Algorithmes pour l'analyse de régions régulatrices dans le génome d'eucaryotes supérieurs

Matthieu Defrance

Algorithmes pour l'analyse de régions régulatrices dans le génome d'eucaryotes supérieurs

THÈSE

présentée et soutenue publiquement le 13 décembre 2006

pour l'obtention du

Doctorat de l'Université des Sciences et Technologies de Lille

(spécialité informatique)

par

Matthieu Defrance

Composition du jury

Président :		
Rapporteurs :	Bernard Jacq, Directeur de Recherche CNRS	IBDM, Université Aix-Marseille II
	David J. Sherman, Maître de Conférences HDR	LaBRI, Université de Bordeaux I
Examinateurs :	Clarisse Dhaenens-Flipo, Professeur	LIFL, Université de Lille I
	Ralf Blossey, Directeur de Recherche CNRS	IRI, Lille
	Thierry Lecroq, Professeur	LITIS, Université de Rouen
Directeur :	Hélène Touzet, Chargé de Recherche CNRS HDR	LIFL, Université de Lille I

UNIVERSITÉ DES SCIENCES ET TECHNOLOGIES DE LILLE
Laboratoire d'Informatique Fondamentale de Lille — UPRESA 8022
U.F.R. d'I.E.E.A. – Bât. M3 – 59655 VILLENEUVE D'ASCQ CEDEX
Tél. : +33 (0)3 28 77 85 41 – Télécopie : +33 (0)3 28 77 85 37 – email : direction@lifl.fr

Table des matières

Notations

Chaîne de Markov

- MM_k : chaîne de Markov d'ordre k.
- T : matrice de transition.
- S : vecteur stationnaire.

Motifs

- Σ : alphabet $\{A, C, G, T\}$.
- b : lettre de l'alphabet Σ.
- p_b : fréquence de la lettre b.
- u : mot sur l'alphabet Σ.
- v : mot sur l'alphabet Σ.
- l : longueur du mot u.
- u_i : lettre i du un mot $u = u_1, ..., u_l$.
- τ : seuil sur le score.

Séquence génomique

- ϕ : séquence génomique.
- L : longueur de la séquence ϕ.
- ϕ_i : élément i de la séquence $\phi = \phi_1, ..., \phi_L$.
- Φ : ensemble de séquences génomique = $\Phi = \{\phi^1, ..., \phi^n\}$.
- N : taille totale des séquences ($\sum_i |\phi_i|$).

Matrice de comptage

- M : matrice de comptage.
- $M_{b,i}$: élément de la ligne b et colonne i de la matrice M.
- m : longueur de la matrice de comptage M.
- W : matrice de poids.
- $W_{b,i}$: élément de la ligne b et colonne i de la matrice W.
- w : longueur de la matrice de poids W.
- $P_{b,i}$: fréquence associée à la lettre b et la position i.

Fenêtre

- $[i, j]$: une fenêtre encadrée par les positions i et j (incluses).
- δ : largeur de la fenêtre $[i, j]$ ($=j - i + 1$).
- X_{ij} : variable aléatoire associée au comptage des occurrences dans la fenêtre $[i, j]$.
- Y_{ij} : variable aléatoire associée au score des occurrences dans la fenêtre $[i, j]$.
- k : nombre d'occurrences pour un motif donné.
- k_i : nombre d'occurrences à la position i d'une séquence pour un motif donné.
- μ : nombre d'occurrences attendues.
- μ^n : nombre d'occurrences attendues pour la séquence n.
- μ_i : nombre d'occurrences attendues à la position i.
- μ_{ij} : nombre d'occurrences attendues dans la fenêtre $[i, j]$.

Score

- s_i : log score à la position i.
- S_i : score total à la position i.

Introduction

L'ADN, parfois appelé "molécule de la vie", est présent dans chacune des cellules qui constituent un organisme vivant. Il est le support de l'information génétique qui gouverne le fonctionnement des cellules. Sa structure en double chaîne complémentaire lui permet en se répliquant, de transmettre au cours des générations cellulaires, le patrimoine génétique d'un organisme. C'est grâce à cette information que se construisent les molécules actives au sein des cellules. Une chaîne d'ADN est constituée par une succession de briques élémentaires, les bases.

Si l'on envisage l'ADN du point de vue de l'enchaînement des bases, l'ADN propre à un organisme peut être considéré comme un "livre" écrit avec un alphabet à quatre lettres, A, C, G et T. Avec la mise à disposition des premières séquences d'ADN, cette vision des choses, bien qu'extrêmement "simple", a ouvert la voie à de nouvelles possibilités de traitement bio-informatique. De nouvelles problématiques se sont ainsi développées sur base de l'algorithmique du texte : la comparaison de séquences, l'annotation de séquences, la recherche de motifs, l'indexation de séquences, ...

L'ensemble de l'ADN propre à un organisme, son patrimoine génétique, est rassemblé dans son génome. Dans celui-ci, certaines portions codent plus spécifiquement pour des molécules fonctionnelles : ce sont les gènes. L'expression d'un gène donné est un mécanisme complexe, conditionné par le contexte. Cela dépend par exemple de l'état métabolique, du tissu dans lequel la cellule se trouve. C'est cette expression différenciée qui permet, par exemple, à une cellule du foie d'avoir un fonctionnement différent de celle du muscle, tout en possédant le même patrimoine génétique.

Comprendre les mécanismes qui participent au contrôle de l'expression des gènes - *la régulation de l'expression* - est une tâche essentielle pour la compréhension du fonctionnement d'une cellule. En particulier, la "compréhension" de certaines maladies à facteurs génétiques passe par une connaissance plus fine des mécanismes régulant l'expression des gènes. Mais c'est une tâche difficile, car la régulation y intervient à plusieurs niveaux.

C'est sur la régulation de la première phase de l'expression, l'initiation de la transcription, que l'on dispose actuellement de la connaissance la plus précise. En particulier, l'action de protéines régulatrices - les facteurs de transcription - joue un rôle important dans l'initiation de la transcription. Ces facteurs, pour pouvoir agir, se fixent à l'ADN sur des sites spécifiques

appelés sites de fixation de facteurs de transcription. Ces sites ou éléments *cis*-régulateurs correspondent à de courts fragments d'ADN comportant une certaine forme de variabilité [81]. L'identification de ces sites de fixation est une étape importante dans la compréhension de la régulation.

De nombreuses techniques expérimentales permettant d'analyser l'expression au niveau transcriptionnel existent (retard sur gel, empreinte à DNase, immuno-précipitation, ChIP-on-chip, puces à ADN, ...). La recherche, avec ces techniques expérimentales de signaux de régulation *in vitro* et a fortiori *in vivo*, pose des contraintes de moyens et de temps importantes. Ces techniques sont, de ce fait, difficilement "adaptables" à des recherches purement exploratoires. Une autre piste est alors de rechercher des éléments *cis*-régulateurs *in silico*, par bio-informatique. Mais cette tâche est difficile du fait de la taille des données et de la nature variable des éléments recherchés.

Une des façons d'appréhender le problème de la recherche d'éléments *cis*-régulateurs est de considérer ce problème comme celui de la recherche de motifs dans un texte. Cette formulation, bien que réductrice vis-à-vis de la complexité des phénomènes biologiques sous-jacents, possède l'avantage d'offrir un angle d'étude propre à l'analyse informatique sur des bases établies d'algorithmique du texte. Dans ce cadre, un motif modélise les courts fragments d'ADN reconnus par un facteur de transcription donné. Un exemple de sites reconnus par le facteur p65 est donné dans la figure 1 :

```
      cGGGAATTTCCaacag
       GGGAATTTCCccggcct
      cGGGAATTTCCgcttc
       TGGAATTTCCggcgcc
gccggggTGGAATTTCC
     gcCGGAATTTCCggtg
 gcacccGGGGATTTCC
 atgatcGGGACTTTCC
  aagatCGGAGTTTCC
       GGGAATTCCCgcatcca
tgcttgaCGGACTTTCC
 ccgattCGGACTTTCC
   tcgcGGGGAATTCCgg
tacggccTGGGGTTTCC
 aggtgtGGGGATTCCC
  ggagtGGGGTTTTCCc
 gcacccGGGGGATTCC
 ctgccgGTGGGTTTCCc
```

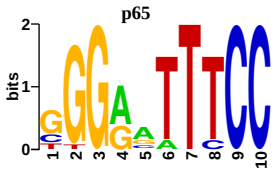


FIG. 1: Exemple de sites reconnus par le facteur de transcription p65 tiré de la banque JASPAR [72].

Parmi les différents modèles disponibles pour appréhender ces motifs, les plus courants sont certainement les matrices de comptage [30, 66, 82]. Les matrices de comptage donnent lieu à la construction de modèles qui reposent sur des bases de théorie de l'information et de la thermodynamique [82]. Malheureusement, la recherche brute d'éléments *cis*-régulateurs à partir de matrices de comptage conduit à un grand nombre de prédictions erronées. La raison

est que le contenu informationnel des motifs est intrinsèquement faible. Pour remédier à cette faible spécificité, d'autres informations doivent être prises en compte.

Une première solution pour améliorer la qualité des prédictions est d'utiliser la génomique comparative [92, 96]. L'idée est la suivante : si l'on considère un ensemble de gènes provenant d'organismes proches, ces gènes doivent partager des mécanismes de régulation, qui se manifestent par des signaux régulateurs conservés sur les séquences. L'identification de ces signaux est rendue possible par la pression de sélection qui s'exerce différemment sur ces régions fonctionnelles. Cette solution comporte des contraintes fortes quant à la disponibilité d'un nombre suffisant de génomes à une "bonne" distance phylogénétique. Eddy [21] a par exemple montré que le nombre d'espèces nécessaires pour effectuer une analyse comparative efficace était inversement proportionnel à la taille des signaux recherchés.

Une autre solution pour prendre en compte davantage d'informations est de rechercher des signaux partagés par un ensemble de gènes co-exprimés. On peut en effet penser que les séquences promotrices associées à ces ensembles de gènes doivent posséder des motifs régulateurs sur-représentés par rapport à d'autres jeux neutres. Dans ce cas, le problème peut se formuler de la manière suivante : rechercher dans les séquences promotrices, les motifs pour lesquels le nombre d'occurrences est significativement plus élevé que ce qui est attendu par hasard. De nombreuses stratégies de recherche ont été proposées pour répondre à ce problème. En particulier, des approches exactes énumératives lorsque la structure des motifs recherchés est simple [50, 59, 79, 88] ou encore des approches limitant l'espace des motifs à des banques de matrices [1, 34].

Le travail de cette thèse s'inscrit dans ce cadre général de la recherche d'éléments *cis*-régulateurs. Nous proposons d'envisager d'autres pistes pour répondre à ce problème. Certains facteurs de transcription sont connus comme ayant une affinité positionnelle forte [63, 87]. Une idée est de prendre en compte la conservation spatiale des sites de fixation des facteurs de transcription, lorsque cela est pertinent, par exemple par rapport au site d'initiation de la transcription. Un autre point, qu'il est possible d'améliorer, concerne la façon dont la conservation entre espèces est prise en compte en génomique comparative. Habituellement, la stratégie utilisée pour rechercher des éléments conservés, consiste à filtrer les régions analysées sur base d'un alignement. Cette stratégie rend difficile la recherche de courts fragments conservés dans des régions qui le sont moins.

Au vu de ces éléments, nous proposons une nouvelle stratégie de recherche locale, qui marie conservation spatiale et conservation entre espèces. Cette approche permet de répondre au problème de la recherche de motifs sur-représentés localement lorsque l'environnement de recherche est hétérogène, c'est-à-dire pour des séquences pouvant provenir d'organismes différents ou de régions différentes. Nous proposons pour cela un moyen d'appréhender et de mesurer statistiquement la qualité d'une sur-représentation locale. Nous fournissons également une méthode efficace pour rechercher ce type de sur-représentation. Cette stratégie est mise en œuvre dans le logiciel TFM-Explorer, que nous avons évalué sur plusieurs jeux de données humain, murin et du rat.

Ce mémoire de thèse est divisé en 4 chapitres.

Dans le premier chapitre, nous introduisons le contexte biologique du problème : la régulation de l'expression des gènes eucaryotes. Nous y précisons les deux principales étapes de l'expression des gènes : la transcription et la traduction ainsi que les différents niveaux où la régulation peut avoir lieu. Ensuite, nous détaillons le niveau sur lequel nous allons travailler : la régulation transcriptionnelle. En particulier, nous étudions l'influence des facteurs de transcription et de leurs sites de fixation. Enfin, nous présentons différentes techniques expérimentales permettant d'analyser la régulation transcriptionnelle au niveau de l'interaction ADN-protéine et de la production d'ARN messagers.

Dans le deuxième chapitre, nous présentons les modèles et méthodes algorithmiques pour la recherche d'éléments *cis*-régulateurs. Nous commençons par présenter comment modéliser, dans un premier temps, les séquences génomiques par des chaînes de Markov et, dans un deuxième temps, les sites de fixation de facteur de transcription par des matrices de comptage et des chaînes consensus. En nous servant de ces modèles, nous présentons les stratégies de recherche utilisant la génomique comparative [44, 96] ainsi que les méthodes recherchant des motifs sur-représentés, dans un ensemble de séquences [1, 34, 86].

Dans le troisième chapitre, nous discutons des limites des approches comparatives actuelles et nous présentons une piste complémentaire pour la recherche de signaux régulateurs : la conservation spatiale. Nous proposons un nouveau modèle pour la sur-représentation permettant de tenir compte de manière souple de la conservation spatiale et entre espèces des éléments *cis*-régulateurs. Pour cela, les notions de *sur-représentation locale* et de *fenêtre locale* sont introduites. Nous expliquons comment évaluer la significativité d'une fenêtre locale, en tenant compte de la variabilité des modèles de fond.

Enfin, dans le quatrième et dernier chapitre, nous expliquons comment mettre en œuvre les idées développées dans le chapitre précédent, avec un algorithme d'identification de fenêtres locales significatives. Cet algorithme est implémenté dans un logiciel appelé TFM-Explorer, utilisant des bases de données de matrices de comptage. Nous terminons ce chapitre sur une note exploratoire en introduisant d'autres pistes applicatives de la méthode. En particulier, nous présentons quelques éléments pour l'inférence de motifs localement sur-représentés.

Chapitre 1

L'expression des gènes et sa régulation

Sommaire

La régulation de l'expression des gènes est la modulation de la production de molécules fonctionnelles à partir des gènes. Elle est à la base du contrôle de la structure et de la fonction des cellules. C'est par ces mécanismes de régulation que les cellules d'un même organisme, bien que possédant les mêmes gènes, peuvent avoir un fonctionnement différencié en fonction du contexte et du temps. Dans ce chapitre, nous rappelons quelques notions de génétique, en particulier ce qui est utile pour comprendre le développement de cette thèse : les mécanismes d'expression et de *régulation de l'expression* des gènes chez les organismes eucaryotes. La première section est consacrée à des rappels généraux sur la définition de gène et des mécanismes d'expression associés. En particulier, la première étape de ces mécanismes - la transcription de l'ADN en ARN - y est détaillée. La deuxième section décrit les mécanismes de régulation de l'expression des gènes en s'attardant sur une voie importante de la régulation : la régulation transcriptionnelle. Enfin, la dernière section présente différentes techniques expérimentales permettant de mesurer l'expression des gènes.

1.1 ADN, gène et génome

La cellule est l'unité structurale élémentaire sur laquelle se fondent les organismes vivants. On distingue en fonction du type de compartimentation de ces cellules, deux grandes catégories d'organismes : les procaryotes, comme les bactéries, et les eucaryotes comme par exemple les levures, les plantes ou les animaux. C'est sur cette deuxième catégorie d'organismes, les eucaryotes, que se concentre cette thèse.

Les eucaryotes sont des organismes composés d'une ou plusieurs cellules présentant un noyau délimité par une enveloppe nucléaire. Ce noyau isole du reste de la cellule le patrimoine génétique, c'est-à-dire l'information qui régit la production des éléments nécessaires au fonctionnement de la cellule. Ce patrimoine est supporté par une molécule double brin : l'acide désoxyribonucléique (ADN). L'ADN est composé d'une chaîne de briques élémentaires : *les nucléotides* (notés A, C, G et T) appareillés par paires (A↔T et G↔C) pour former une structure en forme de double hélice.

Chaque chaîne de nucléotides formant la double hélice d'ADN est orientée dans un sens défini en fonction de la position des liaisons entre nucléotides (5' et 3'). La lecture d'une séquence d'ADN est effectuée dans le sens 5'→3'.

Chez les eucaryotes, l'ADN est présent dans le noyau des cellules sous forme compactée en une structure appelée *chromatine*. Cette structure joue un rôle important dans l'accessibilité de l'information génétique et donc dans sa possibilité d'expression (régulation). Plusieurs niveaux de compactage organisent la structure de l'ADN. Le premier niveau, le nucléosome, est formé par l'enroulement de l'ADN autour de complexes protéiques, appelés *les histones*. Ensuite, à un deuxième niveau, la chromatine s'enroule sur elle-même pour former une structure en zig-zag. Enfin certaines portions se condensent en "super-boucles". C'est cette alternance de structures de chromatine condensées et diffuses qui forme le *chromosome*. La hiérarchie de structures est présentée dans la figure 1.1.

L'ensemble des chromosomes constitue le *génome* d'une espèce. La taille du génome est fortement variable d'un organisme à un autre. Le génome humain est par exemple composé de 23 chromosomes et contient un peu plus de trois milliards de paires de nucléotides, alors que celui de la levure (*Saccharomyces cerevisiae*) ne contient que 13 millions de paires de nucléotides répartis sur 16 chromosomes. On distingue dans le génome d'une espèce, des parties codantes - *les gènes* - qui sont utilisées pour produire des protéines, et des portions non codantes qui contiennent notamment des informations permettant de réguler l'expression de parties codantes. Un gène à protéine peut se définir comme une portion de l'ADN d'un organisme qui permet, lorsqu'elle est exprimée, de produire une protéine. Le nombre de gènes et la proportion codante du génome présentent une forte disparité d'un organisme à l'autre. Par exemple les 30 000 gènes présents dans le génome humain ne représentent que 3% de sa taille (en nombre de nucléotides), alors que les 6 000 gènes présents dans le génome de la levure représentent plus de 70 % de sa taille.

Les protéines sont les éléments essentiels de la cellule : elles sont nécessaires à sa constitution et à son fonctionnement. Au sein d'une cellule, la synthèse des protéines est le résultat de l'interaction entre l'information génétique (portée par le génome) et le contexte (milieu extracellulaire, état métabolique, tissu dans lequel elle est présente). Des mécanismes complexes de

[1] *source de l'image* : `http ://www.nature.com/nature/journal/v421/n6921/images/nature01411-f1.2.jpg`

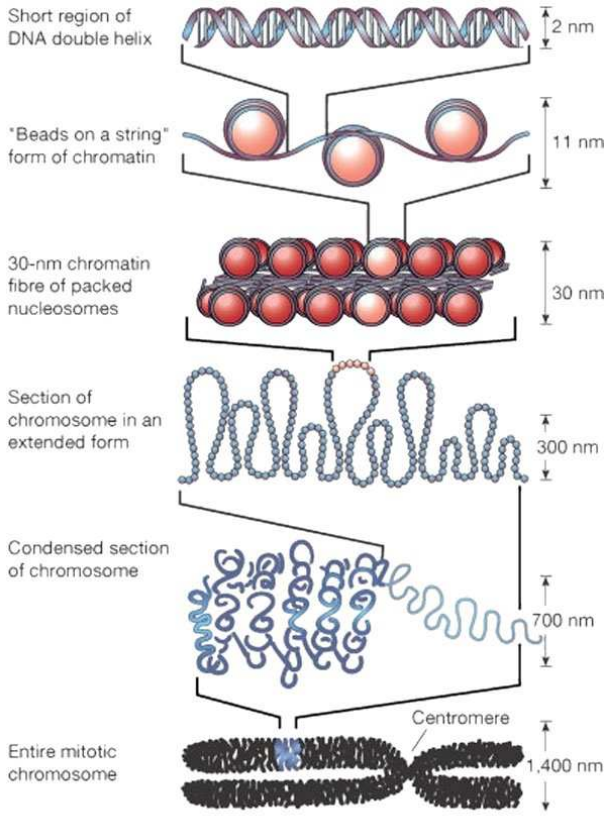


FIG. 1.1: Détails sur la structure de l'ADN[1].

régulation permettent d'exprimer ou de réprimer, en fonction du contexte, l'expression d'une partie de l'information génétique, et ainsi de moduler la production des protéines présentes dans la cellule.

1.2 L'expression des gènes

Les mécanismes d'expression et leur régulation font intervenir une succession d'étapes. Nous présentons dans cette section les éléments essentiels permettant d'appréhender ces mécanismes. En particulier, nous concentrons cette présentation sur les mécanismes de régulation de la transcription et leurs implications au niveau de l'ADN.

1.2.1 La synthèse des protéines

L'expression des gènes codant pour des protéines consiste en une succession de deux grandes étapes qui vont permettre de produire, à partir de l'ADN, des protéines : la transcription et la traduction. Lors de la *transcription*, une molécule intermédiaire - l'ARN messager (ARNm) - est synthétisée dans le noyau en utilisant la séquence d'ADN d'un gène comme modèle. Puis l'ARN messager subit une phase de maturation et d'épissage afin de produire un ARN mature qui pourra être traduit en protéine. Lors de cette phase, les régions non co-

dantes de l'ARN, nommées introns, sont excisées pour ne conserver que les portions codantes, appelées exons. L'ARN messager ainsi obtenu est ensuite transporté à l'extérieur du noyau pour être traduit en protéine. Lors de cette deuxième étape de *traduction*, les triplets de nucléotides de l'ARN sont traduits en acides aminés et assemblés pour former une protéine. Ces différentes étapes sont présentées dans la figure 1.2.

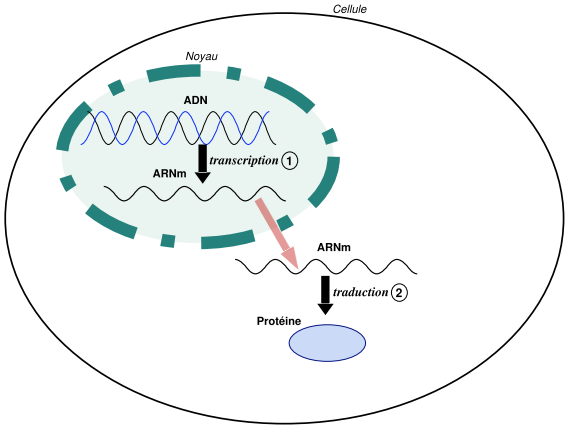


FIG. 1.2: Schéma général de l'expression des gènes chez les eucaryotes.

La transcription

La transcription peut se définir comme le mécanisme qui permet la synthèse d'une séquence d'ARN à partir de la séquence d'ADN d'un gène. Ce mécanisme fait intervenir différentes protéines - les facteurs de transcription - et un complexe protéique servant à la synthèse de l'ARN : l'ARN polymérase. Ces protéines forment, en se fixant à l'ADN, un appareil de transcription appelé appareil basal. L'appareil basal se déplace alors le long de l'ADN et synthétise la molécule d'ARN en utilisant l'ADN comme matrice.

La traduction

La traduction est la seconde étape du processus d'expression qui permet de produire une protéine à partir d'un ARNm. C'est lors de cette étape que l'ARNm est interprété, c'est-à-dire traduit, en chaîne d'acides aminés qui forment la protéine. Chaque acide aminé est décodé à partir de triplets de nucléotides présents sur la séquence d'ARN : les codons. Le code génétique définit le système de correspondance entre les codons et les acides aminés. Ce code est, à quelques exceptions près, le même chez tous les organismes.

Le mécanisme de traduction fait intervenir différents éléments : le ribosome et les ARN de transfert (ARNt). Après s'être fixé à l'extrémité de l'ARNm, le ribosome se déplace de codon en codon le long de l'ARN. Il associe à chacun des codons un ARNt qui apporte l'acide aminé correspondant. Les acides aminés sont successivement incorporés dans une chaîne par liaison peptidique afin de former la protéine finale.

Régulation de l'expression

La synthèse des protéines est contrôlée en fonction du contexte dans lequel se trouve la cellule. Cette régulation, ou modulation de l'expression, intervient à tous les niveaux de la synthèse des protéines. En particulier, chez les eucaryotes, elle peut intervenir au niveau de :

- l'activation de la structure chromatinienne ;
- l'initiation de la transcription ;
- l'étape de maturation de l'ARN ;
- l'étape de transport de l'ARN en dehors du noyau ;
- l'étape de traduction ;
- la dégradation des objets (ARN messager et protéines).

Lorsqu'un gène est actif, c'est-à-dire lorsque sa structure chromatinienne est présente sous forme non condensée, une part importante de régulation a lieu au niveau transcriptionnel et plus particulièrement lors de la phase d'initiation.

Nous concentrons cette introduction sur ce niveau de régulation pour deux raisons. D'une part, le contrôle de la traduction de l'ADN en ARN conditionne les étapes suivantes de l'expression. D'autre part, c'est sur ce mode de régulation qu'il existe actuellement le plus de connaissances, de techniques expérimentales et de données disponibles.

1.2.2 Régulation transcriptionnelle

Lors de la transcription d'un brin d'ARN à partir de l'ADN, un complexe protéique joue un rôle clé : l'ARN polymérase. Nous détaillons son fonctionnement, et son interaction avec d'autres acteurs majeurs de l'initiation de la transcription : les facteurs de transcription.

ARN polymérase

Les ARN polymérases sont des protéines de grande taille composées d'une dizaine de sous-unités. On distingue, chez les eucaryotes, trois polymérases différentes : les ARN polymérases I qui synthétisent les grands ARN ribosomiques (ARNr 28S, 18S et 5,8S), les ARN polymérases II qui synthétisent les ARN pré-messagers (ARNm) et la plupart des petits ARN nucléaires (ARNsn), et les ARN polymérases III qui synthétisent les ARN de transfert (ARNt), les ARN ribosomiques 5S et les petits ARN nucléaires.

La transcription d'un brin d'ARN est classiquement décomposée en trois phases :

- initiation : l'ARN polymérase se fixe sur l'ADN au niveau de sites spécifiques appelés promoteurs en recrutant des facteurs de transcription ;
- élongation : l'ARN polymérase se déplace le long de l'ADN matrice et synthétise l'ARN complémentaire ;
- terminaison : un signal spécifique provoque l'arrêt de la synthèse d'ARN.

Ces trois phases sont présentées dans la figure 1.3. Du fait de la machinerie transcriptionnelle, la transcription est réalisée dans un seul sens. Le brin d'ADN 3'→5' est utilisé comme matrice pour produire un ARN correspondant au brin codant orienté dans le sens 5'→3'. Si l'on considère la séquence d'ADN codante, la région promotrice d'un gène correspond à la région

située dans la partie 5' de l'ADN. Le déplacement de la polymérase se faisant dans la direction 5'→3' de la séquence codante, la région promotrice se définit comme région en amont du gène.

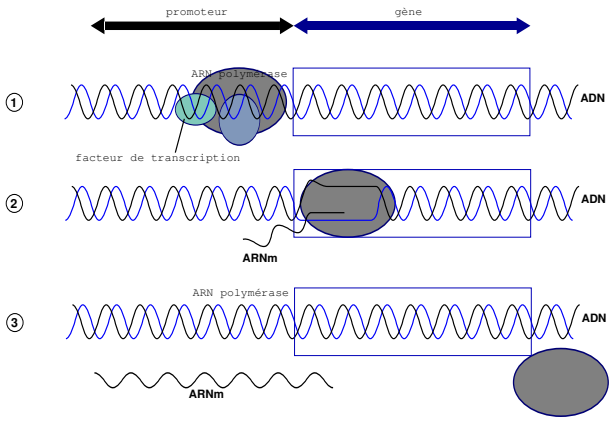


FIG. 1.3: Schéma général de la transcription de gènes chez les eucaryotes.

Facteurs de transcription

L'arrimage de l'ARN polymérase sur l'ADN nécessite la présence préalable de protéines particulières appelées *facteurs généraux de transcription*. Dans un premier temps, ces facteurs se fixent de manière séquentielle en amont du gène à transcrire dans une région appelée promoteur en formant un complexe. Ensuite ce complexe va recruter l'ARN polymérase et la positionner au niveau d'un site localisé en amont du gène, appelée site d'initiation de la transcription. La transcription peut alors commencer. En association des facteurs généraux, d'autres facteurs, les *facteurs de transcription spécifiques*, vont intervenir et influer de manière positive ou négative sur la transcription. Il convient ici de distinguer précisément les facteurs de transcription généraux ou basaux, dont la présence est requise pour initier la transcription, des facteurs spécifiques qui possèdent une action régulatrice sur l'expression propre à chaque gène.

D'un point de vue biochimique, les facteurs de transcription sont des protéines possédant des domaines de fixation à l'ADN et des domaines d'activation de la transcription. Les domaines de fixation vont leur permettre de se fixer à l'ADN sur de courtes séquences spécifiques : les sites de fixation de facteurs de transcription. Un exemple de facteur de transcription est donné dans la figure 1.4.

Dans l'assemblage ADN-protéine, on appelle également élément *trans*-régulateur le facteur de transcription, et élément *cis*-régulateur le motif nucléique reconnu. Les sites peuvent correspondre aux séquences de fixation de plusieurs facteurs ce qui dans ce cas définit un module *cis*-régulateur.

L'interaction ADN-protéine repose sur une forme de complémentarité entre la composition nucléique du site de fixation et le site actif de la protéine régulatrice [41, 53, 58]. Les sites sur lesquels se fixent les facteurs de transcription sont de courts segments d'ADN (une dizaine de

[2] *source de l'image* : `http ://web.uconn.edu/mcb201/F10-40.JPG`

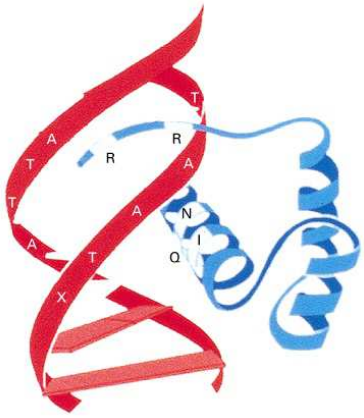


FIG. 1.4: Exemple de facteur (protéine à homéodomaine) et d'interaction à l'ADN[2].

nucléotides) qui présentent une forte variabilité au niveau de leurs séquences nucléiques. Des différences importantes peuvent être observées lorsque l'on compare les différents fragments de séquences correspondant à différents sites de fixation d'un facteur donné. Cette variabilité peut s'expliquer par différentes raisons. Tout d'abord certaines bases composant les sites de fixation n'interviennent peu ou pas dans la chimie de fixation. D'autre part, certains sites de fixation peuvent avoir, suivant leurs fonctions ou leurs implications dans la régulation, une affinité différente avec les facteurs [52].

1.2.3 Structure d'un gène

Nous avons vu qu'un gène pouvait se définir comme une portion d'ADN servant de modèle à la synthèse d'une molécule. Les différents mécanismes que nous avons décrits (initiation de la transcription, maturation de l'ARNm, traduction) se traduisent sur l'ADN par une succession de signaux qui structurent le gène. La portion d'ADN transcrite en ARN est délimitée, d'une part, par un site d'initiation de la transcription, et d'autre part, par le terminateur. Entourant le site d'initiation de la transcription se trouve le promoteur. C'est dans cette zone qu'interagissent les éléments nécessaires à l'initiation de la transcription et que l'ARN polymérase est recrutée. C'est également dans cette zone qu'intervient une large part des signaux spécifiques permettant aux facteurs de transcription de se fixer à l'ADN. Chez les eucaryotes, on distingue à l'intérieur de la région transcrite en ARN, une succession de portions d'ADN codantes (les exons) et non codantes (les introns). Ces portions non-codantes sont éliminées (excision) au cours d'une des phases d'expression (la maturation de l'ARN messager). Chaque partie codante ou exons sera décodée lors du processus de synthèse de la protéine. Le début du premier exon et la fin du dernier exon sont délimités par des signaux particuliers : les codons start et stop. Cette structure est présentée dans la figure 1.5.

