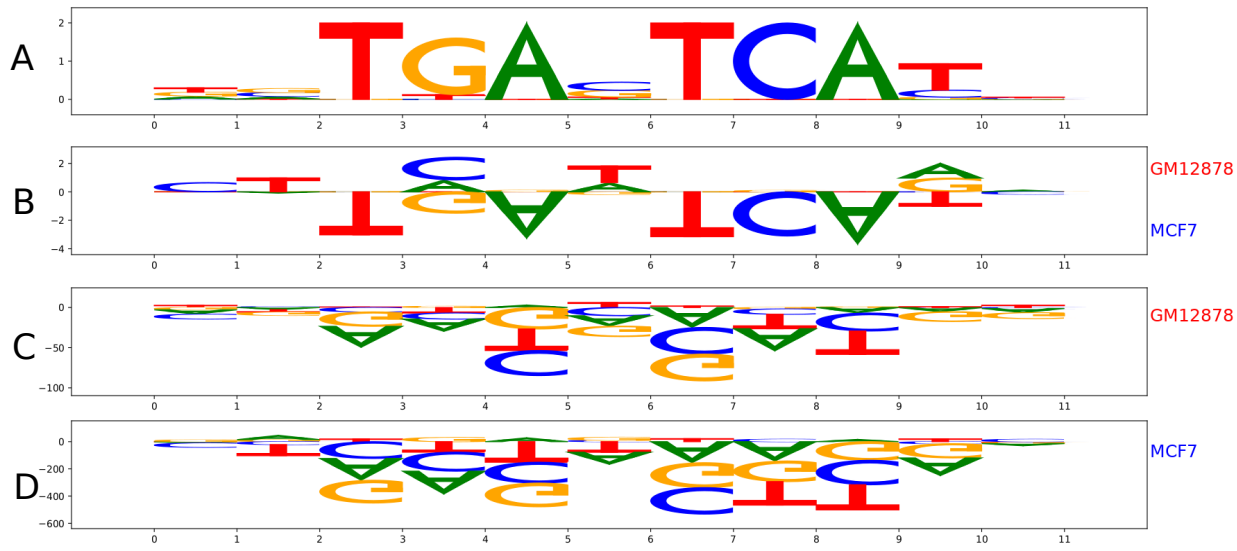

Figure 8.7 – *Représentation de différents logos des motifs de FOS dans GM12878 et MCF-7.* **A** : Logo du motif MA0476.1 FOS présent dans JASPAR. **B** : logo de DM entraîné pour distinguer la fixation de FOS entre GM12878 ($Y = 1$) et MCF7 ($Y = 0$). **C** : logo de la PWM optimisée par DAMO dans GM12878. **D** : logo de la PWM optimisée par DAMO dans MCF7. Sur la figure D, le motif canonique se retrouve dans les poids positifs de la matrice ce qui n'est pas vrai sur la figure C. Ces poids étant petits par rapport à la valeur absolue des poids négatifs ils sont assez peu visibles sur le logo. Cependant, dans la majorité des cas ils peuvent être retrouvés en regardant le nucléotide manquant dans les poids négatifs.

positifs par Unibind.

8.5 Application aux spécificités de traitements

Dans cette section, nous nous intéressons à une problématique un peu différente de celle traitée jusqu'à présent. En effet, jusqu'ici, la condition qui variait d'une classe à l'autre était le type cellulaire. Maintenant nous proposons de discriminer des classes de séquences de même type cellulaire mais où le traitement utilisé sur les expériences diffère. Nous nous plaçons donc dans un problème de classification où toutes les séquences sont fixées par un même facteur de transcription cible, dans un même type cellulaire, mais où les classes diffèrent par le traitement utilisé. Pour cela, nous utilisons un jeu de données différent de celui présenté dans la problématique au Chapitre 5 mais extrait une fois encore des données Unibind. Les mêmes procédures de traitements des données, décrites au chapitre 5.2, ont été appliquées. Le jeu de donnée contient 15 expériences sur 8 facteurs de transcriptions différents, dans 7 types cellulaires. La Figure 8.8 représente les distributions des AUROC obtenues sur l'ensemble de test avec les différentes méthodes associées à TFscope. On voit sur cette figure que les résultats semblent en moyenne supérieurs en terme de performance que ce que l'on obtient sur les spécificités cellulaires. Il faut cependant garder à l'esprit que ces résultats sont obtenus sur un échantillon de données bien plus petit. L'ordre de performances des méthodes semble être conservé, mis à part pour l'information d'environnement nucléotidique qui semble contenir moins d'informations pour distinguer les séquences. Là encore, cela peut être un effet du faible nombre d'expériences.

Prenons pour exemple, le problème de classification des séquences fixées par ESR1 dans le type cellulaire T47D (voir Figure 8.9). La classe positive correspond aux séquences dans les cellules sans traitement tandis que la classe négative correspond aux séquences dans les cellules avec un traitement progestérone. TFscope obtient une AUROC de 0,95. Les types d'information les plus importants sont DM et les cofacteurs. On peut ensuite observer sur la Figure 8.10 que les régions d'intérêts semblent être courtes et centrées autour du site de fixation du TF cible. Les variables les plus importantes sont toutes liées aux cofacteurs, mis à part DM. Les motifs associés aux facteurs de transcription cible (ESR1) et ceux de la même famille (ESR2) ont des coefficient positifs, de même que les motifs associés à la famille HOX (HOXB13) et Foxk1. Le modèle va donc avoir tendance à prédire l'absence de traitement quand ces motifs sont rencontrés. À noter que les motifs HOXB13 et Foxk1 ont certaines ressemblances puisqu'ils comportent beaucoup de répétitions du nucléotide A et un T. À l'inverse, le modèle prédit le traitement progestérone quand les motifs associés à AR et NR3C2 sont observés dans les séquences (coefficients négatifs).

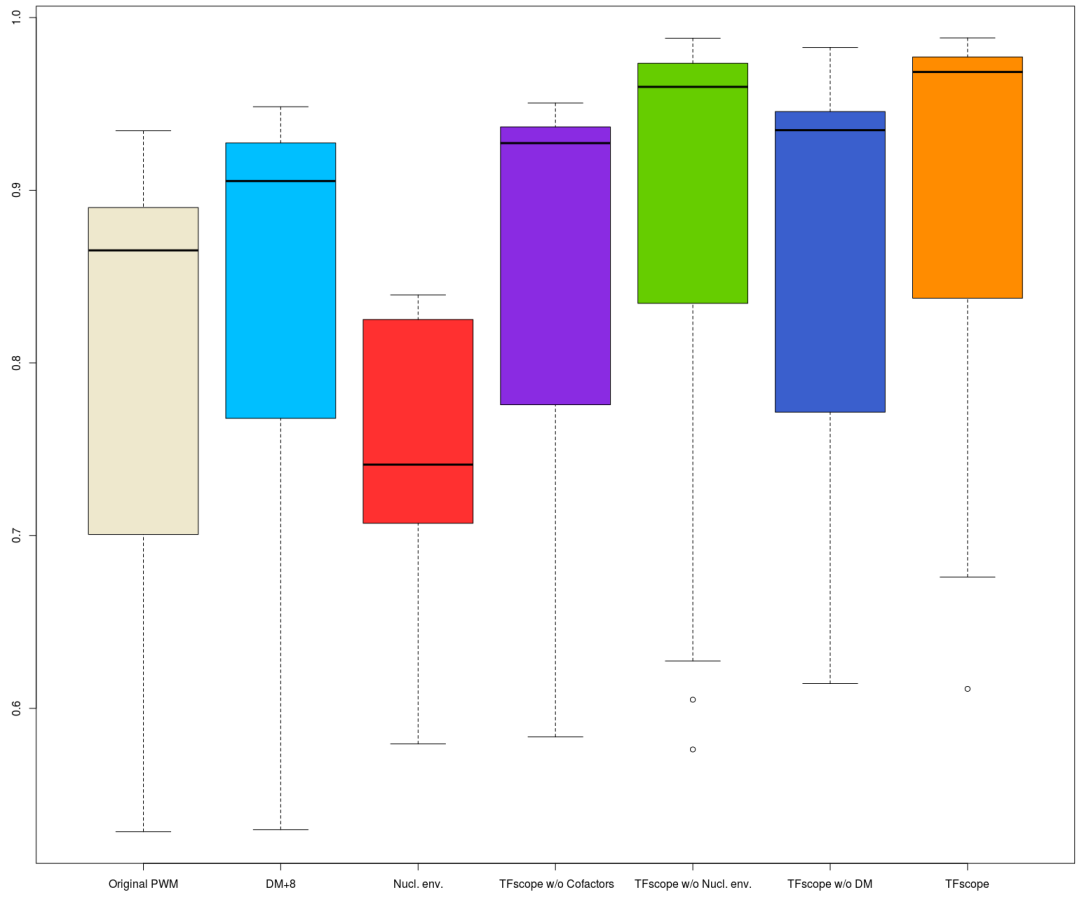

Figure 8.8 – *Boîtes à moustaches des distributions d'AUROC des différentes méthodes réalisées sur 15 expériences.*
Les AUROC sont calculées sur un même ensemble de test indépendant représentant 30% des séquences.

On retrouve sur le logo en Figure 8.11.B une bonne partie du motif original (présent dans JASPAR, Figure 8.11.A) dans les coefficients positifs. Il semblerait donc que l'absence de traitement conserve un motif plus proche du motif original. À noter que nous observons des conclusions similaires sur une autre expérience de classification entre traitements sur ESR1 dans T47D. Dans cette expérience, la classe positive correspond aux séquences dans les cellules sans traitement et la classe négative correspond aux séquences dans les cellules avec le traitement r5020, qui est un agoniste de la progestérone. Ces résultats apparaissent sur le logo de DM en Figure 8.11.C, ainsi que dans les résultats de TFscope (Figures disponibles en annexes : A.17 et A.18).

Dans l'article H.Mohammed *et al.* Nature 2015 [131], dont sont issues ces données, les auteurs ont montré que la progestérone modifiait les préférences de fixation de ESR1 et que les sites de fixations gagnés sous progestérone étaient distinguables par la présence de motif de réponse à la progestérone (PRE). Nos résultats semblent montrer que les séquences fixées par ESR1 sous progestérone sont enrichies en motif AR et NR3C2, qui sont d'autres récepteurs d'hormone stéroïde et dont les PWM présentent de fortes similarités avec le motif PRE décrit par Mohammed *et al.*. Notons que JASPAR ne contient pas de PWM pour le TF PR et que nos expériences ne pouvaient donc pas identifier de motif PRE directement. De manière intéressante, nous enrichissons les observations de H.Mohammed *et al.* en montrant que le motif ESR1 canonique tel que décrit dans la PWM JASPAR est plutôt retrouvé dans les séquences fixés par ESR1 sans progestérone (Figure 8.10), un résultat confirmé par DM (Figure 8.11).

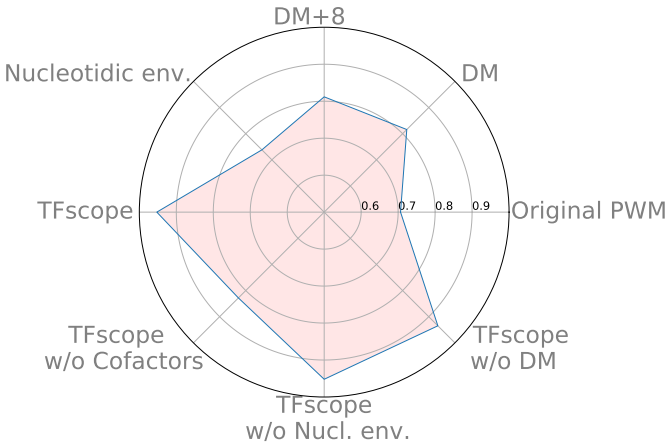

FIGURE 8.9 – *Graphique radar mesurant les AUROC des différentes approches et leurs combinaisons.*
Expérience : facteur de transcription ESR1 dans le type cellulaire T47D sans traitement ($Y = 1$) et avec traitement progestérone ($Y = 0$). Les AUROC sont obtenues sur un même ensemble de test représentant 30% des séquences. « Original PWM » correspond à l'utilisation de la PWM présente dans JASPAR et utilisée dans Unibind (MA0112.3 ESR1). « DM » et « DM+8 » sont les modèles motif discriminant avec respectivement 0 et 4 positions flanquantes de chaque coté. « Nucleotidic env. » correspond aux variables de DExTER et « Cofactors » aux variables de PosTFcoop.

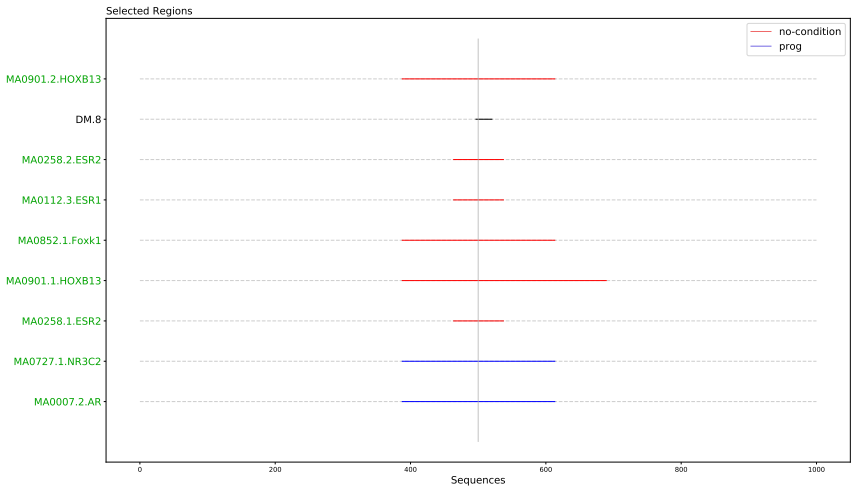

FIGURE 8.10 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur ESR1, dans le type cellulaire T47D sans traitements ($Y = 1$) et avec traitement progestérone ($Y = 0$).*

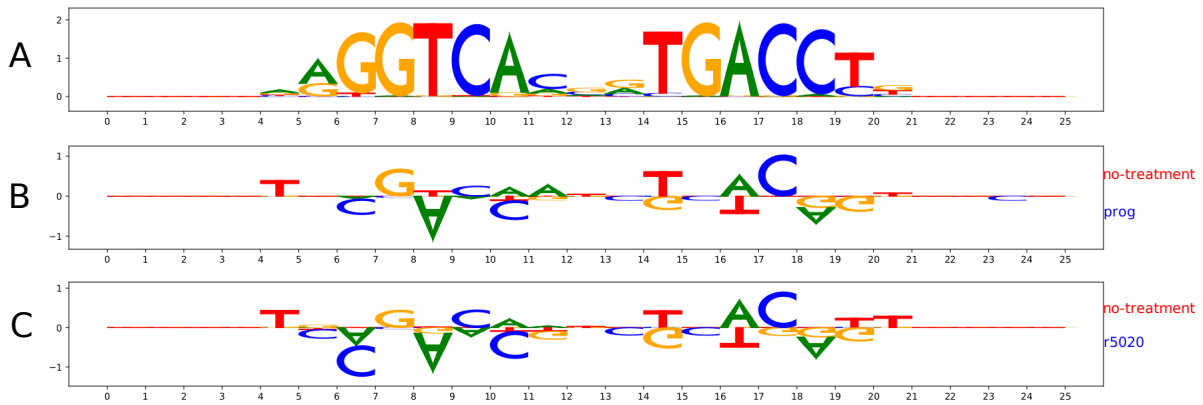

FIGURE 8.11 – *Représentation de différents logos associés à ESR1.*
A : Logo de la PPM MA0112.3 ESR1 présente dans JASPAR. **B :** Logo de DM sur la classification entre traitements de ESR1 : sans traitement contre progestérone, type cellulaire : T47D. **C :** Logo de DM sur la classification entre traitements de ESR1 : sans traitement contre r5020, type cellulaire : T47D.

