UNIVERSITAT DE BARCELONA

Detection of Transcription Factor Binding Sites by Means of Multivariate Signal Processing Techniques

Erola Pairó Castiñeira

Aquesta tesi doctoral està subjecta a la llicència ***Reconeixement 3.0. Espanya de Creative Commons***.

Esta tesis doctoral está sujeta a la licencia ***Reconocimiento 3.0. España de Creative Commons***.

This doctoral thesis is licensed under the ***Creative Commons Attribution 3.0. Spain License***.

Detection of Transcription Factor Binding Sites by Means of Multivariate Signal Processing Techniques

Erola Pairó Castiñeira

PhD in Engineering and advanced technologies

Department of electronics, University of Barcelona
Institute of BioEngineering for Catalonia
Supervisors: Santiago Marco Colás and Alexandre Perera Lluna

Abstract

Gene expression is a complex and highly regulated process. Most of the regulation is controlled by short DNA sequences that can be bound by some proteins called transcription factors (TF). Binding to these sites the transcription factors can start the transcription of mRNA, stop it, or just control the amount of mRNA produced. The DNA binding sites of these transcription factors have some specific characteristics: (1) They are short sequences (2) They can be located anywhere in the genome and (3) they are degenerated, which means that some mutations in the binding site sequence do not alter its binding functionality. These characteristics made impossible to look for a specific sequence in a specific DNA region and, in order to find these binding sites, first they have to be modelled.

Due to the importance of gene expression in the study of cell differentiation and its implication in some genetic diseases, many computational models and experimental processes trying to describe binding site motifs and then look for them into a genome have appeared in the last 10 years. The computational models can be divided into two main groups: the motif discovery methods which try to find binding sites within a set of co-regulated sequences without previous knowledge and the motif search methods which model the binding sites using previous known motifs and then try to locate binding sequences that fit the model.

The focus of this thesis is to use the conversion from symbolical to numerical DNA and the previous knowledge of binding site sequences in order to construct models for DNA motifs. In this context, known multivarite signal processing techniques can be the ideal tools to construct models that can take into account interdependences without needing a large number of sequences or a high computational time.

First a characterization of the transcription factors was performed, using

the relationship TF-gene and also the complexity of the binding sites, and then two different detectors were built.

The first detector converts the DNA motif matrix into a numerical matrix and uses a Principal Component Analysis (PCA) to model the binding sites. The information of the interdependences is calculated using the covariance which is a second order statistics. The Q-residuals of the PCA model can be used to distinguish between binding sites and genomic sequences.

The disadvantage of this first model is that it is difficult to interpret. Converting the DNA symbolical matrix into a DNA numerical cube allows the application of PARAFAC which has a unique solution and is, therefore, easier to interpret it. Since the PARAFAC models have a biological meaning, their scores can be combined with the PARAFAC Q-residuals in order to construct a quadratic detector.

A tots els Miquels Pairó, i també a la meva àvia Magda.

Acknowledgements

First of all I would like to thank my thesis supervisors Dr. Alexandre Perera and Dr. Santiago Marco for giving me the opportunity of doing this thesis and guiding me through the whole process.
Most of the work included in the thesis was done in close collaboration with Joan Maynou, who participated into the creation of the R software and some analysis. Without his work this thesis wouldn't have been possible.
During the thesis I was working in the Intelligent Signal Processing (ISP) group in the University of Barcelona, the Artificial Olfaction group in the IBEC and the SysBio group in the Politechnical University of Catalonia, I have coincided with many people over all these years and I can say that i learned something from each one of them, so I would like to thank them for all their help and support. More than just a professional journey, a thesis is also a personal adventure with good and bad moments, and inevitably some of the people sharing the adventure with you will become more than working colleagues, friends. I would like to mention specially Benjamin Auffarth for always pushing and challenging me, Victor Pomareda for his conversations about life and work, Didier Domínguez and Luis Fernández with whom luckily I can always count on. Also Oriol Canela and Marc Webber who were not part of the working group, but with whom the conversations at lunch time were both refreshing and useful.

I spent some months in the University of Copenhagen, the faculty of Life Sciences doing a PhD Stage. I would like to thank all the people in the food technology group for their help during my months there, specially Rasmus Bro and Jose Amigo who were my supervisors during the stage.

Finally I would like to thank my family and friends for their constant support during these years. My mother Roser Castiñeira, my father Miquel Pairó, my brother and his girlfriend Miquel and Gemma and my sister Nú ria, thank you very much.

Contents

List of Figures

LIST OF FIGURES

List of Tables

1

Introduction

The genetic information of every living organism is contained in the double-helix Deoxyribonucleic Acid (DNA). This information is encoded in a four letter alphabet composed by the four DNA nucleotides: A, C, G, T.
The basic unit of genetic information is a gene, a relatively short string of DNA nucleotides which contains the information necessary to create proteins, responsible for most of the cellular processes. The Central Dogma of molecular biology establishes that the DNA information stored in a gene is first passed to the Ribonucleic acid (RNA) in a process called transcription, and then translated into proteins.
Although this general view of the information flowing from gene to RNA and then to protein gives a basic idea of the concepts of gene expression it does not account for the great complexity of the process. Actually, just a 10% of the human genome is composed by genes. Many of the remaining 90% contains functional elements that control when, where, and in which amount a protein is needed in the cell. This regulation of the gene expression is crucial because many processes like cell differentiation and some responses to specific signals need a control of the expressed proteins at each specific time.
A more general view is given by the figure 1.1 where the basic steps that go from DNA to the final protein are explained. In the first part of the figure it can be seen that the gene is preceded by a non-coding upstream region, in blue, where the elements that control the transcription bind. This region is called promoter of the gene. The gene itself also contains some non-coding regions called introns (also represented in blue) interspersed within the exons or coding regions. When the transcription of the gene is triggered by the binding of some proteins to the promoter, the RNA polymerase