[3] *source de l'image* : `http ://images.clinicaltools.com/images/gene/geneelementstext.jpg`

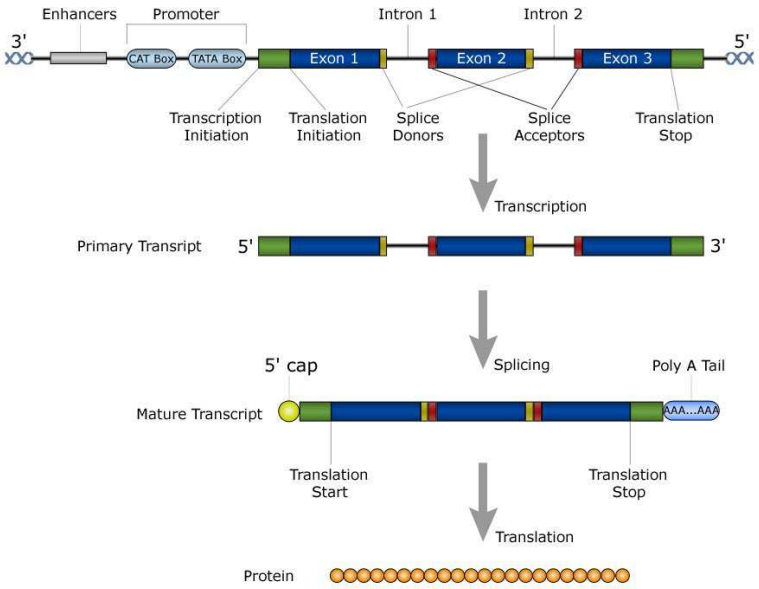


FIG. 1.5: Détails sur la structure d'un gène eucaryote[3].

1.3 Techniques expérimentales pour l'analyse de l'expression

Pour pouvoir comprendre concrètement les mécanismes intervenant dans la régulation de la production des protéines, il faut disposer de données expérimentales. Le mécanisme transcriptionnel est le mécanisme de régulation sur lequel on dispose actuellement du plus de données. Au cours des dernières décennies, de nombreuses techniques permettant de mesurer l'expression au niveau de l'interaction ADN-facteur et au niveau des ARN messagers ont été développées. Parmi ces techniques, on distingue :

- les techniques classiques, qui permettent d'analyser finement les interactions protéines-ADN sur des portions sélectionnées de l'ADN (analyse des sites de fixation des facteurs de transcription) ;
- et les techniques haut-débit, qui permettent d'analyser le niveau d'expression d'un large ensemble de gènes.

Nous détaillons dans cette section, les techniques de mesure et d'analyse de l'expression les plus courantes. En particulier, celles classiques de retard sur gel, d'empreinte à DNase et d'immuno-précipitation, et celles haut-débit de puces à ADN et de ChIP-on-chip.

1.3.1 Retard sur gel

La technique de retard sur gel (plus formellement connue sous le nom EMSA - Electrophoretic mobility shift assays) est une technique permettant d'identifier les propriétés de fixation de protéines à un fragment d'ADN [23]. Elle permet par exemple d'identifier les régions de fixation d'un facteur de fixation donné. Cette technique va consister à mesurer la vitesse

de migration d'un fragment d'ADN dans un gel soumis à un champ électrique. Elle repose sur le principe suivant : un fragment d'ADN seul migre plus vite qu'un fragment complexé à des protéines (les facteurs de transcription dans notre cas). Il est ainsi possible d'étudier l'affinité protéines-ADN en effectuant des mesures comparatives de la vitesse de migration de fragments dans différentes conditions (complexé, non complexé, ...). Cette technique est considérée comme fortement sensible pour l'étude des propriétés de fixation des facteurs sur l'ADN.

1.3.2 Empreinte à la DNase I

La technique d'empreinte à la DNase I permet d'étudier les interactions protéine-ADN en fournissant une empreinte du site de fixation de la protéine sur l'ADN [25]. Elle repose sur l'utilisation d'une nucléase (la DNase de type I) pour permettre une digestion partielle des séquences d'ADN sur lesquelles se sont fixées les protéines à étudier. C'est dans ce cas, la présence de ces protéines qui empêche la digestion par la nucléase des fragments sur lesquels elles sont fixées. Ces fragments fournissent une "empreinte" élargie (résolution d'une dizaine de bases) du site de fixation du facteur étudié. Cette technique possède le désavantage d'être moins sensible que le retard sur gel et de nécessiter une quantité plus importante de protéines. Néanmoins, elle permet d'identifier certaines interactions non détectables par d'autres techniques.

1.3.3 Immuno-précipitation de chromatine (ChIP)

La technique d'immuno-précipitation de chromatine permet d'identifier *in vivo* les fragments d'ADN fixés par les facteurs de transcription [90]. Cette technique se décompose en trois étapes. La première étape consiste en l'immobilisation, ou le pontage (par une irradiation aux Ultra-Violets ou par un traitement au formaldéhyde) des protéines fixées *in vivo* sur l'ADN. Dans un deuxième temps, l'ADN est découpé en courts fragments par sonication. Enfin, les fragments d'ADN sur lesquels se sont fixées les protéines sont isolés par affinité immuno-précipitation. On obtient alors un ensemble de fragments d'ADN correspondant aux segments où les facteurs étaient fixés *in vivo*. Ces fragments peuvent alors être clonés, puis séquencés pour analyse.

1.3.4 Puces à ADN

Les puces à ADN permettent de mesurer systématiquement le niveau d'expression d'un ensemble de gènes au sein d'une population de cellules [18]. Plus précisément, elles sont utilisées pour détecter les fragments de séquences d'ARN présents dans un échantillon donné. Pour cela un ensemble de fragments d'ADN connus, appelés sondes, est fixé sur un support par précipitation. Ensuite, par un système de marquage sur les fragments à analyser, le niveau d'hybridation de toutes les sondes, c'est-à-dire le taux de sondes qui vont s'associer aux fragments présents dans l'échantillon pour reformer une double hélice, va être mesuré. Il est ainsi possible de déduire les niveaux d'expression des gènes de l'échantillon. La figure 1.6 présente un exemple de puce à ADN composée de 40 000 sondes.

[4] *source de l'image* : `http ://upload.wikimedia.org/wikipedia/en/0/0e/Microarray2.gif`

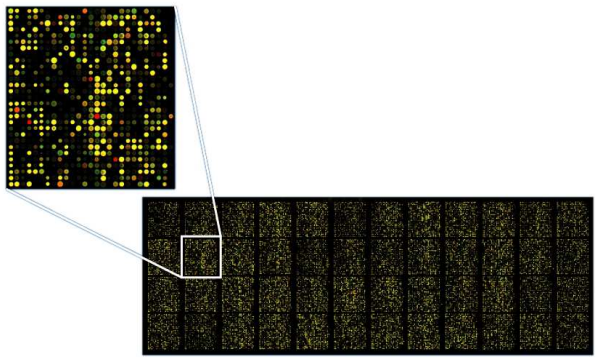


FIG. 1.6: Exemple de puce à ADN[4].

Nous présentons ici le principe historique d'une puce à ADN. Ce principe "de base" a été au fil du temps amendé de nombreuses manières pour produire une large diversité de puces dont la présentation dépasserait le cadre de ce document.

Une puce à ADN est constituée d'un petit support solide, généralement une lame de verre, sur lequel un ensemble de fragments d'ADN (les sondes) ont été chimiquement fixés en des points distincts du support. Ces fragments d'ADN, utilisés comme représentants de gènes ou de portions du génome, sont placés sur le support de manière géométrique par un système robotisé. Une puce peut contenir de quelques centaines de fragments d'ADN à plusieurs dizaines de milliers. Les différentes étapes de la réalisation d'une expérience de puce à ADN sont les suivantes :

1. extraction des ARN messagers d'une population de cellules ;
2. sélection et amplification des fragments à déposer sur la puce (oligos courts, longs ou très longs) ;
3. marquage (radioactif, fluorescence) ;
4. dépôt sur la puce ;
5. hybridation des fragments déposés avec les sondes ;
6. lecture de la puce ;
7. analyse des données.

Chaque fragment d'ADN présent sur la puce (les sondes) va permettre de fixer de manière spécifique, par hybridation, les fragments complémentaires (les cibles) présents dans l'échantillon à analyser. L'hybridation, quand elle a lieu, peut être mise en évidence par des procédés optiques ou radioactifs lorsque les cibles ont été préalablement marquées. Enfin en analysant les signaux lumineux ou radioactifs, au moyen de systèmes d'analyse d'images, il est possible d'identifier les fragments cibles présents dans l'échantillon. La quantification de ces signaux permet d'obtenir une image précise du niveau d'expression des différents fragments de gènes de l'échantillon analysé. La difficulté majeure repose alors sur l'analyse de la quantité importante de données produites par la puce.

Les puces à ADN peuvent comporter un ou deux canaux de lecture : les puces bi-canal et les puces mono-canal. Les puces bi-canal permettent de mesurer l'expression différentielle des gènes dans deux échantillons, comme un échantillon de référence et un échantillon d'expérience. Ainsi, il est possible de connaître précisément quels sont les gènes sur ou sous

exprimés dans un échantillon par rapport à l'autre. Cela permet par exemple, de comparer les différences d'expression entre des cellules malades et des cellules saines. Dans ce cas, les deux échantillons sont marqués en utilisant deux fluorochromes différents (Cy3 et Cy5) et hybridés sur la même puce.

1.3.5 ChIP-on-chip

La technique ChIP-on-chip (ChIP sur puce) propose de déterminer le spectre d'action d'une protéine *in vivo* à l'échelle génomique [13, 35, 70, 93]. Elle combine pour cela la technique d'immuno-précipitation de chromatine et de puce à ADN. Il est ainsi possible d'étudier l'ensemble des sites sur lesquels un facteur de transcription donné se fixe *in vivo*. Le principe de fonctionnement peut se décomposer en différentes étapes. Dans un premier temps, les protéines fixées *in vivo* sur l'ADN sont immobilisées (par formaldéhyde). L'ADN est alors extrait et découpé en fragments par sonication. Les fragments sur lesquels les protéines sont fixées sont isolés par immuno-précipitation. Ensuite, les protéines sont détachées et les fragments d'ADN sont marqués avant d'être hybridés sur la puce.

Chapitre 2

Analyse bio-informatique des motifs régulateurs

Sommaire

Nous avons vu dans le chapitre précédent le rôle important des facteurs de transcription dans la régulation transcriptionnelle. La compréhension du contrôle exercé par ces facteurs dans la régulation d'un gène passe par l'identification des sites d'interaction avec l'ADN. Différentes techniques expérimentales permettant de répondre à cette question sont actuellement disponibles (section 1.3). Mais leur utilisation reste contraignante et coûteuse. Les puces à ADN nécessitent par exemple de disposer d'une annotation des gènes pour pouvoir rechercher les facteurs communs à un ensemble de gènes co-régulés. Les informations fournies par ces techniques ouvrent toutefois la voie à un travail de modélisation des sites de fixation de facteurs de transcription.

Dans le cadre de la recherche d'éléments régulateurs, une des questions habituellement posée est la suivante : étant donné un ensemble de gènes, quels sont les facteurs susceptibles d'être impliqués dans la régulation de ces gènes, et quels sont leurs sites de fixation associés ? Dans ce contexte, l'analyse bio-informatique des motifs régulateurs apportent des méthodes prédictives pour aider à la découverte et à la localisation d'éléments *cis*-régulateurs. Cette recherche nécessite de disposer de modèles pour le texte (les séquences génomiques) d'une part et des modèles pour les motifs (les sites de fixation) d'autre part.

Dans ce chapitre, nous commençons par présenter comment modéliser les séquences génomiques à l'aide de modèles Markoviens (section 2.1) et comment modéliser les motifs à l'aide de matrices et de consensus (section 2.2). Nous montrons ensuite comment rechercher des sites de fixation en utilisant ces modèles (section 2.3). En section 2.4, nous discutons de la qualité des prédictions et des limites de la recherche "brute". Nous exposons alors deux moyens pour améliorer la pertinence des prédictions : la génomique comparative en section 2.5 et la recherche de motifs sur-représentés dans un ensemble de séquences co-exprimées (section 2.6).

2.1 Modélisation des séquences génomiques

Lorsque l'on s'intéresse aux signaux de régulation, on cherche à extraire des motifs se détachant du contexte génomique. Dans ce cadre, une des questions à aborder est la suivante : comment comparer une observation à ce qui peut être attendu par hasard étant donné le contexte ? Pour pouvoir répondre à cette question il est nécessaire de modéliser le contexte, c'est-à-dire le comportement attendu de la séquence génomique étudiée. Un modèle adapté au problème de recherche de mots exceptionnels doit par exemple permettre de pouvoir estimer simplement la probabilité d'occurrence d'un mot ou d'un motif approché ainsi que sa loi de comptage attendue. Il sert ainsi de contexte de référence lorsque l'on cherche à évaluer la qualité d'une observation exceptionnelle. Nous présentons dans cette section les modèles homogènes probabilistes, basés sur la composition en mots, les plus couramment employés pour modéliser les séquences nucléiques : les modèles de Bernoulli et les modèles de Markov. Ces modèles possèdent l'avantage d'être bien établis et simples à manipuler.

À partir de maintenant, nous considérons qu'une séquence d'ADN est un texte écrit sur un alphabet à quatre lettres : $\Sigma = \{A, C, G, T\}$. La question posée par la modélisation est alors : comment définir une séquence ϕ de variables aléatoires à valeurs dans Σ qui ait un comportement proche d'une séquence réelle vis à vis des motifs étudiés ?

2.1.1 Chaînes de Bernoulli

Une chaîne de Bernoulli est une séquence de variables aléatoires indépendantes (sans mémoire) $X_1, X_2, ..., X_L$. En considérant que les variables X_i prennent leurs valeurs dans l'ensemble fini $\sum = \{A, C, G, T\}$, une chaîne de Bernoulli peut se définir par un vecteur correspondant aux probabilités d'apparition de chaque valeur $[P(A), P(C), P(G), P(T)]$. Un modèle de Bernoulli correspond donc simplement à ne retenir pour une séquence que sa composition en nucléotides. La fréquence d'un mot $u_1, ..., u_l$ se calcule simplement comme le produit des probabilités d'apparition de chaque lettre u_i du mot dans le modèle :

$$P(u) = \prod_{i=1}^{l} P(u_i)$$

Le principal intérêt de ce modèle repose sur sa simplicité d'utilisation. Néanmoins il semble peu adapté à la modélisation des séquences génomiques réelles qui comportent un ordre de conservation plus important (codons dans les parties codantes...).

2.1.2 Chaînes de Markov

Une chaîne de Markov est une séquence de variables aléatoires avec mémoire finie. Plus formellement, une séquence Markovienne de variables aléatoires X_i vérifie la propriété suivante :

$$P(X_n = u_n | X_{n-1} = u_{n-1}, ..., X_1 = u_1) = P(X_n = u_n | X_{n-1} = u_{n-1})$$

Cela signifie que la connaissance de la valeur de la variable X_{n-1} est à elle seule suffisante pour connaître la valeur de la variable X_n. Si l'on considère que les variables X_i prennent leurs valeurs dans l'ensemble fini $\Sigma = \{A, C, G, T\}$, une séquence nucléique $s = x_1, ..., x_L$ peut être modélisée par un modèle Markovien [69] que l'on notera MM. On peut également définir des modèles de Markov avec une mémoire d'ordre k. Dans ce cas, il est nécessaire de connaître les valeurs des variables X_{n-k}, ..., X_{n-1} pour connaître la valeur de la variable X_n. On note alors par MM_k un modèle Markovien d'ordre ou de mémoire k. Il est possible par un changement d'alphabet de réduire un modèle d'ordre k sur Σ à un modèle d'ordre 1. Dans ce cas, l'élément $v_i, ..., v_{i+k-1}$ présent à la position i ne dépend que de celui présent à la position $i-1$ $(v_{i-1}, ..., v_{i+k-2})$.

Un modèle Markovien d'ordre k correspond à la définition de toutes les probabilités de transition $T_{uv} = P(v|u)$ où u et v sont des mots de longueur k. Ces transitions sont habituellement définies par une matrice de transition T. Une chaîne de Bernoulli peut être vue comme une chaîne de Markov particulière sans mémoire (d'ordre 0). Dans ce cas, toutes les lignes de la matrice de transition sont identiques (une ligne est donnée par le vecteur de probabilité de la séquence de Bernoulli).

Pour une séquence donnée, la matrice de transition peut se construire en considérant les transitions qui maximisent la probabilité de la séquence pour la matrice. Cela s'obtient simplement à partir de la fréquence d'observation des mots de longueur $k+1$ présents dans la séquence. Un exemple de matrice de transition d'ordre 1 construite à partir de la séquence

`ACTATAGGACTTAGCCTT` est donné ci-dessous :

$$T = \begin{pmatrix} 0 & 0.4 & 0.4 & 0.2 \\ 0 & 0.25 & 0 & 0.75 \\ 0.33 & 0.33 & 0.33 & 0 \\ 0.6 & 0 & 0 & 0.4 \end{pmatrix}$$

où $T_{0,0}$ correspond à la transition $T_{A \to A} = 0$, $T_{0,1}$ à la transition $T_{A \to C} = 0.4$, etc.

Pour pouvoir évaluer la probabilité d'un mot $u_1, ..., u_k, ..., u_n$ dans un modèle de Markov d'ordre k, il faut commencer par déterminer la probabilité du préfixe $u_1, ..., u_k$. Cette probabilité peut être obtenue en utilisant le vecteur stationnaire S défini par $S = S \times T$ pour former la loi initiale (ce qui revient à $P(X_i = u_k) \to V(u_k)$ lorsque $i \to \infty$). Étant donné le vecteur stationnaire et la matrice de transition, la probabilité du mot $u_1, ..., u_k, ..., u_n$ se calcule de la manière suivante :

$$P(w) = S(u_1, ..., u_k) \times T(u_1, ..., u_k, u_2, ..., u_{k+1}) \times ... \times T(u_{l-k-1}, ..., u_{l-1}, u_{l-k}, ..., u_l)$$

Une des difficultés lorsque l'on modélise une séquence par un modèle Markovien concerne le choix de l'ordre du modèle à utiliser. L'ordre doit, d'une part, être suffisamment élevé pour modéliser le comportement étudié de la séquence, et d'autre part, rester raisonnable pour pouvoir être construit de manière fiable sans sur-adaptation. En effet, pour pouvoir construire un modèle d'ordre k, il faut pouvoir disposer d'une séquence de taille suffisante. Le nombre de paramètres à estimer est de $3 \times \Sigma^k$ (pour chaque ligne de la matrice, trois fréquences sont mesurées, la quatrième est déduite du fait de $\Sigma P = 1$). Idéalement il faut disposer de données de taille 1 000 fois plus grande que le nombre de paramètres à estimer. Il faut par exemple disposer idéalement d'une séquence d'environ 10 000 bases pour établir un modèle d'ordre 1 et de 200 000 bases pour établir un modèle d'ordre 3.

Thijs et co-auteurs [85] ont évalué l'influence de la modélisation des séquences sur la qualité de prédiction d'éléments *cis*-régulateurs. Ils ont mesuré les variations du comportement de leur méthode de découverte de motifs lorsque l'ordre de la chaîne de Markov utilisée pour modéliser les séquences augmentait. Pour cela, ils ont mesuré sur différents jeux de données réels (promoteurs avec des boîtes GC) et simulé l'écart entre le nombre de motifs prédits et le nombre de motifs effectivement présents dans les jeux. Ils ont ainsi pu mettre en évidence le gain apporté par l'utilisation d'un modèle d'ordre suffisamment élevé pour modéliser le comportement des séquences génomiques. Plus particulièrement, ce sont des modèles d'ordre 3 ou 4 qui leur ont permis d'obtenir les résultats les plus satisfaisants. Ils ont également noté la dégradation des performances lorsque des modèles d'ordre trop élevé (ordre 5) étaient employés, ce qui se justifie essentiellement par une quantité insuffisante de données disponibles pour établir un modèle de cet ordre.

2.2 Modélisation des sites de fixation de facteurs de transcription

Après avoir modélisé le contexte, nous présentons les modèles pour les éléments *cis*-régulateurs. Il existe de nombreuses façons de modéliser les courtes séquences correspondant

aux sites de fixation de facteurs de transcription. Ces modélisations doivent permettre de prendre en compte la variabilité des motifs reconnus par un facteur de transcription donné. On citera les représentations courantes suivantes : consensus, expression régulière, modèle de Markov caché, matrices de comptage.

Afin de fixer plus précisément les idées, différentes séquences correspondant aux sites reconnus par le facteurs p65 sont présentées dans la figure 2.1. Ces séquences sont issues de la base de données JASPAR [72] et correspondent à 18 sites observés chez l'humain.

```
       cGGGAATTTCCaacag
        GGGAATTTCCccggcct
       cGGGAATTTCCgcttc
        TGGAATTTCCggcgcc
gccggggTGGAATTTCC
     gcCGGAATTTCCggtg
 gcacccGGGGATTTCC
 atgatcGGGACTTTCC
  aagatCGGAGTTTCC
       GGGAATTCCCgcatcca
tgcttgaCGGACTTTCC
 ccgattCGGACTTTCC
   tcgcGGGGAATTCCgg
tacggccTGGGGTTTCC
 aggtgtGGGGATTCCC
  ggagtGGGGTTTTCCc
 gcacccGGGGGATTCC
 ctgccgGTGGGTTTCCc

       GGGRATTTCC
```

Fig. 2.1: Exemple de sites et consensus pour le facteur de transcription p65.

Le point de départ à la construction d'un modèle est la famille de séquences correspondant aux différentes occurrences référencées du site. Généralement, c'est à partir de l'alignement multiple des différentes séquences que le modèle est construit (par exemple l'alignement présenté dans la figure 2.1). Nous détaillons successivement différentes représentations se basant sur cet alignement, et discutons des avantages et contraintes imposés par ces représentations.

2.2.1 Représentation consensus

Une des représentations les plus couramment employées pour modéliser un motif est la représentation consensus dégénérée. Cette représentation utilise un alphabet étendu, appelé code IUPAC (tableau 2.1), qui permet de prendre en compte une certaine forme de variabilité dans les séquences en considérant qu'une position peut être décrite par un seul, deux, trois ou quatre nucléotides différents. La construction du consensus peut se faire de la manière suivante [22] : pour chaque position de l'alignement (sans gap), si un des nucléotides est présent dans une proportion au moins égale à 60%, alors ce nucléotide est utilisé pour modéliser la colonne.

Dans le cas contraire, si deux nucléotides sont présents dans des proportions au moins égales à 35%, alors c'est le code correspondant au code dégénéré du couple qui est utilisé. De la même manière, un code dégénéré pour un triplet est utilisé lorsque chaque élément du triplet est présent dans une proportion supérieure à 20%. Enfin pour représenter une position non conservée, la lettre N est utilisée. Par exemple, il est possible de construire le consensus : `GGGRATTTCC` à partir des séquences présentées dans la figure 2.1.

La représentation consensus d'un motif présente l'intérêt d'être simple à manipuler et à mettre en œuvre. Elle permet par exemple une comparaison visuelle rapide de différentes familles de sites. Néanmoins cette représentation engendre une perte importante d'information vis-à-vis de l'alignement dont elle est issue. Le biais de composition pour les différents nucléotides possibles à une position donnée de l'alignement n'est, dans ce cas, pas pris en compte.

Code	Description
A	Adenine
C	Cytosine
G	Guanine
T	Thymine
U	Uracil
R	Purine (A or G)
Y	Pyrimidine (C, T, or U)
M	C or A
K	T, U, or G
W	T, U, or A
S	C or G
B	C, T, U, or G (not A)
D	A, T, U, or G (not C)
H	A, T, U, or C (not G)
V	A, C, or G (not T, not U)
N	Any base (A, C, G, T, or U)

Tab. 2.1: Code IUPAC.

2.2.2 Représentation matricielle

Une autre représentation habituellement utilisée pour modéliser la variabilité de chaque position du site de fixation est la *matrice de comptage*. En utilisant l'alignement multiple des différents sites, il est aisé de construire la matrice de comptage : celle-ci correspond simplement, pour chaque position de l'alignement, au nombre de nucléotides de chaque type rencontrés. Un exemple de matrice de comptage pour les sites du facteur p65 est donné dans la figure 2.2.

La représentation matricielle est une représentation basée sur l'alignement, à la fois simple et plus générale que les motifs consensus. Une première limitation de ce type de représentation est l'hypothèse d'indépendance entre colonnes sur laquelle elle repose. Il est toutefois possible d'étendre cette représentation afin de prendre en compte les nucléotides non plus de manière individuelle mais par couple (di-nucléotides) [26].

A	0	0	0	11	10	2	0	0	0	0
C	4	0	0	0	3	0	0	2	18	18
G	11	17	18	7	4	0	0	0	0	0
T	3	1	0	0	1	16	18	16	0	0

FIG. 2.2: Exemple de matrice de comptage pour le facteur p65.

Cependant, Benos et co-auteurs [5, 6] ont montré que le fait de considérer une contribution indépendante de chaque position du motif constituait une bonne approximation de la nature des interactions ADN-protéines.

Une autre limitation majeure de la représentation matricielle concerne la rigidité qu'elle impose aux sites pouvant être reconnus. Elle interdit par exemple des motifs en plusieurs parties comportant un espacement variable. Ceci peut être inadapté à la détection de sites de fixation pour certaines familles de facteurs. Une autre solution pour modéliser les sites est par exemple d'utiliser un modèle de Markov caché (HMM) [20]. Mais dans ce cas, on se heurte souvent au manque de données appropriées pour construire le modèle.

2.2.3 Base de données de matrices

De larges banques de modèles de sites de fixation représentés par des matrices sont disponibles. La banque de données spécialisée PlantCARE [45] référence des sites et modèle de sites pour les plantes. Nous détaillons ci-après les deux banques générales les plus employées : TRANSFAC [94] et JASPAR [72].

TRANSFAC

TRANSFAC [94] est une base de données d'éléments eucaryotes *cis*-régulateurs et *trans*-régulateurs. La plupart des données présentes dans TRANSFAC sont extraites d'une compilation bibliographique. La compilation de données a débuté en 1998, pour être ensuite informatisée en 2000. Actuellement deux versions sont disponibles : la version publique (TRANSFAC 7.0) et la version payante proposée par la société BIOBASE. La base TRANSFAC est constituée d'un ensemble de données sur les sites de fixation, les gènes, les facteurs de transcription et les matrices. La version publique de TRANSFAC contient 398 matrices de comptage construites principalement à partir d'observations *in vitro*.

JASPAR

La base de données JASPAR [72] est une base de sites de fixation de facteurs de transcription ouverte. Cette base, créée en 2004, est portée par deux laboratoires : Center for Molecular Medecine and Therapeutics (University of British Columbia) et le Karolinska Institutet. Les sites y sont modélisés par des matrices. Deux points importants distinguent cette base des autres : d'une part les données présentes sont non redondantes, et d'autre part du fait de sa nature "open source" les données peuvent être utilisées sans aucune restriction. La base JASPAR est constituée des trois sous-unités suivantes :

- JASPAR CORE qui contient un ensemble non redondant de matrices (123) construit à partir d'un ensemble de sites de fixation eucaryotes expérimentalement vérifiés.
- JASPAR FAM qui contient un ensemble de méta-modèles décrivant les propriétés partagées par les sites de fixation.
- JASPAR PHYLOFACTS est une sous-base qui contient un ensemble de matrices (174) construits à partir de séquences phylogénétiquement conservées.

Dans la suite de ce document le terme JASPAR fera référence à l'unité JASPAR CORE.

2.2.4 Visualisation : Sequence Logos

Les matrices de comptage s'accompagnent d'outils pour leur visualisation. Schneider et co-auteurs [75] ont défini la conservation pour une position comme la différence entre l'entropie maximale et l'entropie observée. Cette conservation R_i se définit de la manière suivante :

$$R_i = 2 - \left(\sum_{b \in \{A,C,G,T\}} -f_b \log_2 f_b \right)$$

où f_b représente la fréquence observée du symbole b à la position i. Il est possible de définir une représentation graphique basée sur le contenu informationnel de chaque position. La représentation graphique proposée par Schneider et Stephens [74] repose, pour chaque colonne de la matrice, sur l'empilement de lettres dont la hauteur totale correspond au contenu informationnel et où la hauteur de chaque lettre dépend de la proportion du nucléotide considéré à cette position. Un exemple de ce type de représentation obtenu avec le logiciel WebLogo [17] pour la matrice p65 (figure 2.2) est donné ci-dessous (figure 2.3).

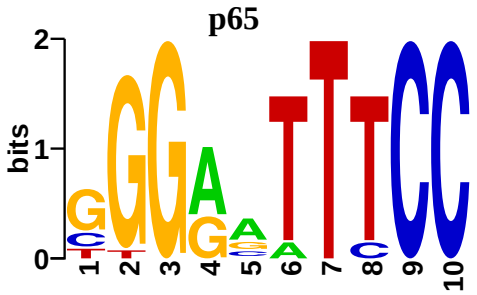


FIG. 2.3: Représentation Sequence Logos de la matrice JASPAR p65. Ce logo indique la conservation du motif à chaque position.

2.3 De la matrice de comptage à la recherche de motifs

La recherche de sites potentiels à l'aide de matrices de comptage repose sur l'idée suivante : l'affinité entre un site nucléique potentiel et le facteur de transcription peut être mesurée à l'aide d'un score. Ce score représente alors le degré d'interaction que peut avoir une séquence d'ADN avec la protéine régulatrice. Il doit permettre de distinguer un vrai site d'un fragment

non actif. Par ce moyen, il est possible de caractériser une certaine part du processus de reconnaissance des protéines régulatrices pour le motif *cis*-régulateur.

Habituellement, pour rendre plus efficace et simplifier la recherche d'occurrences, la matrice de comptage est transformée en matrice de poids. La matrice de poids se définit comme une matrice dont les éléments sont les poids à utiliser pour calculer le score d'un mot u et ainsi mesurer le degré d'affinité de ce mot avec le motif décrit par la matrice M. Nous présentons deux définitions permettant de construire une matrice de poids à partir d'une matrice de comptage : l'approche log-odd score [19] et la matrice entropie [66]. Ces deux constructions reposent sur l'hypothèse d'additivité du score, c'est-à-dire l'hypothèse selon laquelle chaque position du motif contribue de manière indépendante au score total.

2.3.1 Matrices entropie

La première définition que nous présentons, proposée par Quandt et co-auteurs [66], est basée sur l'entropie. La construction de la matrice repose sur l'idée suivante : pour une position donnée, c'est-à-dire pour une colonne de la matrice de comptage, les éléments sont pondérés par une valeur C_i, proportionnelle à l'entropie de la position. Pour chaque position i cette pondération se calcule de la manière suivante :

$$C_i = \frac{100}{\log 5} \left[\sum_{b \in \{A,C,G,T,gap\}} P_{b,i} \log P_{b,i} + \log 5 \right]$$

où $P_{b,i}$ représente la probabilité d'observer la lettre b à la position i. Cette probabilité se déduit simplement de la matrice de comptage, en considérant les fréquences des éléments colonne par colonne. Dans ce calcul, le terme $\log 5$ est utilisé pour rendre l'entropie positive et le facteur $\frac{100}{\log 5}$ pour normaliser les poids entre 0 et 100. En utilisant cette pondération C_i, la matrice de poids se construit de la manière suivante :

$$W_{b,i} = \frac{C_i \times M_{b,i}}{\sum_{i=1}^{m} C_i \times \max_{b \in \{A,C,G,T\}} M_{b,i}}$$

où le poids d'un élément est normalisé par la somme des poids maximums de chacune des colonnes.