Chapitre 9

Discussion et perspectives

Dans ce manuscrit, nous avons étudié les spécificités de fixation des facteurs de transcription dans différents types cellulaires chez l'homme. Cette étude a pris la forme d'un problème de classification entre deux classes de séquences, où toutes les séquences sont fixées par un facteur de transcription cible et seul le type cellulaire dans lequel il est fixé varie entre les classes. Nous avons pour cela utilisé et développé différentes méthodes basées sur les informations contenues dans les séquences. Ces informations, extraites sous forme de variables explicatives, ont ensuite été utilisées dans des modèles de régression linéaire logistique pénalisée pour optimiser la prédiction et déterminer quels types d'information sont les plus discriminants. Grâce à ces modèles, nous avons pu prouver qu'il est possible de distinguer la fixation des TF dans différents types cellulaires, uniquement en utilisant l'information contenue dans les séquences. De plus, nos modèles étant interprétables, ils nous permettent de mieux comprendre comment est régie la spécificité de fixation entre types cellulaires.

La question de l'interprétabilité d'un modèle est primordiale dans l'étude de données biologiques, car dans la majorité des cas on ne cherche pas seulement à prédire mais plutôt à savoir comment prédire. Les modèles basés sur du *deep learning* obtiennent de bons résultats de prédiction et peuvent apprendre à utiliser des informations inconnues ou négligées par la communauté ; cependant il peut être difficile d'analyser celles-ci. Parmi les travaux développé pour analyser ces informations, on peut par exemple citer DeepLIFT [126], qui quantifie l'importance de chaque nucléotide pour la prédiction du modèle à partir d'une séquence donnée en calculant un score d'importance. TF-MoDISco [132] est un autre outil qui permet d'interpréter un modèle. Il utilise le score d'importance sur des nucléotides

(calculé par exemple grâce à DeepLIFT) afin de les regrouper pour former de nouveaux motifs. Même si ces outils permettent une interprétation d'une séquence particulière, il est difficile d'apporter une interprétation globale du fonctionnement du réseau sur l'ensemble des séquences. Dans les modèles que nous avons développés, nous sommes certes limités par les informations que nous choisissons d'inclure, mais nous avons un contrôle sur ces informations et nous sommes capables de quantifier l'importance de chacune des informations présentes. De plus, il est tout à fait envisageable de rajouter d'autres informations basées sur la séquence, par exemple des prédictions d'une structure locale de l'ADN, comme DNAshape [94].

Dans nos expériences nous avons utilisé les données Unibind [35] pour essayer de ne conserver que des fixations directes de TF. Parmi l'ensemble des données Unibind disponibles, nous avons mis en place différentes procédures pour réduire le nombre d'expériences à étudier (=502) en ne conservant que les plus intéressantes dans nos analyses. Pour cela, nous avons utilisé un score de dissimilarité et n'avons gardé que les expériences dépassant un certain seuil. Ces seuils sont bien sûr sujets à discussion car ils ont été choisis de façons assez arbitraires. Il pourrait donc être intéressant de relancer les mêmes analyses sur des expériences sélectionnées avec des seuils plus permissifs. Ensuite, parmi toutes les expériences étudiées nous avons pu identifier des expériences « trop faciles » à distinguer. Les données de ChIP-seq qui composent ces expériences pourrait donc contenir de forts biais expérimentaux. Si ces biais existent bel et bien, ils pourraient aussi apparaître dans une moindre mesure sur d'autres expériences de classification. Cependant il est difficile de quantifier ces biais dans les expériences. Une façon de limiter au maximum l'existence de ces biais expérimentaux est de n'associer que des données de ChIP-seq issues du même laboratoire/expérimentateur. Néanmoins, certaines des expériences présentes dans notre jeu de données sont constituées à partir de données de ChIP-seq issues du même laboratoire, et ont pourtant des bonnes performances (voir Figure A.19).

Afin d'analyser l'information contenue strictement dans le motif de fixation du facteur de transcription étudié, nous avons développé le modèle DM. Nous avons pu montrer qu'il surpasse les performances des motifs originaux présents dans la base de données JASPAR sur notre problématique, et qu'il permet d'identifier les différences de motifs entre les classes. De plus, l'ajout d'une pénalisation de type L1 dans son apprentissage facilite son interprétation. Cependant, il n'est pas adapté à tous types de problèmes et ne permet

par exemple pas d'être utilisé pour scanner les séquences. Comparé à DAMO sur notre problématique, il apparaît plus performant et beaucoup plus interprétable. Cependant il serait intéressant de les comparer sur d'autres problématiques comme par exemple sur la problématique classique de séquence fixées/non-fixées, ou d'adapter DM à de nouvelles problématiques. Il serait aussi intéressant de comparer ses performances à d'autres motifs discriminants tels que STREME [86].

Notre motif discriminant, utilisé sur la problématique de spécificité de fixation entre types cellulaire, a montré que le motif d'un facteur de transcription est variable d'un type cellulaire à l'autre. Il est pour l'instant assez difficile de connaître le type cellulaire qui a été utilisé pour apprendre les PWM présentes dans JASPAR, ou de savoir si ces PWM sont issues de données contenant plusieurs types cellulaires. Obtenir cette information pourrait permettre d'apporter des analyses complémentaires à notre étude. Par exemple, en comparant le type cellulaire dans lequel la PWM a été apprise à ceux présents dans chaque expérience. En effet, si le type cellulaire utilisé pour apprendre la PWM de JASPAR est aussi utilisé dans l'une des deux classes d'une expérience de classification, cela change l'interprétation que l'on pourrait faire des résultats, et pourrait expliquer les cas extrêmes que nous avons étudiés en section 8.4. De manière plus générale, il semble nécessaire de construire des motifs spécifiques aux types cellulaires pour chaque facteur de transcription. Il pourrait par exemple être intéressant d'avoir une base de données où chaque TF a un premier motif « moyen » appris sur l'ensemble des types cellulaires disponible, ainsi que des motifs spécifiques pour chaque tissu. Le premier motif permettrait de pouvoir visualiser la fixation du TF sans spécificité ou pourrait être utile si le type cellulaire recherché par l'utilisateur n'est pas encore disponible. Les motifs tissus spécifiques permettrons d'augmenter la prédiction des modèles construits avec ces nouvelles PWM, dans les types cellulaires correspondants.

Une autre information considérée est l'information de position des cofacteurs du TF cible. Le modèle proposé, PosTFcoop, permet d'extraire des variables correspondant aux scores maximums de toutes les PWM présentes dans JASPAR dans une région propre à chacune d'entre elles. Avec ce modèle, nous montrons que l'information de présence des cofacteurs et l'information de position de ces cofacteurs par rapport au TF cible sont utiles pour distinguer la fixation du facteur de transcription cible.

Les PWM décrivent la fixation d'un facteur de transcription. Cependant dans des modèles comme les nôtres, qui basent leur prédiction sur une combinaison de nombreux TF, certaines PWM peuvent être sélectionnées uniquement pour capturer l'environnement nucléotidique et non pour décrire la fixation des facteurs de transcription qu'elles modélisent. Séparer l'information des cofacteurs et l'information d'environnement nucléotidique a donc été une question récurrente de cette thèse. Pour cela, nous considérons des variables d'environnement nucléotidiques obtenues avec la méthode DExTER, et nous avons utilisé ces variables à différentes étapes de nos analyses pour réduire au maximum l'effet de l'environnement nucléotidique sur l'information des cofacteurs.

Une autre problématique importante de la méthode PosTFcoop est la corrélation entre variables dues aux fortes resemblances entre motifs de fixation de TF appartenant à la même famille. Il peut donc être difficile de déterminer si le motif est sélectionné pour la fixation du TF auquel il est associé dans la base de données JASPAR ou pour celle d'un TF de la même famille. En effet, on voit clairement avec DM que le motif de fixation peut être variable entre types cellulaires. La fixation d'un TF dans un type cellulaire particulier pourrait ainsi être mieux modélisée par le motif associé à un autre TF appartenant à la même famille. Pour résoudre ce problème il serait possible de considérer l'information des familles de cofacteurs au lieu des cofacteurs. La méthode « RSAT : matrix clustering » [133] permet de regrouper les motifs de fixation PWM en famille en fonction de leur proximité. Utiliser RSAT pourrait nous permettre de réduire les PWM considérées en utilisant une PWM par famille. Nous réduirions ainsi le nombre de variables à extraire en entrée de la méthode PosTFcoop ainsi que dans le modèle LASSO associé. En revanche en utilisant ces regroupements, nous pourrions perdre de l'information si par exemple un TF cible est réellement fixé en présence de deux TF d'une même famille à des positions différentes. Une autre manière de faire, plus sophistiquée, serait de joindre les regroupements de motifs RSAT à un modèle GroupLASSO [52], pour procéder à une sélection de familles de cofacteurs.

Les trois types d'informations considérés sont combinés en un modèle appelé TFscope. Ce modèle obtient de très bonnes performances sur l'ensemble des expériences considérées (AUROC médiane = 0.868). Nous montrons ainsi que ces informations peuvent être combinées pour améliorer les prédictions et qu'elles ne sont pas redondantes. TFscope permet en outre de quantifier l'importance de chacune des informations présentes. Nous

observons que l'information génomique importante pour discriminer les types cellulaires est variable d'une expérience à l'autre, montrant que la spécificité de fixation des facteurs de transcription se manifeste sous différentes formes. De plus, nous voyons que pour certains facteurs de transcription un seul type d'information semble dominant, alors que pour d'autres le type d'information le plus important change suivant l'expérience.

Concernant l'étude de comparaison avec l'ouverture de la chromatine on a pu observer que cette information ne permet pas toujours de distinguer les types cellulaires. De plus, nous observons que cette information n'est pas identique à celle qui est capturée par TFscope. Il sera cependant nécessaire de réaliser cette comparaison sur un plus grand nombre d'expériences afin de confirmer ou non les résultats.

Dans le Chapitre 8, nous avons également réalisé une étude en se plaçant cette fois dans un problème de classification où les séquences sont fixées par un TF dans un type cellulaire particulier et seul le traitement change entre les classes. Les résultats obtenus sur un petit nombre d'expériences sont très encourageants. Cette approche, capable d'identifier les préférences de fixation des TF sous traitement pourrait trouver des applications cliniques, en identifiant par exemple des mutations somatiques responsables d'une absence de réponse dans les génomes de patients. Là encore, ces résultats sont obtenus sur peu d'expériences différentes, et il est donc nécessaire de répéter cette étude sur plus de données.

Une problématique que nous avons peu étudié est celle présentée dans le travail préliminaire du Chapitre 4, c'est-à-dire la problématique de classification entre séquences fixées dans un même type cellulaire avec le même traitement, mais par deux facteurs de transcription différents. C'est une problématique très pertinente, puisque l'on sait que les facteurs de transcription d'une même famille partagent souvent des motifs très similaires. La méthode TFscope pourrait tout a fait être utilisée dans ce cadre et permettre d'identifier les déterminants génomiques expliquant les différences de fixation entre deux TF d'une même famille.

Il est important de noter que dans les problèmes de classification classiques entre séquences fixées/non-fixées (voir section 3.3), la question du choix de l'ensemble des séquences non-fixées (background) est toujours importante [134], puisqu'elle peut, à elle

seule, modifier complètement les performances obtenues. Comme problématique liée au background, on peut par exemple citer le biais en GC [135] dans les séquences. Il peut donc être nécessaire de corriger ce biais avant de lancer toute analyse, sans quoi l'information biologiquement importante pourrait être masquée dans les modèles. Au contraire, dans les 3 problèmes de classification dont nous avons discuté, le fait d'avoir deux classes fixées par le facteur de transcription nous permet d'éviter ce problème de choix du background. Il serait intéressant d'étudier une autre problématique de classification, où la classe positive contiendrait des séquences fixées par le TF cible dans un type cellulaire et la classe négative correspondrait aux séquences fixées par le même TF cible dans plusieurs autres types cellulaires. Une telle problématique permettrait d'étudier les spécificités propres au type cellulaire en classe positive, et non les différences entre deux types cellulaires comme cela a été fait dans ce manuscrit. En revanche, cela conduirait à prêter attention aux types cellulaires présents dans les séquences négatives, le choix du background serait donc une part importante de cette problématique.

Aussi, il pourrait être pertinent d'étudier la conservation des spécificités de fixation des facteurs de transcription dans les types cellulaires. Par exemple, trouver certaines spécificités entre types cellulaires similaires chez la souris et chez l'homme pourrait permettre de mieux comprendre leurs fonctions. Les données ReMap2022 [136] comporterons des données sur la souris et permettront d'étudier cette conservation.

Pour améliorer TFscope, il serait possible d'appliquer DM dans la région associée aux cofacteurs déterminée par PosTFcoop (en utilisant la PWM associée). Par exemple, nous pourrions nous concentrer sur les variables de cofacteurs sélectionnées avec le modèle PosTFcoop. Ces variables étant chacune associée à une région et une PWM, il est possible d'utiliser DM dans cette région en prenant cette PWM comme entrée. Cela serait inclus entièrement dans la phase d'apprentissage optimisant ainsi les variables associées aux cofacteurs et remplaçant les anciennes dans le modèle TFscope. Puis les performances de ce nouveau modèle seraient calculées sur un ensemble de test indépendant. Cela pourrait permettre d'étudier les spécificités cellulaires de fixation des cofacteurs du TF cible en plus d'augmenter les performances.

Nous avons eu l'occasion de mettre en valeur ce travail avec plusieurs posters : en janvier 2019 à Barcelone (SMPGD), en octobre 2019 à Aussois (ProbGen) et en juin 2020

à JOBIM. De plus, un premier article présentant ce travail a été présenté dans les proceedings de la conférence JOBIM 2021. Nous travaillons actuellement à la rédaction d'un article de revue afin de publier ce travail.

En conclusion, cette étude nous permet de mieux comprendre la fixation des facteurs de transcription au travers de différentes conditions (types cellulaires, traitements, etc...), grâce à plusieurs méthodes performantes et interprétables développées au cours de cette thèse. Nous espérons que cette compréhension des mécanismes de fixations permettra d'avancer sur la recherche des pathologies liées à ceux-ci.

Bibliographie

[1] Tong Ihn Lee and Richard A. Young. Transcriptional regulation and its misregulation in disease. *Cell*, 152(6) :1237–1251, March 2013. ISSN 1097-4172. doi : 10.1016/j.cell.2013.02.014.

[2] Zeba Wunderlich and Leonid A. Mirny. Different gene regulation strategies revealed by analysis of binding motifs. *Trends in Genetics*, 25(10) :434–440, October 2009. ISSN 0168-9525. doi : 10.1016/j.tig.2009.08.003. URL `https://www.sciencedirect.com/science/article/pii/S0168952509001656`.

[3] Jie Wang, Jiali Zhuang, Sowmya Iyer, XinYing Lin, Troy W. Whitfield, Melissa C. Greven, Brian G. Pierce, Xianjun Dong, Anshul Kundaje, Yong Cheng, Oliver J. Rando, Ewan Birney, Richard M. Myers, William S. Noble, Michael Snyder, and Zhiping Weng. Sequence features and chromatin structure around the genomic regions bound by 119 human transcription factors. *Genome Research*, 22(9) :1798–1812, September 2012. ISSN 1549-5469. doi : 10.1101/gr.139105.112.