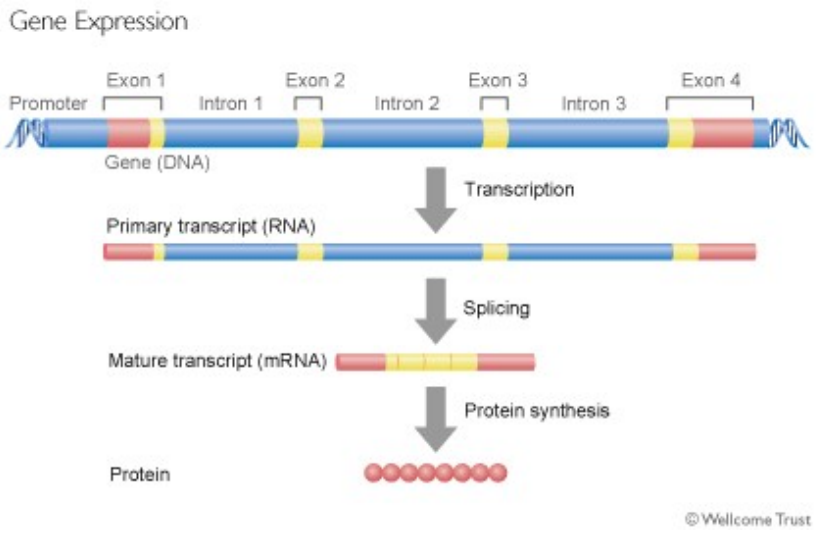


Figure 1.1. General description of the processes that lead from DNA to the protein, where the non-coding sequences are represented blue, and the exons in red and yellow. The DNA sequence is composed by the promoter and the gene which, in turn, is composed by introns and exons. Both introns and exons are transcribed and then the mRNA is modified by the splicing process and the mature mRNA is translated into a protein. Source: http://www.ncbi.nlm.nih.gov/probe/docs/applexpression/

creates a complementary RNA sequence. The result of the transcription of the gene is the messenger RNA (mRNA) but, as it has still to be processed, in eukaryotes it is known as the pre-mature mRNA or pre-mRNA which is shown in the second step of the figure 1.1.

The introns are cut from the pre-mRNA and then the exons are combined to form the mature mRNA. This process, known as alternative splicing, accounts for the large diversity of proteins present in the eukaryotic organisms, which largely exceeds the number of genes or transcript units. The mature RNA is then translated outside the nucleus where it can be further processed. The final step shown in the figure 1.1 is the protein synthesis or the translation from mRNA to protein. The ribosome travels along the mRNA translating the information into an aminoacid chain using the genetic code, where each unit of three nucleotides, a codon, encodes for an aminoacid. The genetic code also includes some start and stop codons and is degenerated because different codons can code for the same aminoacid. The resultant polypeptide folds into a functional protein.

To unravel the basic questions of gene expression, the genomes of the living organisms

started to be studied many years ago. In 1995 the first genome of a free living organism, the bacterium *Haemophilus influenzae* was published by Craig Venter's laboratory (Fleischmann et al., 1995). Since then, a genomics revolution started and the genome of many organisms has been sequenced. The list of sequenced organisms has grown enormously in the last years, specially since the publication of the human genome (Lander et al., 2001). Nowadays, many of the questions of biology can only be addressed using computational analysis of large data. For this reason the published genomes are available in large Internet databases and biologists, computer scientists and statisticians collaborate to study them.

18 years ago the International Nucleotide Sequence Database (INSD) was created with the aim to collect and exchange all publicly available DNA data in the different databases. INSD is a collaboration between the Database of Japan (DDBJ), the European Molecular Biology Labs (EMBL), and the Genbank which is funded by the U.S. National Institute of Health. The data is exchanged between the three databases in a daily basis and the major contributors are the individual scientists and the genomic project groups. Each database uses its own standard format, but other formats such as FASTA, which is the accepted format in all the analysis software, are considered in all databases.

EMBL database (Kanz et al., 2005) is located at the European Bioinformatics Institute (EBI). It has a web-based interface and it includes tools (Blitz, Fasta, Blast) to allow external users to compare their own sequence against the available data. Most of these tools are part of the European molecular biology open software suite (EMBOSS), which is a collection of open-source packages created in order to allow bioinformatics analysis (Rice et al., 2000). DDBJ is the main nucleotide database in Asia and its data comes mostly form Japanese researchers (Miyazaki et al., 2004). Genbank (Benson et al., 2012) is accessible through the NCBI Entrez retrieval system, which integrates data from the major DNA and protein sequence databases along with taxonomy, genome, mapping, protein structure and domain information, and the biomedical journal literature via PubMed. A new release of the database appears every two months.

Some of the most useful databases for bioinformaticians and computational biologists are part of the ENSEMBL project (Flicek et al., 2011). Started in 1999, this project includes a gene database, with gene ID, the name and the location in the genome, a variation database that can show homologs and alignments along many genomes and

also a regulatory database which includes information in a cell-type and in a general basis about the regulatory regions within a genome and the elements that can be found there. It is updated regularly and in the current version (ensembl76) there is information for more than 60 species. Apart from the databases, the ENSEMBL webtool allows the use of many tools as biomart which can proportionate a lot of extra information as the protein or transcript ID for a gene, map it to some external IDs or find homologs and orthologous genes.
The UCSC Genome browser contains the sequence and working draft for a large collection of genomes. In its genome browsers, there are also annotations about the genes contained, the splicing sites, the conservation of the sequence among organisms as well as tools to analyse the genomes. It is mostly important for being the home of the ENCODE project whose aim is to annotate all the functional sites within the Human genome using experimental data (Raney et al., 2011) and of the Neanderthal project which has information about the Neanderthal genome and its similarity with the human genome (Green et al., 2010).
Databases of some functional parts of the genome also exist. The Eukaryotic Promoter Database (EPD) is an annotated non-redundant collection of eukaryotic POL II promoters, for which the transcription start site has been determined experimentally (Périer et al., 2000). It is structured in a way that facilitates the extraction of promoter subsets. The 2012 release contains 4806 promoters of several organisms. The annotations include many cross-references to EMBL, swiss-prot (a protein database), other databases and bibliographic references. Since most of the TFBS are located at the promoter sequence of the regulated genes, this database is of particular interest when studying TFBS.