2.3.2 Matrices log-odd

Une autre approche pour construire la matrice de poids est celle basée sur le log-odd score [30, 80, 82]. Dans ce cas, le poids d'un élément est déterminé par le logarithme du ratio de la fréquence du nucléotide dans la matrice par rapport à sa fréquence dans la séquence génomique. Afin d'éviter la sur-adaptation de la matrice, un pseudo-poids β est utilisé pour compenser le poids d'un élément par sa fréquence génomique. Si l'on note n le nombre de séquences, la construction de la matrice log-odd s'écrit comme suit :

$$W_{b,i} = \log \frac{(P_{b,i} + \beta f_b)/(n + \beta)}{f_b}$$

où f_b est la fréquence attendue pour la lettre b. Un exemple de construction de type de matrice est donné dans la figure 2.4.

Cette formulation du score peut être reliée à l'énergie de liaison ADN-protéine [6, 7, 32]. Elle présente donc l'avantage de reposer à la fois sur un modèle bien établi de la théorie de l'information et sur un modèle physique définissant l'énergie de liaison avec l'ADN.

A	0	0	0	11	10	2	0	0	0	0
C	4	0	0	0	3	0	0	2	18	18
G	11	17	18	7	4	0	0	0	0	0
T	3	1	0	0	1	16	18	16	0	0

$$\downarrow \quad W_{b,i} = \log \frac{(P_{b,i}+0.25)/(n+1)}{0.25}$$

A	-2.94	-2.94	-2.94	0.86	0.77	-0.75	-2.94	-2.94	-2.94	-2.94
C	-0.11	-2.94	-2.94	-2.94	-0.38	-2.94	-2.94	-0.75	1.35	1.35
G	0.86	1.29	1.35	0.42	-0.11	-2.94	-2.94	-2.94	-2.94	-2.94
T	-0.38	-1.35	-2.94	-2.94	-1.35	1.23	1.35	1.23	-2.94	-2.94

FIG. 2.4: Construction d'une matrice de poids à partir d'une matrice de comptage.

2.3.3 Problème de recherche de motifs spécifiques par une matrice

Lorsque l'on dispose d'une matrice de poids W, le score d'un mot u se calcule simplement en sommant les poids associés aux lettres du mot. Ce qui s'exprime de la manière suivante :

$$S(u, W) = \sum_{i=1}^{w} W_{u_i,i}$$

où w représente la longueur de la matrice W.

L'algorithme de recherche des occurrences peut alors se définir de la manière suivante :
Entrée : une séquence ϕ, une matrice de poids W, un seuil τ.
Sortie : l'ensemble des positions i de ϕ pour lesquelles le score $S(\phi_i, ..., \phi_{i+m-1}, W)$ est supérieur au seuil τ.
Dans ce cadre, l'algorithme de recherche "naïf" peut s'exprimer de la manière suivante : calculer pour chaque position de la séquence d'entrée S, le score de la matrice à l'aide de la fonction de score. Toutes les positions pour lesquelles le score est supérieur à un seuil fixé au préalable, sont considérées comme des occurrences de la matrice dans la séquence. Les logiciels Patser[31] et MatInspector [66] implémentent par exemple cette méthode.

Patser
Le logiciel Patser[31] est un logiciel basé sur le log-odd score développé par Hertz et Stormo qui permet de rechercher les occurrences potentielles de sites dans un ensemble de séquences. Il accepte en entrée soit une matrice de poids soit une matrice de comptage. Dans ce dernier cas la matrice de comptage est convertie en matrice de poids log-odd.

MatInspector
Le logiciel MatInspector [14, 66] est un logiciel pour l'identification de sites de fixation utilisant

une large bibliothèque de matrices (TRANSFAC). Il introduit différents concepts : un sous-score basé sur le cœur de matrice et un système de seuil optimisé. De nombreux programmes utilisent la stratégie développée dans MatInspector pour effectuer leur phase de recherche de sites.

Plus récemment, pour répondre au problème de la recherche de sites potentiels à plus large échelle, différentes améliorations ont été proposées. En particulier, des stratégies basées sur la construction d'un index du texte [4] ou d'un index des motifs [48] ont été développées.

2.4 Qualité des prédictions

Une fois un site potentiel prédit, il faut se poser la question de la significativité statistique de cette prédiction, c'est-à-dire évaluer quelle est la chance d'obtenir une occurrence de ce type, par hasard. Nous définissons dans ce cas la P-valeur d'une occurrence comme suit : étant donnée une occurrence de score s, quelle est la probabilité d'obtenir par hasard une occurrence de score supérieur ou égal à s. En notant $P(z)$ la probabilité d'observer une occurrence de score z, la P-valeur peut se calculer de la manière suivante :

$$P(X \geq s) = \int_{z=s}^{\infty} P(z)dz$$

Pour pouvoir répondre à cette question, il faut établir une distribution statistique du score. Dans cette section, nous présentons comment déterminer la distribution du score d'un site potentiel. Nous explorons également différents moyens pour prendre en compte le problème des tests multiples dans la mesure de la qualité d'une prédiction et discutons aussi de la pertinence biologique d'une prédiction. Pour établir la distribution statistique du score d'une matrice, plusieurs options sont possibles :

- approximation de la distribution par une loi de probabilité classique ;
- construction par calcul de la distribution exacte du score pour une modélisation donnée des séquences ;
- construction de manière empirique de la distribution du score à partir d'un grand nombre d'observations.

Nous allons successivement détailler chacune de ces méthodes.

2.4.1 Approximation de la distribution du score par une loi analytique

Une des façons les plus simples d'approximer la distribution du score est d'utiliser une loi analytique connue pour laquelle on estime les paramètres. Si l'on considère une séquence génomique modélisée par un modèle de Bernoulli $[P(A), P(C), P(G), P(T)]$, l'espérance et la variance de la distribution du score se calculent simplement de la manière suivante :

$$\mu = \sum_{i=1}^{w} \sum_{b \in \{A,C,G,T\}} P(b) \times W_{b,i}$$

$$\sigma^2 = \left(\sum_{i=1}^{w} \sum_{b \in \{A,C,G,T\}} P(b) \times W_{b,i}^2 \right) - \mu^2$$

Lorsque l'on recherche des sites avec des matrices comme modèles, seuls les scores élevés ($\geq \tau$) sont pris en compte. Il convient de définir une statistique adaptée aux valeurs extrêmes. Il est possible d'utiliser une distribution de Pareto généralisée [62], lorsque l'on cherche à approximer la queue d'une distribution dans le cadre de la théorie des valeurs extrêmes. Cette approximation est l'une des plus couramment employées pour décrire des données générées par un système qui ne conserve que les valeurs supérieures à un seuil de coupure τ. Dans ce cas, pour modéliser le comportement à décroissance exponentielle de la queue de distribution, la distribution de Pareto suivante peut être utilisée :

$$P(X) = \frac{1}{\sigma} e^{-\frac{X-\tau}{\sigma}}$$

où σ est le paramètre de la distribution et τ le seuil de coupure sur le score.

Pour fixer les idées sur le profil d'une distribution de score, quatre distributions sont données dans la figure 2.5. Elles correspondent aux distributions observées des scores au-dessus du seuil de coupure τ pour quatre matrices fortement différentes, issues de TRANSFAC. Le score employé ici est un log-odd score tel que défini précédemment (section 2.3.2). On constate que pour les matrices courtes ou strictes (`STAT5A_03` et `HMGIY_Q6`) les distributions sont fortement morcelées. En effet, le nombre de scores différents produits par ces matrices est relativement réduit (très inférieur à 4^m).

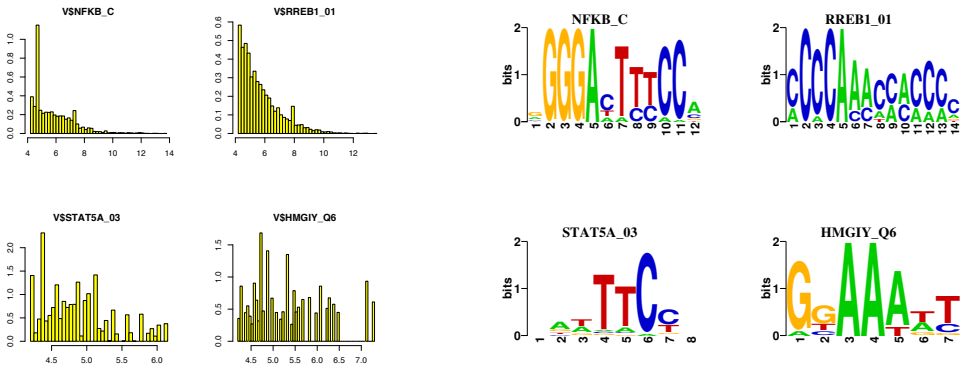


FIG. 2.5: Exemples de distributions observées pour le score des matrices `NFKB_C`, `RREB1_01`, `STAT5A_03`, `HMGIY_Q6`

L'approximation de distribution du score par une loi analytique, bien que simple à mettre en œuvre, semble peu adaptée lorsque les matrices sont relativement courtes et le seuil τ élevé (matrice `HMGIY_Q6` de la figure 2.5). Dans ce cas, d'autres stratégies doivent être employées.

2.4.2 Distribution exacte

Une autre approche pour obtenir la distribution du score est de construire cette distribution de manière exacte, en se donnant un modèle de génération pour les séquences. Claverie et Audic [16] ont proposé un calcul par programmation dynamique de la distribution du score lorsque les séquences sont modélisées par un modèle de Bernoulli. Plus récemment, Huang et

co-auteurs [36] ont proposé un moyen de calculer cette distribution lorsque le modèle était Markovien. Nous présentons ici, une version de l'algorithme de programmation dynamique de Claverie généralisée aux modèles Markoviens (le modèle de Bernoulli étant considéré comme un modèle Markovien de taille nulle).

Discrétisation du score
Une des premières contraintes lors de la construction de la distribution discrète du score est le choix du niveau de discrétisation du score. Cette discrétisation doit permettre d'avoir un niveau de résolution élevé pour les hautes valeurs de score, tout en réduisant le nombre de valeurs à considérer, afin de rendre la distribution facile à manipuler et à stocker. Un des moyens les plus commodes pour discrétiser le score est de définir le nombre de valeurs possibles a priori. On pourra par exemple définir le score discret à partir du niveau de discrétisation ϵ de la manière suivante :

$$S_{discret} = round\left(\frac{\epsilon S}{S_{max} - S_{min}}\right)$$

où ϵ, le paramètre de discrétisation, représente le nombre de scores possibles.

Algorithme
Étant donné un modèle de Markov d'ordre k, défini par sa matrice de transition T et son vecteur stationnaire S, et une matrice de poids W, l'algorithme de programmation dynamique calcule la probabilité des scores discrets associés à W. Pour cela, une table des probabilités, associant la probabilité du mot u de score s, est construite dynamiquement. La construction de cette table se fait de manière itérative sur la longueur de la matrice et sur l'ordre total des mots u de taille k. Nous désignons par i la position dans la matrice W, et par u l'ordre du mot dans $[0..4^k - 1]$ (par exemple pour les mots de longueur 2, AA=0, AC=1, ...). En posant $j = u/4 - (u \div 4)$, $u' = 4\left(4^k - (u \div 4^{k-1})\right) + j$ et $s' = s + W_{j,i}$, la formule de récurrence s'écrit de la manière suivante :

$$P(s', u') = \begin{cases} P(s', u') + P(s, u) \times T_{u,j} & \\ S_{u'} & \text{si } s' = 0 \\ 0 & \text{si } u' = 0 \end{cases} \tag{2.1}$$

Une fois la table construite, la probabilité du score s s'obtient simplement en sommant tous les éléments de la table associés au score s :

$$P(s) = \sum_x P(s, x)$$

Implémentation
Nous présentons maintenant un exemple d'implémentation pour cet algorithme. Pour construire la table des fréquences P avec un modèle de Markov d'ordre k il est nécessaire de posséder une mémoire de taille $\alpha \times 4^k$ où α représente le nombre de valeurs discrètes du score. Il est possible d'utiliser comme structure mémoire intermédiaire, une table à deux entrées $P_a[u][s]$ où la clé u correspond à l'indice d'un mot de longueur k sur l'ordre total des mots et s une valeur du score intermédiaire à la position i de la matrice. A chaque pas de l'itération, il est nécessaire de posséder deux structures mémoire : celle du pas courant i notée

P_b et celle du pas précédent notée P_a. La construction de la distribution peut se décomposer de la manière suivante :

- une phase d'initialisation. Lors de cette phase la structure mémoire P_a est initialisée à l'aide du vecteur stationnaire S pour tous les mots de taille k et les scores nuls ($P_a[u][0] = S[u]$) ;
- une phase de construction itérative de la table mémoire P. Lors de chaque itération i la structure P_b est reconstruite à partir de P_a et de la matrice de transition T en balayant tous les scores et tous les mots de taille k (mémoire Markovienne) ;
- une phase de finalisation, lors de laquelle la table des fréquences est construite à partir de la structure mémoire. Chaque élément de la table des fréquences est obtenu par sommation des probabilités de tous les suffixes de taille k ayant produit ce score.

Algorithme 1 : Algorithme pour le calcul de la distribution du score d'une occurrence dans une chaîne de Markov

Entrées : Vecteur stationnaire S, matrice de transition T, ordre du modèle k, matrice de poids W
Sorties : Distribution du score P
// Initialiser le tableau des scores pour les mots de taille k
$P_a[*][0] = 0$
pour u *dans* 4^k **faire**
  $P_a[u][0] = S[u]$
pour i *dans* $1..l$ **faire**
  $P_a = [][]$
  pour j *dans* $0..3$ **faire**
    pour u *dans* 4^k **faire**
      pour t *dans clés* $P_a[u]$ **faire**
        // Calcul du score en valeur entière
        $s = t + W[j, i]$
        // indice du mot u' = suffixe de u concaténé avec lettre j
        $u' = 4\left(4^k - (u \div 4^{k-1})\right) + j$
        $P_b[u'][s] = P_b[u'][s] + P_a[u][t] \times T[u, j]$
  $P_a = P_b$
// Finalisation
pour s *dans Scores* **faire**
  $P[s] = \sum_u (P_a[][s])$
retourner P

2.4.3 Distribution empirique

Dans certains cas, lorsque la séquence génomique ne peut être correctement modélisée par un modèle Markovien, il est possible de construire de manière empirique la distribution du score. La distribution est alors construite de manière triviale en regroupant les scores discrets de tous les mots de taille m présents dans un large ensemble de séquences. Les fréquences

pour chacun des scores en sont ensuite déduites. Une des limitations les plus critiques de cette méthode est la taille des données à utiliser et de ce fait le temps de calcul nécessaire pour établir une distribution avec une résolution suffisante. Malgré ces limitations cette méthode présente l'avantage de fournir une distribution adaptée aux séquences étudiées sans faire d'hypothèse sur la façon de les modéliser. Dans la pratique, on utilisera une large portion du génome pour établir la distribution.

2.4.4 Tests multiples

La P-valeur permet d'estimer la probabilité de faire une prédiction donnée. Lorsque l'on réalise un ensemble de prédictions, des tests multiples, la P-valeur est calculée pour chacune de ces prédictions. Dans ce cas, la probabilité d'obtenir par hasard une prédiction avec une P-valeur au moins aussi bonne qu'une valeur donnée, augmente avec le nombre de test réalisés. Habituellement, la P-valeur obtenue est "compensée" pour prendre en compte les effets liés aux tests multiples : la E-valeur.

Si l'on considère une séquence génomique de longueur L et un motif de longueur l, le nombre de positions analysées est $N = L - l + 1$. Pour une prédiction donnée, la E-valeur correspond au nombre d'occurrences avec un score au moins aussi bon qu'un seuil fixé que l'on s'attend à trouver par hasard, dans des conditions équivalentes aux tests effectués. Par exemple, une prédiction avec une E-valeur de 1×10^{-3} correspond à une prédiction que l'on s'attend à obtenir une fois par hasard si l'on effectue 1 000 prédictions. Il existe plusieurs façons de compenser la P-valeur pour tenir compte des problèmes inhérents aux tests multiples. Les techniques les plus courantes utilisées sont les suivantes : la correction de Bonferroni et la correction basée sur le taux de faux positifs découverts.

Correction de Bonferroni

Une des approches les plus simples pour prendre en compte les tests multiples est la correction introduite par Bonferroni [10]. Elle consiste à corriger la P-valeur individuelle de chaque test par le nombre total de tests effectués. Si l'on note N le nombre de tests effectués et que l'on considère les tests indépendants, la E-valeur d'un test donné au sens de Bonferroni s'écrit simplement de la manière suivante :

$$\text{E-valeur} = N \times \text{P-valeur}$$

Correction par taux de faux positifs

Lorsque les tests réalisés ne sont pas indépendants, la correction de Bonferroni est un mauvais estimateur de la correction à utiliser. Dans ce cas, il est possible d'utiliser une correction basée sur le taux de faux positifs découverts. Cette méthode se base sur un contrôle de la proportion de faux positifs générés par les tests. Pour ce faire, la proportion de mauvaises prédictions est évaluée sur un large jeu de données dans des conditions similaires à celles utilisées pour traiter le jeu d'étude. Cette technique possède l'avantage d'être plus sensible que l'approche traditionnelle, car elle s'adapte de manière précise à la façon dont sont générées les prédictions. Mais, de ce fait, possède l'inconvénient de nécessiter un recalcul de la correction

lorsque les conditions sont modifiées. C'est le cas, par exemple, lorsque des jeux de tailles différentes sont utilisés.

2.4.5 Pertinence biologique

Nous avons parlé du modèle des matrices, de la manière d'y associer un système de score et des interprétations de ce score. Nous finissons cette partie en discutant de la qualité prédictive intrinsèque des matrices au regard de leur pertinence biologique. Cela nous amène à présenter la "qualité" d'une matrice, puis à traiter du problème de spécificité des prédictions.

Qualité d'une matrice

La qualité d'une matrice est déterminée par son caractère discriminant, c'est-à-dire sa capacité à prédire des occurrences correctes. Différentes approches ont été développées pour évaluer cette qualité. Une des premières approches a été fournie par Schneider [75]. Elle repose sur le contenu informationnel du modèle matriciel. En considérant la matrice de comptage M, le contenu informationnel se définit de la manière suivante :

$$I(M) = \sum_{i=1}^{m} \left(2 + \sum_{b \in \{A,C,G,T\}} P_{b,i} \log_2 P_{b,i} \right)$$

où m représente la longueur de la matrice et $P_{b,i}$ représente la fréquence de la lettre b à la position i de la matrice de comptage. Dans ce cas, un bon motif est un motif dont le contenu informationnel est élevé.

Une des limitations de cet évaluateur est liée à la considération faite sur l'équiprobabilité d'apparition des nucléotides dans les séquences étudiées (chaque nucléotide possède la même probabilité d'apparition). Afin de prendre en compte cette variabilité, le calcul du contenu informationnel a été étendu [75, 82] pour incorporer la fréquence attendue de chaque nucléotide. Le score peut alors se définir comme l'entropie relative (ou distance de Kullback-Leibler) du modèle matriciel du motif par rapport au modèle de fond :

$$I_{seq}(M) = \sum_{i=1}^{m} \sum_{b \in \{A,C,G,T\}} \left(P_{b,i} \log_2 \frac{P_{b,i}}{p_b} \right)$$

où p_b représente la fréquence attendue de la lettre b.

Ce contenu informationnel peut être relié à l'énergie moyenne d'interaction ADN-protéines des sites ayant servi à construire la matrice [82, 83].

Limites de la recherche à l'aide de matrices

La recherche d'éléments régulateurs, en ne prenant en compte que la séquence, comporte d'importantes limitations et est insuffisante pour appréhender toute la complexité des mécanismes de régulation transcriptionnelle. La proportion de faux positifs peut devenir dans ce cas rédhibitoire. Par exemple, un motif approché de longueur 6 possédant une erreur peut

être rencontré par hasard, toutes les 1 000 bases, dans une séquence. Dans ce cadre, la plupart des sites prédits n'ont pas d'activité *in vitro* et doivent être considérés comme de fausses prédictions. Wasserman et Sadelin ont désigné ce manque de spécificité le "futility theorem" [92].

Le fait d'éliminer les mauvaises matrices n'est pas suffisant pour répondre au problème de la recherche d'éléments régulateurs. En effet, les prédictions sont intrinsèquement non spécifiques du fait de la modélisation et de la façon dont le problème biologique est traité (seule la séquence nucléique est considérée). Détecter des liaisons observables *in vitro* n'est pas suffisant pour connaître les sites actifs *in vivo*.

Pour pouvoir contrer cette faible spécificité des prédictions une des solutions est de travailler avec plus d'informations : utiliser des séquences partageant des éléments pour réduire le nombre de faux positifs. Deux pistes peuvent être envisagées : celle de la génomique comparative et celle de la recherche de motifs sur-représentés dans les séquences promotrices de gènes co-régulés. Nous présentons ces approches dans les sections 2.5 et 2.6.

2.5 Génomique comparative

Pour augmenter la sélectivité des méthodes de détection de signaux régulateurs, il peut être pertinent de prendre en compte la conservation entre espèces. Cette approche repose sur l'idée suivant laquelle les signaux régulateurs sont soumis à une pression sélective plus forte que le reste des régions non codantes. Il est alors intéressant de considérer les éléments régulateurs conservés entre plusieurs organismes [96].

Nous détaillons dans cette section, l'application de la génomique comparative, au problème de la recherche de signaux *cis*-régulateurs. Dans un premier temps, nous précisons et replaçons cette problématique dans le cadre plus général de la génomique comparative. Ensuite, nous explorons le problème d'empreinte phylogénétique et d'alignement de séquences. Enfin, nous discutons des contraintes en termes de distance phylogénétique et de nombre d'espèces nécessaires.

2.5.1 Conservation des éléments régulateurs

Au départ, concentré sur l'analyse des régions codantes (identification des frontières exons-introns ...), le champ d'étude de la génomique comparative s'est progressivement étendu à l'analyse des régions non codantes et des signaux régulateurs. L'utilisation de la génomique comparative pour la recherche de signaux régulateurs repose sur l'idée suivante : des gènes orthologues (chez des espèces partageant un ancêtre commun) situés à une distance évolutive "suffisante" doivent avoir en commun une partie de leurs mécanismes de régulation du fait de la pression de sélection qui s'est appliquée aux éléments régulateurs. En effet, comme pour les séquences codantes, on peut considérer que les signaux régulateurs fonctionnels sont soumis à une pression sélective plus importante que le reste des régions non codantes. Il est possible dans ce cas de rechercher les signaux régulateurs uniquement dans les régions régulatrices fortement conservées entre orthologues. Deux étapes sont alors nécessaires pour rechercher des régions conservées :

– en premier lieu, la recherche de gènes orthologues chez différentes espèces ;

– et ensuite, l'extraction de régions conservées par alignement.

Les premières extensions se sont concentrées sur l'analyse comparative à l'échelle génomique. Par exemple, les études menées par Oeltjen et co-auteurs [57] sur les génomes de l'homme et de la souris dans une large région relative à la kinase BTK, ont permis de mettre en évidence des portions conservées correspondant à la fois à des gènes et à des régions non codantes jouxtant le premier exon de ces gènes. Plus récemment, Wassermann et co-auteurs [92] ont mené une analyse comparative des sites de fixation connus de facteurs spécifiques au muscle du squelette chez l'humain le rat et la souris. Cette étude a permis de montrer que 98% des sites de fixation expérimentalement vérifiés étaient confinés dans 19% des fragments de séquences humaines conservées entre l'humain et les orthologues chez la souris et le rat.

Les contraintes imposées par la première étape sont déterminantes vis à vis de la capacité de l'approche à détecter les signaux régulateurs. Hardison et co-auteurs [29], ont par exemple montré, en comparant le génome humain avec celui d'autres espèces, que le choix de l'espèce la plus adaptée était dépendant de la région où étaient situées les régions étudiées. Cette question est abordée dans la section 2.5.3.

2.5.2 Empreinte et masquage phylogénétique

Lorsque l'on dispose d'un ensemble de gènes orthologues, le problème est alors de rechercher les éléments conservés entre ces séquences. Dans ce cadre, nous précisons les notions d'empreinte et de masquage phylogénétique et nous détaillons les contraintes que ces méthodes induisent sur la stratégie d'alignement de séquences à employer.

Empreinte phylogénétique

Le terme *empreinte phylogénétique* désigne l'approche qui cherche à identifier des éléments régulateurs en comparant les séquences génomiques d'espèces proches. La technique par empreinte phylogénétique a d'abord été réalisée manuellement en comparant visuellement l'alignement de séquences orthologues. Cette technique a ensuite évolué vers un traitement automatisé plus systématique.

Masquage phylogénétique

Lorsque l'on dispose d'un nombre suffisant d'espèces très proches (humain/primates), il est possible de rechercher des signaux de régulation par *masquage phylogénétique*. Le masquage phylogénétique consiste à rechercher des éléments divergents chez les espèces étudiées. Si l'on considère un ensemble de séquences alignées, un masquage consiste à "mettre de côté" les régions pour lesquelles un nucléotide diverge chez au moins une des espèces. Ceci permet alors de fournir des zones parfaitement conservées chez le groupe : les régions non masquées. Cette technique présente l'avantage sur celle par empreinte, de permettre de découvrir des signaux qui ne sont pas présents chez une espèce plus éloignée (humain/souris). Cependant cette technique nécessite d'avoir à disposition un nombre important de génomes d'espèces très proches.

Boffelli et co-auteurs [9] ont utilisé cette technique en comparant des séquences humaines avec celles d'un large ensemble de primates non humains pour découvrir des régions fonctionnelles dans le génome humain et des éléments régulateurs spécifiques aux primates.

Alignement de séquences

Lorsque l'on dispose d'un ensemble de gènes orthologues, la deuxième étape consiste généralement à aligner les séquences promotrices afin d'obtenir des régions fortement conservées. Afin de compléter les programmes classiques d'alignement de type Needleman et Wunsch [54], différentes heuristiques d'alignement à large échelle ont été proposées. La méthode AVID [11] propose par exemple de répondre à la problématique à échelle génomique en fournissant un moyen efficace d'alignement global pouvant travailler sur de longues séquences (plusieurs centaines de kilo bases). On citera également, les algorithmes alignement spécialisés LAGAN and Multi-LAGAN [12].

En se basant sur l'alignement obtenu, les régions fortement conservées peuvent être extraites. Une des stratégies possibles consiste à extraire les régions fortement conservées en fonction du score local d'alignement ou du pourcentage d'identité. Wasserman et co-auteurs proposent par exemple une méthode d'extraction de régions fortement conservées basée sur le pourcentage d'identité et un système de fenêtres glissantes [34].

ConSite
Le logiciel ConSite [44, 73] propose de rechercher dans un couple de séquences des sites de fixation potentiels modélisés par des matrices de comptage. La méthode sur laquelle repose le logiciel peut se décomposer en trois étapes :

1. le programme aligne (si besoin) le couple de séquences orthologues fournies en entrée ;
2. ensuite, les deux séquences sont analysées individuellement afin de rechercher des sites potentiels à l'aide de matrices de haute qualité ;
3. enfin, les sites prédits sont comparés, et seuls sont conservés ceux présents, pour une même matrice, à des positions équivalentes dans les séquences.

Les auteurs ont ainsi montré sur différents jeux de données, que l'ajout d'une séquence orthologue dans la recherche de signaux régulateurs permettait de réduire significativement le taux de faux positifs (réduction de 85% du nombre de sites détectés) sans trop compromettre la sensibilité (diminution de 72.5% à 65.5%).

2.5.3 Choix des espèces

Une des contraintes majeures lorsque l'on utilise la génomique comparative concerne le nombre et le type d'organismes différents nécessaires pour pouvoir rechercher les signaux régulateurs partagés. Cette question peut se poser de la manière suivante : quels génomes en terme de nombre et de distance phylogénétique sont nécessaires pour réaliser une analyse comparative des signaux régulateurs d'une espèce donnée. Pour pouvoir répondre à ce problème d'identification, Eddy propose un modèle mathématique capturant des éléments essentiels de la génomique comparative [21]. Deux résultats importants peuvent être tirés de l'analyse de ce modèle :

- pour une distance évolutive donnée, le nombre de génomes nécessaires pour effectuer une analyse comparative est inversement proportionnel à la taille des signaux recherchés ;
- à distance évolutive proche, le nombre de génomes nécessaires est inversement proportionnel à leur distance phylogénétiques avec l'espèce étudiée.

Ces résultats impliquent ainsi de devoir recourir à un nombre important de génomes lorsque l'on recherche des éléments régulateurs avec une forte résolution (une dizaine de bases), adaptée aux éléments *cis*-régulateurs. Une autre conséquence concerne le nombre de génomes requis en fonction de la nature de la comparaison qui est réalisée. Une approche par masquage phylogénique, qui nécessite des espèces proches, sera délicate à réaliser du fait du nombre d'espèces nécessaires pour garantir une résolution suffisante. Par exemple, Eddy souligne que plus de vingt espèces sont souhaitables pour réaliser une analyse comparative avec une résolution d'une dizaine de bases. Dans le cas de l'approche par empreinte phylogénique, le nombre d'espèces nécessaires pour obtenir la même résolution est nettement inférieur. Plus précisément, pour un taux de faux positifs fixé au maximum à 1×10^{-4} et un taux de faux négatif inférieur à 0.01 (99% d'éléments prédits), 16 espèces à distance équivalente humain-souris sont nécessaires pour effectuer une comparaison par empreinte phylogénique à une résolution de 8 nucléotides. Ce nombre est réduit à 3 lorsque la résolution désirée est de l'ordre de 50 nucléotides.

Il ressort également de cette étude que, concernant la distance évolutive nécessaire, il n'existe pas d'optimum clair. Cependant une distance trop faible a tendance à défavoriser la découverte de signaux conservés.

La conclusion principale que l'on peut tirer peut se poser de la manière suivante : une augmentation du nombre d'espèces prises en compte a pour conséquence d'augmenter la stringence et la résolution de l'approche comparative.

2.6 Recherche de motifs sur-représentés

Une autre façon d'améliorer la recherche de signaux *cis*-régulateurs est de poser l'hypothèse suivante : si l'on recherche des signaux régulateurs dans les régions promotrices d'un ensemble de gènes co-régulés ou co-exprimés, il doit exister des caractéristiques communes au niveau de la séquence promotrice de ces gènes. Dans ce cas, on doit observer une sur-représentation des motifs *cis*-régulateurs détectés dans les régions en amont de ces gènes. Cette question a donné lieu à de nombreux travaux algorithmiques qui sont détaillés dans cette section. Avant de détailler le problème et les solutions permettant d'y répondre, nous précisons les ensembles de gènes sur lesquels nous allons travailler : les gènes co-régulés et co-exprimés.

Gènes co-régulés
Des gènes pour lesquels il existe des mécanismes communs de régulation sont appelés gènes co-régulés. Ces gènes partageant des éléments régulateurs, l'utilisation des séquences en amont de ces gènes doit permettre d'extraire des signaux communs.