[4] Shane Neph, Jeff Vierstra, Andrew B. Stergachis, Alex P. Reynolds, Eric Haugen, Benjamin Vernot, Robert E. Thurman, Sam John, Richard Sandstrom, Audra K. Johnson, Matthew T. Maurano, Richard Humbert, Eric Rynes, Hao Wang, Shinny Vong, Kristen Lee, Daniel Bates, Morgan Diegel, Vaughn Roach, Douglas Dunn, Jun Neri, Anthony Schafer, R. Scott Hansen, Tanya Kutyavin, Erika Giste, Molly Weaver, Theresa Canfield, Peter Sabo, Miaohua Zhang, Gayathri Balasundaram, Rachel Byron, Michael J. MacCoss, Joshua M. Akey, M. A. Bender, Mark Groudine, Rajinder Kaul, and John A. Stamatoyannopoulos. An expansive human regulatory lexicon encoded in transcription factor footprints. *Nature*, 489(7414) :83–90, September 2012. ISSN 1476-4687. doi : 10.1038/nature11212. URL `https://www.nature.com/articles/nature11212`. Bandiera_abtest : a Cc_license_type : cc_y Cg_type : Nature Research Journals Number : 7414 Primary_atype : Research Publisher : Nature Publishing Group Subject_term : Genetic techniques ;Genomics ;Transcription Subject_term_id : genetic-techniques ;genomics ;transcription.

[5] Julia Zeitlinger. Seven myths of how transcription factors read the cis-regulatory code. *Current Opinion in Systems Biology*, September 2020. ISSN 2452-3100. doi : 10.1016/j.coisb.2020.08.002. URL `http://www.sciencedirect.com/science/article/pii/S2452310020300305`.

[6] Walther Flemming. *Zellsubstanz, Kern und Zelltheilung*. F.C.W. Vogel, Leipzig, 1882. Open Library ID : OL20780192M.

[7] Robert K. McGinty and Song Tan. Nucleosome Structure and Function. *Chemical Reviews*, 115(6) :2255–2273, March 2015. ISSN 0009-2665. doi : 10.1021/cr500373h. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4378457/.

[8] Boyan Bonev and Giacomo Cavalli. Organization and function of the 3D genome. *Nature Reviews Genetics*, 17(11) :661–678, November 2016. ISSN 1471-0064. doi : 10.1038/nrg.2016.112. URL https://www.nature.com/articles/nrg.2016.112/. Bandiera_abtest : a Cg_type : Nature Research Journals Number : 11 Primary_atype : Reviews Publisher : Nature Publishing Group Subject_term : Chromatin analysis ;Chromatin structure ;Gene expression Subject_term_id : chromatin-analysis ;chromatin-structure ;gene-expression.

[9] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. Molecular Biology of the Cell. 4th edition. *Molecular Biology of the Cell. 4th edition*, 2002. URL https://www.ncbi.nlm.nih.gov/books/NBK26887/. Publisher : Garland Science.

[10] Max E. Wilkinson, Clément Charenton, and Kiyoshi Nagai. RNA Splicing by the Spliceosome. *Annual Review of Biochemistry*, 89(1) :359–388, June 2020. ISSN 0066-4154. doi : 10.1146/annurev-biochem-091719-064225. URL https://www.annualreviews.org/doi/10.1146/annurev-biochem-091719-064225. Publisher : Annual Reviews.

[11] Jian-Wei Wei, Kai Huang, Chao Yang, and Chun-Sheng Kang. Non-coding RNAs as regulators in epigenetics (Review). *Oncology Reports*, 37(1) :3–9, January 2017. ISSN 1021-335X. doi : 10.3892/or.2016.5236. URL https://www.spandidos-publications.com/10.3892/or.2016.5236. Publisher : Spandidos Publications.

[12] David S. Latchman. Transcription factors : An overview. *The International Journal of Biochemistry & Cell Biology*, 29(12) :1305–1312, December 1997. ISSN 13572725. doi : 10.1016/S1357-2725(97)00085-X. URL https://linkinghub.elsevier.com/retrieve/pii/S135727259700085X.

[13] Yimeng Yin, Ekaterina Morgunova, Arttu Jolma, Eevi Kaasinen, Biswajyoti Sahu, Syed Khund-Sayeed, Pratyush K. Das, Teemu Kivioja, Kashyap Dave, Fan Zhong, Kazuhiro R. Nitta, Minna Taipale, Alexander Popov, Paul A. Ginno, Silvia Domcke, Jian Yan, Dirk Schübeler, Charles Vinson, and Jussi Taipale. Impact of cytosine methylation on DNA binding specificities of human transcription factors. *Science (New York, N.Y.)*, 356(6337) : eaaj2239, May 2017. ISSN 1095-9203. doi : 10.1126/science.aaj2239.

[14] Kenneth S. Zaret and Jason S. Carroll. Pioneer transcription factors : establishing competence for gene expression. *Genes & Development*, 25(21) :2227–2241, November 2011. ISSN 1549-5477. doi : 10.1101/gad.176826.111.

[15] David S. Latchman. Transcription factors : bound to activate or repress. *Trends in Biochemical Sciences*, 26(4) :211–213, April 2001. ISSN 0968-0004. doi : 10.1016/S0968-0004(01)01812-6. URL https://www.cell.com/trends/biochemical-sciences/abstract/S0968-0004(01)01812-6. Publisher : Elsevier.

[16] C. O. Pabo and R. T. Sauer. Transcription factors : structural families and principles of DNA recognition. *Annual Review of Biochemistry*, 61 :1053–1095, 1992. ISSN 0066-4154. doi : 10.1146/annurev.bi.61.070192.005201.

[17] François Spitz and Eileen E. M. Furlong. Transcription factors : from enhancer binding to developmental control. *Nature Reviews. Genetics*, 13(9) :613–626, September 2012. ISSN 1471-0064. doi : 10.1038/nrg3207.

[18] Ekaterina Morgunova and Jussi Taipale. Structural perspective of cooperative transcription factor binding. *Current Opinion in Structural Biology*, 47 :1–8, December 2017. ISSN 0959-440X. doi : 10.1016/j.sbi.2017.03.006. URL `https://www.sciencedirect.com/science/article/pii/S0959440X17300088`.

[19] Franziska Reiter, Sebastian Wienerroither, and Alexander Stark. Combinatorial function of transcription factors and cofactors. *Current Opinion in Genetics & Development*, 43 : 73–81, April 2017. ISSN 0959-437X. doi : 10.1016/j.gde.2016.12.007. URL `https://www.sciencedirect.com/science/article/pii/S0959437X17300059`.

[20] Arttu Jolma, Yimeng Yin, Kazuhiro R. Nitta, Kashyap Dave, Alexander Popov, Minna Taipale, Martin Enge, Teemu Kivioja, Ekaterina Morgunova, and Jussi Taipale. DNA-dependent formation of transcription factor pairs alters their binding specificity. *Nature*, 527(7578) :384–388, November 2015. ISSN 1476-4687. doi : 10.1038/nature15518.

[21] Sangjin Kim, Erik Broströmer, Dong Xing, Jianshi Jin, Shasha Chong, Hao Ge, Siyuan Wang, Chan Gu, Lijiang Yang, Yi Qin Gao, Xiao-dong Su, Yujie Sun, and X. Sunney Xie. Probing Allostery Through DNA. *Science*, 339(6121) :816–819, February 2013. doi : 10.1126/science.1229223. URL `https://www.science.org/lookup/doi/10.1126/science.1229223`. Publisher : American Association for the Advancement of Science.

[22] Richard I. Sherwood, Tatsunori Hashimoto, Charles W. O'Donnell, Sophia Lewis, Amira A. Barkal, John Peter van Hoff, Vivek Karun, Tommi Jaakkola, and David K. Gifford. Discovery of directional and nondirectional pioneer transcription factors by modeling DNase profile magnitude and shape. *Nature Biotechnology*, 32(2) :171–178, February 2014. ISSN 1546-1696. doi : 10.1038/nbt.2798.

[23] Jie Wang, Jiali Zhuang, Sowmya Iyer, XinYing Lin, Troy W. Whitfield, Melissa C. Greven, Brian G. Pierce, Xianjun Dong, Anshul Kundaje, Yong Cheng, Oliver J. Rando, Ewan Birney, Richard M. Myers, William S. Noble, Michael Snyder, and Zhiping Weng. Sequence features and chromatin structure around the genomic regions bound by 119 human transcription factors. *Genome Research*, 22(9) :1798–1812, September 2012. ISSN 1549-5469. doi : 10.1101/gr.139105.112.

[24] Majid Kazemian, Hannah Pham, Scot A. Wolfe, Michael H. Brodsky, and Saurabh Sinha. Widespread evidence of cooperative DNA binding by transcription factors in Drosophila development. *Nucleic Acids Research*, 41(17) :8237–8252, September 2013. ISSN 1362-4962. doi : 10.1093/nar/gkt598.

[25] Maria D. Chikina and Olga G. Troyanskaya. An effective statistical evaluation of ChIPseq dataset similarity. *Bioinformatics (Oxford, England)*, 28(5) :607–613, March 2012. ISSN 1367-4811. doi : 10.1093/bioinformatics/bts009.

[26] Victor Levitsky, Elena Zemlyanskaya, Dmitry Oshchepkov, Olga Podkolodnaya, Elena Ignatieva, Ivo Grosse, Victoria Mironova, and Tatyana Merkulova. A single ChIP-seq dataset is sufficient for comprehensive analysis of motifs co-occurrence with MCOT package. *Nucleic Acids Research*, 47(21) :e139–e139, December 2019. ISSN 0305-1048. doi : 10.1093/nar/gkz800. URL `https://doi.org/10.1093/nar/gkz800`.

[27] Can Sönmezer, Rozemarijn Kleinendorst, Dilek Imanci, Guido Barzaghi, Laura Villacorta, Dirk Schübeler, Vladimir Benes, Nacho Molina, and Arnaud Regis Krebs. Molecular Co-occupancy Identifies Transcription Factor Binding Cooperativity In Vivo. *Molecular Cell*, December 2020. ISSN 1097-2765. doi : 10.1016/j.molcel.2020.11.015. URL `http://www.sciencedirect.com/science/article/pii/S1097276520307930`.

[28] Jimmy Vandel, Océane Cassan, Sophie Lèbre, Charles-Henri Lecellier, and Laurent Bréhélin. Probing transcription factor combinatorics in different promoter classes and in enhancers. *BMC Genomics*, 20(1) :103, February 2019. ISSN 1471-2164. doi : 10.1186/s12864-018-5408-0. URL `https://doi.org/10.1186/s12864-018-5408-0`.

[29] Raluca Gordân, Ning Shen, Iris Dror, Tianyin Zhou, John Horton, Remo Rohs, and Martha L. Bulyk. Genomic regions flanking E-box binding sites influence DNA binding specificity of bHLH transcription factors through DNA shape. *Cell Reports*, 3(4) :1093–1104, April 2013. ISSN 2211-1247. doi : 10.1016/j.celrep.2013.03.014.

[30] Iris Dror, Tamar Golan, Carmit Levy, Remo Rohs, and Yael Mandel-Gutfreund. A widespread role of the motif environment in transcription factor binding across diverse protein families. *Genome Research*, 25(9) :1268–1280, January 2015. ISSN 1088-9051, 1549-5469. doi : 10.1101/gr.184671.114. URL `https://genome.cshlp.org/content/25/9/1268`. Company : Cold Spring Harbor Laboratory Press Distributor : Cold Spring Harbor Laboratory Press Institution : Cold Spring Harbor Laboratory Press Label : Cold Spring Harbor Laboratory Press Publisher : Cold Spring Harbor Lab.

[31] M. A. White, C. A. Myers, J. C. Corbo, and B. A. Cohen. Massively parallel in vivo enhancer assay reveals that highly local features determine the cis-regulatory function of ChIP-seq peaks. *Proceedings of the National Academy of Sciences*, 110(29) :11952–11957, July 2013. ISSN 0027-8424, 1091-6490. doi : 10.1073/pnas.1307449110. URL `http://www.pnas.org/cgi/doi/10.1073/pnas.1307449110`.

[32] Iris Dror, Tamar Golan, Carmit Levy, Remo Rohs, and Yael Mandel-Gutfreund. A widespread role of the motif environment in transcription factor binding across diverse protein families. *Genome Research*, 25(9) :1268–1280, September 2015. ISSN 1088-9051, 1549-5469. doi : 10.1101/gr.184671.114. URL `http://genome.cshlp.org/lookup/doi/10.1101/gr.184671.114`.

[33] Yong Zhang, Tao Liu, Clifford A. Meyer, Jérôme Eeckhoute, David S. Johnson, Bradley E. Bernstein, Chad Nusbaum, Richard M. Myers, Myles Brown, Wei Li, and X. Shirley Liu.

Model-based Analysis of ChIP-Seq (MACS). *Genome Biology*, 9(9) :R137, September 2008. ISSN 1474-760X. doi : 10.1186/gb-2008-9-9-r137. URL https://doi.org/10.1186/gb-2008-9-9-r137.

[34] Aseel Awdeh, Marcel Turcotte, and Theodore J. Perkins. WACS : improving ChIP-seq peak calling by optimally weighting controls. *BMC Bioinformatics*, 22 :69, February 2021. ISSN 1471-2105. doi : 10.1186/s12859-020-03927-2. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7885521/.

[35] Marius Gheorghe, Geir Kjetil Sandve, Aziz Khan, Benoit Ballester, and Anthony Mathelier. A map of direct TF–DNA interactions in the human genome. *Nucleic Acids Research*, 47 (4) :13, 2019.

[36] Marcel Geertz and Sebastian J. Maerkl. Experimental strategies for studying transcription factor-DNA binding specificities. *Briefings in Functional Genomics*, 9(5-6) :362–373, December 2010. ISSN 2041-2657. doi : 10.1093/bfgp/elq023.

[37] Divyanshi Srivastava and Shaun Mahony. Sequence and chromatin determinants of transcription factor binding and the establishment of cell type-specific binding patterns. *Biochimica et Biophysica Acta (BBA) - Gene Regulatory Mechanisms*, 1863(6) : 194443, June 2020. ISSN 1874-9399. doi : 10.1016/j.bbagrm.2019.194443. URL http://www.sciencedirect.com/science/article/pii/S1874939919300392.

[38] Bing Li, Michael Carey, and Jerry L. Workman. The Role of Chromatin during Transcription. *Cell*, 128(4) :707–719, February 2007. ISSN 0092-8674, 1097-4172. doi : 10.1016/j.cell.2007.01.015. URL https://www.cell.com/cell/abstract/S0092-8674(07)00109-2. Publisher : Elsevier.

[39] Esteban O. Mazzoni, Shaun Mahony, Michael Closser, Carolyn A. Morrison, Stephane Nedelec, Damian J. Williams, Disi An, David K. Gifford, and Hynek Wichterle. Synergistic binding of transcription factors to cell-specific enhancers programs motor neuron identity. *Nature Neuroscience*, 16(9) :1219–1227, September 2013. ISSN 1546-1726. doi : 10.1038/nn.3467.

[40] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32) :15849–15854, August 2019. ISSN 0027-8424, 1091-6490. doi : 10.1073/pnas.1903070116. URL http://www.pnas.org/lookup/doi/10.1073/pnas.1903070116.

[41] Tin Kam Ho. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8) :832–844, August 1998. ISSN 1939-3539. doi : 10.1109/34.709601. Conference Name : IEEE Transactions on Pattern Analysis and Machine Intelligence.

[42] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*. Adaptive Computation and Machine Learning series. MIT Press, Cambridge, MA, USA, December 2001. ISBN 978-0-262-19475-4.

[43] S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2) :129–137, March 1982. ISSN 1557-9654. doi : 10.1109/TIT.1982.1056489. Conference Name : IEEE Transactions on Information Theory.

[44] G. Enderlein. McCullagh, P., J. A. Nelder : Generalized linear models. Chapman and Hall London – New York 1983, 261 S., £ 16,–. *Biometrical Journal*, 29(2) : 206–206, 1987. ISSN 1521-4036. doi : 10.1002/bimj.4710290217. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/bimj.4710290217`. _eprint : https ://onlinelibrary.wiley.com/doi/pdf/10.1002/bimj.4710290217.