1.1 Gene Regulation

The regulation of gene expression can take place at every step, starting at the genomic level with e.g. gene methylation, and continuing through the transcriptional level, the translational level until the post-translational level through protein degradation or modification. Most of the proteins produced just in a tissue or in response to a signal are regulated at the transcriptional level, because a failure in this first step make all

the others redundant.

1.1.1 Transcriptional Regulation

The accessibility of DNA sequences to the transcriptional machinery can be controlled by how the DNA sequences are packed in the cell, this makes chromatin structure a first and important step in transcriptional regulation of gene expression. The second step is mediated by some short DNA sequences which are bound by specific proteins called transcription factors (TF).

1.1.1.1 Chromatin mediated regulation

DNA binds to histones in a structure that is known as chromatin. The basic unit of chromatin is the nucleosome, which is composed by the DNA wrapped 2 times in 8 histone molecules. This structure is further compacted into a solenoid in order to prevent genes to transcript. In the active genes the nucleosome structure is simpler or nucleosomes are simply removed (Mohd-Sarip and Verrijzer, 2004).
In order to allow TF to have access to the regulated genes, the chromatin structure around the gene should be modified. This is usually done by the chromatin remodelling factors, which can act in three different ways:

- Altering the association of the histone molecules in the chromosome which allows the TF to bind.
- Moving the nucleosome along the DNA.
- Displacing the nucleosome to another DNA molecule.

The modification of the histones is not only achieved through the participation of the remodelling factors. Histones have a complex pattern of post-transcriptional modifications which interact with each other and alter chromatin structure. One example is histone acetylation, which is usually found in regions where chromatin is not tightly packed.

1.1.1.2 Transcription Factor Biding sites

The cell transcriptional machinery needs a signal in order to know where and when a protein is needed. A gene embedded in a random DNA would not be expressed because this signal would not be available. The proteins responsible to give the signal are the transcription factors (TF) which bind to specific DNA sequences, the transcription factor binding sites (TFBS). TFBS are usually short sequences, with no more than 20 bp, and degenerated. This means that some non-identical, but similar, sequences can have the same functionality. Once a TF is bound to a specific site in the DNA, it interacts with other TFs and also with other molecules in order to signal the amount of protein needed. One of the main characteristics of TFs is the cooperation between them in order to regulate gene expression.

TFBS are mostly located at the upstream region of the transcription start site (TSS) which is called promoter of the gene. In this region two main kinds of binding site sequences can be found. Those sequences which are directly involved in the process of transcription and those that are only found on some specific genes. An example of the first class of DNA binding sites is the TATA box which is found in almost all genes 30 bp upstream of the TSS and that is known to give information about the location of the TSS. Examples of the second kind of binding sites are the ones that are involved in the transcription of a gene in a specific tissue or following a specific signal.

Some other binding site sequences are far from the transcription start site and can be more than 10 Kbp upstream of the regulated gene. Most of them are enhancers or silencers of transcription. Some others are insulators which do not alter directly the expression of a gene but block alterations of the DNA structure induced by enhancers and silencers in order to prevent the transcription of some other gene to be altered. (Latchman, 2008). In the figure 1.2 the basic picture of transcription is shown with the involvement of TFBS and chromatin structure. As it can be seen, some DNA regions have a high density of nucleosomes which make the access to the DNA for TF and other molecules difficult. The chromatin remodelling factors make some other regions less tightly packed. In these less-packed regions there are some binding modules where TF can bind and collaborate to control the transcription. The TATA-box complex, formed by many transcription associated factors (TAF), indicates where the transcription start

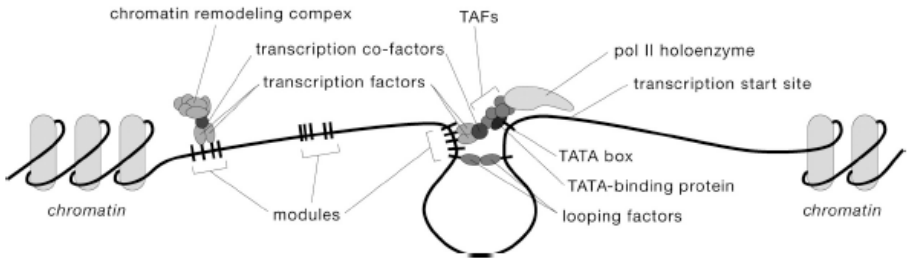


Figure 1.2. Schema of the mechanisms involved in the transcriptional regulation. The chromatin remodelling factors unpacked the chromatin in the regions where the genes should be expressed. The unpacking allows the transcription factor modules to be bound by the collaborating transcription factor binding sites and the TATA-box that indicates the initiation of transcription (Sandve, 2008).

site of the gene is.