Gènes co-exprimés
Les techniques à haut rendement, comme par exemple les puces à ADN, permettent de mesurer le niveau d'expression d'un large ensemble de gènes en parallèle. Il est dans ce cadre possible

d'extraire des gènes qui partagent des profils d'expressions similaires. Un ensemble de gènes exprimés dans des conditions similaires, c'est à dire possédant le même profil d'expression, est appelé ensemble de gènes co-exprimés. La co-expression des gènes peut donner un point d'entrée pertinent sur la compréhension des mécanismes de régulation.

2.6.1 Formalisation du problème

Avant de nous intéresser aux méthodes de recherche de sur-représentation, nous définissons plus formellement le problème de recherche de motifs sur-représentés. En première approche, le problème peut se décrire de la manière suivante :
Entrée : un ensemble $\Phi = \{\phi^1, ..., \phi^n\}$ de séquences régulatrices.
Sortie : l'ensemble des motifs pour lesquels le nombre d'occurrences est significativement plus élevé que ce qui est attendu par hasard.
Cette formulation nécessite d'être précisée sur deux points : d'une part sur ce qu'est le comportement attendu par hasard, et d'autre part sur la nature du motif recherché.

Le comportement attendu par hasard pour le nombre d'occurrences d'un motif sous-entend de préciser la loi de comptage attendu pour le motif.

Concernant les motifs, nous allons distinguer les classes suivantes : les motifs exacts, les motifs avec erreurs, les motifs matriciels. La classe de motifs peut également être déterminée par une base de motifs comme par exemple les motifs matriciels disponibles dans la base de donnée TRANSFAC. Habituellement la prédiction de motifs exceptionnels est réalisée en deux étapes :

- recherche et comptage de motifs d'intérêt sur les séquences ;
- classement des motifs en fonction du caractère exceptionnel de leur comptage.

C'est le choix de la classe de motifs qui va conditionner la stratégie de recherche. On désigne généralement par *méthode d'inférence*, les méthodes comportant uniquement des contraintes sur la structure du motif lui-même. Dans ce cas, les motifs sont construits *ab initio* à partir de l'ensemble de séquences. Par opposition au terme "inférence", on désigne par *recherche à l'aide de modèle*, la recherche de motifs utilisant un ensemble de motifs pré-établis, comme par exemple ceux présents dans la base de donnée TRANSFAC. Le choix d'une classe de motif conditionne les algorithmes à mettre en œuvre.

Lorsque l'espace de recherche associé à la classe de motif n'est pas trop grand, des algorithmes exacts énumératifs sont envisageables. C'est le cas pour la recherche de motifs exacts ou avec erreurs. Nous présentons ces algorithmes dans la section 2.6.2 pour les motifs exacts et 2.6.3 pour les motifs avec erreurs. Pour des motifs matriciels de comptage, la recherche exhaustive n'est pas réalisable du fait de la nature NP-complet du problème [46]. Dans ce cas, des heuristiques de type Expectation-Maximization ou de Gibbs sampling sont mises en œuvre (section 2.6.4). Enfin, lorsque l'on recherche des motifs avec l'aide d'une banque de modèles, des stratégies efficaces peuvent être employées (section 2.6.5). Le tableau ci-dessous (2.2) récapitule les méthodes qui sont présentées dans cette section. Nous présentons successivement ces deux différentes approches dans les sections suivantes.

Oligo-analysis [88]	Détection d'oligonucléotides sur-représentés basée sur le comptage. Calcul de la P-valeur basé sur une loi binomiale.
YMF [78, 79]	Enumération des motifs avec erreur et espace en son centre. Évaluation de la significativité par Z-score.
Weeder [59, 60]	Enumération des motifs avec erreur utilisant un arbre des facteurs.
RMES [68]	Énumération des mots par comptage. Approximation de la loi de comptage pour le calcul de la P-valeur par une loi de Poisson composée, calculée dans un modèle de Markov.
QuickScore [67]	Énumération des mots par comptage. Calcul exact de la P-valeur basé sur les séries génératrices.
SMILE [50]	Énumération des motifs structurés avec erreur utilisant un arbre des facteurs. Évaluation de la significativité des motifs par Z-score.
MEME [3]	Expectation-Maximization avec grand nombre de points de départ.
Gibbs sampler [55]	Gibbs sampling.
TOUCAN [1]	Recherche à l'aide d'une base de matrices (TRANSFAC ou JASPAR). Calcul de la P-valeur basé sur une loi binomiale.
OTFBS [97]	Recherche à l'aide d'une base de matrices (TRANSFAC). Calcul de la P-valeur basé sur une loi binomiale.
oPOSSUM [34]	Recherche à l'aide d'une base de matrices (JASPAR). Filtrages des sites par génomique comparative (humain/souris). Évaluation de la significativité par Z-score ou probabilité exacte de Fischer.

TAB. 2.2: Algorithmes d'inférence et de recherche de motifs.

2.6.2 Motifs exacts

La recherche de mots exacts transposée à celle d'éléments régulateurs entraîne des contraintes fortes quant à la nature des éléments qui peuvent être détectés. En effet, les signaux de régulation présentent habituellement une variabilité dans leur composition. Dans ce cas il ne s'agit pas de trouver des sites variables mais des traces de sites à l'aide de sous mots plus courts et particulièrement conservés.

Les méthodes par comptage de mots exacts sont les méthodes les plus simples à mettre en œuvre lorsque l'on recherche des motifs sur-représentés dans un ensemble de séquences. Généralement, ces méthodes procèdent en deux étapes :

1. comptage de tous les mots de longueur l présents dans l'ensemble de séquences Φ ;
2. évaluation de la significativité statistique de chacun des mots en fonction d'un contexte de fond donné.

La première étape de l'analyse est facile. Une table de hachage correspondant au nombre d'occurrences dans Φ pour chacun des mots de longueur l est tout d'abord établie. La difficulté réside alors dans l'évaluation statistique de la significativité de chacun des mots en fonction du nombre de ses occurrences et du contexte.

Le contexte doit définir le comportement ou comptage attendu dans des conditions neutres. Habituellement, ce comportement est établi soit de manière empirique (à partir d'un jeu de référence) soit en utilisant un modèle de type Markovien (qui peut être construit à partir de l'ensemble Φ des séquences étudiées ou d'un jeu de référence). Une fois le modèle de référence déterminé, il convient d'établir une statistique pour le comptage des occurrences de chaque mot. Suivant les conditions d'étude, différentes statistiques et moyens de les évaluer sont possibles.

Lorsque que l'on considère uniquement les occurrences non chevauchantes de motifs, la statistique de comptage peut être décrite par une loi binomiale [88]. Étant donné un nombre d'occurrences k observées dans Φ pour le mot u, la P-valeur de ce mot peut se définir comme la probabilité d'observer au moins k fois u dans le modèle de fond. En posant p_u la fréquence attendue du mot u, ceci se note de la manière suivante :

$$P(occ\{u\} \geq k) = \sum_{j=k}^{N} \binom{j}{N} p_u^j (1 - p_u)^{(N-j)} \tag{2.2}$$

Une autre approche pour décrire la statistique de comptage des mots exceptionnels, en tenant compte des chevauchements entre occurrences, est d'utiliser une loi de Poisson composée. Un moyen d'évaluer cette distribution, lorsque le modèle de fond est Markovien, est proposé dans la méthode RMES [68]. Il est également possible d'évaluer la statistique de comptage des mots dans un modèle de Markov en utilisant la théorie des larges déviations. Le logiciel LD-SPatt [56] propose par exemple une implémentation de ce type d'approche.

2.6.3 Motifs avec erreurs

Afin d'étendre les motifs exacts, dont la rigidité est mal adaptée à la recherche de signaux régulateurs, de nombreuses méthodes utilisant des motifs avec erreurs ont été développées. Van Helden et co-auteurs [89] ont par exemple proposé de rechercher des motifs en forme de dyades (deux motifs exacts espacés). De manière plus générale, un motif avec erreurs peut se définir à partir d'un motif exact de longueur l pour lequel certaines positions sont indéterminées, c'est-à-dire pouvant accepter chacun des 4 nucléiotides A, C, G ou T. Par exemple, le motif `ACNANGT` est un motif de longueur 7 contenant 2 erreurs.

Nous présentons ici trois méthodes exactes permettant de rechercher des motifs comportant des erreurs : YMF, SMILE et Weeder. Pour chacune de ces méthodes, les contraintes

imposées aux positions des erreurs sont différentes, ce qui implique une structure générale différente pour le motif. Nous détaillons dans chacun des cas la nature de cette structure et la stratégie de recherche associée.

YMF

Le logiciel YMF (Yeast Motif Finder) [78, 79] propose de rechercher les motifs avec erreurs espacés. Ce type de motif est défini comme un mot écrit avec l'alphabet IUPAC restreint et peut comporter un nombre fixé d'espaces, ou positions non conservées, en son milieu. Plus formellement, un motif est défini par un préfixe α et un suffixe β écrits avec l'alphabet $\{A, C, G, T, R, Y, S, W\}$ et éventuellement avec un certain nombre d'espaces (N dans le code IPUPAC) situés entre le préfixe et le suffixe. Si l'on note N^s les s espaces composant le motif, la structure du motif peut s'écrire $\alpha N^s \beta$.

Étant donné la structure $\alpha N^s \beta$, les paramètres en entrée du problème sont le nombre d'espaces s et la longueur cumulée l du suffixe et du préfixe. La première étape de l'algorithme consiste à construire une table référençant le nombre d'occurrences rencontrées dans les séquences d'entrée pour tous les motifs vérifiant la structure précédente.

Une fois le comptage des occurrences effectué, les motifs sont classés suivant leur Z-score. Le Z-score d'un motif est calculé en fonction du nombre de ses occurrence présentes dans les séquences d'entrée en évaluant le nombre attendu et son écart type dans un modèle de fond Markovien.

SMILE

Marsan et Sagot [50] ont proposé un algorithme exact de recherche de motifs structurés, appelés p-boîtes. La structure d'une p-boîte est la suivante : une collection ordonnée de p éléments séparés des positions indéterminées où chacun de ces éléments ou boîtes est un mot pouvant comporter une substitution. Un motif est donc défini par la donnée des p boîtes $(m^1, ..., m^p)$ et des $p-1$ intervalles $d^1, ..., d^{p-1}$. L'objectif de l'algorithme est de déterminer les boîtes m^i étant donné le nombre p et les $p-1$ intervalles. Pour résoudre ce problème, l'algorithme procède en deux étapes. Dans un premier temps, un arbre des suffixes est construit à partir des séquences génomiques. Puis, dans un deuxième temps, cet arbre est parcouru de manière simultanée et récursive pour tous les motifs vérifiant la structure donnée. La stratégie adoptée pour réduire la complexité du parcours de l'arbre est d'effectuer des sauts arrière pour chacun des intervalles d^i. La méthode employée, pour parcourir l'arbre des suffixes pour chacune des boîtes du motif, est celle décrite dans [71]. Cette méthode effectue un parcours en profondeur pour tous les motifs de taille k (ici une boite) comportant e erreur. La complexité peut s'écrire de la manière suivante : $O(N|\Sigma|^e k^e)$ où $|\Sigma|^e k^e)$ correspond à la taille du voisinage à distance e de m et N la taille totale des séquences.

Pour terminer, un test de significativité est appliqué à chacun des motifs trouvés. Deux moyens pour calculer la significativé d'un motif structuré sont mis en œuvre : un calcul de Z-score et un calcul du χ^2 comparant la table de contingence observée du motif et la table de comptage attendu.

Weeder

Une approche alternative s'appuyant également sur le parcours de l'arbre des suffixes introduit avec SMILE [71] a été proposée par Pavesi [59, 60]. Cette approche permet en fixant des contraintes sur les positions relatives de l'erreur, d'étendre l'énumération à des motifs de grande taille. Ces contraintes sont les suivantes : étant donné le taux d'erreur $\epsilon < 1$, le préfixe p du motif doit contenir au plus $\lceil \epsilon |p| \rceil$ erreurs.

Pour un motif donné, l'étape d'extraction se déroule de manière itérative en incorporant les chemins valides, c'est-à-dire les chemins possédant moins de $\lceil \epsilon |p| \rceil$ erreurs. Cette contrainte permet de réduire la complexité de l'étape d'extraction de motifs de manière significative sans compromettre la recherche de motifs pertinents. Si l'on note $|\Sigma|$ la taille de l'alphabet et N la longueur totale des séquences ($N = \sum_i |\delta_i|$), la complexité sans contrainte sur l'erreur est la suivante $O(|\Sigma|^e m^e N)$ où $|\Sigma|^e m^e$ représente le nombre de chemins à distance e pour un motif donné (e erreurs sur m positions). En fixant le taux d'erreur ϵ la complexité devient : $O(|\Sigma|^{\epsilon m} \lceil 1/\epsilon \rceil^{\epsilon m} N)$.

Chaque motif extrait est enfin classé en fonction de sa significativité qui est évaluée par un Z-score dépendant du nombre d'occurrences attendues.

2.6.4 Motifs matriciels

Nous avons vu qu'un des modèles les plus "adaptés" pour décrire un site de fixation était le modèle matriciel. Les approches d'inférence de matrice reposent sur la construction d'alignements optimaux de fragments de séquences. De nombreuses heuristiques, principalement basées sur des algorithmes Expectation-Maximization, ont été proposées pour répondre à ce problème (Gibbs sampler [55], MEME [3]). Un des avantages de ce type de méthode est de considérer une structure plus souple que les motifs avec erreurs : une matrice de comptage. Ces heuristiques sont généralement sensibles au bruit et aux conditions initiales. Comme dans le cas des méthodes énumératives, la significativité des motifs produits est habituellement évaluée dans un deuxième temps. Le problème peut se formuler de la manière suivante : étant donné un ensemble de séquences $\Phi = \{\phi^1, ..., \phi^n\}$, trouver la matrice M de longueur m qui maximise la vraisemblance de M dans Φ.

Si l'on considère un ensemble de n séquences et un entier m correspondant à la longueur du motif à trouver, la stratégie consiste à trouver un mot u^k de longueur m pour chaque séquence ϕ^k tel que la similarité entre ces mots soit maximale. Lorsqu'ils sont alignés, ces mots permettent de former une matrice de fréquences M. Si l'on note $a^1, ..., a^n$ les positions des mots u^k dans chacune des séquences ϕ^1, ..., ϕ^n, le problème consiste à trouver ces positions.

Algorithme EM

Si l'on considère le problème de recherche de motifs dans un ensemble de séquences comme un problème de données manquantes, où les données manquantes sont les positions $a^1, ..., a^n$, une première solution pour résoudre le problème est d'utiliser un algorithme de type Expectation-Maximization [43]. En partant d'un ensemble de positions $a^1, ..., a^n$ aléatoirement choisies, l'algorithme construit de manière itérative la matrice M, en utilisant les deux étapes suivantes : une étape d'estimation de probabilité $P(M, i^k)$ d'observer la matrice M en chaque

position i de la séquence ϕ^k, et une étape de maximisation où les positions fournissant la probabilité maximale (a^k) sont utilisées pour reconstruire la matrice. Les étapes nécessaires à la construction de M sont détaillées dans l'algorithme 2. L'algorithme s'arrête lorsqu'il y a convergence, c'est-à-dire lorsque la probabilité d'observer M aux positions $a^1, ..., a^n$ ne peut plus être améliorée.

Algorithme 2 : Principe de l'algorithme Expectation Maximization pour la recherche de motifs

Entrées : un ensemble de séquences $\Phi = \{\phi^1, ..., \phi^n\}$, une taille de mot m
Sorties : un modèle M
1. Sélectionner aléatoirement des positions $a^1, ..., a^n$ dans Φ
2. Créer une matrice de fréquence M à partir de $a^1, ..., a^n$
3. **pour** i^k *de* 1 *à* $L - m + 1$ *dans* ϕ^k **faire**
   $P(M, i^k) = \prod_{j=0}^{m} M_{\phi_{i^k+j}, j}$
4. Choisir les nouvelles positions a^k qui maximisent $\prod_k P(M, i^k)$
5. Répéter 2., 3. et 4. jusqu'à convergence
retourner M

Les algorithmes EM sont déterministes pour un point de départ donné. La convergence vers un optimum local est garantie. Un des moyens habituellement mis en œuvre pour tenter d'échapper aux optima locaux est de lancer la recherche un grand nombre de fois en utilisant des points de départ différents.

Ce type d'approche est mise en œuvre dans la méthode MEME (Multiple EM for Motif Elicitation) [3]. Une des particularités de cette méthode est d'employer un très grand nombre de points de départ et d'effectuer une itération pour chacun de ces points afin de sélectionner de "bonnes" matrices de départ. Le logiciel "The Improbizer" [2] utilise également un algorithme EM tout en permettant d'évaluer la vraisemblance du motif dans le jeu de séquence de manière relative par rapport à un modèle de fond Markovien.

Gibbs sampling

Afin de se départir des maxima locaux des méthodes EM classiques, Lawrence et co-auteurs [42] ont proposé une adaptation à la recherche de motifs nucléiques de l'algorithme EM non déterministe de Gibbs sampling. L'algorithme procède de manière itérative en supprimant à chaque itération une séquence contribuant à l'alignement. La stratégie peut s'écrire comme indiqué dans l'algorithme 3.

L'algorithme de Gibbs sampling donne de bons résultats dans de nombreux cas spécifiques (convergence vers l'optimum global). Néanmoins, dans certains cas lorsque les signaux sont difficiles à extraire l'algorithme peut converger vers un maximum local.

L'approche peut être étendue, pour prendre en compte la présence d'un nombre quelconque de motifs par séquence. Dans le cadre de la recherche de signaux de régulation, la version de Gibbs sampling implémentée dans AlignAce [38] permet de rechercher l'ensemble des motifs possédant des sur-représentations dans un ensemble de séquences. Une autre adaptation de

Algorithme 3 : Principe de l'algorithme de Gibbs sampling pour la recherche de motifs

Entrées : un ensemble de séquences $\Phi = \{\phi^1, ..., \phi^n\}$, une taille de mot m
Sorties : un modèle M
1. Sélectionner aléatoirement des positions $a^1, ..., a^n$ dans Φ
2. Sélectionner aléatoirement une séquence ϕ^k
3. Créer une matrice de fréquence M à partir de $a^1, ..., a^n$ sans a^k
4. **pour** i^k *de* 1 *à* $L - m + 1$ *dans* ϕ^k **faire**
   $P(M, i^k) = \prod_{j=0}^{m} M_{\phi_{i^k+j}, j}$
5. Choisir les nouvelles positions a^k proportionnellement à $P(M, i^k)$
6. Répéter 2., 3., 4. et 5. jusqu'à convergence
retourner M

l'algorithme est mise en œuvre dans la méthode GLAM [24] qui permet de produire des motifs sans spécifier a priori la taille des motifs recherchés : la taille des motifs est également optimisée. Une autre amélioration proposée dans MotifSampler [85] concerne la modélisation de la séquence. Ici un modèle de Markov spécifique à l'organisme étudié est utilisé pour modéliser la séquence de fond.

2.6.5 Recherche de motifs à l'aide d'un ensemble de candidats

Dans certains cas, par exemple lorsque les séquences étudiées sont longues, il peut être difficile de découvrir des motifs régulateurs par une approche brute d'inférence. Tompa et co-auteurs [86] ont par exemple montré la difficulté à rechercher des motifs régulateurs par inférence en comparant les résultats de nombreuses méthodes sur différents jeux de données, réels et simulés, pour différents organismes (levure, souris, humain, ...). Une des solutions pour répondre à la difficulté du problème est alors de réduire l'espace de recherche en recherchant des motifs régulateurs parmi un ensemble de candidats connus a priori. Un des avantages de ce type d'approche est de permettre une analyse exhaustive des occurrences des motifs du fait de la taille raisonnable de l'espace de recherche. De plus, les techniques comme par exemple celle du chIP-on-chip doivent permettre de disposer, à terme, de bases de sites de meilleure qualité et en plus grand nombre. Nous détaillons ici trois logiciels représentatifs de cette catégorie utilisant des banques de matrices comme ensemble de candidats : TOUCAN [1], OTFBS [97] et oPOSSUM [34].

TOUCAN

TOUCAN [1] est une suite logicielle dédiée à la découverte et à l'analyse d'éléments *cis*-régulateurs partagés par un ensemble de séquences. Cette suite est composée de différents logiciels qui couvrent un large spectre de l'analyse des signaux *cis*-régulateurs. Nous présentons ici les composants de cette suite qui concernent la recherche de motifs sur-représentés à partir d'une base de matrices. La recherche s'effectue en deux étapes. La première étape consiste à rechercher des motifs connus à l'aide du logiciel MotifScanner. La seconde étape consiste à évaluer la significativité des motifs prédits en fonction de la sur-représentation de leurs occurrences.

La stratégie de recherche proposée dans MotifScanner [1] repose sur le concept suivant : maximiser la probabilité d'observer la séquence génomique fournie en entrée étant donné un modèle de site matriciel et un modèle de fond Markovien. Dans ce cadre, les portions de la séquence correspondant aux occurrences potentielles de sites sont modélisées par la matrice de comptage et les autres portions par un modèle de fond Markovien. Si l'on considère, par exemple, une seule occurrence de motif à la position a dans la séquence ϕ, la probabilité d'observer la séquence se calcule de la manière suivante :

$$P(\phi|a, M, B) = \prod_{i=1}^{a-1} P(\phi_i|B) \prod_{j=1}^{m} P(M_{\phi_{a+j},j}) \prod_{i=a+m}^{L} P(\phi_i|B)$$

où $P(\phi_i|B)$ représente la probabilité d'observer la lettre ϕ_i dans le modèle de fond B et $P(M_{\phi_{a+j},j})$ la probabilité d'observer la lettre ϕ_{a+j} à la position j de la matrice de comptage.

Pour localiser les instances d'un modèle donné, l'algorithme procède en deux étapes : étant donné une séquence ϕ, un modèle de site M et un modèle de fond B, l'algorithme va tout d'abord calculer le nombre d'instances Q le plus probable. Ensuite les Q positions possédant les scores les plus élevés sont extraites. L'évaluation du nombre d'instances est calculé en maximisant la vraisemblance de la séquence dans une alternance de deux modèles. La vraisemblance pour Q occurrences peut se noter de la manière suivante :

$$E_{\phi,M,B}[Q] = \sum_{k=0}^{\infty} k \times P(Q = k|\phi, M, B)$$

où k représente le nombre d'instances du motif M dans la séquence ϕ et $P(Q = k|\phi, M, B)$ représente la probabilité d'observer c instances, étant donné la séquence ϕ, le modèle M et le modèle de fond B. Cette probabilité est calculée pour les différentes valeurs de k de manière itérative. Le schéma général de l'algorithme est donné ci-dessous (4).

Algorithme 4 : Algorithme de MotifScanner

Entrées : une séquence ϕ, une matrice M, un modèle de fond B, un seuil ϵ
Sorties : l'ensemble des occurrences de M dans ϕ
Initialiser $P(Q = 0|\phi, M, B)$ et $P(Q = 1|\phi, M, B)$
tant que $P(Q = i|\phi, M, B) > \epsilon$ **faire**
  $i \leftarrow i + 1$
  pour k *dans* $0..i$ **faire**
    mettre à jour $P(Q = k|\phi, M, B)$
Calculer $E_{\phi,M,B}[Q]$
retourner *les Q positions de meilleur score*

Pour évaluer la fréquence attendue de chaque matrice, la fréquence des sites potentiels dans un jeu de référence est mesurée. Les séquences de références utilisées sont extraites de différentes banques de données (Eukariotic Promoter Database [64] ou des régions régulatrices extraites de Ensembl [37]). La P-valeur de chaque matrice (des occurrences de) est alors estimée en utilisant une loi binomiale pour modéliser le comptage attendu.

La significativité d'un motif est alors calculée en corrigeant, la P-valeur par le nombre N de motifs utilisés de la manière suivante :

$$sig(occ\{M\}) = -\log_{10}(N \times P(occ\{M\} \geq k))$$

L'algorithme retourne la liste des motifs M les plus significatifs (classée suivant le coefficient de significativité sig).

OTFBS

OTFBS [97] est un logiciel de recherche de motifs sur-représentés utilisant les matrices de TRANSFAC comme modèles de sites. Similairement à TOUCAN, la méthode procède en deux étapes pour extraire les motifs sur-représentés. Dans un premier temps, en utilisant l'ensemble des matrices de TRANSFAC, des sites potentiels sont recherchés dans le jeu de séquences à l'aide de l'algorithme MatInspector (section **??**). Dans un deuxième temps, une P-valeur mesurant la sur-représentation des occurrences, est calculée pour chaque matrice en utilisant un test binomial de significativité. OTFBS utilise un ensemble de séquences de contrôle issues de la base EPD [64], afin de calculer la distribution de fréquences attendues pour chaque motif. Étant donné un seuil de similarité τ et une matrice M, la distribution de fréquences est calculée pour un ensemble de seuils (100) de la manière suivante :

$$p_{M,\tau} = S_{\geq\tau}/N_{control}$$

où $S_{\geq\tau}$ représente le nombre d'occurrences trouvées avec un score supérieur ou égal τ par MatInspector dans les séquences de contrôle et $N_{control}$ le nombre de nucléotides présents dans le jeu de contrôle. La P-valeur, probabilité d'observer au moins k occurrences de M dans le jeu de séquences étudiées, est alors calculée en approximant la loi de comptage des occurrences par une loi binomiale ayant $p_{M,\tau}$ comme fréquence attendue.

oPOSSUM

Le logiciel oPOSSUM [34] est un logiciel de recherche de sites de fixation sur-représentés chez la souris et l'humain utilisant la génomique comparative. Il se distingue des deux programmes précédents par la prise en compte de la conservation entre les séquences promotrices humain/souris afin d'améliorer la pertinence des prédictions. oPOSSUM est composé de deux unités : une base de données des sites conservés entre l'humain et la souris et une unité statistique pour évaluer les sur-représentations. L'utilisateur fournit en entrée un ensemble de gènes et un ensemble de matrices (sélectionnées dans la base JASPAR). Le logiciel retourne la liste des matrices les plus significatives pour ce jeu de données, c'est-à-dire les matrices pour lesquelles il existe une sur-représentation des occurrences dans les régions conservées avoisinant le site d'initiation de la transcription. Ces deux unités : base de données et unité statistique sont détaillées ci-après.

La base de données des sites conservés est construite à l'aide des matrices de JASPAR et d'un ensemble de séquences orthologues homme-souris. La construction de la base peut se décomposer en trois étapes :

1. **Récupération d'un ensemble de gènes orthologues humain-souris à partir de la base Ensembl.** Pour chacun des gènes, les séquences (répétitions masquées)

correspondant à 5 000 bases en amont du site d'initiation de la transcription et à 5 000 bases dans la partie 3' du gène sont utilisées. Seules les paires de séquences correspondant à des gènes orthologues sont conservées.

2. **Récupération de l'empreinte phylogénétique pour les couples de séquences orthologues.** Un alignement des couples de séquences est réalisé. Seules sont prises en compte les régions fortement conservées entre l'homme et la souris, pour lesquelles le taux de conservation est d'au moins 75%.
3. **Détection des sites potentiels.** En utilisant les matrices présentes dans JASPAR l'ensemble des régions conservées est analysé. Afin de réduire le nombre de faux positifs, seules les matrices possédant un contenu informationnel suffisant sont utilisées (supérieur à 8 bits). Le système de score employé est un système de log-score tel que défini précédemment 2.3.2. Seuls les sites avec un score supérieur au seuil de coupure, présents à la fois dans les deux séquences du couple orthologue, sont conservés dans la base de données.

Deux mesures statistiques de la sur-représentation des sites dans l'ensemble de séquences d'entrée sont utilisées : le Z-score et la probabilité exacte de Fischer.

Chapitre 3

Sur-représentations locales

Sommaire

Nous avons vu dans le chapitre précédent que la recherche d'éléments *cis*-régulateurs pouvait être améliorée lorsque l'on considérait, d'une part, la conservation entre espèces (génomique comparative), et d'autre part, leur sur-représentation dans des séquences co-exprimées. Dans ce chapitre, nous explorons la possibilité de prendre en compte d'autres types d'information dans la recherche de signaux régulateurs. Nous discutons tout d'abord de l'intérêt de prendre en compte la conservation spatiale des sites de fixation. Cette conservation spatiale nous amène à formuler le problème de recherche de *sur-représentations locales*. Nous montrons ensuite comment combiner génomique comparative et sur-représentation locale de manière souple, en s'affranchissant des contraintes habituellement introduites par l'alignement de séquences. Nous concluons ce chapitre en proposant un calcul de la significativité d'une fenêtre locale lorsque le modèle de fond est *hétérogène*.

3.1 Sites de fixation et localité

La fonctionnalité d'un site de fixation peut être conditionnée par son positionnement sur le promoteur. Ce conditionnement peut être relatif au site d'initiation de la transcription ou aux sites d'autres facteurs se combinant pour former des complexes de régulation. Nous commençons par détailler l'importance de ces contraintes spatiales dans les mécanismes de régulation. Puis, nous introduisons la notion de région spatiale pour un ensemble de séquences avec la définition de fenêtre positionnelle.

3.1.1 Facteurs et spécificité positionnelle

Différents facteurs de transcription sont nécessaires à l'initiation de la transcription chez les eucaryotes. En particulier, certains facteurs se fixent dans la région proximale afin de permettre à l'ARN polymérase de débuter la transcription. Le promoteur obéit à une organisation spécifique. Par exemple :

- les boîtes TATA, fixées par le facteur TBP, sont situées à environ 25 bases du site d'initiation ;
- les boîtes CCAAT, fixées par le facteur CTF, sont situées à environ 80 bases du site d'initiation ;
- les boîtes GC, fixées par le facteur Sp1, sont situées à environ 90 bases du site d'initiation.

Concernant les boîtes TATA, Gralla et co-auteurs [28] ont montré chez *E. coli* que la position des éléments régulateurs le long du promoteur avait une influence sur la manière dont la régulation par ces éléments opérait. Les gènes activés par le facteur TBP possèdent souvent des mécanismes d'expression dépendant du tissu et du contexte. Récemment, Ponjavic et al. [63] ont montré en étudiant la distribution de la distance entre la boîte TATA et le site d'initiation de la transcription dans le génome de vertébrés, que la position relative de la boîte TATA était corrélée à la tissu-spécificité de l'expression. Cette préférence positionnelle a également été mise en évidence de manière plus large chez les eucaryotes [61].

D'autre part, Tsunoda et co-auteurs [87] ont mis en évidence l'affinité positionnelle des sites potentiels de fixation pour différents facteurs tirés de TRANSFAC en utilisant les séquences promotrices tirées de EPD [64]. Pour cela, ils ont mesuré systématiquement les variations du nombre de sites prédits le long de séquences alignées sur le site d'initiation de la transcription (fenêtres de 20 bases). Les cinq facteurs dont l'affinité positionnelle mesurée, c'est-à-dire l'écart entre le nombre de sites prédits et attendus est la plus forte, sont dans ce cas : TATA, MEF2, Oct-1, GC-box, Barbie Box. Le tableau 3.1 donne la liste des dix facteurs pour lesquelles l'affinité positionnelle est la plus forte.

3.1.2 Positionnement des facteurs collaboratifs

La conservation de la position peut également être relative à un autre facteur. Nous avons vu que les éléments *cis*-régulateurs pouvaient être structurés en unités modulaires. Chacune de ces unités est constituée par un ensemble de facteurs de transcription et de sites de fixation qui sont organisés spatialement le long du promoteur [40, 95]. Un exemple de module de régulation est donné dans le figure 3.1.