[45] Oded Maimon and Lior Rokach. *Data Mining and Knowledge Discovery Handbook*. Springer Science & Business Media, May 2006. ISBN 978-0-387-25465-4.

[46] David Arthur and Sergei Vassilvitskii. k-means++ : The Advantages of Careful Seeding. page 11.

[47] Robert Tibshirani. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society : Series B (Methodological)*, 58(1) :267–288, January 1996. ISSN 00359246. doi : 10.1111/j.2517-6161.1996.tb02080.x. URL `http://doi.wiley.com/10.1111/j.2517-6161.1996.tb02080.x`.

[48] Trevor Hastie, Sami Tibshirani, and Jerome Friedman. *Elements of Statistical Learning : data mining, inference, and prediction. 2nd Edition.* 2009. URL `https://web.stanford.edu/hastie/ElemStatLearn/`.

[49] Krzysztof C. Kiwiel. *Methods of descent for nondifferentiable optimization*, volume 1133 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1985. ISBN 978-3-540-15642-0. doi : 10.1007/BFb0074500. URL `https://mathscinet.ams.org/mathscinet-getitem?mr=797754`.

[50] Peter Bühlmann and Sara van de Geer. Generalized linear models and the Lasso. In *Statistics for High-Dimensional Data*, pages 45–53. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-20191-2 978-3-642-20192-9. doi : 10.1007/978-3-642-20192-9_3. URL `http://link.springer.com/10.1007/978-3-642-20192-9_3`. Series Title : Springer Series in Statistics.

[51] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 67(2) :301–320, April 2005. ISSN 1369-7412, 1467-9868. doi : 10.1111/j.1467-9868.2005.00503.x. URL `https://onlinelibrary.wiley.com/doi/10.1111/j.1467-9868.2005.00503.x`.

[52] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 68(1) : 49–67, 2006. ISSN 1467-9868. doi : 10.1111/j.1467-9868.2005.00532.x. URL `https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2005.00532.x`. _eprint : https ://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2005.00532.x.

[53] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 67(1) :91–108, February 2005. ISSN 1369-7412, 1467-9868. doi : 10.1111/j.1467-9868.2005.00490.x. URL `https://onlinelibrary.wiley.com/doi/10.1111/j.1467-9868.2005.00490.x`.

[54] Jeff Heaton. Ian Goodfellow, Yoshua Bengio, and Aaron Courville : Deep learning. *Genetic Programming and Evolvable Machines*, 19(1) :305–307, June 2018. ISSN 1573-7632. doi : 10.1007/s10710-017-9314-z. URL `https://doi.org/10.1007/s10710-017-9314-z`.

[55] Zhen Shen, Wenzheng Bao, and De-Shuang Huang. Recurrent Neural Network for Predicting Transcription Factor Binding Sites. *Scientific Reports*, 8(1) :15270, October 2018. ISSN 2045-2322. doi : 10.1038/s41598-018-33321-1.

[56] Divyanshi Srivastava, Begüm Aydin, Esteban O. Mazzoni, and Shaun Mahony. An interpretable bimodal neural network characterizes the sequence and preexisting chromatin predictors of induced TF binding. *bioRxiv*, page 672790, June 2020. doi : 10.1101/672790. URL `https://www.biorxiv.org/content/10.1101/672790v2`. Publisher : Cold Spring Harbor Laboratory Section : New Results.

[57] Yao Xue and Nilanjan Ray. Cell Detection with Deep Convolutional Neural Network and Compressed Sensing. August 2017.

[58] Yu Zhang, Peter Tiňo, Aleš Leonardis, and Ke Tang. A Survey on Neural Network Interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, pages 1–17, 2021. ISSN 2471-285X. doi : 10.1109/TETCI.2021.3100641. Conference Name : IEEE Transactions on Emerging Topics in Computational Intelligence.

[59] Felipe Albrecht, Markus List, Christoph Bock, and Thomas Lengauer. DeepBlueR : large-scale epigenomic analysis in R. *Bioinformatics*, 33(13) :2063–2064, July 2017. ISSN 1367-4803. doi : 10.1093/bioinformatics/btx099. URL `https://doi.org/10.1093/bioinformatics/btx099`.

[60] Žiga Avsec, Melanie Weilert, Avanti Shrikumar, Amr Alexandari, Sabrina Krueger, Khyati Dalal, Robin Fropf, Charles McAnany, Julien Gagneur, Anshul Kundaje, and Julia Zeitlinger. Deep learning at base-resolution reveals motif syntax of the cis-regulatory code. *bioRxiv*, page 737981, August 2019. doi : 10.1101/737981. URL `https://www.biorxiv.org/content/10.1101/737981v1`. Publisher : Cold Spring Harbor Laboratory Section : New Results.

[61] Seonwoo Min, Byunghan Lee, and Sungroh Yoon. Deep learning in bioinformatics. *Briefings in Bioinformatics*, 18(5) :851–869, September 2017. ISSN 1477-4054. doi : 10.1093/bib/bbw068.

[62] Gary D Stormo. DNA binding sites : representation and discovery. page 8, 2000.

[63] Michael Brown, Richard Hughey, Anders Krogh, I. Saira Mian, Kimmen Sjölander, and David Haussler. Using Dirichlet Mixture Priors to Derive Hidden Markov Models for Protein Families. In *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*, pages 47–55. AAAI Press, 1993.

[64] David Martin, Vincent Maillol, and Eric Rivals. Fast and Accurate Genome-Scale Identification of DNA-Binding Sites. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 201–205, December 2018. doi : 10.1109/BIBM.2018.8621093.

[65] Michael Burrows and David Wheeler. A Block-Sorting Lossless Data Compression Algorithm. Technical report, DIGITAL SRC RESEARCH REPORT, 1994.

[66] P. Ferragina and G. Manzini. Opportunistic data structures with applications. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 390–398, November 2000. doi : 10.1109/SFCS.2000.892127. ISSN : 0272-5428.

[67] Charles E. Grant, Timothy L. Bailey, and William Stafford Noble. FIMO : scanning for occurrences of a given motif. *Bioinformatics (Oxford, England)*, 27(7) :1017–1018, April 2011. ISSN 1367-4811. doi : 10.1093/bioinformatics/btr064.

[68] Timothy L. Bailey, Mikael Boden, Fabian A. Buske, Martin Frith, Charles E. Grant, Luca Clementi, Jingyuan Ren, Wilfred W. Li, and William S. Noble. MEME Suite : tools for motif discovery and searching. *Nucleic Acids Research*, 37(suppl_2) :W202–W208, July 2009. ISSN 0305-1048. doi : 10.1093/nar/gkp335. URL `https://doi.org/10.1093/nar/gkp335`.

[69] Stuart Geman and Donald Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. page 21, 1984.

[70] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society : Series B (Methodological)*, 39(1) :1–22, September 1977. ISSN 00359246. doi : 10.1111/j.2517-6161.1977.tb01600.x. URL `https://onlinelibrary.wiley.com/doi/10.1111/j.2517-6161.1977.tb01600.x`.

[71] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wootton. Detecting subtle sequence signals : a Gibbs sampling strategy for multiple alignment. *Science (New York, N.Y.)*, 262(5131) :208–214, October 1993. ISSN 0036-8075. doi : 10.1126/science.8211139.

[72] Xuhua Xia. Position Weight Matrix, Gibbs Sampler, and the Associated Significance Tests in Motif Characterization and Prediction. *Scientifica*, 2012 :917540, 2012. ISSN 2090-908X. doi : 10.6064/2012/917540. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3820676/`.

[73] Frederick P. Roth, Jason D. Hughes, Preston W. Estep, and George M. Church. Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nature Biotechnology*, 16(10) :939–945, October 1998. ISSN 1546-1696. doi : 10.1038/nbt1098-939. URL `https://www.nature.com/articles/nbt1098-939`. Bandiera_abtest : a Cg_type : Nature Research Journals Number : 10 Primary_atype : Research Publisher : Nature Publishing Group.

[74] Charles E. Lawrence and Andrew A. Reilly. An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins : Structure, Function, and Bioinformatics*, 7(1) : 41–51, 1990. ISSN 1097-0134. doi : 10.1002/prot.340070105. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.340070105`. _eprint : https ://onlinelibrary.wiley.com/doi/pdf/10.1002/prot.340070105.

[75] T. L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proceedings. International Conference on Intelligent Systems for Molecular Biology*, 2 :28–36, 1994. ISSN 1553-0833.

[76] Timothy L. Bailey, James Johnson, Charles E. Grant, and William S. Noble. The MEME Suite. *Nucleic Acids Research*, 43(W1) :W39–W49, July 2015. ISSN 0305-1048. doi : 10.1093/nar/gkv416. URL `https://doi.org/10.1093/nar/gkv416`.

[77] Y. J. Hu, S. Sandmeyer, C. McLaughlin, and D. Kibler. Combinatorial motif analysis and hypothesis generation on a genomic scale. *Bioinformatics (Oxford, England)*, 16(3) : 222–232, March 2000. ISSN 1367-4803. doi : 10.1093/bioinformatics/16.3.222.

[78] Saurabh Sinha. Discriminative motifs. In *Proceedings of the sixth annual international conference on Computational biology*, RECOMB '02, pages 291–298, New York, NY, USA, April 2002. Association for Computing Machinery. ISBN 978-1-58113-498-8. doi : 10.1145/565196.565234. URL `https://doi.org/10.1145/565196.565234`.

[79] X. Shirley Liu, Douglas L. Brutlag, and Jun S. Liu. An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments. *Nature Biotechnology*, 20(8) :835–839, August 2002. ISSN 1087-0156. doi : 10.1038/nbt717.

[80] Andrew D. Smith, Pavel Sumazin, and Michael Q. Zhang. Identifying tissue-selective transcription factor binding sites in vertebrate promoters. *Proceedings of the National Academy of Sciences of the United States of America*, 102(5) :1560–1565, February 2005. ISSN 0027-8424. doi : 10.1073/pnas.0406123102.

[81] Saurabh Sinha. On counting position weight matrix matches in a sequence, with application to discriminative motif finding. *Bioinformatics*, 22(14) :e454–e463, July 2006. ISSN 1367-4803. doi : 10.1093/bioinformatics/btl227. URL `https://doi.org/10.1093/bioinformatics/btl227`.

[82] Mike J. Mason, Kathrin Plath, and Qing Zhou. Identification of context-dependent motifs by contrasting ChIP binding data. *Bioinformatics (Oxford, England)*, 26(22) :2826–2832, November 2010. ISSN 1367-4811. doi : 10.1093/bioinformatics/btq546.

[83] Peter Huggins, Shan Zhong, Idit Shiff, Rachel Beckerman, Oleg Laptenko, Carol Prives, Marcel H. Schulz, Itamar Simon, and Ziv Bar-Joseph. DECOD : fast and accurate discriminative DNA motif finding. *Bioinformatics (Oxford, England)*, 27(17) :2361–2367, September 2011. ISSN 1367-4811. doi : 10.1093/bioinformatics/btr412.

[84] Timothy Bailey. DREME : Motif discovery in transcription factor ChIP-seq data. *Bioinformatics*, 27 :1653–1659, June 2011. doi : 10.1093/bioinformatics/btr261.

[85] Ronak Y. Patel and Gary D. Stormo. Discriminative motif optimization based on perceptron training. *Bioinformatics*, 30(7) :941–948, April 2014. ISSN 1367-4803. doi : 10.1093/bioinformatics/btt748. URL https://doi.org/10.1093/bioinformatics/btt748.

[86] Timothy L Bailey. STREME : accurate and versatile sequence motif discovery. *Bioinformatics*, (btab203), March 2021. ISSN 1367-4803. doi : 10.1093/bioinformatics/btab203. URL https://doi.org/10.1093/bioinformatics/btab203.

[87] Sven Heinz, Christopher Benner, Nathanael Spann, Eric Bertolino, Yin C. Lin, Peter Laslo, Jason X. Cheng, Cornelis Murre, Harinder Singh, and Christopher K. Glass. Simple Combinations of Lineage-Determining Transcription Factors Prime cis-Regulatory Elements Required for Macrophage and B Cell Identities. *Molecular Cell*, 38(4) :576–589, May 2010. ISSN 1097-2765. doi : 10.1016/j.molcel.2010.05.004. URL https://www.sciencedirect.com/science/article/pii/S1097276510003667.

[88] Morgane Thomas-Chollier, Matthieu Defrance, Alejandra Medina-Rivera, Olivier Sand, Carl Herrmann, Denis Thieffry, and Jacques van Helden. RSAT 2011 : regulatory sequence analysis tools. *Nucleic Acids Research*, 39(Web Server issue) :W86–91, July 2011. ISSN 1362-4962. doi : 10.1093/nar/gkr377.

[89] In The Brain and F. Rosenblatt. The Perceptron : A Probabilistic Model for Information Storage and Organization, 1958.

[90] Shuxiang Ruan and Gary D. Stormo. Comparison of discriminative motif optimization using matrix and DNA shape-based models. *BMC Bioinformatics*, 19(1) :86, March 2018. ISSN 1471-2105. doi : 10.1186/s12859-018-2104-7. URL https://doi.org/10.1186/s12859-018-2104-7.

[91] Shuxiang Ruan and Gary D. Stormo. Inherent limitations of probabilistic models for protein-DNA binding specificity. *PLoS computational biology*, 13(7) :e1005638, July 2017. ISSN 1553-7358. doi : 10.1371/journal.pcbi.1005638.

[92] Anthony Mathelier and Wyeth W. Wasserman. The Next Generation of Transcription Factor Binding Site Prediction. *PLOS Computational Biology*, 9(9) :e1003214, September 2013. ISSN 1553-7358. doi : 10.1371/journal.pcbi.1003214. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003214. Publisher : Public Library of Science.

[93] Tianyin Zhou, Lin Yang, Yan Lu, Iris Dror, Ana Carolina Dantas Machado, Tahereh Ghane, Rosa Di Felice, and Remo Rohs. DNAshape : a method for the high-throughput prediction of DNA structural features on a genomic scale. *Nucleic Acids Research*, 41(Web Server issue) :W56–W62, July 2013. ISSN 0305-1048. doi : 10.1093/nar/gkt437. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3692085/.

[94] Anthony Mathelier, Beibei Xin, Tsu-Pei Chiu, Lin Yang, Remo Rohs, and Wyeth W. Wasserman. DNA shape features improve transcription factor binding site predictions in vivo. *Cell systems*, 3(3) :278–286.e4, September 2016. ISSN 2405-4712. doi : 10.1016/j.cels.2016.07.001. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5042832/.

[95] Jerome Friedman, Trevor Hastie, Robert Tibshirani, and others. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[96] Babak Alipanahi, Andrew Delong, Matthew T. Weirauch, and Brendan J. Frey. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nature Biotechnology*, 33(8) :831–838, August 2015. ISSN 1546-1696. doi : 10.1038/nbt.3300. URL `https://www.nature.com/articles/nbt.3300`. Bandiera_abtest : a Cg_type : Nature Research Journals Number : 8 Primary_atype : Research Publisher : Nature Publishing Group Subject_term : Computational biology and bioinformatics ;Gene regulatory networks ;Genome informatics Subject_term_id : computational-biology-and-bioinformatics ;gene-regulatory-networks ;genome-informatics.

[97] Jian Zhou and Olga G Troyanskaya. Predicting effects of noncoding variants with deep learning–based sequence model. *Nature Methods*, 12(10) :931–934, October 2015. ISSN 1548-7091, 1548-7105. doi : 10.1038/nmeth.3547. URL `http://www.nature.com/articles/nmeth.3547`.

[98] ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414) :57–74, September 2012. ISSN 1476-4687. doi : 10.1038/nature11247.