Transcription factors are proteins with different 3-dimensional structures and different binding domains, which is the part contacting DNA. They can be classified according to the main characteristics into four basic superclasses: zinc coordinated, Basic domains, Helix-turn-Helix and β-scaffold. This four superclasses can be divided in classes, families and sub-families according to the different binding domains. The main TF families are:

- **Homeobox:** All the transcription factors of the homeobox family have a region of homology of approximately 180 bp. This region encodes for 60 amino-acids that form an helix-turn-helix motif which is the binding region.
- Cys_2His_2 **zinc finger:** These transcription factors have from 7 to 11 atoms of zinc per molecule, which makes zinc the crucial component of the structure. The binding region is composed by multiple fingers consisting in an α-helix and an anti-parallel β-sheet. The name is because each finger also has 2 cysteines and 2 histidines.
- **Multi-cysteine zinc finger:** These TF are activated by forming a hormone-receptor complex, and are an example of TF activated by a specific signal.
- **Leucine zipper:** Leucine zipper are characterized by a Leucine rich region in which successive Leucine residues occur every seventh amino-acid. The leucine

rich region is not the DNA binding domain, but it has an indirect structural role because it forms two symmetric dimers in the adjacent regions which will be the basic DNA binding domains.

- **Helix-loop-helix:** The helix-loop-helix is formed by two amphipatic helices containing all the charged amino-acids separated by a non-helical loop. It has a similar role than leucine zipper allowing dimerization.

Most of the transcriptions factors can be classified into one of the described structural families, but other families exist with different binding domains and also some relationships between domains can exist, even if they are not frequent.

1.1.1.3 Post-transcriptional Regulation

The result of the transcription is an mRNA molecule which includes the transcription of the coding regions, or exons, and the non-coding regions or introns. This is called the pre-mRNA. As soon as the $5'$ end of the nascent transcript is available, the pre-mRNA processing starts.
The first step is the mRNA capping which consists in altering the 5' end of the mRNA in order to prevent its degradation. Still in the nucleus and co-transcriptionally a second step is performed. The pre-mRNA is modified by means of the alternative splicing. Alternative splicing is a process that separates the introns and the exons of the transcript, allowing the introns to combine within them in order to form different mature mRNAs that will finish in different expressed proteins. This process is responsible for the large diversity of proteins in most eukaryotic organisms. Alternative splicing is controlled by the binding of the spliceosome, which is a complex of some transcriptional proteins and small RNAs (sRNAs) to the RNA splicing sites. Similarly to the transcription factor binding sites, the splicing sites accept some variation in their sequences without loosing their function, but they usually have a more well conserved 5' and 3' sites which are considered the core of the splicing site sequences. (Proudfoot et al., 2002; Wang and Burge, 2008).
The mature RNA is then transferred out of the nucleus in order to be translated to proteins. But mRNA levels are only partly correlated with protein expression levels (with a 40% or a 50% of correlation), because there also is a strong translational regulation. The translational regulation is specially important when the cell processes need some

abrupt change in protein expression, which is common in cell response to stress or cell apoptosis (Mata et al., 2005). This regulation is mainly due to the effect of microRNA (miRNA) that are small regions of RNA (typically of 20 bp) which bind to partially complementary sites in the mRNA and repress its expression (Hammell, 2010).
Further steps in the post-transcriptional regulation are the regulation of protein activity, for example via the kinase phosphorylation which is a hallmark in signalling cascades, or protein degradation.

1.2 Experimental determination of binding sites

As the identification of binding sites is a major step in the comprehension of the protein synthesis, many experimental methods try to characterize them. The variability in the methods is huge: some of them are applied genome-wide while others can be only applied to promoter sequences, some of them find regions where protein-DNA interactions are possible without knowing the specific protein while others are only useful for a specific transcription factor (Elnitski et al., 2006).
The most known technique to find regions of protein-DNA binding is the Deoxyribonuclease sensitivity, while, when a specific transcription factor is searched genome-wide, the most widely used method is the chromatin immunoprecipitation.

1.2.1 DNaseI Sensitivity

Deoxyribonuclease (DNase) is an enzyme that catalyses the hydrolytic cleavage of DNA. The degree of response from DNA to DNase can be classified as generalized sensitivity or hypersensitivity. Generalized nuclease sensitivity appears in all the expressed genes and is correlated with relatively large regions of open chromatin due to the presence of acetylated histones. Hypersensitivity appears in short DNA stretches (from 100bp to 400 bp) with extreme sensitivity to the cleavage effects of the enzyme. This effect is related to functional non-coding regions: promoters, enhancers, silencers, origins of replication, recombination elements and structural sites of centromers and telomers. It is associated with the removal of nucleosomes or the presence of modified histones (e.g. methylated) because they reduce the affinity from DNA to nucleosomes. DNaseI Hypersensitivity (DHS) has been widely used as a method to discover the presence of

a binding site when the specific protein is not known because it is a good indicator of the presence of an active promoter of a gene.
There are different methods to calculate the DNase sensitivity and the accuracy varies among them, from an error of 500 bp in the first methods of 1980 to the near nucleotide resolution using quantitative Polymerase chain reaction (qPCR) (Crawford et al., 2004). More recent techniques have been developed that allow the use of the DNaseI sensitivity in a genome-wide scale such as quantitative chromatin profiling (Dorschner et al., 2004) and massively parallel signature sequencing (Thurman et al., 2012).

1.2.2 Promoter analyses

Gene expression experiments can measure the production of a reported protein in response to cis-acting regulatory signals, for example using fluorescent proteins. When a enhancer is inserted into the promoter sequence of the gene, it produces a gain of function whereas introducing a mutation in a known binding site can produce a loss of gene production. The main disadvantage of these experiments is that the created cell lines will not provide an *in vivo* environment for the cell.
The study of *in vivo* gene expression is more technically difficult but it can provide conclusions that are not possible with cultured cells, such as the action of a specific transcription factor in a specific biological pathway (Hallikas et al., 2006).

1.2.3 Protein Binding assays

1.2.3.1 EMSAs

Electrophoretic mobility shift assay (EMSA) are the historically way to report the interactions between DNA and proteins. It is based on the idea that the mobility of a protein-DNA complex is less than the mobility of the free DNA. Usually these assays are performed for qualitative purposes but under some conditions quantitative data of the binding strength can also be retrieved from the experiments (Hellman and Fried, 2007). In the assay, solutions of protein and nucleic acids are combined and the mixtures are then subjected to electrophoresis through a polycramide gel. Typically the protein bound DNA will migrate slower than the free nucleic acid. The technique is simple and robust but it also has some disadvantages. The most important ones

are that the electrophoretics shifts depend on more things than simply the molecular weight of the bound protein, that the electrophoresis is not performed in a chemical equilibrium of the protein-DNA complexes and that, once a protein-DNA complex is found, it is not straightforward to find the binding sites within the genome.