Facteur	Fenêtre	Affinité
TATA	-40, -20	6.25
MEF2	-40, -30	5.92
TATA(B)	-40, -20	5.89
Oct-1	-40, -30	5.07
GC-box	-70, -50	5.04
Barbie Box	-40, -20	4.27
CdxA	-40, -20	4.21
Pbx-1	-40, -30	3.96
Brn-2	-220, -210	3.84
Sox-5	-70, -60	3.74

TAB. 3.1: Facteurs possédant la plus forte affinité positionnelle, d'après [87].

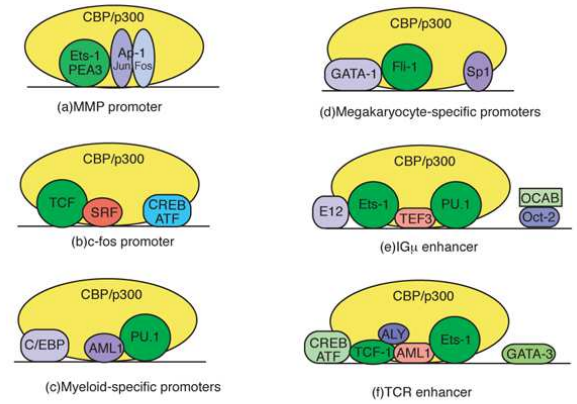


FIG. 3.1: Exemples de modules *cis*-régulateurs[1].

Une piste de recherche pour détecter ces modules est de prendre en compte la conservation spatiale, entre les sites de fixation composant le module. Le logiciel CREME [76] propose par exemple de trouver des co-occurrences de facteurs, dans les régions non codantes conservées entre l'humain et la souris.

3.1.3 Fenêtre locale

Au vu des observations précédentes, il semble pertinent de prendre en compte la sur-représentation d'éléments régulateurs de manière locale. Pour cela, nous définissons la notion de fenêtre spatiale délimitant un ensemble d'éléments *cis*-régulateurs.

Lorsque l'on considère un ensemble de séquences et un motif, une fenêtre peut se définir comme une région englobant un ensemble d'occurrences du motif. Ces régions peuvent être courtes lorsque le facteur possède une forte spécificité spatiale (par exemple pour les boîtes TATA) ou longues dans le cas contraire. Une région est alors caractérisée par une position

[1] *source de l'image* : `http ://www.brc.riken.jp/lab/dna/en/GENESETBANK/ets_txnf.png`

de début et de fin dans un référentiel commun à l'ensemble des séquences. Ce référentiel est déterminé en utilisant des positions clairement définies dans chacune des séquences. On peut par exemple considérer, comme en 3.1.1, le site d'initiation de la transcription comme référence ou, comme en 3.1.2, le site de fixation d'un autre facteur.

De manière plus formelle, une fenêtre est définie de la manière suivante : étant donné un facteur de transcription, un ensemble $\Phi = \{\phi^1, ..., \phi^n\}$ de séquences alignées sur une position de référence et un ensemble de sites repérés sur ces séquences par leur position de début, une fenêtre est un intervalle $[i, j]$ et un nombre k de sites dont les positions sont comprises entre i et j.

Pour fixer les idées, un exemple de fenêtre contenant 6 sites est donné dans la figure 3.2. Les occurrences prédites sont représentées par des rectangles et la fenêtre représentée par la boîte en pointillés. Ici les séquences sont alignées sur le site d'initiation de la transcription.

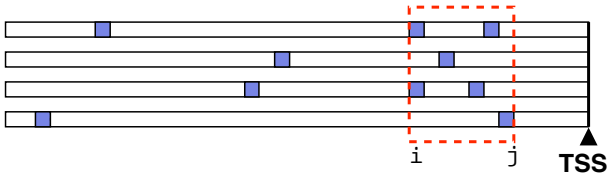


FIG. 3.2: Exemple de fenêtre locale.

Dans le contexte de fenêtre locale, le problème de sur-représentation devient : trouver les fenêtres $[i, j]$ pour lesquelles il existe un motif sur-représenté. C'est que nous appelons le problème de *sur-représentation locale*. Par défaut, toutes les méthodes de sur-représentation que nous avons présentées dans la section 2.6 sont des méthodes globales.

3.2 Prise en compte de différentes espèces

En section 2.5, nous avons vu que la plupart des méthodes tirant partie de la génomique comparative nécessitent deux étapes : la recherche de couples de séquences orthologues chez des espèces à distance phylogénétique suffisante, et la recherche de régions non codantes conservées. Nous allons détailler successivement les contraintes et les limitations imposées par ces deux étapes préalables à la recherche de sites de fixation.

3.2.1 Recherche de gènes orthologues

Plusieurs contraintes interviennent lorsque l'on cherche à utiliser des gènes orthologues. Une première limite concerne la disponibilité des données. En effet, pour un gène d'une espèce donnée, il n'existe pas toujours de gène orthologue disponible et de plus il n'est pas toujours possible d'automatiser la récupération d'orthologues pour un ensemble de gènes donné.

Une autre limitation inhérente à la disponibilité de séquences orthologues concerne le choix des espèces à utiliser. Comme nous l'avons détaillé dans le chapitre précédent la distance phylogénétique entre les espèces sélectionnées joue un rôle important (section 2.5.3). Les

espèces les plus distantes vont conditionner la nature des éléments qui peuvent être trouvés [21, 77].

3.2.2 Limite de l'approche par alignement

Une fois les séquences orthologues sélectionnées, il faut identifier les zones fortement conservées. Cela se fait généralement par alignement de couples d'orthologues. Les stratégies développées dans rVista [49] ou ConSite [44] ou plus récemment oPOSSUM [34] se basent sur ce moyen de filtrage. Dans ce cas, seules les régions pour lesquelles la qualité des alignements est suffisante, sont analysées.

Ce type d'approche fournit de bons résultats lorsque les signaux sont fortement conservés entre les organismes considérés. Malheureusement, cela rend difficile la prédiction de signaux de régulation situés dans des régions faiblement ou partiellement conservées. L'étape d'alignement préalable nécessaire à ces approches se fait au détriment de la sensibilité de la méthode. À titre d'illustration, nous avons effectué des prédictions de sites de fixation à l'aide du logiciel ConSite [44], sur un couple de séquences orthologues humain-souris pour lesquels les mécanismes d'expression sont connus. Il s'agit de deux gènes spécifiques au muscle du squelette, `NM_001927` et `NM_010043`, sur lesquels nous donnons plus de détails dans la section 4.2.1. La figure 3.3 donne les résultats.

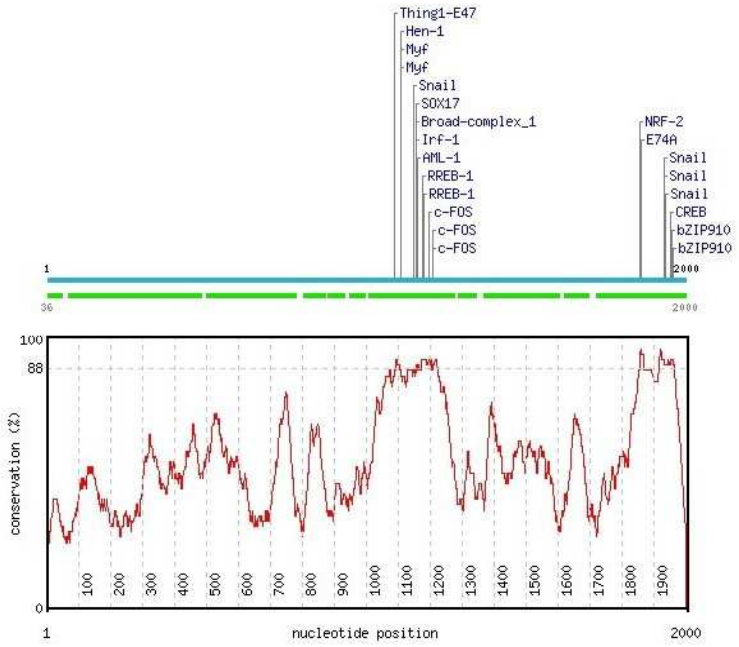


FIG. 3.3: Prédictions faites par CONSITE pour le couple de séquences `NM_001927` et `NM_010043`.

Si l'on considère, le promoteur du gène humain `MN_001927`, trois sites de fixation vérifiés expérimentalement sont connus : MEF2, avec un site à la position -887 et un à la position -76, et Myf avec un site à la position -927. En utilisant la séquence murine orthologue `MN_010043`,

le logiciel CONSITE fournit les prédictions présentées dans la figure 3.3. Deux sites sont détectés pour le facteur Myf dans la région $[-1000 : -900]$ ce qui correspond à un des deux sites connus pour Myf. Malheureusement aucun site n'est prédit pour MEF2 et le site Myf situé dans la région $[-100 : 0]$ n'est pas non plus détecté. Ceci est résumé dans le tableau suivant 3.2 :

Facteur	**Sites connus**	**Sites détectés**
MEF2	-887, -76	
Myf	-927	-900

TAB. 3.2: Positions des sites détectés et des sites connus pour le gène `MN_001927` et les facteurs spécifiques au muscle (relativement au site d'initiation de la transcription).

Sur cet exemple, l'alignement contraint la qualité des régions qui sont désignées comme conservées.

Par ailleurs, Sidow [77] indique par exemple que l'alignement de séquences ne qualifie pas "pleinement" la conservation mais permet simplement l'inférence limitée aux éléments conservés les plus contraints.

D'autre part, l'alignement n'est en général pas suffisant pour pouvoir capturer de courts segments conservés, correspondants aux éléments régulateurs, lorsque les régions avoisinantes le sont moins. Une justification à cette limitation est par exemple apportée par Moses et co-auteurs [51], qui ont montré expérimentalement que chez la levure *Saccharomyces cerevisiae* les portions de séquences correspondant aux sites de fixation de facteurs évoluaient moins vite que les régions avoisinantes.

Pour répondre aux limitations induites par l'alignement, nous proposons une approche permettant de tirer parti, de manière plus souple, de la conservation entre espèces. Au lieu de rechercher des éléments régulateurs sur-représentés uniquement dans les régions fortement conservées, nous allons rechercher des sur-représentations locales. Cela est rendu possible par l'utilisation de fenêtres positionnelles. Il est dans ce cas, seulement nécessaire de pouvoir combiner plusieurs espèces, donc plusieurs modèles de fond adaptés au problème.

3.3 Modèle hétérogène pour le comptage des sites

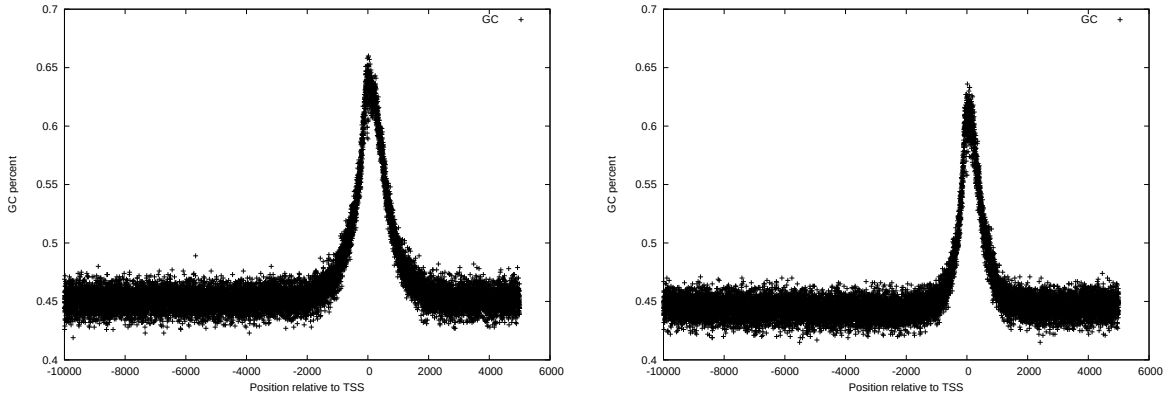
Pour pouvoir rechercher des éléments régulateurs sur-représentés tout en prenant en compte les deux formes de conservation que nous avons introduites précédemment, à savoir la conservation spatiale et la conservation entre espèces, il faut définir une modélisation adaptée. Nous considérons ici, un modèle pour le comptage définissant le nombre d'occurrences que l'on s'attend à trouver par défaut dans une fenêtre $[i, j]$ pour un ensemble de séquences (provenant éventuellement de différentes espèces). Ce modèle doit prendre en compte la variabilité du contexte à deux niveaux :

- le long des séquences elles-mêmes (intra-séquence) ;
- et entre les séquences (inter-séquences).

Nous détaillons successivement ces deux niveaux de variabilité.

3.3.1 Prise en compte de la variabilité intra-séquence

La composition du génome est connue comme étant fortement hétérogène. On peut par exemple observer une augmentation importante de la composition en dinucléotides GC (îlots CpG) dans les régions en amont du site d'initiation de la transcription. La figure 3.4 montre la variation mesurée de la composition en nucléotides G et C d'un ensemble de 4 000 séquences pour deux organismes : l'humain et la souris. En analysant cette figure, on peut constater la singularité de la région proximale. Alors que la composition est AT riche pour les régions éloignées du site d'initiation (plus de 200 bases), la composition devient riche en G et C dans la région proximale avec un pic au niveau du site d'initiation. Cette composition variable le long des séquences doit être prise en compte pour la détection de motifs sur-représentés. Par exemple, l'évolution de la densité le long des séquences du nombre de sites prédits pour trois matrices lorsque le modèle est homogène est reporté dans la figure 3.5. On peut constater un fléchissement ou une augmentation de la densité de sites prédits dans la région proximale, suivant le contenu G et C de la matrice considérée.



Promoteurs extraits du génome humain

Promoteurs extraits du génome de la souris

FIG. 3.4: Variation de la composition en nucléotides G et C des séquences promotrices.

Les méthodes de "sur-représentation" présentées en 2.6 se basent sur un modèle de fond homogène : un même modèle le long des séquences. Cette approche est correcte tant que l'on travaille sur des régions courtes. Pour pouvoir travailler sur plusieurs fenêtres, il faut disposer de modèles de fond adéquats. Une des façons, de construire un modèle adapté à la variation de la composition des séquences, est de considérer une succession de modèles uniformes, appropriés aux régions décrites. Deux étapes sont nécessaires pour construire ce type de modèle : le découpage des séquences de référence, puis la construction de modèles sur les segments de séquences.

3.3.2 Prise en compte de la variabilité inter-séquences

Le second niveau de variabilité, à prendre en compte dans le modèle de fond, concerne les variations rencontrées entre les séquences. Cette variation, d'une séquence à l'autre, peut être induite par les régions du génome considérées (région riche en nucléotides G et C par exemple) ou bien encore par les différences entre génomes eux-mêmes. En effet, la composition

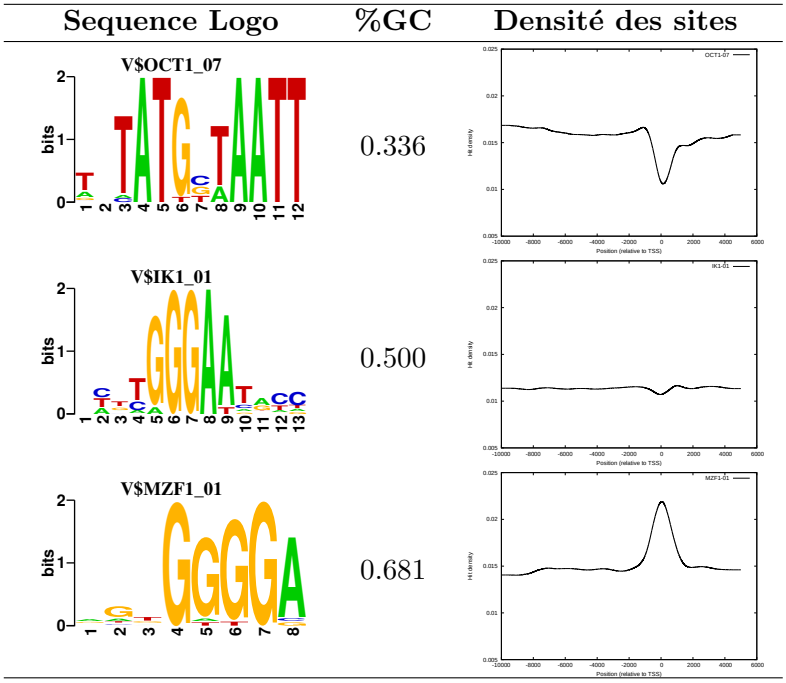


FIG. 3.5: Variation de la densité des sites prédits lorsque le modèle est homogène.

d'un génome est fortement variable d'une espèce à l'autre. Par exemple, le génome humain comporte un pourcentage en G et C de 39.7%, alors que cette proportion est de 42% chez le poulet et 51.7% chez la levure. Pour pouvoir traiter simultanément différentes espèces, le modèle de fond doit permettre de tenir compte de ces différences. Ceci est rendu possible par la combinaison de modèles. Un modèle de fond spécifique est associé à chaque séquence du jeu de données suivant l'organisme duquel elle provient. Les différents modèles sont ensuite combinés afin de représenter la composition de l'échantillon étudié.

3.4 Significativité d'une fenêtre locale

Nous avons défini ce qu'était une fenêtre locale et avons montré comment prendre en compte la variabilité des séquences dans le modèle de fond. Au vu de ces éléments, nous présentons comment évaluer la significativité d'une fenêtre à travers sa P-valeur en construisant une loi de comptage adaptée à ces modèles. Nous décrivons tout d'abord, comment il est possible d'approximer cette loi dans une fenêtre locale par une loi de Poisson et discutons de la pertinence de cette approximation. Ensuite, nous présentons comment évaluer la P-valeur d'une fenêtre locale à partir de cette approximation. Enfin nous expliquons comment incorporer le score individuel de chaque occurrence dans le calcul de la P-valeur, et discutons des avantages et des limites de cet apport.

3.4.1 Distribution statistique du comptage des sites potentiels

Pour analyser la loi de comptage des sites d'un facteur de transcription, nous faisons deux hypothèses sur les sites potentiels associés à une matrice. La première est que ces sites sont

rares. Cette première hypothèse est légitime car les matrices sont prévues pour être utilisées avec des seuils élevés, correspondant à des fréquences d'occurrences faibles. La seconde est que ces sites sont non-chevauchants. Ce qui est le cas lorsque l'on ne considère que la première occurrence pour un train d'occurrences. Dans ce cas, nous proposons d'approximer la distribution du comptage par une distribution de Poisson [69]. Si l'on note X_{ij} la variable aléatoire qui décrit le comptage dans la fenêtre $[i, j]$ et λ_{ij} le nombre d'occurrences attendues dans la fenêtre $[i, j]$, la probabilité d'observer k occurrences dans $[i, j]$ s'écrit :

$$P(X_{ij} = k) = \frac{\lambda_{ij}^k e^{-\lambda_{ij}}}{k!}$$

Afin de valider cette approximation, nous avons évalué l'écart entre ce modèle théorique et différentes observations sur des jeux de données. Le Quantile-Quantile plot du comptage des occurrences obtenues pour la matrice TRANSFAC `CREL_01` est donné dans la figure 3.6. On constate sur cet exemple une bonne adéquation entre l'observation et l'approximation utilisée.

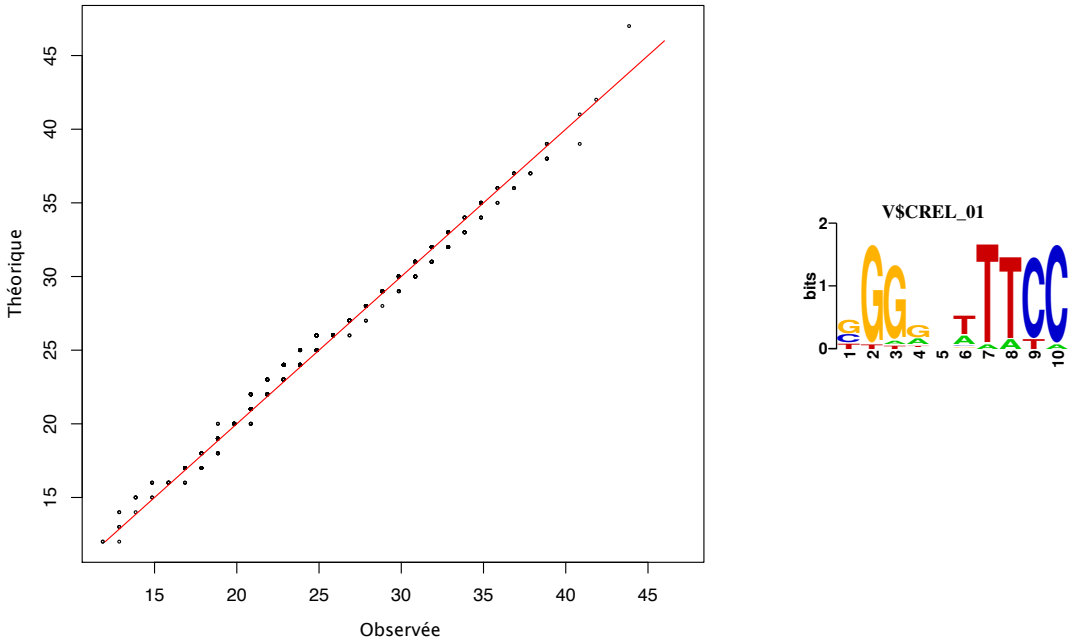


FIG. 3.6: Quantile-Quantile plot entre la distribution de comptage observée et approximée par une loi de Poisson pour la matrice `CREL_01`.

On peut se demander de manière plus systématique si cette approximation est adaptée pour l'ensemble des matrices TRANSFAC et JASPAR. Il existe différents moyens de mesurer systématiquement l'écart entre une distribution observée et une distribution théorique. On citera par exemple le test de Kolmogorov ou encore le test d'ajustement du χ^2. Nous allons considérer ici une des mesures les plus conventionnelles : le test d'ajustement du χ^2. En

formulant l'hypothèse H0 selon laquelle la distribution du comptage des occurrences pour une matrice donnée dans une fenêtre $[i, j]$ suit une distribution de Poisson, ce test permet de répondre à la question : l'hypothèse H0 est-elle vraisemblable ? Il est ainsi possible de mesurer pour l'ensemble des matrices la qualité de cette adéquation. Habituellement, ce test est réalisé en fixant un seuil α correspondant au risque de rejeter à tort H0. Le test du χ^2 est réalisé en comparant la distribution empirique du comptage mesuré sur un large ensemble de séquences à la distribution approximée par une loi de Poisson. Pour obtenir la distribution empirique du comptage, nous avons utilisé un jeu de 1 000 gènes humains pris dans l'ensemble des gènes annotés RefSeq [65] disponibles dans le projet Génome Browser de l'UCSC [39]. Ensuite, en utilisant l'ensemble des modèles de sites (matrices) de vertébrés disponibles dans la version publique des bases de données JASPAR et TRANSFAC (79 et 243 matrices), nous avons recherché de manière exhaustive, dans une région donnée, l'ensemble des sites potentiels non chevauchants sur les deux brins de notre ensemble de séquences. Les résultats obtenus pour les seuils de significativité α : 0.1, 0.05 et 0.01 sont donnés dans la table 3.3. Ce tableau peut se lire de la manière suivante : pour un seuil α fixé par exemple à 0.1 (10%), idéalement si l'ensemble des matrices suivaient une loi de Poisson, on s'attendrait à trouver (90%) des matrices dans cette catégorie (72% ou 68% dans notre cas). Pour un seuil fixé à 5%, une large proportion des matrices passent le test (80% pour les matrices issues de JASPAR).

Jeu de matrices	$\alpha > 0.1$	$\alpha > 0.05$	$\alpha > 0.01$
JASPAR	72%	80%	87%
TRANSFAC (public)	68%	74%	83%

Tab. 3.3: Ajustement de la loi de Poisson pour la loi de comptage des sites potentiels.

Une première constatation favorable est qu'en fixant un seuil α à 10%, pour 72% des matrices de JASPAR et 68% des matrices de TRANSFAC, on ne rejette pas l'hypothèse H0 correspondant à la distribution de Poisson. On constate d'autre part que pour une proportion non négligeable des matrices (23% pour JASPAR, 27% pour TRANSFAC), l'approximation de la loi de comptage des sites potentiels par une loi de Poisson doit être rejetée en fixant un seuil relativement faible (1%) pour α.

3.4.2 P-valeur d'une fenêtre

Si l'on désigne par k le nombre de sites potentiels présents dans la fenêtre $[i, j]$, il est possible de définir la P-valeur de cette fenêtre comme la probabilité d'observer une région au moins aussi dense, c'est-à-dire contenant au moins k sites. Pour pouvoir calculer explicitement cette probabilité, nous utilisons l'approximation de Poisson de la section précédente. Nous présentons dans un premier temps un calcul de la P-valeur basé sur une approximation de Poisson dans le cadre d'un *ensemble homogène de séquence*, c'est-à-dire utilisant un même modèle (mais variable le long des séquences) pour toutes les séquences. Dans un deuxième temps, afin de permettre de prendre en compte l'hétérogénéité entre les séquences (différentes espèces par exemple), nous présentons une extension du calcul de la P-valeur dans un cadre *ensemble hétérogène de séquences*, c'est-à-dire utilisant un modèle différent pour chaque séquence du jeu de données.

P-valeur pour un ensemble homogène de séquences

Si l'on note X_{ij} la variable aléatoire qui décrit le comptage dans la fenêtre $[i, j]$, la probabilité $P(X_{ij} \geq k)$ d'observer au moins k occurrences dans cette fenêtre s'exprime de la manière suivante :

$$P(X_{ij} \geq k) = 1 - \sum_{z=0}^{k-1} \frac{(\delta n \lambda)^z}{z!} e^{-n\delta\lambda}$$

où λ correspond à la probabilité d'obtenir une occurrence à une position dans la fenêtre $[i, j]$ dans une séquence, δ la largeur de la fenêtre et n le nombre de séquences. Ce paramètre peut être obtenu à partir du modèle de fond associé aux séquences (valeur de l'espérance locale).

P-valeur pour un ensemble hétérogène de séquences

Nous présentons maintenant un moyen d'étendre le calcul de la P-valeur pour un ensemble hétérogène de séquences. Pour cela, nous allons considérer la distribution de la somme des comptages individuels où chacune des séquences apporte une contribution basée sur un modèle différent. Dans ce cas, les distributions sont prises en compte de manière indépendante pour chacune des séquences. La distribution du comptage des occurrences pour la $x^{\text{ième}}$ séquence est approximée par une loi de Poisson de paramètre λ^x. Ce paramètre peut se calculer à partir du modèle de fond associé à la séquence x (l'espérance locale). Lorsque l'on considère la distribution de la somme des comptages, la P-valeur peut s'écrire de la manière suivante :

$$P(X_{ij} \geq k) = 1 - \sum_{z=0}^{k-1} \frac{(\delta \sum_{x=1}^{n} \lambda^x)^z}{z!} e^{-\delta \sum_{x=1}^{n} \lambda^x}$$

3.4.3 Prise en compte du score des sites potentiels dans le calcul de la P-valeur

Jusqu'à présent, pour estimer la qualité d'une fenêtre, seul le nombre de sites potentiels a été utilisé. La qualité individuelle de chaque site, c'est-à-dire son score n'a pas été prise en compte. Il peut être intéressant de se demander si la prise en compte du score des sites peut permettre d'améliorer la qualité de l'évaluation des fenêtres.

Pour répondre à cette question, nous avons estimé la distribution du score cumulé d'un ensemble de sites. Pour une matrice donnée, cette distribution peut être construite sur la base suivante : à une fenêtre $[i, j]$, nous associons le nombre de sites X_{ij} et le score cumulé Y_{ij}. En considérant la distribution du score d'un site indépendante de la loi de comptage, le score cumulé Y_{ij} d'une fenêtre peut s'exprimer comme la composition de la distribution du comptage avec celle du score. Cette composition peut s'écrire de la façon suivante :

$$Y_{ij} = \sum_{z=1}^{X_{ij}} Y_z$$

La P-valeur peut se définir comme la probabilité d'obtenir dans la fenêtre $[i, j]$ un score cumulé supérieur à y. Cette probabilité se calcule par sommation sur toutes les valeurs de

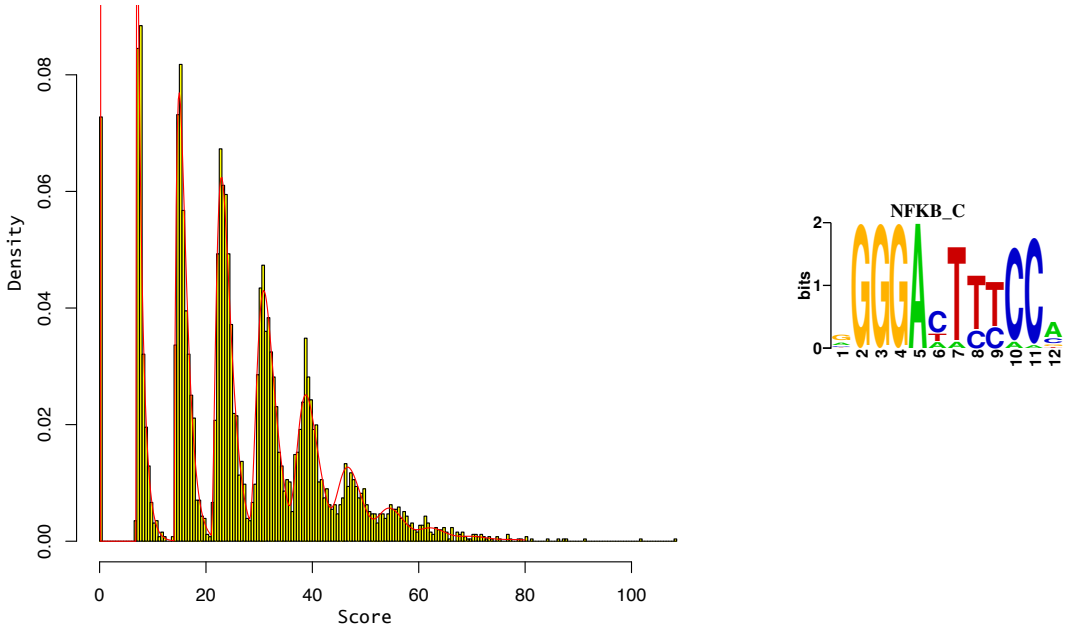


FIG. 3.7: Exemple de distribution du score cumulé pour la matrice NFKB_C.

comptage k possibles. Ceci se note de la manière suivante :

$$P(Y_{ij} \geq y) = \sum_{k=0}^{\infty} P\left[\sum_{z=1}^{k} Y^z \geq y\right] P(X_{ij} = k)$$

Pour pouvoir calculer explicitement la P-valeur à l'aide de la formulation précédente, il faut définir précisément les deux termes (Y et X) de cette expression. En ce qui concerne la distribution du comptage des sites, nous allons comme dans les sections précédentes approximer cette distribution par une loi de Poisson. Nous avons vu également dans la section 2.4 comment approximer la distribution du score d'une occurrence de manière analytique et exacte. L'utilisation d'une distribution discrète exacte du score est coûteuse en calcul, ce qui rend son utilisation peu adaptée à sa composition avec la loi de comptage. Notre objectif n'étant ici que de tester la prise en compte du score de chaque occurrence dans l'évaluateur statistique de significativité, nous allons considérer ici des matrices ayant de bonnes propriétés permettant d'approximer leur distribution de score par une distribution exponentielle (section 2.4).