[99] Roadmap Epigenomics Consortium, Anshul Kundaje, Wouter Meuleman, Jason Ernst, Misha Bilenky, Angela Yen, Alireza Heravi-Moussavi, Pouya Kheradpour, Zhizhuo Zhang, Jianrong Wang, Michael J. Ziller, Viren Amin, John W. Whitaker, Matthew D. Schultz, Lucas D. Ward, Abhishek Sarkar, Gerald Quon, Richard S. Sandstrom, Matthew L. Eaton, Yi-Chieh Wu, Andreas R. Pfenning, Xinchen Wang, Melina Claussnitzer, Yaping Liu, Cristian Coarfa, R. Alan Harris, Noam Shoresh, Charles B. Epstein, Elizabeta Gjoneska, Danny Leung, Wei Xie, R. David Hawkins, Ryan Lister, Chibo Hong, Philippe Gascard, Andrew J. Mungall, Richard Moore, Eric Chuah, Angela Tam, Theresa K. Canfield, R. Scott Hansen, Rajinder Kaul, Peter J. Sabo, Mukul S. Bansal, Annaick Carles, Jesse R. Dixon, Kai-How Farh, Soheil Feizi, Rosa Karlic, Ah-Ram Kim, Ashwinikumar Kulkarni, Daofeng Li, Rebecca Lowdon, GiNell Elliott, Tim R. Mercer, Shane J. Neph, Vitor Onuchic, Paz Polak, Nisha Rajagopal, Pradipta Ray, Richard C. Sallari, Kyle T. Siebenthall, Nicholas A. Sinnott-Armstrong, Michael Stevens, Robert E. Thurman, Jie Wu, Bo Zhang, Xin Zhou, Arthur E. Beaudet, Laurie A. Boyer, Philip L. De Jager, Peggy J. Farnham, Susan J. Fisher, David Haussler, Steven J. M. Jones, Wei Li, Marco A. Marra, Michael T. McManus, Shamil Sunyaev, James A. Thomson, Thea D. Tlsty, Li-Huei Tsai, Wei Wang, Robert A. Waterland, Michael Q. Zhang, Lisa H. Chadwick, Bradley E. Bernstein, Joseph F. Costello, Joseph R. Ecker, Martin Hirst, Alexander Meissner, Aleksandar Milosavljevic, Bing Ren, John A. Stamatoyannopoulos, Ting Wang, and Manolis Kellis. Integrative analysis of 111 reference human epigenomes. *Nature*, 518(7539) :317–330, February 2015. ISSN 1476-4687. doi : 10.1038/nature14248.

[100] Mahmoud Ghandi, Dongwon Lee, Morteza Mohammad-Noori, and Michael A. Beer. Enhanced regulatory sequence prediction using gapped k-mer features. *PLoS computational biology*, 10(7) :e1003711, July 2014. ISSN 1553-7358. doi : 10.1371/journal.pcbi.1003711.

[101] John W. Whitaker, Zhao Chen, and Wei Wang. Predicting the human epigenome from DNA motifs. *Nature Methods*, 12(3) :265–272, March 2015. ISSN 1548-7105. doi : 10.1038/nmeth.3065. URL https://www.nature.com/articles/nmeth.3065. Bandiera_abtest : a Cg_type : Nature Research Journals Number : 3 Primary_atype : Research Publisher : Nature Publishing Group Subject_term : Epigenomics ;Genome informatics ;Genomics ;Machine learning Subject_term_id : epigenomics ;genome-informatics ;genomics ;machine-learning.

[102] David R. Kelley, Jasper Snoek, and John L. Rinn. Basset : learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Research*, 26 (7) :990–999, January 2016. ISSN 1088-9051, 1549-5469. doi : 10.1101/gr.200535.115. URL https://genome.cshlp.org/content/26/7/990. Company : Cold Spring Harbor Laboratory Press Distributor : Cold Spring Harbor Laboratory Press Institution : Cold Spring Harbor Laboratory Press Label : Cold Spring Harbor Laboratory Press Publisher : Cold Spring Harbor Lab.

[103] Robin Andersson, Claudia Gebhard, Irene Miguel-Escalada, Ilka Hoof, Jette Bornholdt, Mette Boyd, Yun Chen, Xiaobei Zhao, Christian Schmidl, Takahiro Suzuki, Evgenia Ntini, Erik Arner, Eivind Valen, Kang Li, Lucia Schwarzfischer, Dagmar Glatz, Johanna Raithel, Berit Lilje, Nicolas Rapin, Frederik Otzen Bagger, Mette Jørgensen, Peter Refsing Andersen, Nicolas Bertin, Owen Rackham, A. Maxwell Burroughs, J. Kenneth Baillie, Yuri Ishizu, Yuri Shimizu, Erina Furuhata, Shiori Maeda, Yutaka Negishi, Christopher J. Mungall, Terrence F. Meehan, Timo Lassmann, Masayoshi Itoh, Hideya Kawaji, Naoto Kondo, Jun Kawai, Andreas Lennartsson, Carsten O. Daub, Peter Heutink, David A. Hume, Torben Heick Jensen, Harukazu Suzuki, Yoshihide Hayashizaki, Ferenc Müller, Alistair R. R. Forrest, Piero Carninci, Michael Rehli, and Albin Sandelin. An atlas of active enhancers across human cell types and tissues. *Nature*, 507(7493) :455–461, March 2014. ISSN 1476-4687. doi : 10.1038/nature12787. URL https://www.nature.com/articles/nature12787. Bandiera_abtest : a Cg_type : Nature Research Journals Number : 7493 Primary_atype : Research Publisher : Nature Publishing Group Subject_term : Gene regulation ;Non-coding RNAs ;Transcriptomics Subject_term_id : gene-regulation ;non-coding-rnas ;transcriptomics.

[104] Michal Levo, Einat Zalckvar, Eilon Sharon, Ana Carolina Dantas Machado, Yael Kalma, Maya Lotam-Pompan, Adina Weinberger, Zohar Yakhini, Remo Rohs, and Eran Segal. Unraveling determinants of transcription factor binding outside the core binding site. *Genome Research*, 25(7) :1018–1029, July 2015. ISSN 1088-9051, 1549-5469. doi : 10.1101/gr.185033.114. URL http://genome.cshlp.org/lookup/doi/10.1101/gr.185033.114.

[105] Robin Andersson, Claudia Gebhard, Irene Miguel-Escalada, Ilka Hoof, Jette Bornholdt, Mette Boyd, Yun Chen, Xiaobei Zhao, Christian Schmidl, Takahiro Suzuki, Evgenia Ntini, Erik Arner, Eivind Valen, Kang Li, Lucia Schwarzfischer, Dagmar Glatz, Johanna Raithel, Berit Lilje, Nicolas Rapin, Frederik Otzen Bagger, Mette Jørgensen, Peter Refsing Andersen, Nicolas Bertin, Owen Rackham, A. Maxwell Burroughs, J. Kenneth Baillie, Yuri Ishizu, Yuri Shimizu, Erina Furuhata, Shiori Maeda, Yutaka Negishi, Christopher J. Mungall, Terrence F. Meehan, Timo Lassmann, Masayoshi Itoh, Hideya Kawaji, Naoto Kondo, Jun Kawai, Andreas Lennartsson, Carsten O. Daub, Pe-

ter Heutink, David A. Hume, Torben Heick Jensen, Harukazu Suzuki, Yoshihide Hayashizaki, Ferenc Müller, Alistair R.R. Forrest, Piero Carninci, Michael Rehli, and Albin Sandelin. An atlas of active enhancers across human cell types and tissues. *Nature*, 507(7493) :455–461, March 2014. ISSN 0028-0836. doi : 10.1038/nature12787. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5215096/`.

[106] Ivan V Kulakovskiy, Ilya E Vorontsov, Ivan S Yevshin, Ruslan N Sharipov, Alla D Fedorova, Eugene I Rumynskiy, Yulia A Medvedeva, Arturo Magana-Mora, Vladimir B Bajic, Dmitry A Papatsenko, Fedor A Kolpakov, and Vsevolod J Makeev. HOCOMOCO : towards a complete collection of transcription factor binding models for human and mouse via large-scale ChIP-Seq analysis. *Nucleic Acids Research*, 46(D1) :D252–D259, January 2018. ISSN 0305-1048. doi : 10.1093/nar/gkx1106. URL `https://doi.org/10.1093/nar/gkx1106`.

[107] Matthew T. Weirauch, Ally Yang, Mihai Albu, Atina G. Cote, Alejandro Montenegro-Montero, Philipp Drewe, Hamed S. Najafabadi, Samuel A. Lambert, Ishminder Mann, Kate Cook, Hong Zheng, Alejandra Goity, Harm van Bakel, Jean-Claude Lozano, Mary Galli, Mathew G. Lewsey, Eryong Huang, Tuhin Mukherjee, Xiaoting Chen, John S. Reece-Hoyes, Sridhar Govindarajan, Gad Shaulsky, Albertha J. M. Walhout, François-Yves Bouget, Gunnar Ratsch, Luis F. Larrondo, Joseph R. Ecker, and Timothy R. Hughes. Determination and inference of eukaryotic transcription factor sequence specificity. *Cell*, 158 (6) :1431–1443, September 2014. ISSN 1097-4172. doi : 10.1016/j.cell.2014.08.009.

[108] Jie Wang, Jiali Zhuang, Sowmya Iyer, Xin-Ying Lin, Melissa C. Greven, Bong-Hyun Kim, Jill Moore, Brian G. Pierce, Xianjun Dong, Daniel Virgil, Ewan Birney, Jui-Hung Hung, and Zhiping Weng. Factorbook.org : a Wiki-based database for transcription factor-binding data generated by the ENCODE consortium. *Nucleic Acids Research*, 41(Database issue) : D171–D176, January 2013. ISSN 0305-1048. doi : 10.1093/nar/gks1221. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3531197/`.

[109] Quy Xiao Xuan Lin, Stephanie Sian, Omer An, Denis Thieffry, Sudhakar Jha, and Touati Benoukraf. MethMotif : an integrative cell specific database of transcription factor binding motifs coupled with DNA methylation profiles. *Nucleic Acids Research*, 47(D1) :D145–D154, January 2019. ISSN 0305-1048. doi : 10.1093/nar/gky1005. URL `https://doi.org/10.1093/nar/gky1005`.

[110] Albin Sandelin, Wynand Alkema, Pär Engström, Wyeth W. Wasserman, and Boris Lenhard. JASPAR an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research*, 32(Database issue) :D91–94, January 2004. ISSN 1362-4962. doi : 10.1093/nar/gkh012.

[111] Oriol Fornes, Jaime A Castro-Mondragon, Aziz Khan, Robin van der Lee, Xi Zhang, Phillip A Richmond, Bhavi P Modi, Solenne Correard, Marius Gheorghe, Damir Baranašić, Walter Santana-Garcia, Ge Tan, Jeanne Chèneby, Benoit Ballester, François Parcy, Albin Sandelin, Boris Lenhard, Wyeth W Wasserman, and Anthony Mathelier. JASPAR 2020 : update of the open-access database of transcription factor binding profiles. *Nucleic Acids Research*, 48(D1) :D87–D92, January 2020. ISSN 0305-1048. doi : 10.1093/nar/gkz1001. URL `https://doi.org/10.1093/nar/gkz1001`.

[112] Nobuo Ogawa and Mark D. Biggin. High-throughput SELEX determination of DNA sequences bound by transcription factors in vitro. *Methods in Molecular Biology (Clifton, N.J.)*, 786 :51–63, 2012. ISSN 1940-6029. doi : 10.1007/978-1-61779-292-2_3.

[113] Christophe Menichelli, Vincent Guitard, Rafael M. Martins, Sophie Lèbre, Jose-Juan Lopez-Rubio, Charles-Henri Lecellier, and Laurent Bréhélin. Identification of long regulatory elements in the genome of Plasmodium falciparum and other eukaryotes. *bioRxiv*, page 2020.06.02.130468, June 2020. doi : 10.1101/2020.06.02.130468. URL `https://www.biorxiv.org/content/10.1101/2020.06.02.130468v1`. Publisher : Cold Spring Harbor Laboratory Section : New Results.

[114] Jacob Benesty, Jingdong Chen, and Yiteng Huang. On the Importance of the Pearson Correlation Coefficient in Noise Reduction. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(4) :757–765, May 2008. ISSN 1558-7924. doi : 10.1109/TASL.2008.919072. Conference Name : IEEE Transactions on Audio, Speech, and Language Processing.

[115] Haomiao Zhou, Zhihong Deng, Yuanqing Xia, and Mengyin Fu. A new sampling method in particle filter based on Pearson correlation coefficient. *Neurocomputing*, 216 :208–215, December 2016. ISSN 0925-2312. doi : 10.1016/j.neucom.2016.07.036. URL `https://www.sciencedirect.com/science/article/pii/S0925231216307925`.

[116] Fabienne Bejjani, Claire Tolza, Mathias Boulanger, Damien Downes, Raphaël Romero, Muhammad Ahmad Maqbool, Amal Zine El Aabidine, Jean-Christophe Andrau, Sophie Lebre, Laurent Brehelin, Hughes Parrinello, Marine Rohmer, Tony Kaoma, Laurent Vallar, Jim R. Hughes, Kazem Zibara, Charles-Henri Lecellier, Marc Piechaczyk, and Isabelle Jariel-Encontre. Fra-1 regulates its target genes via binding to remote enhancers without exerting major control on chromatin architecture in triple negative breast cancers. *Nucleic Acids Research*, 49(5) :2488–2508, March 2021. ISSN 1362-4962. doi : 10.1093/nar/gkab053.

[117] Fabienne Bejjani, Emilie Evanno, Kazem Zibara, Marc Piechaczyk, and Isabelle Jariel-Encontre. The AP-1 transcriptional complex : Local switch or remote command ? *Biochimica et Biophysica Acta (BBA) - Reviews on Cancer*, 1872(1) :11–23, August 2019. ISSN 0304-419X. doi : 10.1016/j.bbcan.2019.04.003. URL `https://www.sciencedirect.com/science/article/pii/S0304419X19300526`.

[118] Cu Nguyen, Jia-Ling Teo, Akihisa Matsuda, Masakatsu Eguchi, Emil Y. Chi, William R. Henderson, and Michael Kahn. Chemogenomic identification of Ref-1/AP-1 as a therapeutic target for asthma. *Proceedings of the National Academy of Sciences*, 100(3) :1169–1173, February 2003. ISSN 0027-8424, 1091-6490. doi : 10.1073/pnas.0437889100. URL `https://www.pnas.org/content/100/3/1169`. Publisher : National Academy of Sciences Section : Biological Sciences.

[119] Rainer Zenz, Robert Eferl, Clemens Scheinecker, Kurt Redlich, Josef Smolen, Helia B. Schonthaler, Lukas Kenner, Erwin Tschachler, and Erwin F. Wagner. Activator protein 1 (Fos/Jun) functions in inflammatory bone and skin disease. *Arthritis Research & Therapy*, 10(1) :201, January 2008. ISSN 1478-6354. doi : 10.1186/ar2338. URL `https://doi.org/10.1186/ar2338`.

[120] P. Matthews Connie, H. Colburn Nancy, and R. Young Matthew. AP-1 a Target for Cancer Prevention. *Current Cancer Drug Targets*, 7(4) :317–324, May 2007. URL https://www.eurekaselect.com/59294/article.

[121] Eitan Shaulian. AP-1 — The Jun proteins : Oncogenes or tumor suppressors in disguise ? *Cellular Signalling*, 22(6) :894–899, June 2010. ISSN 0898-6568. doi : 10.1016/j.cellsig.2009.12.008. URL https://www.sciencedirect.com/science/article/pii/S0898656810000033.