1.2.3.2 ChIP assays

Chromatin immunoprecipitation is the most used experimental technique to determine whether proteins bind to specific regions of the chromatin *in vivo*. The steps of the Chromatin immunoprecipitation experiments which can be observed in the figure 1.3 are the following (Carey et al., 2009):

1. The living cells are cross-linked using formaldehyde which serves to fix protein-DNA interactions and then they are lysed.

2. The chromatin is sheared into short fragments (0.2-1 Kb) using sonication or enzymatic digestion.

3. The protein bound DNA fragments are then immunoprecipitated using the specific antibodies.

4. Cross-linking is reversed.

5. DNA is purified and assayed to determine the sequence bound by the protein.

Two main techniques allow to characterize genome wide binding sites. First the ChIp-chip which combines the ChIP with DNA micro-arrays appeared and, more recently, the ChIP-seq that uses next generation massive parallel sequencing.
In the ChIP-chip experiments (Ren et al., 2000) after the immunoprecipitation a Polymerase chain reaction (PCR) is used in order to to amplify the DNA signal. Then, the IP-enriched DNA is labeled with a fluorescent molecule and genomic DNA prepared from the ChIP input is used as a reference and labeled with a different fluorescent molecule. The two probes are then combined and hybridized to a single DNA microarray. The results of the hybridization allow one to identify which segments of the genome were enriched in the IP. Since the precise location of each arrayed element is known, construction of a genome-wide map of *in vivo* protein-DNA interactions is possible.

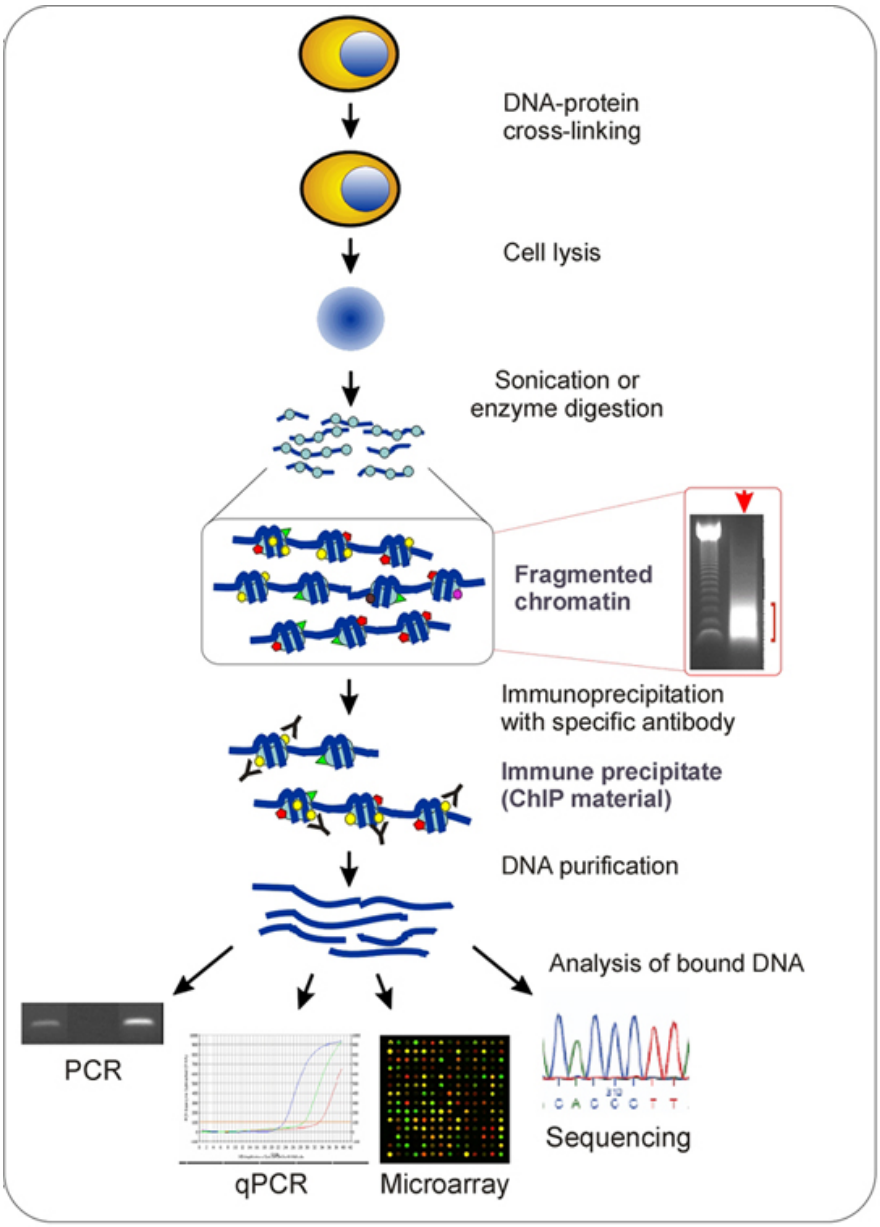


Figure 1.3. Steps of the ChIP experiments. First the DNA is cross-linked with formaldehyde and then the cell lysis is performed. The chromatin is fragmented and the fragments are immunoprecipitated using specific antibodies. Finally DNA is purified and some technique is applied in order to know the DNA sequences bound by the protein (Collas and Dahl, 2008)

Although ChIP-chip allows the genome-wide identification of binding sites, ChIP-seq is now becoming the most used technique because of its high-resolution, cost-effectiveness and the ability to sequence millions of bases in a short time. In this technique, the DNA fragments obtained from the ChIP experiments are sequenced using the next generation genome sequencers and the results are then mapped to the reference chromosome.