À partir de la distribution du score d'un site de seuil τ, il est possible de définir la distribution de la somme des scores de k sites à l'aide de la distribution Gamma de paramètre $y - k\tau, k, \frac{1}{\sigma}$. La P-valeur d'une fenêtre $[i, j]$ de score y peut se calculer en composant une distribution de Poisson avec la distribution exponentielle. Ce qui peut se noter de la manière suivante :

$$P(Y \geq y) = \sum_{k=1}^{\infty} P_{Poisson}(X = k, \lambda) P_{gamma}(Y \geq y - k\tau, k, \frac{1}{\sigma})$$

En explicitant les deux termes de cette expression ($P_{Poisson}(X = k, \lambda)$ et $P_{gamma}(Y \geq y - k\tau, n, \frac{1}{\sigma})$), la P-valeur prend la forme suivante :

$$P(Y \geq y) = \sum_{k=1}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \int_{z=y}^{\infty} \frac{(z - k\tau)^{k-1} e^{-\sigma z}}{\Gamma(k) \frac{1}{\sigma^k}}$$

où Γ représente la fonction Gamma.

La figure 3.7 donne un exemple de distribution observée et calculée pour la matrice `NFKB_C`.

Gain apporté par la prise en compte du score des sites

Maintenant que nous disposons d'un moyen d'évaluer la significativité d'une fenêtre en prenant en compte le score des occurrences, on peut se poser la question suivante : la prise en compte du score permet-elle d'augmenter la discrimination dans la prédiction des éléments sur-représentés ?

Pour tenter de répondre à cette question, nous avons mesuré l'écart entre le score observé et attendu, pour un jeu de données possédant des motifs régulateurs communs expérimentalement vérifiés. Le score attendu a été estimé à partir d'un ensemble de gènes tirés aléatoirement dans le génome. Nous avons choisi pour cela un jeu de gènes cibles du facteur NF-κB. Ce jeu présente l'avantage d'être bien connu et de posséder des éléments régulateurs pour lesquels on dispose de motifs avec de bonnes propriétés. Ce qui permet de disposer d'un cadre où les approximations faites sur la distribution du score et du nombre d'occurrences sont valides. Les résultats obtenus sont donnés dans le tableau 3.4.

Jeu NF-κB	**Échantillon aléatoire**
α=1.0275	α=1.0

Tab. 3.4: Gain apporté par la prise en compte du score.

On constate, d'après ces mesures, que les scores des fenêtres comportant des sites valides (jeu NF-κB) diffèrent peu des scores des fenêtres avec des éléments qui ne le sont pas (2.75 en moyenne). Au vu du gain apporté et du coût de la mise en œuvre (en terme de calcul) imposé par la prise en compte du score, il ne semble pas pertinent d'utiliser le score total des occurrences dans le calcul de la significativité d'une fenêtre.

Chapitre 4

TFM-Explorer

Sommaire

Nous avons présenté dans le chapitre précédent la notion de sur-représentation locale et nous avons montré comment mesurer la significativité d'une fenêtre locale tout en prenant en compte la variabilité des séquences étudiées. Nous décrivons maintenant un algorithme efficace pour l'identification de fenêtres locales (section 4.1). Cet algorithme est mis en œuvre dans le logiciel TFM-Explorer, qui intègre des modèles pour les génomes de l'homme, de la souris et du rat. En section 4.2, nous présentons les résultats de TFM-Explorer sur trois jeux de données, résultats que nous comparons à ceux obtenus avec les logiciels TOUCAN, OTFBS et oPOSSUM, décrits au chapitre 2. Enfin, nous concluons en présentant une piste pour étendre l'approche locale à l'inférence de motifs (section 4.3).

4.1 Recherche de fenêtres avec sur-représentations locales

Le programme TFM-Explorer propose de mettre en œuvre une stratégie répondant au problème de la recherche de fenêtres dans lesquelles les occurrences d'une matrice donnée

sont sur-représentées. Après avoir replacé la méthode dans le schéma général de la recherche de motifs sur-représentés localement, nous détaillons, dans cette section, comment rechercher efficacement des fenêtres locales. Nous montrons ensuite comment corriger la P-valeur pour prendre en compte les tests multiples. Enfin, nous présentons quelques éléments concernant l'implémentation de la méthode.

4.1.1 Schéma général

La stratégie développée dans TFM-Explorer s'articule en trois étapes distinctes (figure 4.1) :

étape 1 : recherche de tous les sites potentiels dans les séquences à l'aide d'un ensemble de matrices (par exemple toutes les matrices de vertébrés de JASPAR).

étape 2 : recherche de fenêtres denses en sites potentiels.

étape 3 : évaluation de la significativité des fenêtres détectées. Pour chacune des ces fenêtres une P-valeur est estimée.

La première étape de la méthode peut être considérée comme indépendante des deux suivantes. En effet, il n'est pas nécessaire de connaître comment les sites potentiels sont prédits pour effectuer la recherche de sur-représentations. Il est possible, dans ce cas, d'utiliser différentes stratégies pour rechercher les sites candidats avant de rechercher des sur-représentations. Nous nous plaçons ici dans un modèle de comptage, c'est-à-dire en ne considérant que le nombre d'occurrences d'un motif. Nous avons vu, dans le chapitre précédent, comment rechercher des sites potentiels et comment mesurer la significativité d'une fenêtre locale. Nous allons maintenant nous intéresser à la recherche de ce type de fenêtre.

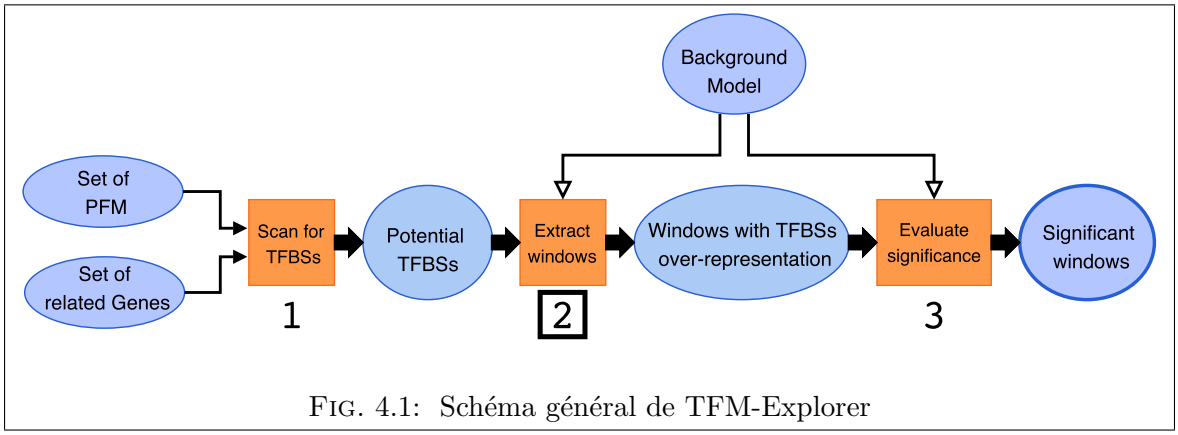


FIG. 4.1: Schéma général de TFM-Explorer

4.1.2 Heuristique pour la recherche de sur-représentations locales

L'approche locale développée ici rend possible la recherche d'éléments régulateurs sans connaissance a priori, ni sur la taille, ni sur la localisation de ces éléments. Cela permet par exemple de rechercher à la fois des éléments régulateurs proximaux et distaux.

Pour rechercher ces fenêtres, plusieurs stratégies sont possibles. La solution la plus naturelle est la recherche exhaustive. Si on note k le nombre total de sites prédits pour l'ensemble des séquences, la stratégie consiste à énumérer toutes les $k^2/2$ fenêtres possibles. Malheureusement, la complexité quadratique de cette approche la rend inadaptée à un traitement efficace sur de longues séquences. Pour réduire cette complexité, une des solutions les plus courantes est de balayer les séquences en utilisant une fenêtre glissante de taille donnée. Cette approche présente le désavantage de fixer a priori une taille de fenêtre de balayage. Au vu de la forte variabilité des séquences dans la région proximale et du comportement différent des facteurs de transcription, il semble important de pouvoir analyser les sur-représentations de manière différenciée suivant la position et la composition de la fenêtre considérée.

Nous présentons ici une méthode heuristique de recherche de fenêtres locales sans contrainte sur la taille ou la position de ces fenêtres. Cette méthode s'articule autour d'un système de score local permettant de mesurer la densité relative des sites à chaque position du référentiel commun aux séquences, et ainsi d'extraire des fenêtres denses en fonction de l'évolution de ce score. La méthode a ainsi la capacité à pouvoir extraire des régions sur-représentées sans connaissance a priori ni sur la taille ni sur la localisation des régions recherchées. Un deuxième point fort de la méthode est de permettre la prise en compte de la variabilité des séquences aussi bien le long des séquences elles-mêmes, qu'entre les séquences.

Score d'une position

La définition du score de balayage passe par la définition d'un score, position par position, sur l'ensemble des séquences alignées. Nous définissons, de manière classique, un log score s_i à la position i comme le rapport de la probabilité d'observer k sites dans le modèle cible sur la probabilité de cette même observation dans un modèle de fond neutre. Le modèle cible t (resp. neutre b) doit permettre de mesurer si le nombre de sites observés à la position i (dans le référentiel commun aux séquences) est probable dans des conditions exceptionnelles (resp. neutres). Le log score s_i peut se formuler de la manière suivante :

$$s_i = \log \frac{P_t(k_i)}{P_b(k_i)}$$

où P_t représente la probabilité d'observer k sites à la position i dans le modèle cible et P_b la probabilité de faire la même observation à la position i dans le modèle de référence.

Ce score est, de par sa construction, positif lorsque l'observation est plus probable dans le modèle cible que dans le modèle de fond. Il convient maintenant de définir plus précisément ces deux modèles et de proposer un moyen de mesurer la probabilité d'une observation dans ces modèles.

Une observation à la position i correspond uniquement à un nombre k de sites. Pour pouvoir mesurer cette probabilité il faut établir une approximation statistique du comptage sous chacune de ces conditions. Le modèle de fond modélise la loi de comptage attendue en chaque position de la séquence pour le motif considéré. Il peut par exemple être construit en approximant la loi de comptage par une loi analytique.

Modèle cible

Contrairement au modèle de fond, le modèle cible doit modéliser le comptage dans le cadre de comptages exceptionnels : les sur-représentations. La définition d'un modèle cible repose sur la définition d'une sur-représentation. En d'autres termes à partir de quel niveau doit-on considérer qu'un motif est sur-représenté ? Une solution simple pour répondre à ce problème est de définir le modèle cible par rapport au modèle de fond. Il est possible de définir le modèle cible de manière différentielle par rapport au modèle de fond en utilisant par exemple un ratio fixe entre l'espérance du comptage dans le modèle cible et le modèle de fond. En notant μ_i l'espérance du comptage dans le modèle de référence à la position i, l'espérance dans le modèle cible peut se définir comme $\lambda_i = \alpha\mu_i$ où α est un ratio supérieur à 1.

Modèle de score pour un ensemble homogène de séquences

Pour approximer la distribution du comptage des occurrences à la position i, nous utilisons, comme en section 3.4.2, une loi de Poisson. Dans ce cas, si on note μ_i l'espérance de la loi de comptage pour le modèle de fond et λ_i l'espérance dans le modèle cible à la position i, le score local exprimé dans un modèle de Poisson se formule de la manière suivante :

$$s_i = \log\left[\left(\frac{\lambda_i}{\mu_i}\right)^{k_i} e^{\mu_i - \lambda_i}\right]$$

Modèle de score pour un ensemble hétérogène de séquences

Une limitation importante des approches précédentes est la prise en compte d'un modèle de fond unique pour toutes les séquences. L'utilisation d'un modèle de fond unique restreint les possibilités d'analyse qui peuvent être faites à un ensemble de séquences provenant du même organisme et qui doivent de plus être localisées dans des régions "équivalentes" des gènes. Afin de prendre en compte l'hétérogénéité des séquences nous allons considérer non plus une distribution unique mais un ensemble de distributions. En posant λ_i^n (resp. μ_i^n) l'espérance attendue du nombre d'occurrences à la position i pour la séquence x dans le modèle cible (resp. dans le modèle de fond), et en considérant les séquences indépendantes, le score s_i s'écrit :

$$s_i = \log\left[\left(\frac{\sum_{x=1}^{n} \lambda_i^x}{\sum_{x=1}^{n} \mu_i^x}\right)^{k_i} e^{\left(\sum_{x=1}^{n} \mu_i^x - \sum_{x=1}^{n} \lambda_i^x\right)}\right]$$

Score total

Le score s_i permet de mesurer à une position donnée la densité en occurrences du jeu de séquences par rapport au modèle de fond. Afin de pouvoir extraire des fenêtres "denses" nous définissons un score total S_i. Ce score possède les deux propriétés suivantes : il est additif (log score) et est minoré par zéro. Il représente l'évolution de la densité en occurrences le long des séquences étudiées. Le score S_i peut se définir par la récurrence suivante :

$$S_i = \max \begin{cases} S_{i-1} + s_i \\ 0 \end{cases}$$

Ce score est calculé de manière incrémentale sur la longueur des séquences dans le référentiel commun aux séquences. D'après la définition précédente, le score S_i évolue de manière croissante lorsque le modèle cible est plus probable que le modèle de fond et de manière décroissante dans le cas contraire. Une fois le score calculé pour toutes les positions i, se pose alors la question de l'extraction des fenêtres (ou régions denses). Il faut pouvoir définir une stratégie d'extraction des fenêtres denses en fonction de l'évolution du score.

Extraction de fenêtres denses

Les fenêtres $[i, j]$ extraites correspondent aux fenêtres délimitées, d'une part, par une position i de score nul ($S_i = 0$) et, d'autre part, par la position j de score S_j maximum précédant la position de score nul suivante. L'extraction de fenêtres peut être réalisée à la volée de manière linéaire par rapport à la longueur des séquences. On peut noter qu'une des conséquences inhérentes à l'utilisation de ce système d'extraction est la production de fenêtres non chevauchantes pour un facteur donné. La figure 4.2 donne un exemple de profil de score et de fenêtres extraites.

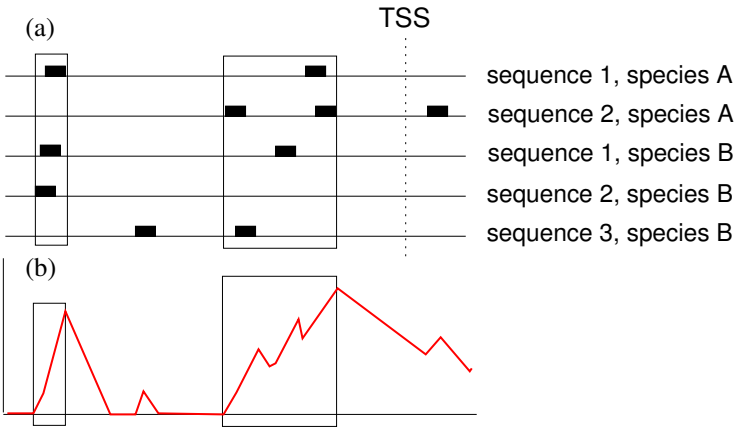


FIG. 4.2: Score total et extraction de fenêtres denses.

4.1.3 Correction de la P-valeur pour le multitest

Pour pouvoir apprécier la qualité d'une P-valeur, dans ce contexte de test multiple, il est nécessaire de la corriger. Dans notre cas, la correction de Bonferroni est délicate à mettre œuvre. En effet la méthode employée pour extraire les fenêtres exclut les fenêtres chevauchantes, elle ne considère qu'un sous ensemble des $\beta L(L-1)/2$ fenêtres potentielles pour les β matrices.

Pour évaluer effectivement la correction à apporter, nous avons mesuré le taux de faux positifs découverts dans des ensembles de séquences tirées aléatoirement dans le génome. Pour

ces jeux de données, on peut considérer qu'il n'existe pas, a priori, de signaux spécifiques à identifier. La méthode de recherche de fenêtres denses peut alors être appliquée à ces jeux considérés comme négatifs. Il est ainsi possible d'estimer, en fonction du taux de faux positifs acceptés le niveau de P-valeur qu'il est possible d'atteindre par hasard. Nous avons ainsi réalisé des prédictions de fenêtres denses pour l'ensemble des matrices de TRANSFAC avec des jeux aléatoires. Ces jeux ont été constitués en sélectionnant aléatoirement dans les génomes de différents organismes (humain et souris) des groupes de 10, 50 et 100 séquences promotrices longues de 2 000 bp. Le pourcentage de tests fournissant une mauvaise prédiction (taux de faux positifs) pour un seuil de P-valeur fixé est donné dans la figure 4.3. D'après cet exemple, on constate que pour un taux de faux positif fixé à 10% le seuil de coupure de la P-valeur s'étend de 1×10^{-6} à 1×10^{-7} pour des jeux de 10 à 100 séquences de 2 000 bp.

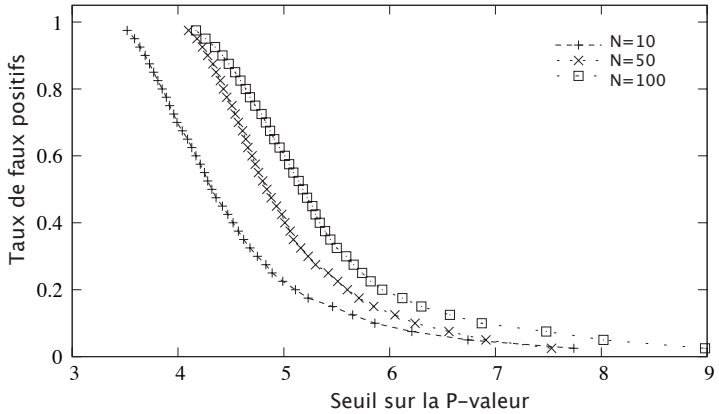


FIG. 4.3: Évolution du taux de faux positifs découverts pour des échantillons aléatoires de 10, 50 et 100 séquences

4.1.4 Réalisation logicielle

La méthode de recherche de sur-représentations locales utilisant des matrices de comptage (section 2.6) a été implémentée dans un logiciel nommé TFM-Explorer (Transcription Factor Matrix Explorer). Une interface à ce logiciel est accessible à partir de l'adresse `http://bioinfo.lifl.fr/TFM-Explorer/`. Cette interface permet à l'utilisateur de rechercher pour un ensemble de séquences et de matrices, les fenêtres possédant des sur-représentations locales par rapport à un modèle de fond construit à partir de différents génomes. La figure 4.4 montre l'interface de saisie. Nous décrivons dans cette section la nature des données que le logiciel accepte en entrée et détaillons les types de données produites en sortie.

Séquences génomiques

Une base de séquences promotrices a été constituée à partir des annotations et génomes disponibles sur le site du Genome Browser de l'UCSC [39]. Il est ainsi possible à partir de l'identifiant d'un gène d'extraire la séquence promotrice de ce gène. La base permet d'utiliser

jusqu'à des séquences de 15 000 bases (10 000 en amont, 5 000 en aval du site d'initiation de la transcription) provenant des dernière versions disponibles des assemblages des génomes de l'homme de la souris et du rat :

- génome humain (hg18) - Mars 2006 ;
- génome de la souris (mm8) - Mars 2006 ;
- génome du rat (rn3) - Juin 2003.

Cela correspond à un ensemble de 24 328 gènes pour le génome humain, 19 343 gènes pour le génome de la souris et 8 314 gènes pour le génome du rat.

Le logiciel accepte en entrée un ensemble de séquences sous l'un des formats suivants :

- un ensemble d'identifiants de gènes au format RefSeq [65] ainsi que la position des régions à extraire relativement au site d'initiation de la transcription ;
- ou un fichier contenant directement des séquences génomiques au format FASTA avec leurs positions relativement au site d'initiation de la transcription.

Modèles de sites

La version actuelle du logiciel propose à l'utilisateur de choisir un ensemble de modèles de sites parmi les modèles matriciels disponibles dans les bases suivantes :

- la version publique de la base TRANSFAC (243 matrices pour les vertébrés) ;
- la dernière version de JASPAR (79 matrices pour les vertébrés).

Sortie du logiciel

Le logiciel fournit en sortie la liste des fenêtres prédites les plus significatives (figure 4.5 bas). A chaque fenêtre sont associés sa position, le facteur impliqué ainsi que sa P-valeur. Il est possible d'obtenir des informations détaillées pour chacune de ces fenêtres :

- un descriptif détaillé de la matrice associée à la fenêtre ;
- des détails statistiques sur les occurrences prédites dans la fenêtre (densité des sites, distribution, ...) ;
- la position des occurrences dans la fenêtre pour chaque séquence.

(figure 4.7)

Le logiciel fournit également la possibilité d'analyser la corrélation entre deux fenêtres prédites (figure 4.6). Un comparatif entre le nombre de sites prédits commun aux fenêtres et le nombre de sites communs que l'on peut espérer trouver, étant donné le profil des deux matrices, est proposé.

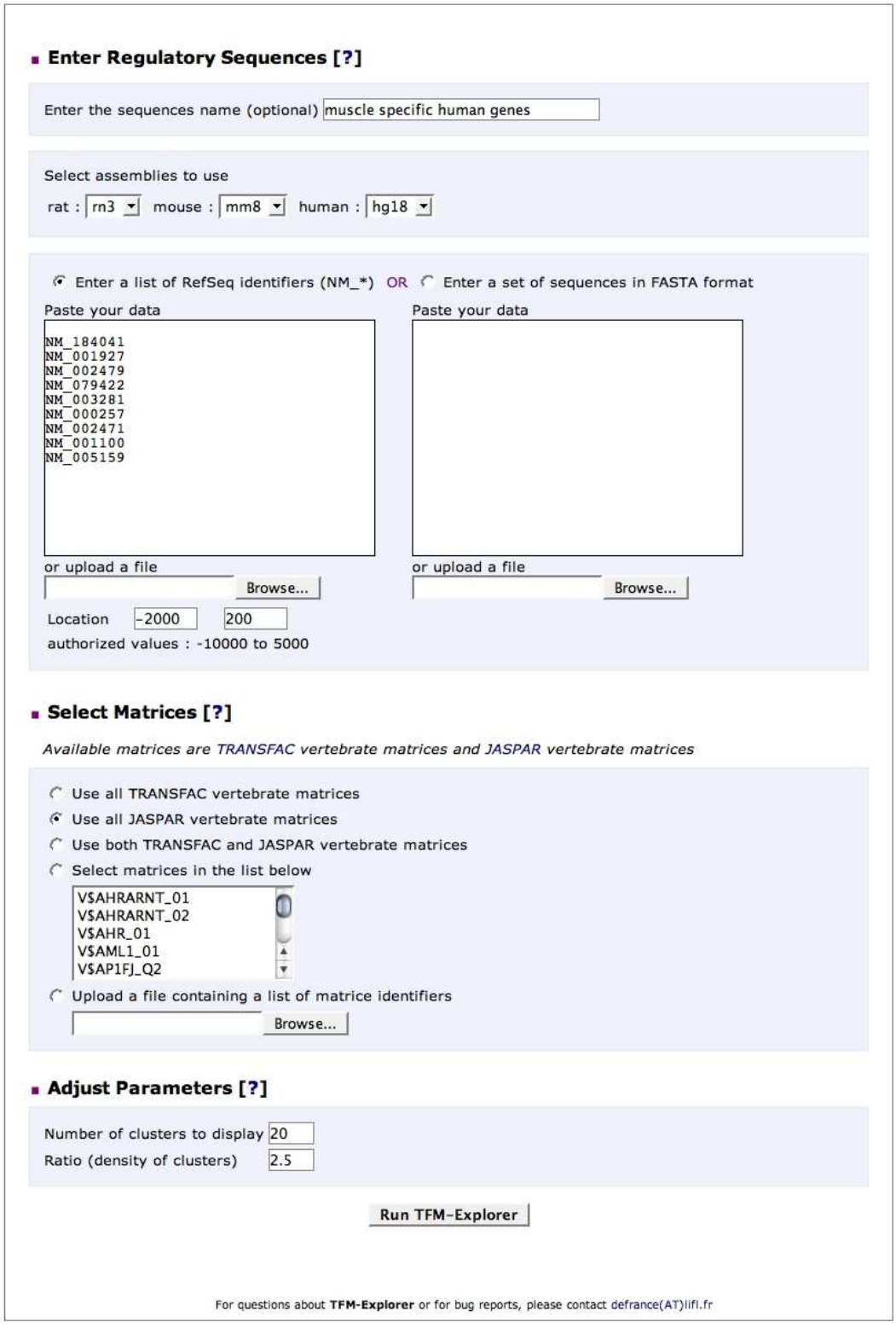
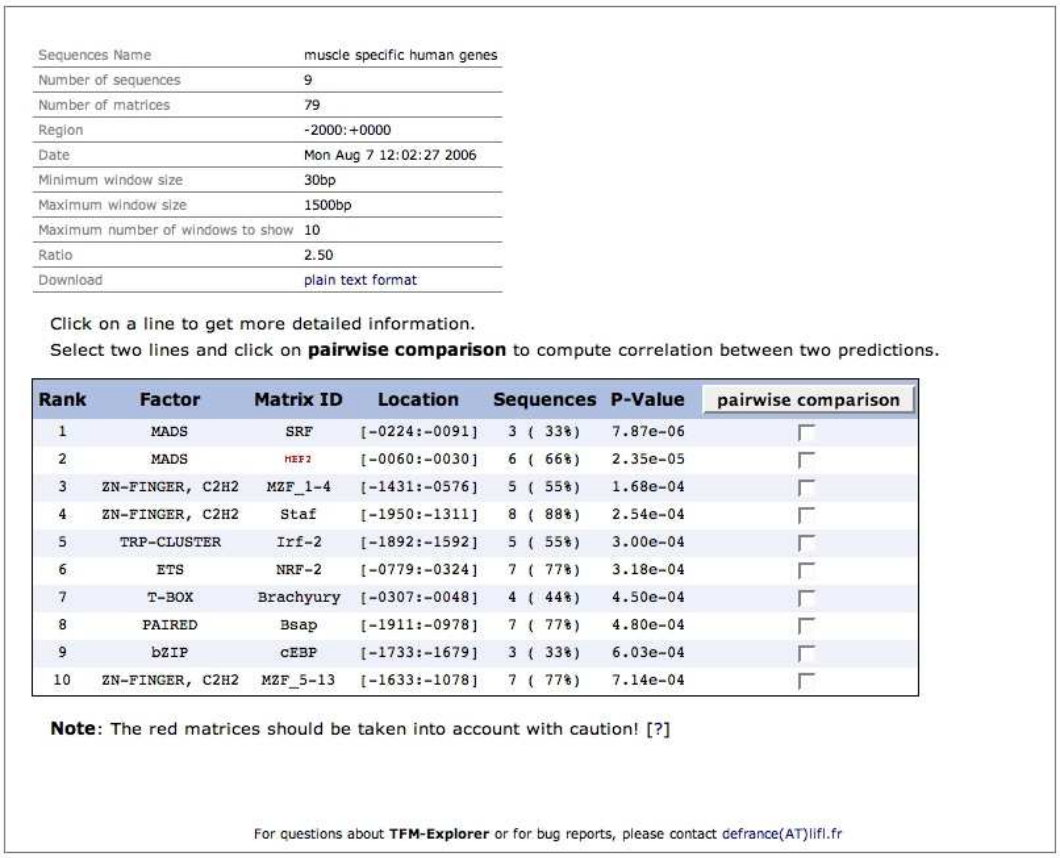


FIG. 4.4: Interface Web pour TFM-Explorer (1).



Sequences Name	muscle specific human genes
Number of sequences	9
Number of matrices	79
Region	-2000:+0000
Date	Mon Aug 7 12:02:27 2006
Minimum window size	30bp
Maximum window size	1500bp
Maximum number of windows to show	10
Ratio	2.50
Download	plain text format

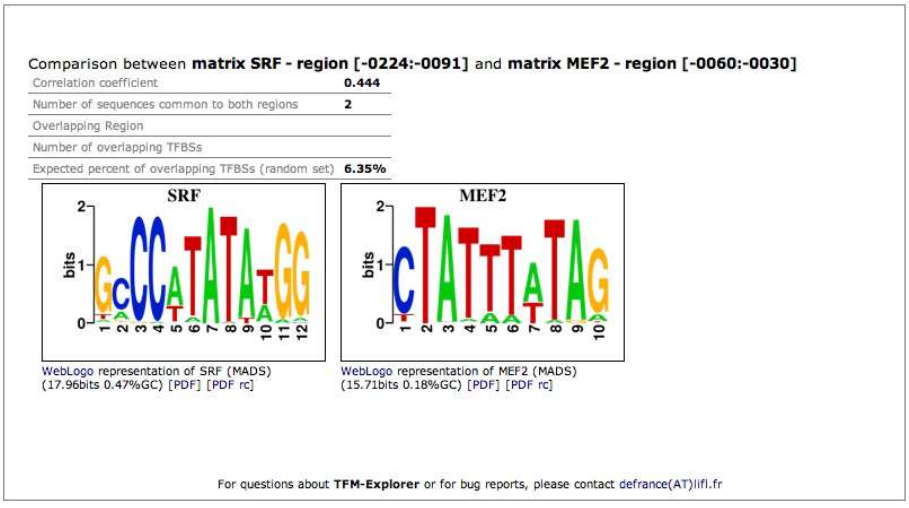
Click on a line to get more detailed information.
Select two lines and click on **pairwise comparison** to compute correlation between two predictions.

Rank	Factor	Matrix ID	Location	Sequences	P-Value	pairwise comparison
1	MADS	SRF	[-0224:-0091]	3 (33%)	7.87e-06	
2	MADS	MEF2	[-0060:-0030]	6 (66%)	2.35e-05	
3	ZN-FINGER, C2H2	MZF_1-4	[-1431:-0576]	5 (55%)	1.68e-04	
4	ZN-FINGER, C2H2	Staf	[-1950:-1311]	8 (88%)	2.54e-04	
5	TRP-CLUSTER	Irf-2	[-1892:-1592]	5 (55%)	3.00e-04	
6	ETS	NRF-2	[-0779:-0324]	7 (77%)	3.18e-04	
7	T-BOX	Brachyury	[-0307:-0048]	4 (44%)	4.50e-04	
8	PAIRED	Bsap	[-1911:-0978]	7 (77%)	4.80e-04	
9	bZIP	cEBP	[-1733:-1679]	3 (33%)	6.03e-04	
10	ZN-FINGER, C2H2	MZF_5-13	[-1633:-1078]	7 (77%)	7.14e-04	

Note: The red matrices should be taken into account with caution! [?]

For questions about **TFM-Explorer** or for bug reports, please contact defrance(AT)lifl.fr

FIG. 4.5: Interface Web pour TFM-Explorer (2).



Comparison between **matrix SRF - region [-0224:-0091]** and **matrix MEF2 - region [-0060:-0030]**

Correlation coefficient	**0.444**
Number of sequences common to both regions	**2**
Overlapping Region	
Number of overlapping TFBSs	
Expected percent of overlapping TFBSs (random set)	**6.35%**

WebLogo representation of SRF (MADS) (17.96bits 0.47%GC) [PDF] [PDF rc]

WebLogo representation of MEF2 (MADS) (15.71bits 0.18%GC) [PDF] [PDF rc]

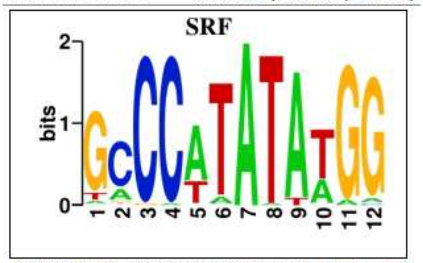
For questions about **TFM-Explorer** or for bug reports, please contact defrance(AT)lifl.fr

FIG. 4.6: Interface Web pour TFM-Explorer (3).