[122] A. S. Dhillon and E. Tulchinsky. FRA-1 as a driver of tumour heterogeneity : a nexus between oncogenes and embryonic signalling pathways in cancer. *Oncogene*, 34(34) :4421–4428, August 2015. ISSN 1476-5594. doi : 10.1038/onc.2014.374.

[123] Xiaoyan Jiang, Hui Xie, Yingyu Dou, Jing Yuan, Da Zeng, and Songshu Xiao. Expression and function of FRA1 protein in tumors. *Molecular Biology Reports*, 47(1) :737–752, January 2020. ISSN 1573-4978. doi : 10.1007/s11033-019-05123-9.

[124] Aaron R. Quinlan and Ira M. Hall. BEDTools : a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6) :841–842, March 2010. ISSN 1367-4803. doi : 10.1093/bioinformatics/btq033. URL https://doi.org/10.1093/bioinformatics/btq033.

[125] Ming Li, Bin Ma, and Lusheng Wang. Finding similar regions in many strings. In *Proceedings of the thirty-first annual ACM symposium on Theory of Computing*, STOC '99, pages 473–482, New York, NY, USA, May 1999. Association for Computing Machinery. ISBN 978-1-58113-067-6. doi : 10.1145/301250.301376. URL https://doi.org/10.1145/301250.301376.

[126] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning Important Features Through Propagating Activation Differences. *arXiv :1704.02685 [cs]*, October 2019. URL http://arxiv.org/abs/1704.02685. arXiv : 1704.02685.

[127] Finale Doshi-Velez and Been Kim. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv :1702.08608 [cs, stat]*, March 2017. URL http://arxiv.org/abs/1702.08608. arXiv : 1702.08608.

[128] Carlo Emilio Bonferroni. Teoria Statistica Delle Classi e Calcolo Delle Probabilità. 1936.

[129] Robert E. Thurman, Eric Rynes, Richard Humbert, Jeff Vierstra, Matthew T. Maurano, Eric Haugen, Nathan C. Sheffield, Andrew B. Stergachis, Hao Wang, Benjamin Vernot, Kavita Garg, Sam John, Richard Sandstrom, Daniel Bates, Lisa Boatman, Theresa K. Canfield, Morgan Diegel, Douglas Dunn, Abigail K. Ebersol, Tristan Frum, Erika Giste, Audra K. Johnson, Ericka M. Johnson, Tanya Kutyavin, Bryan Lajoie, Bum-Kyu Lee, Kristen Lee, Darin London, Dimitra Lotakis, Shane Neph, Fidencio Neri, Eric D. Nguyen, Hongzhu Qu, Alex P. Reynolds, Vaughn Roach, Alexias Safi, Minerva E. Sanchez, Amartya Sanyal, Anthony Shafer, Jeremy M. Simon, Lingyun Song, Shinny Vong, Molly Weaver, Yongqi Yan, Zhancheng Zhang, Zhuzhu Zhang, Boris Lenhard, Muneesh Tewari, Michael O. Dorschner, R. Scott Hansen, Patrick A. Navas, George Stamatoyannopoulos, Vishwanath R. Iyer, Jason D. Lieb, Shamil R. Sunyaev, Joshua M. Akey, Peter J. Sabo, Rajinder Kaul,

Terrence S. Furey, Job Dekker, Gregory E. Crawford, and John A. Stamatoyannopoulos. The accessible chromatin landscape of the human genome. *Nature*, 489(7414) :75–82, September 2012. ISSN 1476-4687. doi : 10.1038/nature11232.

[130] Andy Pohl and Miguel Beato. bwtool : a tool for bigWig files. *Bioinformatics*, 30(11) : 1618–1619, June 2014. ISSN 1367-4803. doi : 10.1093/bioinformatics/btu056. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4029031/`.

[131] Hisham Mohammed, I. Alasdair Russell, Rory Stark, Oscar M. Rueda, Theresa E. Hickey, Gerard A. Tarulli, Aurelien A. A. Serandour, Stephen N. Birrell, Alejandra Bruna, Amel Saadi, Suraj Menon, James Hadfield, Michelle Pugh, Ganesh V. Raj, Gordon D. Brown, Clive D'Santos, Jessica L. L. Robinson, Grace Silva, Rosalind Launchbury, Charles M. Perou, John Stingl, Carlos Caldas, Wayne D. Tilley, and Jason S. Carroll. Progesterone receptor modulates estrogen receptor-alpha action in breast cancer. *Nature*, 523(7560) : 313–317, July 2015. ISSN 0028-0836. doi : 10.1038/nature14583. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4650274/`.

[132] Avanti Shrikumar, Katherine Tian, Žiga Avsec, Anna Shcherbina, Abhimanyu Banerjee, Mahfuza Sharmin, Surag Nair, and Anshul Kundaje. Technical Note on Transcription Factor Motif Discovery from Importance Scores (TF-MoDISco) version 0.5.6.5. October 2018. URL `https://arxiv.org/abs/1811.00416v5`.

[133] Jaime Abraham Castro-Mondragon, Sébastien Jaeger, Denis Thieffry, Morgane Thomas-Chollier, and Jacques van Helden. RSAT matrix-clustering : dynamic exploration and redundancy reduction of transcription factor binding motif collections. *Nucleic Acids Research*, 45(13) :e119, July 2017. ISSN 0305-1048. doi : 10.1093/nar/gkx314. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5737723/`.

[134] Gherman Novakovsky, Manu Saraswat, Oriol Fornes, Sara Mostafavi, and Wyeth W. Wasserman. Biologically relevant transfer learning improves transcription factor binding prediction. *Genome Biology*, 22(1) :280, September 2021. ISSN 1474-760X. doi : 10.1186/s13059-021-02499-5. URL `https://doi.org/10.1186/s13059-021-02499-5`.

[135] Yuval Benjamini and Terence P Speed. Estimation and correction for GC-content bias in high throughput sequencing. page 27.

[136] Fayrouz Hammal, Pierre De Langen, Aurélie Bergon, Fabrice Lopez, and Benoit Ballester. ReMap 2022 : a database of Human, Mouse, Drosophila and Arabidopsis regulatory regions from an integrative analysis of DNA-binding sequencing experiments (Under review).

Annexe

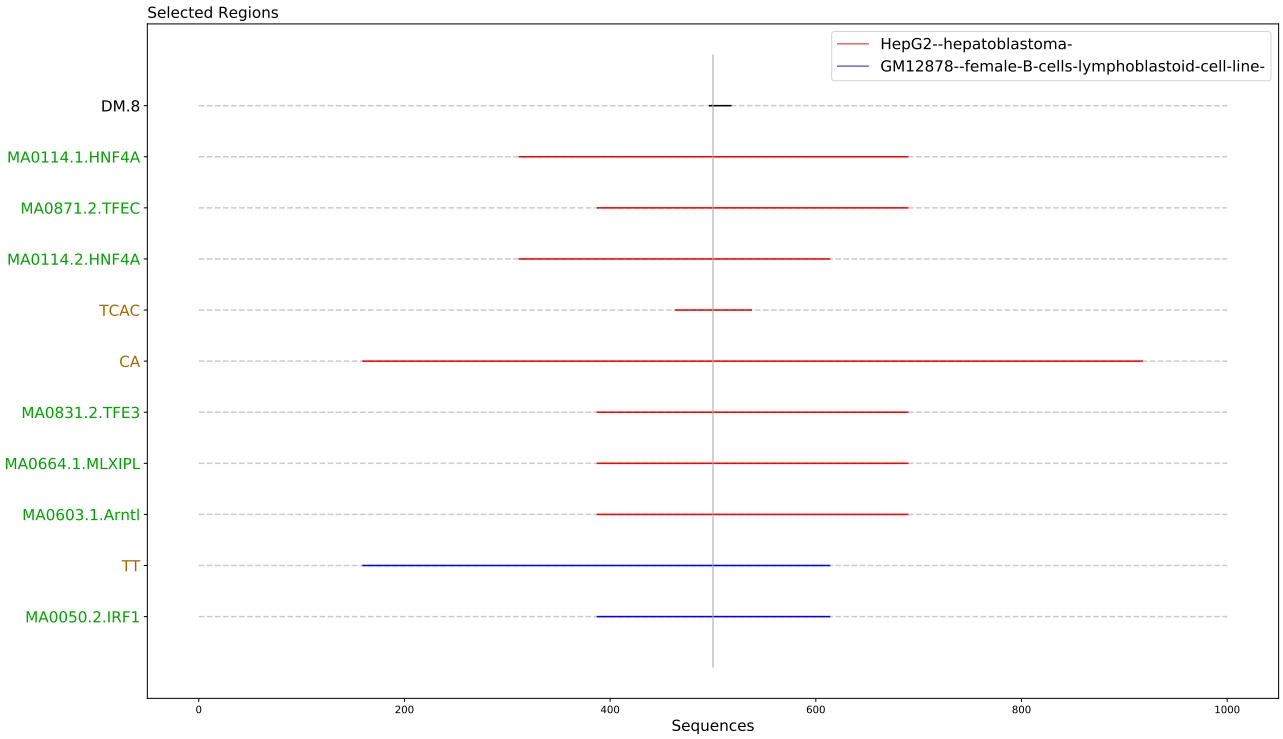

FIGURE A.1 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur USF2, HepG2/GM12878 (A).*

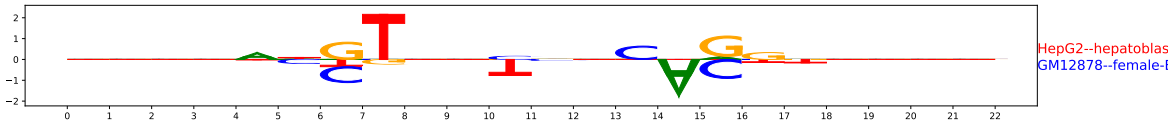

FIGURE A.2 – *Représentation logo de DM sur USF2, HepG2/GM12878 (A).*

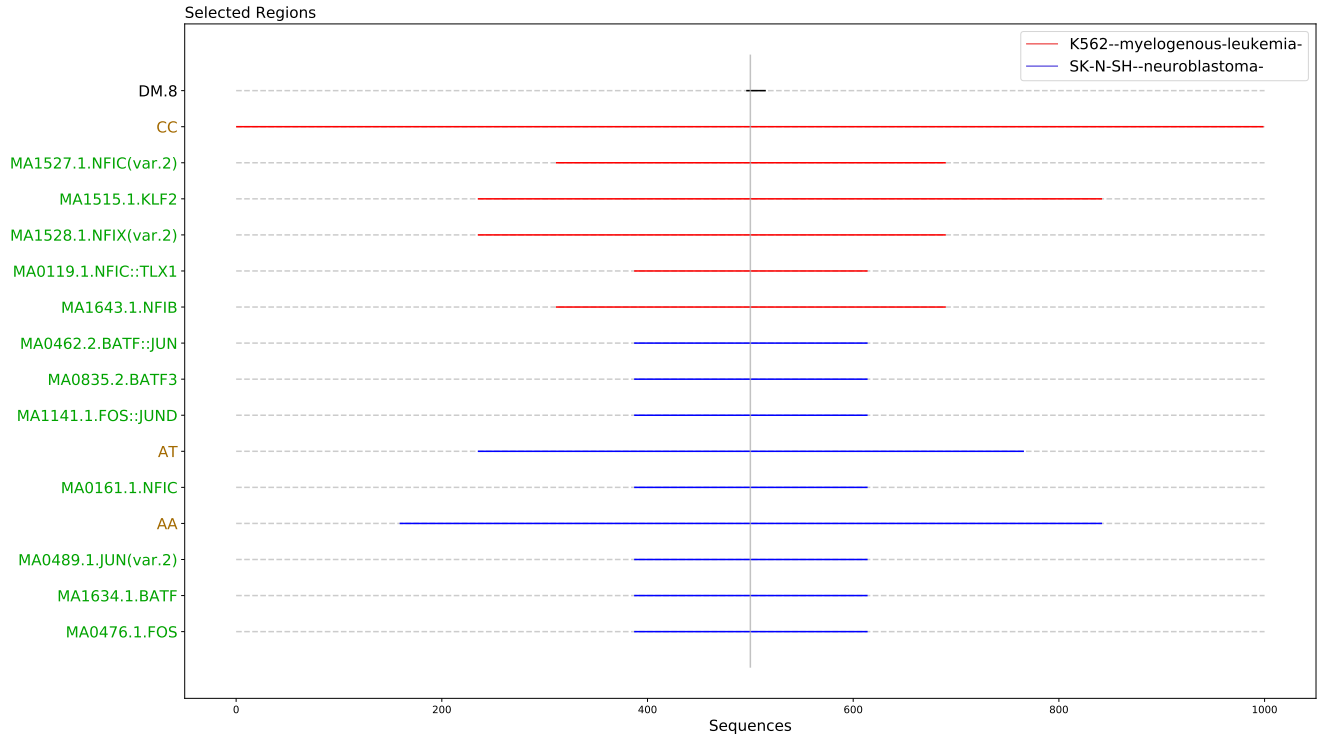

Figure A.3 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur NFIC, K562/SK-N-SH (B).*

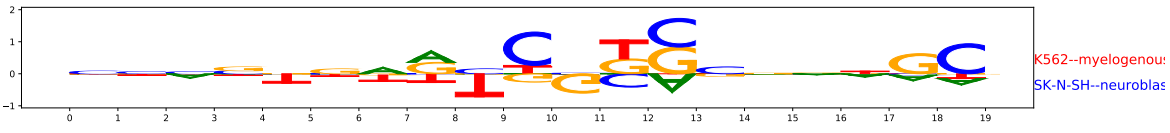

Figure A.4 – *Représentation logo de DM sur NFIC, K562/SK-N-SH (B).*

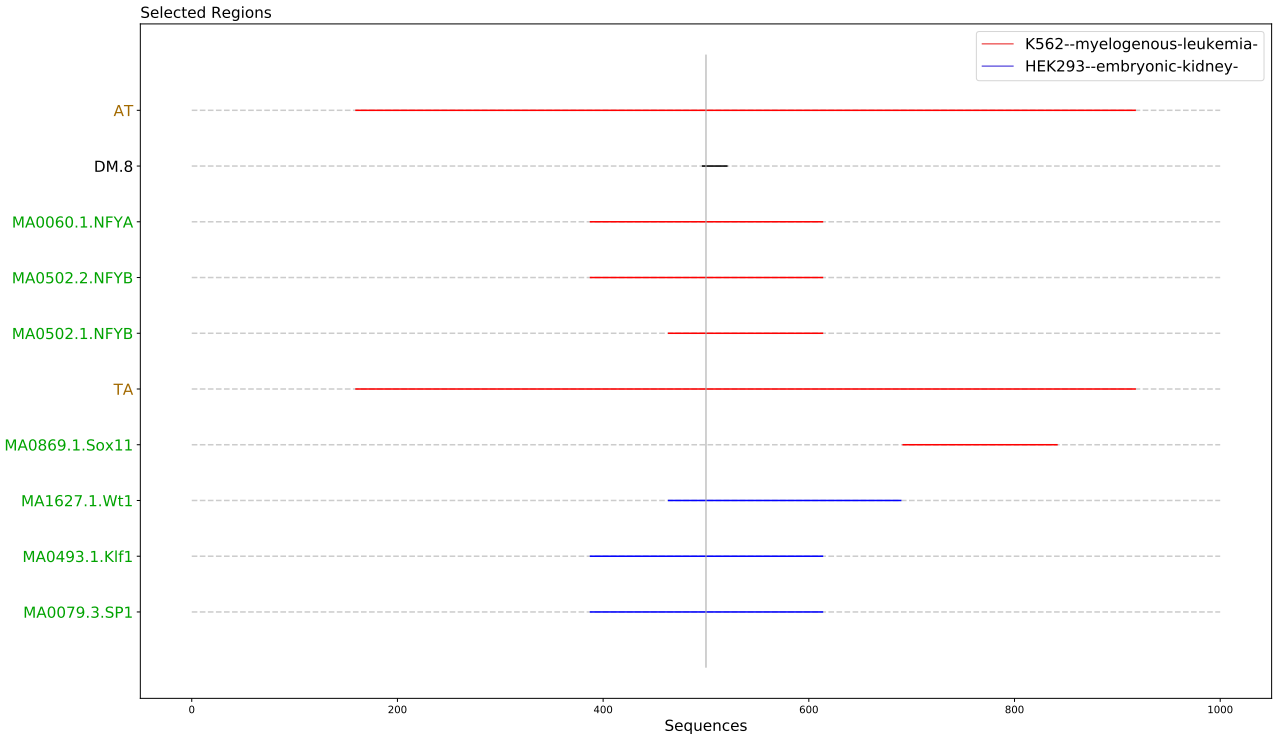

FIGURE A.5 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur SP2, K562/HEK293 (C).*