1.2.4 Transcription Factor Binding Sites Databases

Some databases collect the experimental found binding sites for the transcription factors. The most used TFBS databases are TRANSFAC which has a public and a commercial release and JASPAR which is entirely public. But also smaller databases of transcription factor motifs exists. These databases are useful to model binding sites, to predict their position within the genomes and to construct the transcriptional network that regulates the expression of a gene.

1.2.4.1 TRANSFAC Database

TRANSFAC database is a database of manually annotated and experimentally proven binding sites that also provides data about the consensus sequences, the binding profiles and the regulated genes. The public version of TRANSFAC 7.0 (2005) contains data for 2397 genes and 6133 factors (7915 sites) and is available for non-commercial purposes.

The first version of TRANSFAC database appeared in 1988, when the importance of gene regulation and, specially, of the transcription factor binding sites became obvious. The aim of the database was to incorporate the quickly growing number of binding sites that were collected and map them into the corresponding promoter. Since then, TRANSFAC has become a large database of binding sites. Data in TRANSFAC is organized by transcription factors and in each transcription factor there is included information about all the known sites and the experimental method used to retrieve them, the gene regulated and the position of the binding sites related to the transcription start site (Wingender, 2008).

Few years ago, TRANSFAC became part of the biobase company and the new versions from TRANSFAC are not publicly available. Apart from an increasing collection of

binding sites, the new TRANSFAC versions also include micro-RNA information, because the scope has been expanded from the study of transcriptional networks to the study of gene-regulation networks. Nowadays the TRANSFAC professional database has data available for 18211 factors (including miRNA), 34742 sites and information about the 70869 genes from different organisms controlled by these TF.
Two new databases associated with TRANSFAC have been released, the TRANScompel which studies the physical and functional interactions between transcription factors and the TRANSpath which can be used to study gene pathways.

1.2.4.2 JASPAR Database

JASPAR database is the largest open-access collection of transcription factor binding sites (Mathelier et al., 2014). The JASPAR core database includes a curated non-redundant set of profiles for binding sites of multicellular eukaryotes that come from published articles, mainly fro *in vitro* experiments, but with the development of Chip-seq methods, some published chip-seq datasets have also been added to the database. The current version has profiles from 590 transcription factors from different organisms including vertebrates, plants, fungi, insects, nematodes and urochordata. The data is organized in a matrix way, so it is easy to model the binding motif and look for binding sites within genomic sequences. The data for each transcription factor includes the binding sequences, the name of the transcription factor, its family and the methodology used to construct the matrix.
Besides the JASPAR core database, JASPAR also includes many separate collections from matrix profiles that cannot be included into the core database because they don't fit the criteria. For example the JASPAR family which includes 11 profiles with the shared properties of the structural classes of TF, the JASPAR phylofacts which includes 174 profiles extracted from phylogenetic studies or other databases which are non-TF binding profiles. The total number of profiles including JASPAR core and the other collections is 840.
The new version of JASPAR can also be explored using new developed packages, as BioPython and a new R tool, which allow an easy access to all the information stored in the database.

1.2.4.3 Other Databases

Smaller public databases of binding sites exist. One example is the ABS database which includes data of experimentally verified binding sites identified from the promoters of orthologous vertebrate genes. The database includes a total of 100 orthologous genes and 610 binding sites corresponding to 68 transcription factors (Blanco et al., 2006). Mapper is a database of 1079 built models to describe different transcription factors and it also includes the annotations of their positions within the genome. The data comes from human, mouse, fly and worm genomes (Marinescu et al., 2005).
VISTA is a database of distant-acting enhancers in human and mouse genomes. The enhancer candidates are chosen between highly conserved sequences or ChIP-seq data and then they are verified *in vivo*. When the *in vivo* validation works, a map with the expression of the enhanced genes is also provided (Visel et al., 2007).
Another database, focused in one organism, is the RedFly 2.0 database which incorporates the information about all the verified regulatory modules in *Drosophila melanogaster*, the affected genes and the expression patterns that they direct (Gallo et al., 2011).

1.2.5 Interaction Databases

Cell processes are mainly regulated by complex protein-protein interactions which can be described as protein interacting networks. These interactions can be physical interactions between proteins, genetic interactions, or also interactions known to catalyse consecutive steps in a cell pathway.
Even though the construction of databases that describe these interactions is complicated, the network view of the genome has become increasingly popular and many public databases try to annotate the different protein interactions. Most of them only take into account direct physical interactions, but others try to annotate all the functional interactions between proteins.
This databases can give a systems biology view of the transcriptional regulation, giving information about the physical interactions between binding sites that govern a gene regulation. They also can give information about where in a pathway a transcriptional regulation is important.

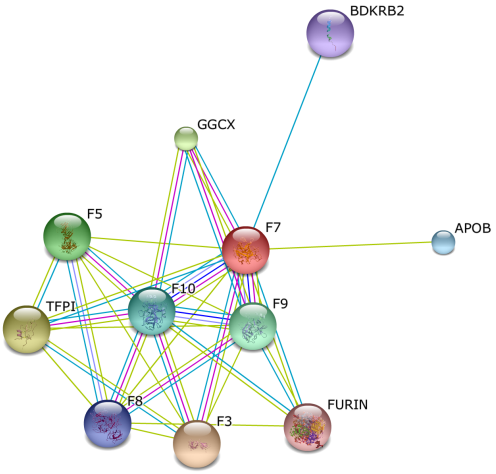


Figure 1.4. Network of protein interactions for the F7 human protein provided by the STRING database. The described interactions come from different sources: experimental verification, text mining, databases, co-expression, etc. Each colour represents a different kind of interaction.