Full results for matrix **SRF** and region **[-0224:-0091]**

Matrix information

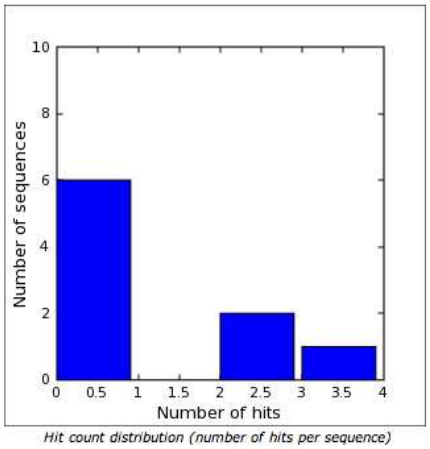
Matrix name	**SRF**
Transcription Factor name	**MADS**
Source	**TRANSFAC OR JASPAR**
Information Content	**17.96 bits (1.50 bits per base)**



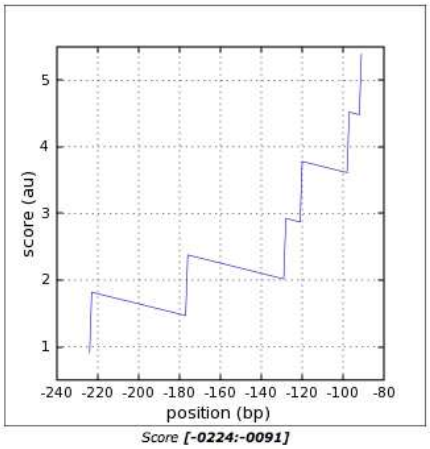
[PDF] WebLogo representation of the matrix **SRF** (**MADS**)

Sequences information

Total number of hits in region **[-0224:-0091]**	**7**
Total number of sequences	**9** (**5.80e-03** TFBS per sequence per bp)
Number of sequences having at least one putative TFBS in region **[-0224:-0091]**	**3** (**1.74e-02** per sequence per bp)
P-value	**7.87e-06**
E-value	**4.76e-01**



Hit count distribution (number of hits per sequence)



*Score **[-0224:-0091]***

GC content

Average GC% for all sequences	**0.53**
Average GC% for the background model	**0.50**
Average GC% for the matrix	**0.47**

List of TFBS for matrix SRF and region [-0224:-0091]

```
NM_079422       01777 -0223 +  3.93 GACCATAAAAGT
NM_079422       01880 -0120 +  2.08 GCCCTAAGAAAG
NM_001100       01776 -0224 +  5.50 CTCCATATACGG
NM_001100       01824 -0176 -  4.70 CCTTCTTTGGTC
NM_001100       01903 -0097 +  9.98 ACCCAAATATGG
NM_005159       01872 -0128 +  2.85 CTCCCTATTTGG
NM_005159       01909 -0091 +  3.81 TTCCTTACATGG
```

For questions about **TFM-Explorer** or for bug reports, please contact defrance(AT)lifl.fr

FIG. 4.7: Interface Web pour TFM-Explorer (4).

4.2 Résultats expérimentaux

L'approche développée dans TFM-Explorer est locale et permet de prendre en compte la conservation entre espèces sans effectuer d'alignements. Une question que l'on peut alors se poser concerne l'apport de ces ajouts du point de vue de la qualité des prédictions. En particulier, on peut se demander comment se comporte cette approche sur des jeux de données biologiques classiques, comparativement aux autres méthodes existantes. Afin de répondre à ces questions, nous présentons dans cette section les résultats sur différents jeux de données, et les comparons avec ceux obtenus avec des méthodes similaires introduites dans la section 2.6.5 : TOUCAN, OTFBS et oPOSSUM. Pour cela trois jeux de données concernant l'humain la souris et le rat pour lesquels les sites de fixation sont connus et vérifiés, ont été utilisés. Ces jeux de données comportent une forte diversité du point de vue de leur taille (de moins d'une dizaine à une centaine de gènes) et de leur composition (des gènes humains seulement et un ensemble mixte humain-souris-rat).

Le premier jeu de données que nous présentons ici est un petit jeu de gènes spécifiques au muscle. Ce jeu, largement employé par la communauté, a servi de point de départ pour vérifier la pertinence de notre approche et pour comparer notre méthode par rapport aux autres logiciels existants.

Le deuxième jeu concerne un ensemble d'une centaine de gènes cibles des facteurs de la famille Rel/NF-κB. Ce jeu a servi de base pour tester l'intérêt de l'approche locale, en particulier, la robustesse de son comportement vis-à-vis de la taille des séquences et des fenêtres à détecter.

Enfin, pour évaluer le comportement des méthodes lorsque plusieurs espèces étaient employées, nous avons évalué les résultats obtenus avec un ensemble mixte de gènes d'histone provenant de différents organismes (humain, rat, souris).

Avant de détailler les résultats obtenus, nous présentons le protocole expérimental concernant la sélection des séquences et l'utilisation des logiciels.

Sélection des séquences

Un jeu de données est constitué d'un ensemble de séquences promotrices repérées par rapport au site d'initiation de la transcription. Nous avons utilisé deux tailles de séquences : 2 000 et 10 000 bp. Dans le premier cas, les séquences correspondent aux 2 000 bp en amont du site d'initiation (région repérée par $[-2000 : 0]$). Dans le deuxième cas, les séquences correspondent aux 5 000 bp en amont du site d'initiation et aux 5 000 bp en aval (région repérée par $[-5000 : 5000]$). Le choix des régions nous a été dictée par la plus grande comptabilité possible entre les logiciels (le logiciel oPOSSUM ne permet que les choix $[-2000 : 0]$, $[-2000 : 2000]$ et $[-5000 : 5000]$).

Pour récupérer les séquences, nous avons utilisé les dernières versions des assemblages de génomes disponibles sur le site du Genome Browser de l'UCSC [39] : humain (hg18) - Mars 2006 ; souris (mm8) - Mars 2006 ; du rat (rn3) - Juin 2003.

Pour chacun des jeux utilisés, nous disposons de facteurs et de sites de fixation connus comme étant spécifiquement impliqués dans la régulation des gènes du jeu. L'évaluation de la

qualité des résultats pour un jeu se fait alors en comparant les prédictions obtenues (facteurs et lorsque cela est possible, les sites) aux données connues pour le jeu.

Utilisation des logiciels

TOUCAN
La version 2.2.5 avec les paramètres par défaut, telle que disponible à l'adresse `http://homes.esat.kuleuven.be/~saerts/software/toucan.php` a été utilisée. Nous avons tout d'abord utilisé MotifScanner pour prédire les sites potentiels dans nos séquences d'entrée en utilisant l'ensemble des matrices de vertébrés de la version publique de TRANSFAC et le modèle de fond EPD(3). Ensuite, nous avons utilisé le module statistique de sur-représentation pour extraire les matrices possédant le nombre d'instances le plus significatif. Lors de l'utilisation du module pour la statistique des sur-représentations, le fichier de fréquence attendu `epd_vertebrates_499_prior0.1.freq` a été utilisé.

OTFBS
La version 1.0 du programme disponible en ligne à l'adresse `http://www.bioinfo.tsinghua.edu.cn/~zhengjsh/OTFBS/` a été utilisée pour les tests. Cette version est basée sur la version publique de TRANSFAC (TRANSFAC 6.0). Afin de rendre les comparaisons plus cohérentes les résultats produits ont été filtrés pour ne prendre en compte que les matrices de vertébrés.

oPOSSUM
La version 1.3 du programme disponible en ligne à l'adresse `http://www.cisreg.ca/cgi-bin/oPOSSUM/opossum` a été utilisé avec l'ensemble des matrices JASPAR (seule base disponible). Les résultats obtenus ont été classés suivant la P-valeur calculée par le test de Fisher.

TFM-Explorer
Concernant TFM-Explorer les paramètres par défaut ont été utilisés.

4.2.1 Gènes spécifiques au muscle

Le premier exemple étudié concerne un ensemble de gènes humains exprimés dans le muscle du squelette. Cet ensemble de gènes est tiré du jeu de référence établi par Wasserman [91]. C'est une version mise à jour du jeu original, avec la rectification des erreurs liées à l'annotation.

Les premières étapes du développement du muscle sont contrôlées par une combinaison d'interactions qui mettent en œuvre des facteurs hélice-boucle-hélice de la famille MyoD (MyF), des facteurs de la famille MADS, des facteurs MEF2 (myocyte enhancer factor-2) et SRF (serum response factor) [47]. D'autres facteurs tels que TEF, MZF ou SP1 contribuent également aux mécanismes d'expression spécifiques au muscle.

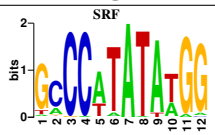
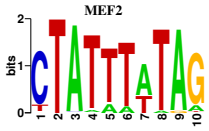
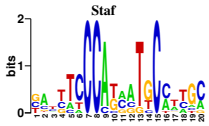
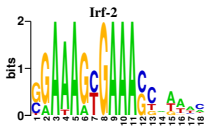
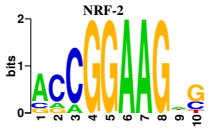
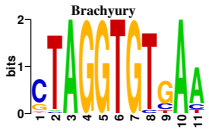
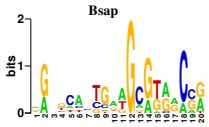
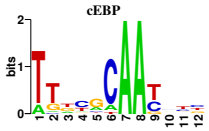
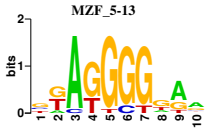
Le jeu utilisé ici est composé de neuf gènes humains pour lesquels des sites de fixation de facteurs spécifiques au muscle du squelette sont expérimentalement vérifiés.En utilisant la

base de données JASPAR, nous avons recherché les facteurs sur-représentés dans les séquences de 2 000 bp en amont de ces gènes.

Les résultats obtenus avec TFM-Explorer sont présentés dans le tableau 4.1. On peut constater que les trois prédictions les plus significatives (SRF, MEF2 et MZF_1) correspondent à des facteurs connus comme impliqués dans la régulation des gènes spécifiques au muscle. Pour chacune de ces prédictions, les régions concernées sont indiquées. Les logos des matrices montrent que les motifs trouvés sont indépendants. Des régions à la fois courtes et proximales ($[-0060 : -0030]$ et $[-0224 : 0091]$) pour MEF2 et SRF, et longues ($[-1431 : -0576]$) pour MZF_1 sont mises en évidence.

Les résultats obtenus avec les autres logiciels sont présentés dans le tableau 4.2. Alors que, parmi les cinq prédictions les plus significatives, TFM-Explorer prédit trois facteurs correctement, OTFBS ne prédit que deux facteurs (MZF1 et MEF2) et ce à la deuxième et troisième place. Dans les mêmes conditions, oPOSSUM et TOUCAN ne prédisent qu'un seul facteur spécifique.

Une des conclusions que l'on peut tirer au vu de ces résultats est la capacité de TFM-Explorer à identifier plusieurs facteurs, qui varient tant par la position de leurs sites de fixation que par l'étendue de leurs régions d'action.

Rank		PWM	Logo	Window	P-value
1	⋆	SRF		[-0224 :-0091]	7.869e-06
2	⋆	MEF2		[-0060 :-0030]	2.350e-05
3	⋆	MZF_1-4		[-1431 :-0576]	1.678e-04
4		Staf		[-1950 :-1311]	2.539e-04
5		Irf-2		[-1892 :-1592]	3.002e-04
6		NRF-2		[-0779 :-0324]	3.180e-04
7		Brachyury		[-0307 :-0048]	4.503e-04
8		Bsap		[-1911 :-0978]	4.800e-04
9		cEBP		[-1733 :-1679]	6.032e-04
10	⋆	MZF_5-13		[-1633 :-1078]	7.141e-04

TAB. 4.1: Résultats pour les gènes spécifiques au muscle
Les facteurs de transcription pour lesquels il existe des sites expérimentalement vérifiés sont notés par une étoile ⋆.

Rank		PWM	Window	P-value
TFM-Explorer				
1	⋆	SRF	[-0224 :-0091]	7.869e-06
2	⋆	MEF2	[-0060 :-0030]	2.350e-05
3	⋆	MZF_1-4	[-1431 :-0576]	1.678e-04
4		Staf	[-1950 :-1311]	2.539e-04
5		Irf-2	[-1892 :-1592]	3.002e-04
6		NRF-2	[-0779 :-0324]	3.180e-04
7		Brachyury	[-0307 :-0048]	4.503e-04
8		Bsap	[-1911 :-0978]	4.800e-04
9		cEBP	[-1733 :-1679]	6.032e-04
10	⋆	MZF_5-13	[-1633 :-1078]	7.141e-04
TOUCAN				
1		HEN1_01		8.567e-02
2	⋆	MEF2_02		1.021e-01
3		RSRFC4_01		1.129e-01
4		TAL1BETAITF2_01		1.311e-01
5		STAT5A_01		1.856e-01
6		TAL1BETAE47_01		2.322e-01
7		YY1_01		2.391e-01
8		STAT5B_01		2.534e-01
9	⋆	MEF2_03		3.056e-01
10		CDC5_01		3.134e-01
OTFBS				
1		YY1_02		2.047e-06
2	⋆	MZF1_02		2.763e-06
3	⋆	MEF2_02		9.493e-06
oPOSSUM				
1	⋆	MEF2		1.768e-04
2		Hen-1		3.730e-04
3		SRY		1.531e-03
4		c-MYB_1		1.780e-03
5		S8		2.983e-03
6		HFH-3		2.994e-03
7	⋆	SP1		3.220e-03
8	⋆	MZF_5-13		3.675e-03
9		Nkx		6.399e-03
10		RORalfa-2		7.747e-03

TAB. 4.2: Résultats pour les gènes spécifiques au muscle.
Les facteurs de transcription pour lesquels il existe des sites expérimentalement vérifiés sont notés par une étoile ⋆.

4.2.2 Gènes cibles des facteurs Rel/NF-κB

Le deuxième jeu de données concerne les gènes cibles des facteurs de la famille Rel/NF-κB. Ce jeu a été établi par la compilation de nombreuses données bibliographiques en collaboration avec des chercheurs de l'IBL. Cette compilation est présentée en détails dans [27].

Les facteurs Rel/NF-κB sont impliqués dans les mécanismes inflammatoires, immunitaires et d'apoptose des cellules. Cinq protéines de cette famille sont connues chez les vertébrés : c-Rel, RelA (p65), RelB, NF-κB1 (p50) et NF-κB2 (p52). Elles correspondent à six matrices dans la base de données TRANSFAC : `CREL_01`, `NFKAPPA50_01`, `NFKAPPAB65_01`, `NFKB_Q6`, `NFKAPPAB_01` and `NFKB_C`. Ces matrices possèdent toutes le consensus suivant : `5'-GGGRNYYYCC-3'`.

Le jeu est constitué d'un ensemble de 99 gènes humains possédant des sites de fixation validés expérimentalement pour au moins un facteur de cette famille. Pour évaluer l'influence de la taille des séquences sur la qualité des prédictions, nous avons testé chacun des logiciels avec différentes tailles de séquences : 2 000 bp ($[-2000 : 0]$) et 10 000 bp ($[-5000 : +5000]$).

Les fenêtres les plus significatives prédites par TFM-Explorer sont présentées dans le tableau 4.3. Une première constatation est que toutes ces fenêtres sont localisées autour du site d'initiation de la transcription, région riche en éléments *cis*-régulateurs. Une deuxième constatation est que les six matrices correspondant aux facteurs de la famille Rel/NF-κB sont présentes dans ces prédictions et que les régions prédites sont en accord avec les positions expérimentalement vérifiées des sites [27]. Les autres prédictions ne correspondent pas à des sites expérimentalement vérifiés. Cependant, excepté pour `CDXA_01`, de nombreux éléments penchent en faveur de leur validité. TFM-Explorer identifie de courtes fenêtres pour le facteur associé à la boîte TATA. La taille et la position de ces fenêtres ($[-56 : -23]$ et $[-55 : -15]$) sont caractéristiques de ce facteur. Les prédictions indiquent que près de 40% des gènes du jeu de données possèdent des boîtes TATA. Ceci peut être comparé à la proportion de gènes à boîte TATA présents dans l'ensemble du génome humain (32%) [84]. Les gènes à activation rapide tels que ceux régulés par les facteurs Rel/NF-κB contiennent fréquemment des boîtes TATA dans leurs promoteurs. En opposant ce type de gènes aux gènes sans boîte TATA (qui possèdent un niveau d'expression plus faible et constant), la relative abondance de gènes à boîte TATA semble être une des caractéristiques de ce jeu de données. Une autre famille de facteurs détectés par TFM-Explorer sont ceux de la famille Sp1. Ces facteurs se fixent sur des sites appelés boîtes GC. Pour ce facteur également, la fenêtre prédite $[-94 : -43]$ est en accord avec les informations connues sur ce facteur. De plus, de nombreux gènes régulés par Rel/NF-κB possèdent en effet dans leur promoteur des boîtes GC, comme par exemple MnSod [8] et les interleukins [33].

Les résultats obtenus par l'ensemble des logiciels pour les deux tailles de séquences sont donnés dans les tables 4.3 et 4.4. Une conclusion importante que l'on peut tirer de ces résultats est la baisse de la qualité des prédictions lorsque la taille des séquences augmente. Seuls TFM-Explorer et oPOSSUM fournissent des résultats pertinents pour les deux tailles de séquences. Le fait qu'oPOSSUM soit résistant aux longues séquences s'explique par son mode de fonctionnement. En effet, il recherche les signaux de régulation à partir d'une base de données de sites conservés entre l'homme et la souris. Il peut donc être indifférent à la longueur des séquences si les éléments conservés ne varient pas. Néanmoins, on peut constater une baisse importance de la P-valeur des prédictions (table 4.4). Dans ces conditions, TFM-Explorer

fournit, grâce à son approche locale, les mêmes prédictions en terme de fenêtre et de P-valeur (pour les fenêtres ne débordant zéro).

Rank		PWM	Window	P-value
TFM-Explorer [-2000 :0]				
1	⋆	NFKAPPAB65_01	[-0520 :-0019]	7.706e-27
2	⋆	NFKAPPAB_01	[-0698 :-0019]	9.418e-20
3		TATA_01	[-0056 :-0023]	1.118e-19
4	⋆	NFKB_C	[-0522 :-0020]	9.148e-19
5		TATA_C	[-0055 :-0015]	4.128e-16
6	⋆	CREL_01	[-0501 :-0020]	3.510e-15
7		CDXA_01	[-0071 :-0018]	4.262e-15
8	⋆	NFKB_Q6	[-0537 :-0021]	3.574e-14
9	⋆	NFKAPPAB50_01	[-0521 :-0019]	1.066e-12
10		SP1_Q6	[-0094 :-0043]	1.451e-11
TOUCAN [-2000 :0]				
1	⋆	NFKAPPAB65_01		1.381e-05
2	⋆	NFKB_C		6.975e-05
3	⋆	NFKAPPAB_01		3.139e-04
4		ARP1_01		1.257e-03
5		SREBP1_01		6.795e-03
6	⋆	NFKB_Q6		4.683e-02
7	⋆	NFKAPPAB50_01		8.661e-02
8		RORA2_01		9.847e-02
9		E47_02		1.628e-01
10		HEN1_01		2.882e-01
OTFBS [-2000 :0]				
1		FOXJ2_01		6.097e-49
2		FOXD3_01		4.229e-45
3		HFH3_01		5.356e-41
4		HNF3B_01		7.352e-35
5		IK2_01		3.031e-20
6		SREBP1_01		1.969e-19
7	⋆	NFKAPPAB65_01		3.708e-19
8	⋆	NFKB_C		8.819e-19
9	⋆	CREL_01		2.571e-18
10		CHOP_01		1.004e-17
oPOSSUM [-2000 :0]				
1	⋆	p65		1.333e-14
2	⋆	NF-kappaB		3.234e-11
3	⋆	c-REL		4.835e-09
4	⋆	p50		3.272e-07
5		SPI-B		5.137e-05
6		c-FOS		1.519e-04
7		Elk-1		2.329e-04
8		deltaEF1		2.877e-04
9		MZF_1-4		3.731e-04
10		Irf-1		6.815e-04

TAB. 4.3: Résultats pour les gènes cibles des facteurs Rel/NF-κB dans la région [-2000 :0]. Les facteurs de transcription pour lesquels il existe des sites expérimentalement vérifiés sont notés par une étoile ⋆.

Rank		PWM	Window	P-value
TFM-Explorer [-5000 :5000]				
1	⋆	NFKAPPAB65_01	[-0520 :+0115]	8.875e-27
2	⋆	NFKAPPAB_01	[-0698 :+0116]	1.026e-20
3	⋆	NFKB_C	[-0522 :-0020]	9.148e-19
4		TATA_01	[-0056 :-0010]	5.585e-18
5	⋆	NFKB_Q6	[-0537 :+0092]	2.241e-16
6		TATA_C	[-0055 :-0015]	4.128e-16
7	⋆	CREL_01	[-0501 :-0020]	3.510e-15
8		CDXA_01	[-0071 :-0018]	4.262e-15
9	⋆	NFKAPPAB50_01	[-0521 :+0012]	8.601e-13
10		SP1_Q6	[-0094 :-0043]	1.451e-11
TOUCAN [-5000 :5000]				
1		HFH3_01		0.0
2		BRACH_01		4.667e-01
3		RORA2_01		8.596e-01
4		NRSF_01		9.956e-01
5		E47_01		1.0
6		VMYB_01		1.0
7		AP4_01		1.0
8		MEF2_01		1.0
9		ELK1_01		1.0
10		EVI1_06		1.0
OTFBS [-5000 :5000]				
oPOSSUM [-5000 :5000]				
1	⋆	p65		1.941e-08
2	⋆	NF-kappaB		1.579e-05
3	⋆	c-REL		7.877e-05
4	⋆	p50		1.510e-04
5		c-FOS		6.236e-04
6		Irf-1		3.301e-03
7		MZF_5-13		5.543e-03
8		MZF_1-4		7.967e-03
9		NRF-2		2.933e-02
10		SPI-B		3.239e-02

TAB. 4.4: Résultats pour les gènes cibles des facteurs Rel/NF-κB dans la région [-5000 :5000]. Les facteurs de transcription pour lesquels il existe des sites expérimentalement vérifiés sont notés par une étoile ⋆.

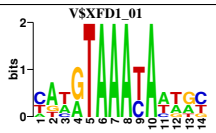
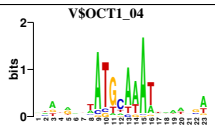
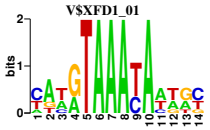
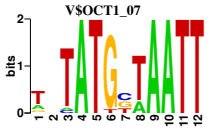
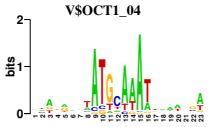
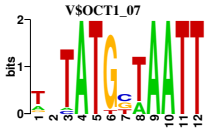
4.2.3 Gènes d'histones

Le dernier jeu présenté ici concerne des gènes codant pour des histones. Les histones sont des protéines impliquées dans la structure chromatinienne de l'ADN. Elles agissent comme des bobines sur lesquelles l'ADN s'enroule pour se compacter. En rendant ou non accessible l'ADN à la machinerie transcriptionnelle, les histones jouent un rôle important dans les mécanismes d'expression. On distingue cinq classes d'histones : H1, H2A, H2B, H3, H4. Ces protéines, et plus particulièrement les histones H3 et H4 ont été fortement conservées au fil de l'évolution. Parmi les motifs *cis*-régulateurs impliqués dans la régulation des histones H3, quatre sont clairement identifiés : CCAAT-box, Oct-1 box, GC-box et AC-box [15]. Exceptés les AC-box pour lesquels aucune entrée n'est disponible dans TRANSFAC, ces motifs correspondent aux matrices suivantes : `NFY_01`, `NFY_C`, `NFY_Q6`, `CAAT_01`, `CAAT_C` pour les CCAAT-box, `OCT1_Q6, OCT1_C, OCT1_0*` pour les boîtes OCT-1, et `SP1_01`, `SP1_Q6` pour les GC-box.

Nous avons évalué l'impact de l'utilisation de gènes provenant de différents organismes pour effectuer les prédictions. Un ensemble de 19 gènes d'histones H3 a été compilé à partir de [15]. Ce jeu est composé de 11 gènes humains, 7 gènes de souris et un gène de rat. Les séquences de 2 000 bp en amont du site d'initiation de la transcription ont été soumises aux différents logiciels en utilisant l'ensemble des matrices de vertébrés de TRANSFAC ou JASPAR le cas échéant. Les résultats obtenus sont présentés dans le tableau 4.5. Deux motifs (CCAAT-box et Oct-1 box) connus comme impliqués dans la régulation des gènes H3 (matrices `NFY_C` et `NFY_Q6`, et `OCT1_04` et `OCT1_07`) sont prédits par TFM-Explorer. Parmi les cinq prédictions les plus significatives faites par TFM-Explorer, la seule matrice non spécifique aux gènes H3 est `XFD1_01`. Cette prédiction peut s'expliquer par le profil de la matrice `XFD1_01`. En effet, il apparaît qu'il est très probable de trouver des occurrences de `XFD1_01` lorsqu'il existe des occurrences de `OCT1_04` ou `OCT1_07` du fait de la similarité des matrices (table 4.6). L'interface de TFM-Explorer permet de comparer ce type de biais en comparant, d'une part, le nombre de sites chevauchants pour un couple de prédictions données et, d'autre part, en calculant le taux de recouvrement théorique [48]. Dans notre exemple, nombre de sites pour `XFD1_01` chevauchent les sites prédits pour `OCT1_07` et `OCT1_04` (respectivement 37% et 53%). Des conclusions similaires peuvent être faites pour la matrice `PBX1_02` et celles correspondant à la boîte CCAAT (`NFY_C` et `NFY_Q6`). Les prédictions réalisées par TOUCAN et OTFBS sont également présentées dans la table 4.5. Pour ce jeu de données aucun résultat n'a été produit par oPOSSUM du fait de l'absence de couples d'orthologues dans sa base de données pour les gènes considérés.

Rank		PWM	Window	P-value
TFM-Explorer				
1	⋆	NFY_C	[-1375 :-0039]	4.757e-24
2	⋆	OCT1_04	[-0588 :-0022]	1.537e-20
3	⋆	NFY_Q6	[-1318 :-0039]	4.026e-16
4	⋆	OCT1_07	[-0574 :-0025]	7.932e-14
5		XFD1_01	[-0890 :-0025]	2.253e-13
6		PBX1_02	[-0491 :-0040]	2.737e-13
7		SRY_02	[-0895 :-0015]	1.803e-12
8		MEF2_04	[-0482 :-0038]	1.826e-12
9		HNF1_01	[-0642 :-0097]	7.089e-12
10		EVI1_04	[-0417 :-0040]	9.277e-12
TOUCAN				
1	⋆	NFY_01		1.364e-08
2	⋆	OCT1_01		1.854e-05
3		GFI1_01		4.506e-05
4		TATA_01		1.315e-03
5	⋆	CAAT_01		1.781e-03
6	⋆	OCT_C		1.018e-02
7		MEF2_02		1.041e-02
8		MEF2_03		1.041e-02
9		NFY_C		1.633e-02
10		CART1_01		2.569e-02
OTFBS				
1		IRF1_01		5.099e-26
2		HFH3_01		6.865e-22
3		FOXJ2_01		1.606e-21
4		MEF2_01		6.896e-20
5		HNF3B_01		1.165e-18
6		MEF2_04		1.243e-18
7		FOXD3_01		3.698e-18
8		MEF2_02		2.964e-15
9		XFD1_01		8.016e-15
10	⋆	NFY_C		6.396e-14

Tab. 4.5: Résultats pour les gènes d'histone H3.
Les facteurs de transcription pour lesquels il existe des sites expérimentalement vérifiés sont notés par une étoile ⋆.

Matrice 1		Matrice 2		Corrélation
XFD1_01		OCT1_04		35.42%
XFD1_01		OCT1_07		19.03%
OCT1_04		OCT1_07		37.34%

TAB. 4.6: Corrélation entre les matrices OCT1_04, OCT1_07et XFD1_01.

4.2.4 Robustesse au bruit

Pour évaluer la robustesse au bruit de TFM-Explorer nous avons mesuré ses qualités de prédictions lorsque les données analysées sont fortement bruitées. Pour cela nous avons construit à partir d'ensembles de gènes co-régulés, des jeux de données artificiels comportant une quantité croissante de bruit en remplaçant les séquences par des séquences sélectionnées aléatoirement dans le même génome que la séquence remplacée.

En partant des deux jeux constitués dans les sections 4.2.1 et 4.2.2, nous avons remplacé de 10% à 90% des séquences par des séquences piochées aléatoirement par palier de 10%. Pour chaque niveau de bruit, 100 jeux de données ont été générés et les prédictions réalisées pour chacun. Une prédiction est considérée comme correcte lorsque la fenêtre prédite la plus significative correspond à un facteur connu comme impliqué dans la régulation du jeu concerné. L'évolution du nombre de prédictions correctes en fonction du niveau de bruit est reporté dans la figure 4.8. On constate que le niveau de bruit qu'un jeu de données peut tolérer est fortement dépendant de la qualité des signaux de régulation présents dans ce jeu. Par exemple, le jeu de haute qualité Rel/NF-κB peut tolérer un niveau de bruit de 50% sans altérer la qualité des prédictions faites. D'autre part, on constate que lorsque le niveau de bruit devient élevé (supérieur à 50%) les signaux de régulation sont progressivement noyés dans le bruit.

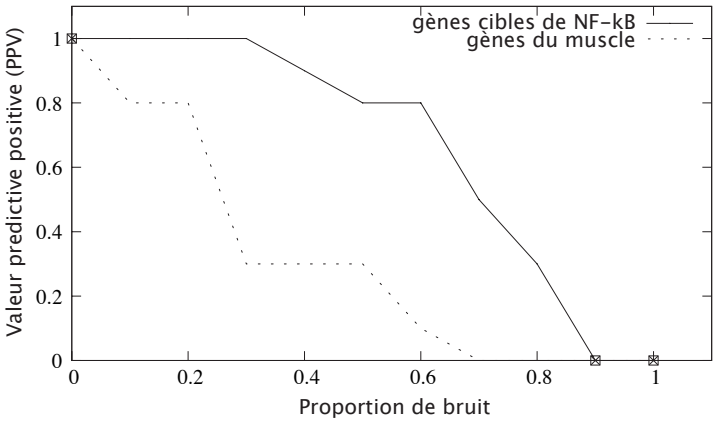


FIG. 4.8: Robustesse au bruit.

4.3 Tentative d'application à l'inférence de motifs

Nous avons présenté dans les sections précédentes une application de la recherche de sur-représentations locales utilisant une base de données de matrices. Dans cette section, nous montrons comment cette méthode peut être étendue à l'inférence de motifs sans connaissance préalable du motif. Nous avons vu au chapitre 2 qu'il existait plusieurs approches d'inférence de motifs sur-représentés, notamment des approches combinatoires comme Oligo-analysis, SMILE, etc. La recherche locale peut améliorer ces approches. Nous illustrons cette stratégie par deux exemples. Le premier reprend le jeu de gènes cibles des facteurs Rel/NF-κB de la section 4.2.2. Le deuxième s'intéresse aux motifs exacts localement sur-représentés dans l'ensemble des promoteurs humains extraits de la base EPD [64].

4.3.1 Motifs avec erreurs localement sur-représentés dans les gènes cibles des facteurs Rel/NF-κB

Le premier jeu de données sur lequel a porté notre investigation concerne les gènes cibles des facteurs Rel/NF-κB (4.2.2). Nous avons vu qu'il était possible d'identifier des sites Rel/NF-κB en ayant recours à la base de données TRANSFAC ou JASPAR. Nous regardons ici si l'approche locale sans connaissance du motif permet également de retrouver ces signaux avec seulement un descripteur de motif.