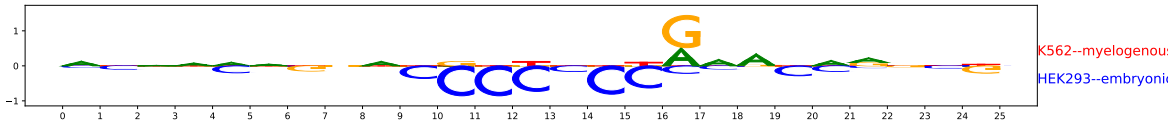

FIGURE A.6 – *Représentation logo de DM sur SP2, K562/HEK293 (C).*

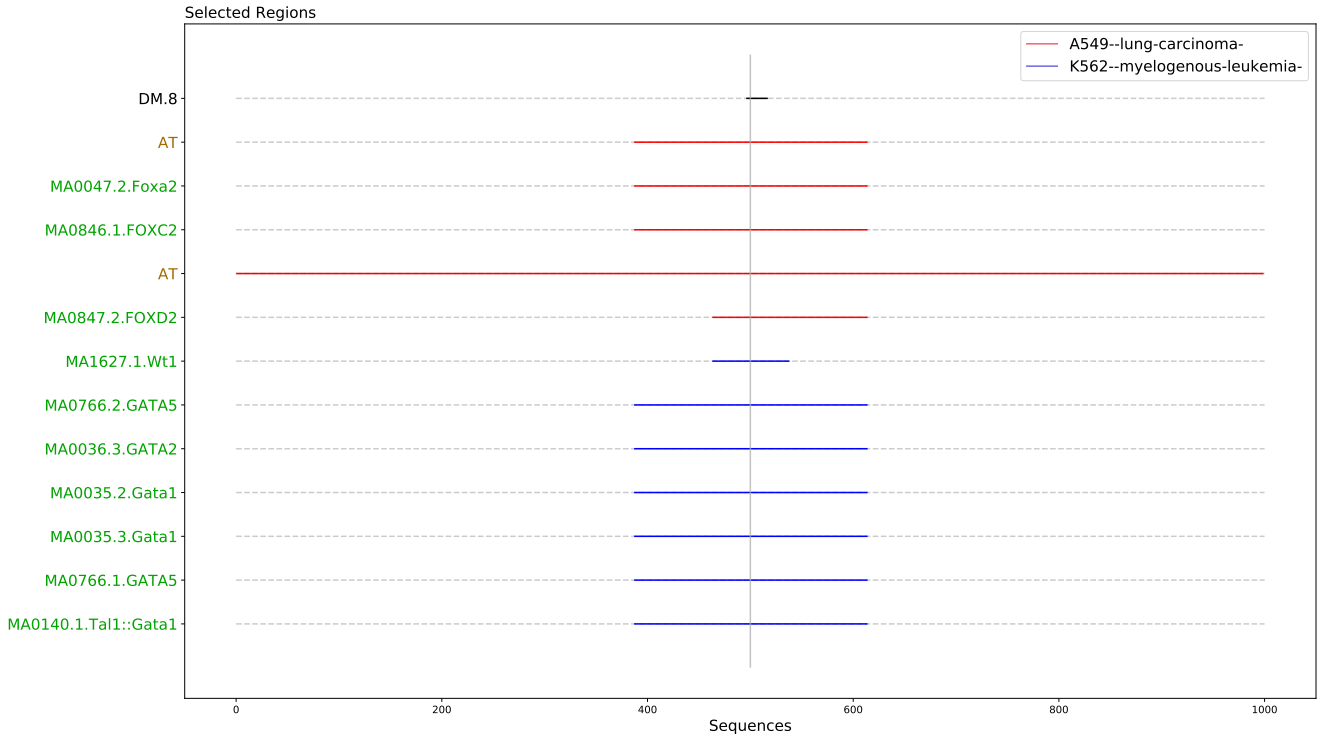

Figure A.7 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur JUNB, A549/K562 (D).*

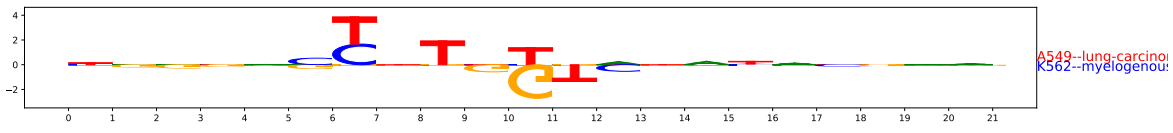

Figure A.8 – *Représentation logo de DM sur JUNB, A549/K562 (D).*

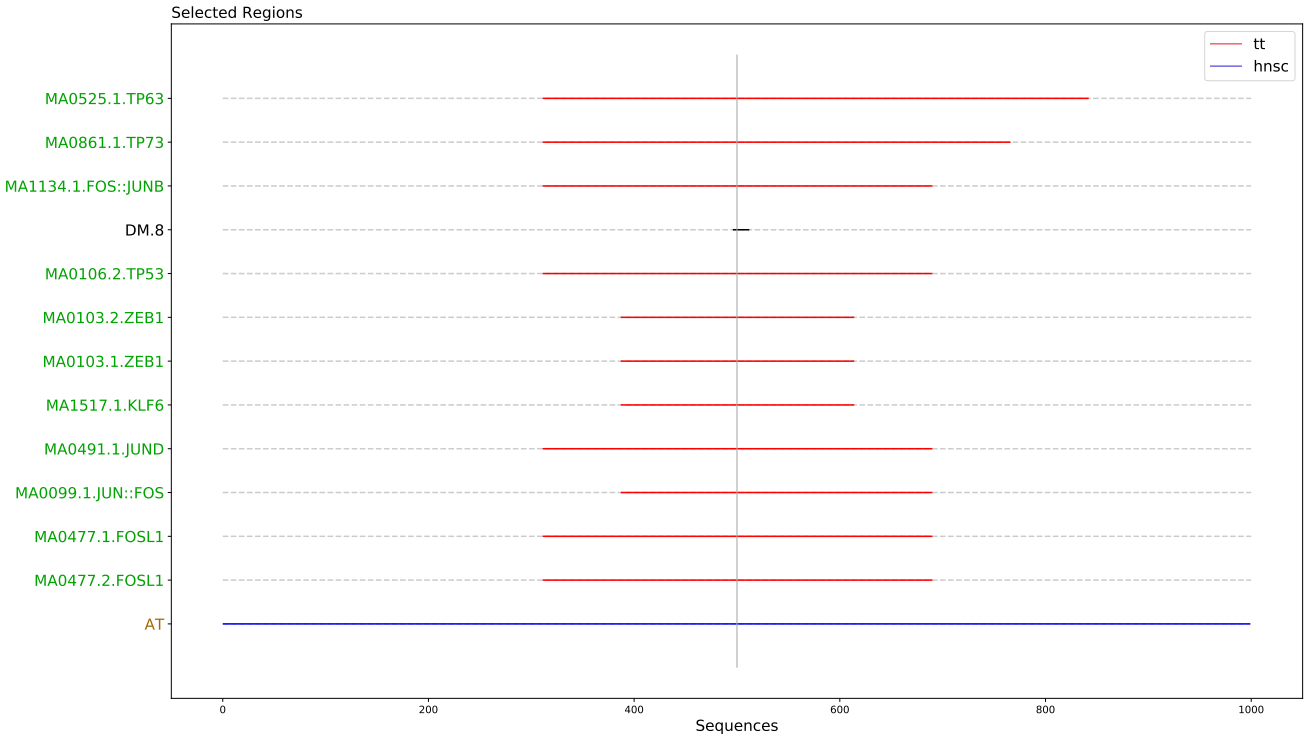

FIGURE A.9 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur SOX2, TT/HNSC (E).*

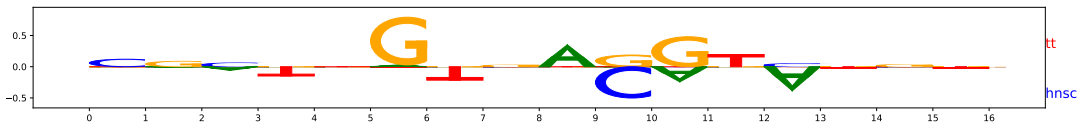

FIGURE A.10 – *Représentation logo de DM sur SOX2, TT/HNSC (E).*

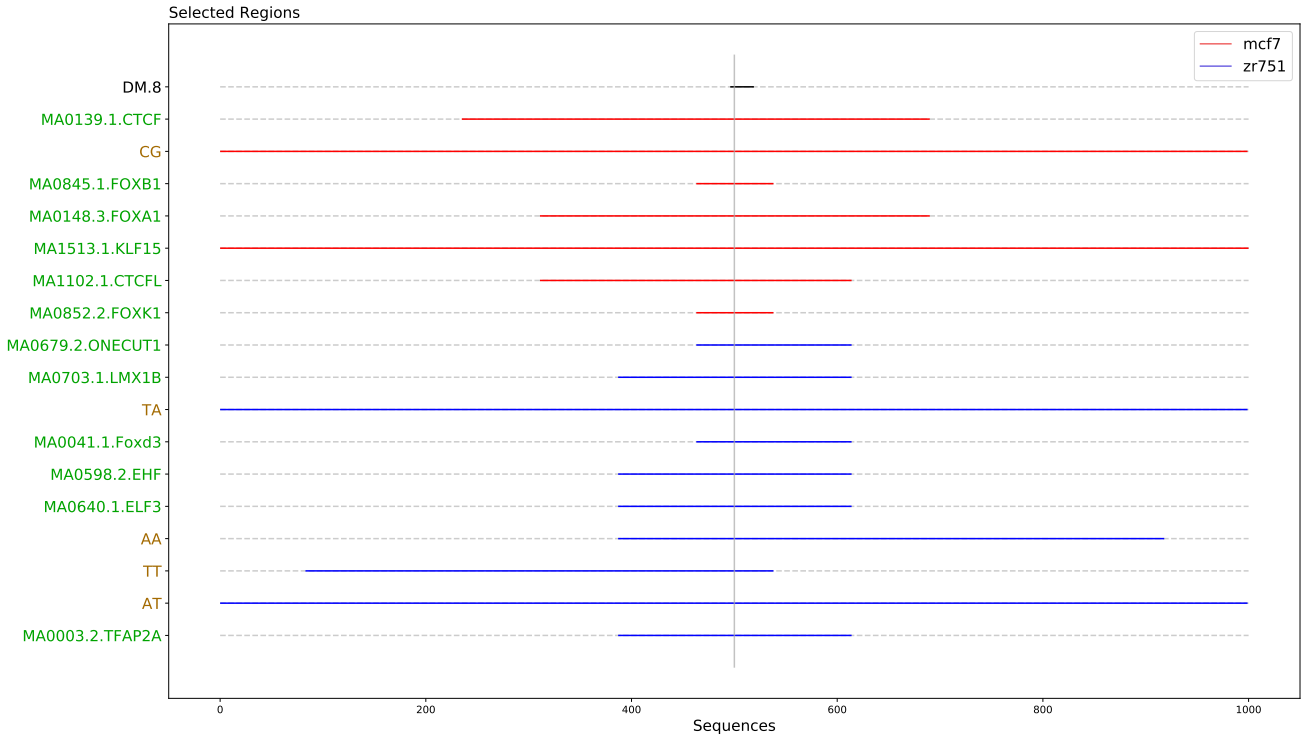

Figure A.11 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur FOXA1, MCF-7/zr75-1 (F).*

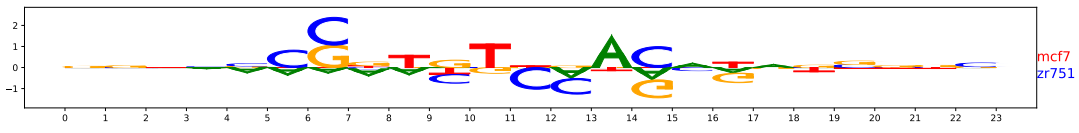

Figure A.12 – *Représentation logo de DM sur FOXA1, MCF-7/zr75-1 (F).*

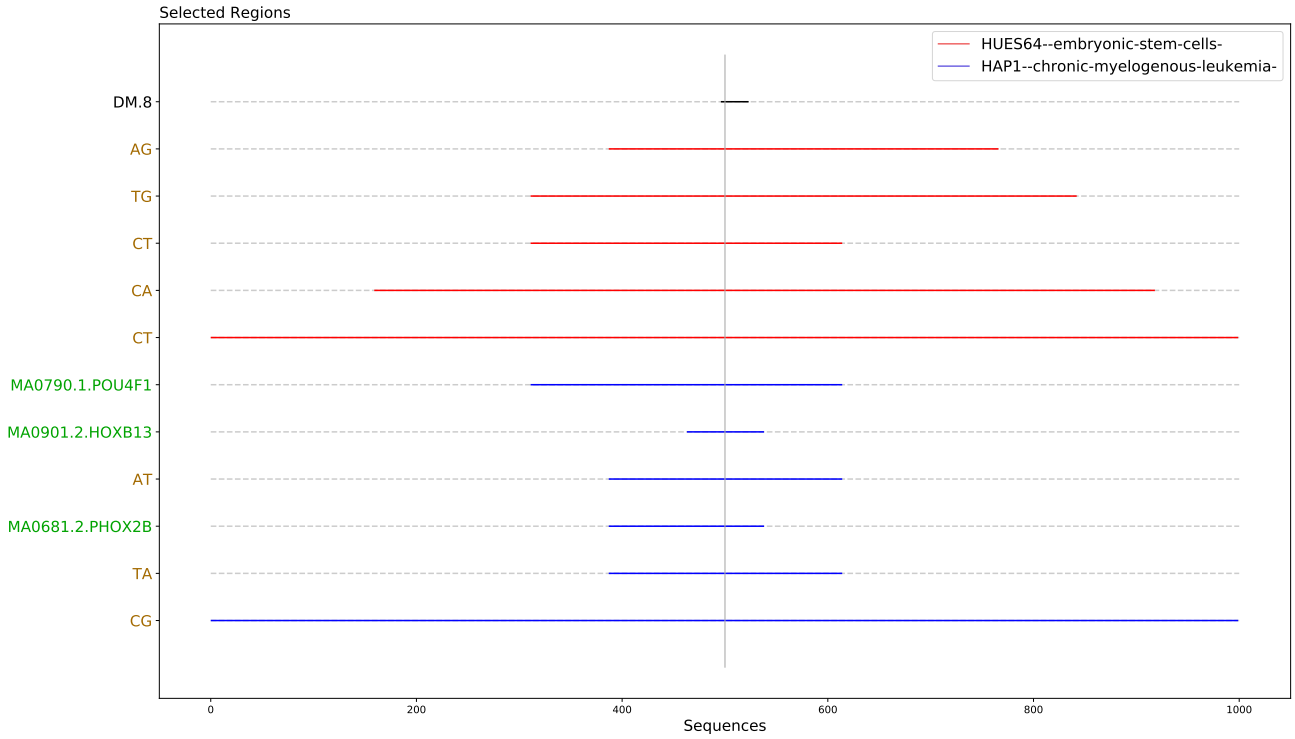

FIGURE A.13 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur CTCF, HUES64/HAP1 (G).*