One of the databases that integrate physical and functional interactions is the search tool for the retrieval of interacting genes (STRING) (Szklarczyk et al., 2011). STRING integrates data from many sources, as experimental data, database search, text mining, co-expression, homology, co-occurrence and neighbourhood, to provide the functional interactions for a given protein. The main advantages of STRING are that it incorporates an scoring scheme to show the reliability of each interaction and also that it has a user-friendly interface. Given a protein, STRING outputs a network of protein interactions and a list of the interacting proteins, its function, the sources of the interaction and the confidence score. This confidence score is benchmarked independently for each source and then a combined score is computed. One example of the network constructed by string can be seen in the figure 1.4 where the F7 interaction network is shown. The different colours of the edges represent different kinds of interactions. The last version of STRING provides data for 1100 genomes, going from bacteria to humans.

On the other hand the molecular interaction (MINT) database focuses on the pure physical protein interactions (Chatr-aryamontri et al., 2007). Unlike in the STRING database, the data in MINT only includes experimental interactions extracted from published papers and the inferred interactions are excluded. Nowadays MINT includes

over 95000 physical interactions between 27461 proteins from 325 organisms. Most of the interactions, a 90% of them, come from genome-wide experiments.
The Biological General Repository for Interaction Datasets (BioGRID) is a public database that includes genetic and physical interactions (Stark et al., 2011). The 3.1.93 release has 375704 non-redundant interactions and 557934 raw interactions from major model organisms as *Saccharomyces cerevesiae*, *Arabidopsis thaliana* and *Homo sapiens*. Current efforts are focused on the areas relevant to human health. It also incorporates a web interface to look for the interactions of a specific protein and to download the data.
Many other databases exists, and many of them are included in PSIQUIC which is focused in molecular interactions. PSIQUIC is a tool that looks at many interaction databases (including protein-protein interactions) and gives the results for all the databases in a single search (Aranda et al., 2011).
A useful application in order to understand the interaction between transcription factors and their regulated genes is the Sabiosciences database, which combines a text mining algorithm with the annotations in the USC genome browser in order to find the regulated genes for a transcription factor.

1.3 ENCODE project

The Encyclopedia of DNA elements(ENCODE) is a project funded by the National Human Genome Research Institute whose aim is to identify all regions of transcription, transcription factor association, chromatin structure and histone modification in the human genome sequence. Summarizing, the project wants to find all the functional sites within the human genome and to make them publicly available, because the comprehension of these sites is of crucial importance in biomedical research.
It started in 2003 with the collaboration of a consortium of computational and laboratory-based scientists. In the pilot phase of the project a 1% of the human genome (30 Mb) was analysed. As many functional genomic elements are only active in certain cell types or in response to certain signals, the analysis was performed using different cell types (ideally it should be performed in all the cell types at every stage of development). This pilot phase was useful in order to evaluate the strategies for identifying various

types of genomic elements by means of high-throughput technologies (Material et al., 2004).

The second phase of the project started in 2007 and lasted 5 years, the objective was to interrogate the complete human genome. In the year 2012 the initial analysis of 1640 datasets involving 147 cell types have been published (Dunham et al., 2012). The results show that 80% of the components of the human genome have at least one biochemical function associated with them, which means that they participate in at least one RNA or chromatin associated event. This is much more than the 8% of bases under negative selection pressure that were expected to be functional from previous estimations. In the ENCODE initial results it is also shown that more than the 99% of the genome lies within a 1.7 Kb distance from a ENCODE annotated element and that a 95% of the genome is less than 8 Kb far from a DNA-protein interaction.

A manual catalogue of coding and non-coding DNA has been constructed, the GENECODE. And it has been observed that the protein-coding genes cover only a 2.94% of the genome, while the transcribed region is much larger. Additionally 119 DNA-binding proteins and some RNA polymerase components have been located in 72 different cell types using ChIP-seq.

A computational study of the ENCODE experimental results correlates quantitatively the RNA production with the chromatin modification and the transcription factor binding. The study of the location of 117 TF in five cell lines states that binding sites are not randomly distributed along the genome and that most transcription factors have collaborative associations that can be measured through co-ocurrence of the sites in the genome. The genome has been divided into 6 genomic regions according to three different criteria: (1) Binding active regions (BAR) and binding inactive regions (BIR) (2) Promoter-proximal regulatory modules (PRM) and gene-distal modules (DRM) and (3) High occupancy of TF regions (HOT) and low occupancy of TF regions (LOT). BAR are regions with a high amount of binding sites, and the presence of them is correlated with the gene density of a DNA region. The HOT and LOT regions are defined according to the region specific likelihood of co-ocurrence of TFBS. This means that HOT regions are defined as regions with a high co-occurrence of TFBS that only co-occur in this region. For instance, co-occurrences like the TATA-box that occur genome-wide are not taking into account in order to define HOT and LOT regions. Most of the HOT regions (approximately a 70%) are within 10 Kbp of a gene and only a 50% of the LOT

regions are close to overlap a gene. In the promoter regions the levels of association are higher than in the intergenic regions, but in the last ones, more specific associations can be found. (Yip et al., 2012)
The individual variations of the genome have been also studied by the project and it has been found that many functional variants within individual genomes lie in non-coding regions, and that most of them are found functional sites. This encourages to perform a whole-genome sequencing instead of a exon sequencing in the study of rare diseases. The study of 4860 single nucleotide polymorphisms (SNPs) associated to a disease by a Genome-wide association study (GWAS) also revealed that a 12% of these SNPs overlap with transcription factor binding sites and that a 34% overlap with DNase hypersensitivity regions (Dunham et al., 2012). These findings enhance the need for tools able to recognize TFBS into large genomic sequences, since the mutations occurring in genes are not sufficient to understand the causes of many diseases or interesting phenotypes.
The results of the functional elements found by the ENCODE project are annotated in the UCSC genome browser (Raney et al., 2011). The annotation includes sequences with quality scores, alignments, signals calculated from the alignments, and in most cases, element or peak calls calculated from the signal data. Each data set is available for visualization and download via the UCSC Genome Browser and it can also be retrieved using a meta-data system that captures the experimental parameters of each assay.