Protocole de recherche

Pour rechercher les motifs sur-représentés dans les régions en amont des gènes cibles des facteurs Rel/NF-κB, nous avons sélectionné comme dans le cas de la recherche avec une base de matrices, des séquences situées dans la région $[-2000 : 0]$ (repérées par rapport au site d'initiation de la transcription). Pour simplifier les recherches, nous avons considéré des motifs avec erreurs ayant la structure suivante : un motif de longueur six pouvant contenir cinq positions consécutives indéterminées. Un motif correspond alors à un motif sans erreur de

longueur six ou à un motif de longueur dix avec cinq positions consécutive indéterminées. L'objectif de la recherche étant de déterminer le gain apporté par l'approche locale, la structure de motif considérée doit permettre de détecter les motifs en dyades associés aux facteurs Rel/NF-κB.

Pour déterminer la significativité d'une sur-représentation locale, nous avons calculé une P-valeur pour chacun des motifs non-chevauchants rencontrés dans les séquences à l'aide d'une loi de Poisson (section 3.4.2). Pour évaluer la fréquence attendue de ces motifs, nous avons mesuré leur fréquence dans un jeu de 2 000 séquences promotrices tirées aléatoirement dans le génome humain. Nous avons ensuite effectué la recherche dans deux cas de figure : dans un premier temps sans appliquer d'approche locale, c'est-à-dire en considérant des fenêtres de taille égale à la longueur des séquences, puis dans un deuxième temps en appliquant une stratégie de recherche locale.

Résultats

Rang		Motif	Fenêtre	P-valeur
Approche locale				
1	^{t}	TATAAA|TTTATA	[-0040 :-0026]	1.64e-16
2	^{t}	ATAAAG|CTTTAT	[-0032 :-0023]	7.47e-10
3	$\star$	GGGANC|GNTCCC	[-0671 :-0025]	9.38e-10
4		ANTCAG|CTGANT	[-0996 :+0090]	2.12e-09
5		GAANCA|TGNTTC	[-0998 :+0091]	3.39e-09
6	^{t}	GNTATA|TATANC	[-0035 :-0024]	5.70e-09
7	$\star$	GGGNCC|GGNCCC	[-1000 :-0020]	1.80e-08
8		AANCAC|GTGNTT	[-0154 :+0090]	2.73e-08
9		CANTCA|TGANTG	[-0455 :+0091]	6.07e-08
10	^{t}	CNATAA|TTATNG	[-0051 :-0029]	1.07e-07
Approche non locale				
1	$\star$	GGGANC|GNTCCC	[-2000 :0]	3.11e-07
2		AGGGAG|CTCCCT	[-2000 :0]	5.63e-07
3		GGGAAA|TTTCCC	[-2000 :0]	7.40e-07
4		AATTCC|GGAATT	[-2000 :0]	1.26e-06
5		ATTTCC|GGAAAT	[-2000 :0]	1.64e-06
6		GAANCA|TGNTTC	[-2000 :0]	3.67e-06
7		CTCTGA|TCAGAG	[-2000 :0]	3.69e-06
8		GACTCA|TGAGTC	[-2000 :0]	5.30e-06
9		GGAANC|GNTTCC	[-2000 :0]	1.39e-05
10		TNGGAA|TTCCNA	[-2000 :0]	2.00e-05

TAB. 4.7: Résultats pour les gènes cibles des facteurs Rel/NF-κB dans la région [-2000 :0]. Les motifs correspondant aux facteurs Rel/NF-κB sont repérés par une étoile $\star$. Ceux correspondant à la boîte TATA sont repérés par un ^{t}.

Les résultats produits pour les deux cas sont donnés dans la table 4.7.

Dans le cas de l'approche locale, deux groupes de motifs sont prédits : les motifs 1, 2 et 4 correspondant au motif `TATAA`, et les motifs 3 et 7 correspondant au consensus associé aux sites Rel/NF-κB. Le consensus commun au premier groupe de motifs (figure 4.9) ainsi que les fenêtres dans lesquelles ils sont prédits, sont révélateurs de la boîte TATA. Comme indiqué dans la section 4.2.2, la présence de boîtes TATA dans les gènes cibles de Rel/NF-κB semble être une caractéristique du jeu. Pour le deuxième groupe, les deux motifs `GGGANNNNNC` et `GGGNNNNNCC` correspondent bien au consensus connu (`GGGRNYYYCC`) pour la famille Rel/NF-κB. Dans le cas de l'approche non locale, seul un motif valide est détecté `GGGANNNNNC` et ce avec une significativité nettement plus faible que pour l'approche locale.

```
4    ATAAAG
1   TATAAA
2 GNTATA
    *****
    TATAA
```

FIG. 4.9: Alignement des motifs correspondant à la boîte TATA.

4.3.2 Mots localement sur-représentés dans les promoteurs humains

Certains motifs régulateurs, comme par exemple les boîtes TATA ou encore les boîtes GC, sont présents dans une large proportion des promoteurs eucaryotes. Suzuki et co-auteurs [84] ont, par exemple, montré que 97% des promoteurs humains possédaient une boîte GC et que près de 32% possédaient une boîte TATA. Ces motifs sont connus comme présentant une forte spécificité spatiale (section 3.1). On peut se demander s'il est possible de détecter ce type d'éléments en recherchant les motifs sur-représentés localement dans un large ensemble de promoteurs. L'idée est alors la suivante : nous allons rechercher les motifs exacts qui présentent la plus forte affinité spatiale ainsi que les fenêtres associées. Pour cela, nous avons utilisé l'ensemble des promoteurs humains provenant de la base de données EPD [64]. L'intérêt de cette base est de fournir un ensemble de séquences promotrices pour lesquelles les sites d'initiation de la transcription sont bien annotés. Nous avons recherché les motifs exacts de longueur six possédant les fenêtres de sur-représentations les plus significatives dans l'ensemble des séquences promotrices entourant le site d'initiation de la transcription (région $[-499 : +100]$).

Stratégie de recherche

Une P-valeur a été calculée comme dans l'exemple précédent à l'aide de l'approximation de Poisson pour chacun des motifs non-chevauchants rencontrés dans les séquences. Nous avons évalué les fréquences attendues pour l'ensemble des motifs à partir du comptage de leurs occurrences sur toute la longueur des séquences. Cette fréquence a ensuite été comparée avec le nombre de sites présents dans chacune des fenêtres locales.

Rang		Motif	Fenêtre	P-valeur
1	^{g}	CCGCCC|GGGCGG	[-0160 :-0045]	3.04e-124
2	^{g}	CCCGCC|GGCGGG	[-0166 :-0044]	1.41e-105
3	^{t}	TATAAA|TTTATA	[-0033 :-0029]	2.90e-96
4	^{g}	CGCCCC|GGGGCG	[-0169 :-0010]	2.09e-93
5	^{g}	CCCCGC|GCGGGG	[-0170 :-0045]	2.39e-86
6		CGGAAG|CTTCCG	[-0085 :+0002]	1.48e-77
7		ATGGCG|CGCCAT	[-0006 :+0031]	1.76e-77
8	^{g}	CGGGGC|GCCCCG	[-0169 :-0041]	1.28e-59
9		CCGGAA|TTCCGG	[-0090 :+0004]	3.19e-58
10		GCCGCC|GGCGGC	[-0024 :+0095]	5.10e-58

TAB. 4.8: Résultats pour les promoteurs tirés de EPD dans la région [-499 :+100].
Les motifs relatifs à la boîte GC sont repérés par un ^{g}. Ceux correspondant à la boîte TATA sont repérés par un ^{t}.

Résultats

Les dix motifs les plus significatifs extraits ainsi que les fenêtres associées sont reportés dans la table 4.8. D'après ces résultats, on peut observer deux classes de motifs : ceux relatifs à la boîte GC (motifs 1, 2, 4, 5 et 8) et ceux relatifs à la boîte TATA (motif 3). Ce dernier motif (`TATAAA`) correspond bien au consensus connu pour la boîte TATA. De plus, la fenêtre $[-33 : -29]$ dans laquelle il est prédit concorde en position et en taille avec la région d'action du facteur associé. Les cinq motifs relatifs à la boîte GC prédits sont également compatibles avec le consensus connu pour cette boîte : `GGGCGG`. Ils correspondent au même motif décalé, le motif le plus significatif (motif 1) formant le cœur ce groupe. La figure 4.10 fournit l'alignement pour ce groupe.

```
2       GGCGGG
1      GGGCGG
4     GGGGCG
8    CGGGGC
5   GCGGGG
      *******
      GGGGCGG
```

FIG. 4.10: Alignement des motifs correspondant à la boîte GC.

Conclusion

La mise à disposition de quantités sans cesse croissantes de données issues du séquençage a ouvert la voie à de nouvelles approches bio-informatiques. Nous avons présenté dans ce document comment il était possible de rechercher des éléments *cis*-régulateurs en considérant le problème de la recherche de signaux transcriptionnels, sous l'angle de l'algorithmique de texte. Nous avons alors montré les limites et avantages des approches actuelles, en particulier en ce qui concerne la génomique comparative et la sur-représentation de motifs. Nous avons également discuté d'un autre type d'information à prendre en compte dans la recherche de signaux : la conservation spatiale. Dans ce cadre, nous avons introduit une nouvelle méthode permettant de tirer parti de ces idées : la recherche de sur-représentations locales dans un contexte multi-organismes. Pour cela, nous avons défini la notion de fenêtre locale et avons donné un moyen d'en calculer la significativité. Enfin, nous avons proposé une méthode heuristique pour extraire efficacement ce type de fenêtres. Cette stratégie, implémentée dans le logiciel TFM-Explorer, a montré sa pertinence sur différents jeux de données.

Une des principales limites à laquelle nous sommes soumis, concerne la nature des données utilisées. Les méthodes présentées dans ce document ne considèrent l'ADN que du point de vue de sa séquence pour prédire des éléments régulant sa transcription. Cela n'est pas sans poser question sur ce que l'on peut espérer comprendre avec cet angle de vision.

La régulation transcriptionnelle est un mécanisme complexe qui intervient dans les trois dimensions d'espace et dans le temps. Une modélisation plus complète des interactions ADN-protéines est sans doute nécessaire pour permettre de mieux comprendre la régulation transcriptionnelle. Mais cette étape est certainement insuffisante pour appréhender complètement ces mécanismes. En effet, la structure spatiale de l'ADN joue un rôle important dans la régulation transcriptionnelle. Elle permet en particulier à certaines parties de l'information génétique de devenir accessibles afin d'être transcrites, ou encore, en rapprochant spatialement des gènes éloignés, de les soumettre aux mêmes protéines régulatrices. La connaissance de ces phénomènes et des mécanismes de régulation transcriptionnelle en général est un vaste champ de recherche qui reste ouvert sous de nombreux aspects.

Malgré cette limitation inhérente à la modélisation du problème, la recherche de signaux régulateurs conservés à partir de séquences permet d'avancer dans la connaissance

des mécanismes de régulation. Nous pensons que l'approche par sur-représentation locale que nous avons introduite pourrait être étendue de plusieurs manières.

Une première extension possible concerne l'espace dans lequel les motifs sont recherchés. La contrainte imposée par l'utilisation d'une banque de modèles matriciels mérite d'être levée. L'inférence de motifs est une tâche plus délicate qui a néanmoins donné quelques succès dans le cas des motifs régulateurs. Dans la fin du dernier chapitre, nous avons présenté quelques pistes pour l'adaptation de l'approche locale à l'inférence de motifs. Il peut sembler intéressant d'étendre cette piste en proposant une méthode d'inférence plus complète basée sur l'approche locale et la conservation entre espèces. La complexité en temps et en espace de ce type de problème impose une réflexion importante sur la nature des structures à utiliser (arbre des suffixes ou autre structure d'index...) et sur les algorithmes à développer.

Une deuxième extension possible concerne le positionnement des séquences, c'est-à-dire le choix d'un repère afin de tirer parti de la conservation locale. Nous avons utilisé le site d'initiation de la transcription comme moyen le plus évident pour définir un repère. Ceci implique de disposer de séquences bien annotées pour que la conservation spatiale ne "fausse" pas les prédictions. D'autres repères doivent pouvoir être utilisés. Une première possibilité, dans le cadre des modules *cis*-régulateurs, est de fixer le repère sur un site de fixation bien établi et de rechercher dans son environnement des sites de facteurs associés. Une autre piste est d'utiliser une région fortement conservée entre orthologues pour définir un point d'accroche.

Bibliographie

[1] S Aerts, G Thijs, B Coessens, M Staes, Y Moreau, and B De Moor. Toucan : deciphering the cis-regulatory logic of coregulated genes. *Nucleic Acids Res*, 31(6) :1753–64, 2003.

[2] W Ao, J Gaudet, W J Kent, S Muttumu, and S E Mango. Environmentally induced foregut remodeling by PHA-4/FoxA and DAF-12/NHR. *Science*, 305(5691) :1743–6, 2004.

[3] T L Bailey and C Elkan. Unsupervised learning of multiple motifs in biopolymers using Expectation Maximization. *Machine Learning*, 21(1-2) :51–80, 1995.

[4] M Beckstette, R Homann, R Giegerich, and S Kurtz. Fast index based algorithms and software for matching position specific scoring matrices. *BMC Bioinformatics*, 7(1) :389, 2006.

[5] P V Benos, M L Bulyk, and G D Stormo. Additivity in protein-DNA interactions : how good an approximation is it ? *Nucleic Acids Res*, 30(20) :4442–51, 2002.

[6] P V Benos, A S Lapedes, and G D Stormo. Is there a code for protein-DNA recognition ? probab(ilistical)ly. . . *Bioessays*, 24(5) :466–75, 2002.

[7] O G Berg and P H von Hippel. Selection of DNA binding sites by regulatory proteins. statistical-mechanical theory and application to operators and promoters. *J Mol Biol*, 193(4) :723–50, 1987.

[8] D Bernard, D Monte, B Vandenbunder, and C Abbadie. The c-Rel transcription factor can both induce and inhibit apoptosis in the same cells via the upregulation of mnsod. *Oncogene*, 21(0950-9232) :4392–402, 2002.

[9] D Boffelli, J McAuliffe, D Ovcharenko, K D Lewis, I Ovcharenko, L Pachter, and E M Rubin. Phylogenetic shadowing of primate sequences to find functional regions of the human genome. *Science*, 299(5611) :1391–4, 2003.

[10] C E Bonferroni. Il calcolo delle assicurazioni su gruppi di teste. In *Studi in Onore del Professore Salvatore Ortu Carboni*, pages 13–60. Rome, 1935.

[11] N Bray, I Dubchak, and L Pachter. AVID : A global alignment program. *Genome Res*, 13(1) :97–102, 2003.

[12] M Brudno, C B Do, G M Cooper, M F Kim, E Davydov, E D Green, A Sidow, and S Batzoglou. LAGAN and Multi-LAGAN : efficient tools for large-scale multiple alignment of genomic dna. *Genome Res*, 13(4) :721–31, 2003.

[13] M J Buck and J D Lieb. ChIP-chip : considerations for the design, analysis, and application of genome-wide chromatin immunoprecipitation experiments. *Genomics*, 83(3) :349–60, 2004.

[14] K Cartharius, K Frech, K Grote, B Klocke, M Haltmeier, A Klingenhoff, M Frisch, M Bayerlein, and T Werner. MatInspector and beyond : promoter analysis based on transcription factor binding sites. *Bioinformatics*, 21(13) :2933–42, 2005.

[15] R Chowdhary, R A Ali, W Albig, D Doenecke, and V B Bajic. Promoter modeling : the case study of mammalian histone promoters. *Bioinformatics*, 21(11) :2623–8, 2005.

[16] J M Claverie and S Audic. The statistical significance of nucleotide position-weight matrix matches. *Comput Appl Biosci*, 12(5) :431–9, 1996.

[17] Gavin E Crooks, Gary Hon, John-Marc Chandonia, and Steven E Brenner. WebLogo : a sequence logo generator. *Genome Res*, 14(6) :1188–90, 2004.

[18] M Dekker. *Handbook of fungal biotechnology.* Dilip K. Arora, 2003.

[19] R Dubin, S Eddy, A Krogh, and G Mitchison. Biological sequence analysis. *Cambridge University Press*, 1998.

[20] S R Eddy. Profile hidden Markov models. *Bioinformatics*, 14(9) :755–63, 1998.

[21] S R Eddy. A model of the statistical power of comparative genome sequence analysis. *PLoS Biol*, 3(1) :e10, 2005.

[22] C Fondrat and A Kalogeropoulos. Approaching the function of new genes by detection of their potential upstream activation sequences in saccharomyces cerevisiae : application to chromosome III. *Curr Genet*, 25(5) :396–406, 1994.

[23] M G Fried. Measurement of protein-DNA interaction parameters by electrophoresis mobility shift assay. *Electrophoresis*, 10(5-6) :366–76, 1989.

[24] M C Frith, Y Fu, L Yu, J-F Chen, U Hansen, and Z Weng. Detection of functional DNA motifs via statistical over-representation. *Nucleic Acids Res*, 32(4) :1372–81, 2004.

[25] D J Galas and A Schmitz. DNAse footprinting : a simple method for the detection of protein-DNA binding specificity. *Nucleic Acids Res*, 5(9) :3157–70, 1978.

[26] N I Gershenzon, G D Stormo, and I P Ioshikhes. Computational technique for improvement of the position-weight matrices for the DNA/protein binding sites. *Nucleic Acids Res*, 33(7) :2290–301, 2005.

[27] K Gosselin, H Touzet, and C Abbadie. NF-*kappa*B target genes, available at http ://bioinfo.lifl.fr/nf-kb/.

[28] J D Gralla and J Collado-Vides. Organization and function of transcriptional regulatory elements. *Cellular and Molecular Biology*, 1996.

[29] R C Hardison. Conserved noncoding sequences are reliable guides to regulatory elements. *Trends Genet*, 16(9) :369–72, 2000.

[30] G Z Hertz, G W 3rd Hartzell, and G D Stormo. Identification of consensus patterns in unaligned DNA sequences known to be functionally related. *Comput Appl Biosci*, 6(2) :81–92, 1990.

[31] G Z Hertz and G D Stormo. Identifying DNA and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*, 15(7-8) :563–77, 1999.

[32] J M Heumann, A S Lapedes, and G D Stormo. Neural networks for determining protein specificity and multiple alignment of binding sites. *Proc Int Conf Intell Syst Mol Biol*, 2(1553-0833) :188–94, 1994.

[33] J Hiscott, J Marois, J Garoufalis, M D'Addario, A Roulston, I Kwan, N Pepin, J Lacoste, H Nguyen, and G Bensi. Characterization of a functional NF-*kappa*B site in the human interleukin 1 beta promoter : evidence for a positive autoregulatory loop. *Mol Cell Biol*, 13(0270-7306) :6231–40, 1993.

[34] S J Ho Sui, J R Mortimer, D J Arenillas, J Brumm, C J Walsh, B P Kennedy, and W W Wasserman. oPOSSUM : identification of over-represented transcription factor binding sites in co-expressed genes. *Nucleic Acids Res*, 33(10) :3154–64, 2005.

[35] C E Horak and M Snyder. ChIP-chip : a genomic approach for identifying transcription factor binding sites. *Methods Enzymol*, 350(0076-6879) :469–83, 2002.

[36] H Huang, M-C J Kao, X Zhou, J S Liu, and W H Wong. Determination of local statistical significance of patterns in Markov sequences with application to promoter element identification. *J Comput Biol*, 11(1) :1–14, 2004.

[37] T Hubbard, D Barker, E Birney, G Cameron, Y Chen, L Clark, T Cox, J Cuff, V Curwen, T Down, R Durbin, E Eyras, J Gilbert, M Hammond, L Huminiecki, A Kasprzyk, H Lehvaslaiho, P Lijnzaad, C Melsopp, E Mongin, R Pettett, M Pocock, S Potter, A Rust, E Schmidt, S Searle, G Slater, J Smith, W Spooner, A Stabenau, J Stalker, E Stupka, A Ureta-Vidal, I Vastrik, and M Clamp. The Ensembl genome database project. *Nucleic Acids Res*, 30(1) :38–41, 2002.

[38] J D Hughes, P W Estep, S Tavazoie, and G M Church. Computational identification of cis-regulatory elements associated with groups of functionally related genes in saccharomyces cerevisiae. *J Mol Biol*, 296(5) :1205–14, 2000.

[39] D Karolchik, R Baertsch, M Diekhans, T S Furey, A Hinrichs, Y T Lu, K M Roskin, M Schwartz, C W Sugnet, D J Thomas, R J Weber, D Haussler, and W J Kent. The ucsc genome browser database. *Nucleic Acids Res*, 31(1) :51–4, 2003.

[40] W Krivan and W W Wasserman. A predictive model for regulatory sequences directing liver-specific transcription. *Genome Res*, 11(9) :1559–66, 2001.

[41] J H Laity, B M Lee, and P E Wright. Zinc finger proteins : new insights into structural and functional diversity. *Curr Opin Struct Biol*, 11(1) :39–46, 2001.

[42] 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*, 262(5131) :208–14, 1993.

[43] C E Lawrence and A A Reilly. An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins*, 7(1) :41–51, 1990.

[44] B Lenhard, A Sandelin, L Mendoza, P Engstrom, N Jareborg, and W W Wasserman. Identification of conserved regulatory elements by comparative genome analysis. *J Biol*, 2(2) :13, 2003.

[45] M Lescot, P Dehais, G Thijs, K Marchal, Y Moreau, Y Van de Peer, P Rouze, and S Rombauts. PlantCARE, a database of plant cis-acting regulatory elements and a portal to tools for in silico analysis of promoter sequences. *Nucleic Acids Res*, 30(1) :325–7, 2002.

[46] M Li, B Ma, and L Wang. Finding similar regions in many strings. *STOC*, pages 473–482, 1999.

[47] Shijie Li, Michael P Czubryt, John McAnally, Rhonda Bassel-Duby, James A Richardson, Franziska F Wiebel, Alfred Nordheim, and Eric N Olson. Requirement for serum response factor for skeletal muscle growth and maturation revealed by tissue-specific gene deletion in mice. *Proc Natl Acad Sci U S A*, 102(0027-8424) :1082–7, 2005.

[48] A Liefooghe, H Touzet, and J-S Varré. Large-scale matching for position weight matrices. In *Combinatorial Pattern Matching*. Lecture Notes in Computer Science, 2006.

[49] G G Loots, I Ovcharenko, L Pachter, I Dubchak, and E M Rubin. rVista for comparative sequence-based discovery of functional transcription factor binding sites. *Genome Res*, 12(5) :832–9, 2002.

[50] L Marsan and M F Sagot. Algorithms for extracting structured motifs using a suffix tree with an application to promoter and regulatory site consensus identification. *J Comput Biol*, 7(3-4) :345–62, 2000.

[51] A M Moses, D Y Chiang, M Kellis, E S Lander, and M B Eisen. Position specific variation in the rate of evolution in transcription factor binding sites. *BMC Evol Biol*, 3(1471-2148 (Electronic)) :19, 2003.

[52] C W Muller. Transcription factors : global and detailed views. *Curr Opin Struct Biol*, 11(1) :26–32, 2001.

[53] L Narlikar and A J Hartemink. Sequence features of DNA binding sites reveal structural class of associated transcription factor. *Bioinformatics*, 22(2) :157–63, 2006.

[54] S B Needleman and C D Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3) :443–53, 1970.

[55] A F Neuwald, J S Liu, and C E Lawrence. Gibbs motif sampling : detection of bacterial outer membrane protein repeats. *Protein Sci*, 4(8) :1618–32, 1995.

[56] G Nuel. LD-SPatt : large deviations statistics for patterns on Markov chains. *J Comput Biol*, 11(6) :1023–33, 2004.

[57] J C Oeltjen, T M Malley, D M Muzny, W Miller, R A Gibbs, and J W Belmont. Large-scale comparative sequence analysis of the human and murine bruton's tyrosine kinase loci reveals conserved regulatory domains. *Genome Res*, 7(4) :315–29, 1997.

[58] C O Pabo and R T Sauer. Transcription factors : structural families and principles of DNA recognition. *Annu Rev Biochem*, 61(0066-4154) :1053–95, 1992.

[59] G Pavesi, G Mauri, and G Pesole. An algorithm for finding signals of unknown length in DNA sequences. *Bioinformatics*, 17 Suppl 1(1367-4803) :S207–14, 2001.

[60] G Pavesi, P Mereghetti, G Mauri, and G Pesole. Weeder Web : discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic Acids Res*, 32(Web Server issue) :W199–203, 2004.

[61] E Perez-Rueda, J D Gralla, and J Collado-Vides. Genomic position analyses and the transcription machinery. *J Mol Biol*, 275(2) :165–70, 1998.

[62] J Pickands. Statistical inference using extreme order statistics. *The Annals of Statistics*, 1975.

[63] J Ponjavic, B Lenhard, C Kai, J Kawai, P Carninci, Y Hayashizaki, and A Sandelin. Transcriptional and structural impact of tata-initiation site spacing in mammalian core promoters. *Genome Biol*, 7(8) :R78, 2006.

[64] V Praz, R Perier, C Bonnard, and P Bucher. The Eukaryotic Promoter Database, EPD : new entry types and links to gene expression data. *Nucleic Acids Res*, 30(1) :322–4, 2002.

[65] K D Pruitt, T Tatusova, and D R Maglott. NCBI Reference Sequence (RefSeq) : a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res*, 33(Database issue) :D501–4, 2005.

[66] K Quandt, K Frech, H Karas, E Wingender, and T Werner. MatInd and MatInspector : new fast and versatile tools for detection of consensus matches in nucleotide sequence data. *Nucleic Acids Res*, 23(23) :4878–84, 1995.

[67] M Régnier and A Denise. Rare events and conditional events on random strings. *Discrete Mathematics and Theoretical Computer Science*, 6(2) :191–214, 2004.

[68] G Reinert and S Schbath. Compound Poisson and Poisson process approximations for occurrences of multiple words in Markov chains. *J Comput Biol*, 5(2) :223–53, 1998.

[69] G Reinert, S Schbath, and M S Waterman. Probabilistic and statistical properties of words : an overview. *J Comput Biol*, 7(1-2) :1–46, 2000.

[70] B Ren, F Robert, J J Wyrick, O Aparicio, E G Jennings, I Simon, J Zeitlinger, J Schreiber, N Hannett, E Kanin, T L Volkert, C J Wilson, S P Bell, and R A Young. Genome-wide location and function of DNA binding proteins. *Science*, 290(5500) :2306–9, 2000.

[71] M F Sagot. Spelling approximate repeated or common motifs using a suffix tree. *Lecture Notes in Computer Science*, 1380 :111–27, 1998.

[72] A Sandelin, W Alkema, P Engstrom, W W Wasserman, and B Lenhard. JASPAR : an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res*, 32(Database issue) :D91–4, 2004.

[73] A Sandelin, W W Wasserman, and B Lenhard. ConSite : web-based prediction of regulatory elements using cross-species comparison. *Nucleic Acids Res*, 32(Web Server issue) :W249–52, 2004.

[74] T D Schneider and R M Stephens. Sequence logos : a new way to display consensus sequences. *Nucleic Acids Res*, 18(20) :6097–100, 1990.

[75] T D Schneider, G D Stormo, L Gold, and A Ehrenfeucht. Information content of binding sites on nucleotide sequences. *J Mol Biol*, 188(3) :415–31, 1986.

[76] R Sharan, I Ovcharenko, A Ben-Hur, and R M Karp. CREME : a framework for identifying cis-regulatory modules in human-mouse conserved segments. *Bioinformatics*, 19 Suppl 1(1367-4803) :i283–91, 2003.

[77] A Sidow. Sequence first. Ask questions later. *Cell*, 111(1) :13–6, 2002.

[78] S Sinha and M Tompa. Discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Res*, 30(24) :5549–60, 2002.

[79] S Sinha and M Tompa. YMF : A program for discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Res*, 31(13) :3586–8, 2003.

[80] R Staden. Computer methods to locate signals in nucleic acid sequences. *Nucleic Acids Res*, 12(1 Pt 2) :505–19, 1984.

[81] G D Stormo. Probing information content of DNA-binding sites. *Methods Enzymol*, 208(0076-6879) :458–68, 1991.

[82] G D Stormo. DNA binding sites : representation and discovery. *Bioinformatics*, 16(1) :16–23, 2000.

[83] G D Stormo and D S Fields. Specificity, free energy and information content in protein-DNA interactions. *Trends Biochem Sci*, 23(3) :109–13, 1998.

[84] Y Suzuki, T Tsunoda, J Sese, H Taira, J Mizushima-Sugano, H Hata, T Ota, T Isogai, T Tanaka, Y Nakamura, A Suyama, Y Sakaki, S Morishita, K Okubo, and S Sugano. Identification and characterization of the potential promoter regions of 1031 kinds of human genes. *Genome Research*, 11(5) :677–84, 2001.

[85] G Thijs, M Lescot, K Marchal, S Rombauts, B De Moor, P Rouze, and Y Moreau. A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics*, 17(12) :1113–22, 2001.

[86] M Tompa, N Li, T L Bailey, G M Church, B De Moor, E Eskin, A V Favorov, M C Frith, Y Fu, W J Kent, V J Makeev, A A Mironov, W S Noble, G Pavesi, G Pesole, M Regnier, N Simonis, S Sinha, G Thijs, J van Helden, M Vandenbogaert, Z Weng, C Workman, C Ye, and Z Zhu. Assessing computational tools for the discovery of transcription factor binding sites. *Nature Biotechnology*, 23(1) :137 – 144, 2005.

[87] T Tsunoda and T Takagi. Automatic extraction of position specific cooccurrence of transcription factor bindings on promoters. *Pac Symp Biocomput*, pages 252–63, 1998.

[88] J van Helden, B Andre, and J Collado-Vides. Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *J Mol Biol*, 281(5) :827–42, 1998.

[89] J van Helden, A F Rios, and J Collado-Vides. Discovering regulatory elements in non-coding sequences by analysis of spaced dyads. *Nucleic Acids Res*, 28(8) :1808–18, 2000.

[90] J Walter and M D Biggin. Measurement of in vivo DNA binding by sequence-specific transcription factors using UV cross-linking. *Methods*, 11(2) :215–24, 1997.

[91] W W Wasserman and J W Fickett. Identification of regulatory regions which confer muscle-specific gene expression. *J Mol Biol*, 278(1) :167–81, 1998.

[92] W W Wasserman, M Palumbo, W Thompson, J W Fickett, and C E Lawrence. Human-mouse genome comparisons to locate regulatory sites. *Nat Genet*, 26(2) :225–8, 2000.

[93] A S Weinmann and P J Farnham. Identification of unknown target genes of human transcription factors using chromatin immunoprecipitation. *Methods*, 26(1) :37–47, 2002.

[94] E Wingender, X Chen, R Hehl, H Karas, I Liebich, V Matys, T Meinhardt, M Pruss, I Reuter, and F Schacherer. TRANSFAC : an integrated system for gene expression regulation. *Nucleic Acids Res*, 28(1) :316–9, 2000.

[95] C H Yuh, H Bolouri, and E H Davidson. Genomic cis-regulatory logic : experimental and computational analysis of a sea urchin gene. *Science*, 279(5358) :1896–902, 1998.

[96] Z Zhang and M Gerstein. Of mice and men : phylogenetic footprinting aids the discovery of regulatory elements. *J Biol*, 2(2) :11, 2003.

[97] J Zheng, J Wu, and Z Sun. An approach to identify over-represented cis-elements in related sequences. *Nucleic Acids Res*, 31(7) :1995–2005, 2003.