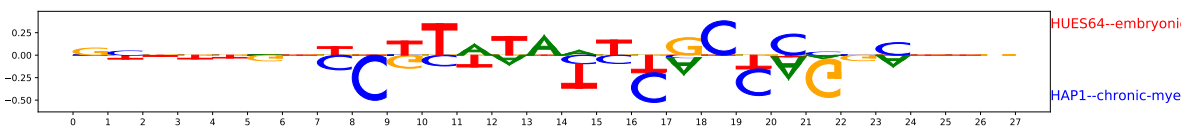

FIGURE A.14 – *Représentation logo de DM sur CTCF, HUES64/HAP1 (G).*

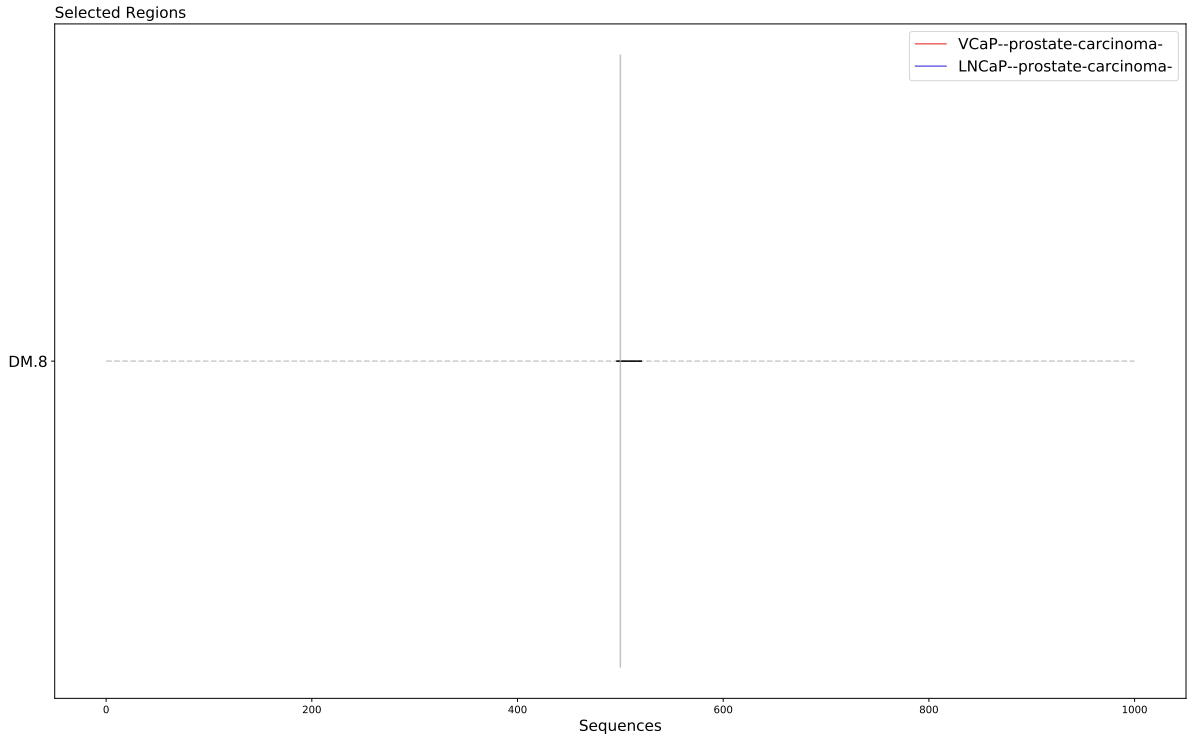

Figure A.15 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur AR, VCaP/LNCaP (H).*

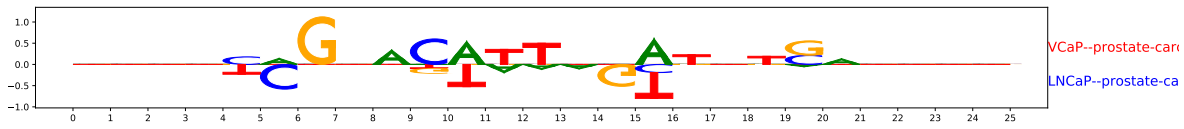

Figure A.16 – *Représentation logo de DM sur AR, VCaP/LNCaP (H).*

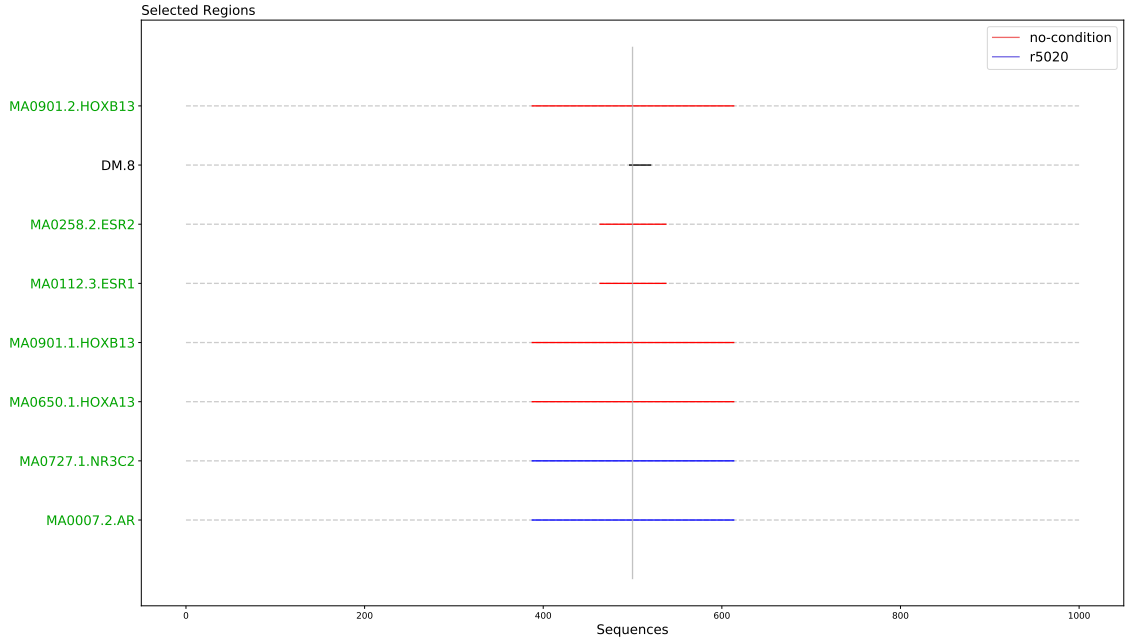

Figure A.17 – *Graphiques des variables sélectionnées et de leur région associée dans TFscope sur ESR1 (T47D), sans traitement/r5020.*

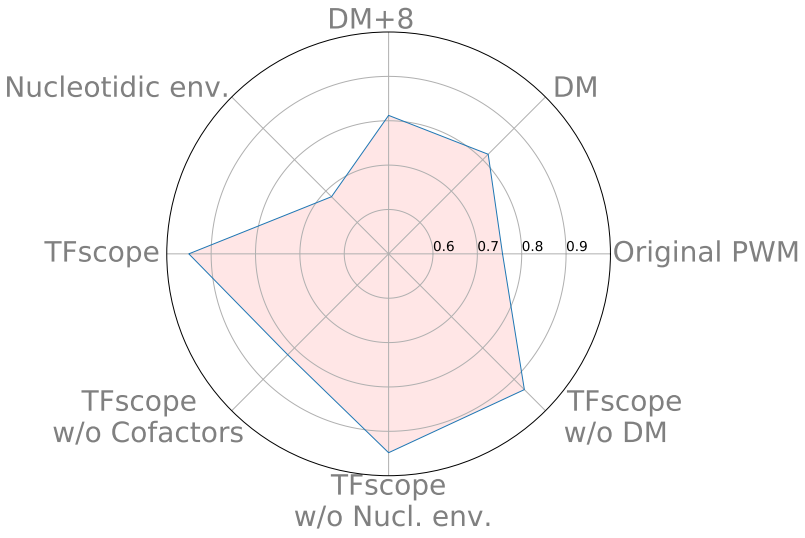

Figure A.18 – *Radar AUROC des méthodes associées à TFscope sur ESR1 (T47D), sans traitement/r5020.*

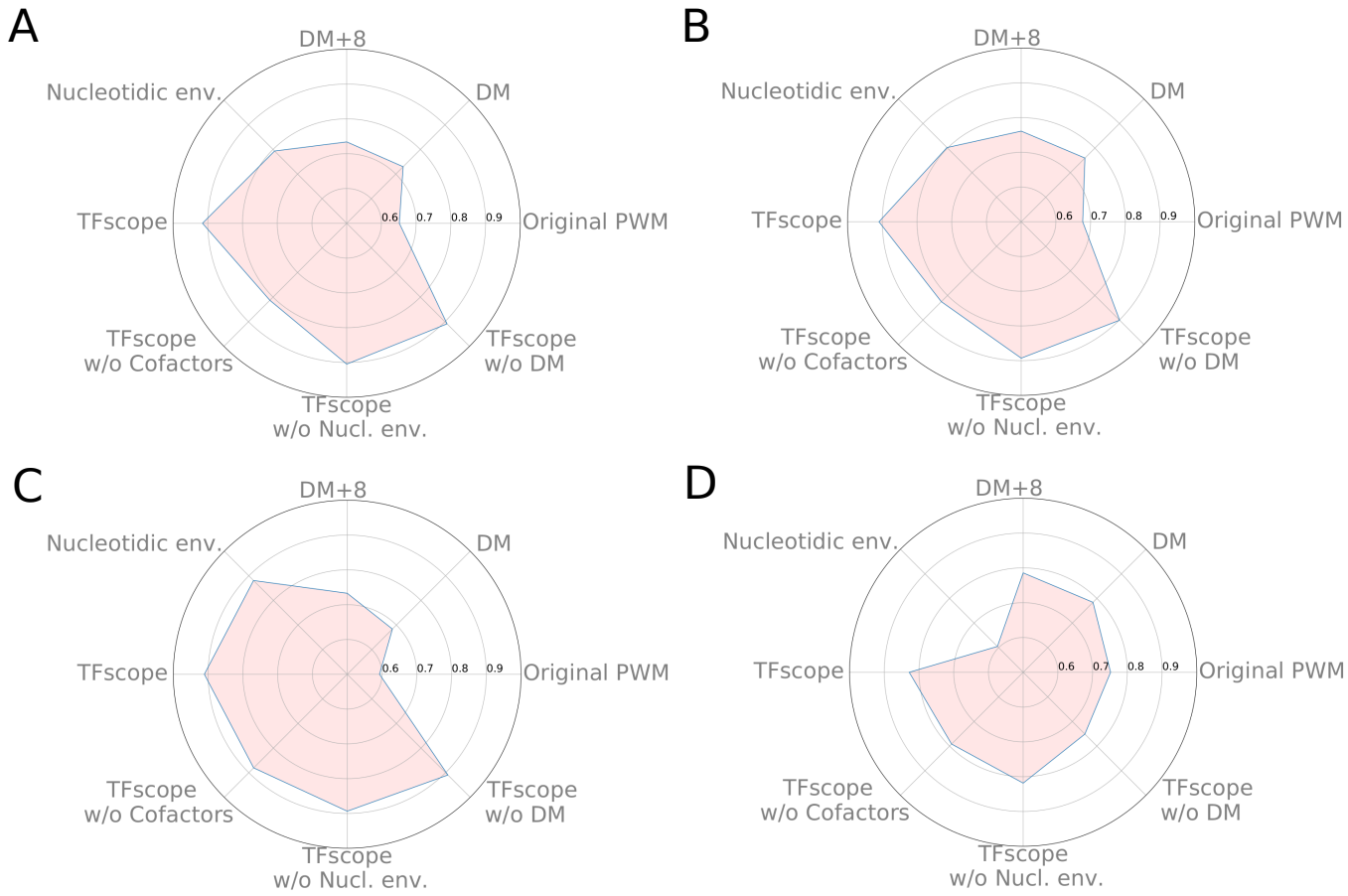

Figure A.19 – *Graphiques radars de différentes expériences dont les données de ChIP-seq sont issues du même laboratoire.*
A : ESR1 (progestérone), T47D/MCF-7. **B** : ESR1 (r5020), MCF-7/T47D. **C** : ESRRA (Heregulin), SKBR3/BT-474. **D** : FOXA1 (etoh24h), VCaP/LNCaP.

Résumé

Dans cette thèse, nous nous intéressons aux déterminants génomiques qui peuvent expliquer les différences de fixation d'un facteur de transcription (TF) particulier entre deux types cellulaires. Les facteurs de transcriptions reconnaissent des sous-séquences particulieres sur lesquelles ils se fixent, l'ensemble de ces sous-séquences est modélisé dans des motifs de fixation. Cependant, le motif de fixation d'un TF ne permet pas d'expliquer entièrement sa fixation. En effet, il n'est pas forcément fixé dès qu'il reconnait son motif de fixation et ne se fixe pas aux mêmes loci en fonction des types cellulaires. Le but de ce travail est donc d'étudier d'autres informations, afin de mieux comprendre la fixation des TF dans différents types cellulaires. Ce problème est étudié dans un cadre de classification supervisée, où les exemples sont des séquences génomiques et les deux classes correspondent aux types cellulaires dans lesquels la séquence est liée par le TF d'intérêt. Les séquences sont décrites par trois types d'informations génomiques qui sont extraites des séquences brutes par trois méthodes dédiées : la spécificité nucléotidique du site de fixation, le contenu nucléotidique autour du site de fixation, et la présence et la position de sites de fixation potentiels d'autres facteurs de transcription coopérants. Toutes ces caractéristiques sont utilisées dans un modèle de régression logistique entraîné avec une vraisemblance pénalisée sur différents problèmes de classification associant un TF dans deux tissus différents. Dans chaque expérience, le modèle est utilisé pour identifier les éléments régulateurs qui sont les plus importants pour les différences de type cellulaire. Nos expériences montrent qu'il est possible de distinguer les sites de fixation spécifiques aux cellules sur la base de la séquence uniquement. De plus, une analyse globale des résultats montre que l'importance relative des trois types d'information dépend fortement du TF et des types cellulaires.

Abstract

In this thesis, we are interested in the genomic determinants that can explain the binding differences of a particular transcription factor (TF) between two cell types. Transcription factors recognise and bind to particular subsequences, the collection of potential subsequences is modelised in binding motifs. However, the binding motif of a TF does not fully explain the binding. Indeed, the TF is not necessarily bound as soon as it recognises its binding motif and does not bind to the same loci depending on the cell type. The aim of this work is therefore to study other information in order to better understand TF binding in different cell types. This problem is studied in a supervised classification framework, where examples are genomic sequences and the two classes correspond to the cell types where the sequence is bound by the TF of interest. Sequences are described by three kinds of genomic features that are extracted from raw sequences by three dedicated methods : the nucleotide specificity of the binding site, the nucleotide content around the binding site, and the presence and position of potential binding sites of other cooperative transcription factors. All these features are used in a logistic regression model trained with penalized likelihood on different classification problems associating one TF in two different tissues. In each experiment, the model is used to identify the regulatory elements that are the most important for cell type differences. Our experiments show that it is possible to distinguish cell specific binding sites on the basis of the sequence only. Moreover a global analysis of the results show that the relative importance of the three kind of information strongly depends on TF and cell types.