1.4 Sequence alignment

Sequences that have evolved from the same ancestor sequence are called homolog sequences. Even though they have diverged due to mutations, insertions and deletions occurred in the different genomes, they are thought to share a similar function and also most of their nucleotides. The concept of sequence alignment appeared in biology in order to find out whether two sequences are homologs, how did they diverge and also if the similarities are still enough to think that they have the same function within different organisms. Two sequences are aligned by writing them into rows. Identical characters are placed in the same column while non-identical characters are considered

a mismatch (mutation) or filled with a gap (insertion or deletion). The alignment with more identical positions between sequences is considered the best alignment.
The alignment can be performed using a pair of sequences, in order to see how similar they are, or with multiple sequences at the same time. This latter approach is useful to do an evolutionary analysis of genomes and also to find the conserved positions in functional sites, such as transcription factors. In some databases transcription factors come as independent sequences with different length, and in those cases, a correct alignment of the sites is a basic step in the construction of a valid model.

1.4.1 Pairwise Alignment

Pairwise alignment is used to find if two sequences are evolutionary related.
A first approximation to that problem can be easily found using dot matrices. To construct a dot matrix one of the two sequences is placed horizontally and the other vertically, the nucleotides in each position are compared and a dot is printed where there is a match. The result is a matrix where sequence matches appear in the form of sequences of dots in the diagonal. (W.Mount, 1998).
Even if this is an easy way to visualize the similarity between two sequences, a score is needed to find the degree of similarity between them. First a score is assigned to matches, mutations and gaps and then the total score is calculated as the sum of scores. The different scores can be represented using DNA substitution matrices. A very simple score function for nucleotides is presented in table 1.1 where each match is considered as a $+1$ and mismatches and gaps are equally treated as -1. Usually the substitution matrices used are the PAM matrices which are based on the evolutionary probability of mutations and gaps.

Once the score of the alignment is calculated, a significance measure is needed, because long sequences will always score higher than short ones despite its similarity. The significance is calculated by taking a set of sequences with the same characteristics of the studied ones and calculating the probability of randomly picking two sequences of this distribution and finding a similar or higher score.

Table 1.1. Simple substitution matrix where the score of each match is +1 and the score of a mismatch, a insertion or a deletion is -1.

	A	C	G	T	-
A	+1	-1	-1	-1	-1
C	-1	+1	-1	-1	-1
T	-1	-1	+1	-1	-1
G	-1	-1	-1	+1	-1
-	-1	-1	-1	-1	NA

1.4.1.1 Global alignment

In global alignment the best alignment between a pair of sequences is found and mutations, deletions and insertions are considered mismatches. To find the best alignment between two sequences of length n, all the possible alignments should be explored. The number of possible alignments is $\binom{2n}{n} = \frac{(2n)!}{(n!)^2} = \frac{2^{2n}}{\sqrt{\pi n}}$ which is large even for small n. This issue can be solved using dynamic programming.

The dynamic programming algorithm used to solve the global alignment problem is the Needleman-Wunsch algorithm (Needleman and Wunsch, 1970). Starting with a score function $S = 0$, the algorithm compares at each step the three possible combinations (a insertion a deletion or a match/mismatch) and chooses the one with higher score, and the new S score is computed. Having the Score $S(i-1, j-1)$, the new score $S(i, j)$ can be found using equation (1.1). This algorithm has been mathematically proven to provide the best alignment given a scoring function. Choosing a good scoring function is, thus, the critical step for alignment (Durbin et al., 1998).

$$S(i,j) = max \begin{cases} S(i-1, j-1) + f(i,j) \\ S(i-1, j) - d \\ S(i, j-1, -d) \end{cases} \tag{1.1}$$

The computational time for the Needleman-Wunsch algorithm increases as $O(nm)$ where n and m are the length of the sequences to align. The computer memory also increases as nm.

1.4.1.2 Local alignment

When aligning pairs of sequences, the most common problem is to compare extended regions of DNA (e.g entire chromosomes corresponding to two different specie). This regions are usually highly diverged sequences and just some small regions are under strong positive selection, the rest of the sequence has many noise that appeared through mutations. In this cases, global alignment would be useless because the only regions that really need to be aligned are the conserved regions.
A dynamic programming to solve the local alignment problem was also proposed, the Smith-Watermans algorithm (Smith and Waterman, 1981). This algorithm is similar to the Needleman-Wunsch algorithm but, when the score S becomes negative, it is set to 0 which means that a new alignment begins.
When the sequences to compare are too large, even the dynamic programming algorithms are too slow and need too many memory to be ran in a computer.
Some heuristic algorithms do not always give the optimal local alignment, but are the best option to match a sequence within a large database. This algorithms, called k-tuple or word algorithms, work in two steps. First they look for words of length l that exactly match a sequence, or that match a sequence over a score S higher than some threshold, and then use dynamic programming to finish the alignment. The first of this heuristics algorithms was the FASTA algorithm (Lipman and Pearson, 1985), and another example is the BLAST algorithm (Altschul et al., 1990) which is now the most used algorithm for local alignment.

1.4.2 Multiple Alignment

Similar genes are widely conserved across divergent species, often performing a similar function, and a simultaneous alignment of sequences of many organisms can find sequence patterns and give an evolutionary history of the sequence. In order to do that, the pairwise alignment methods need to be expanded to multiple sequence alignment (MSA), which tries to find a relationship between more than two sequences.
Most of the ideas from pairwise alignment can be applied also to MSA, such as the concepts of local and global alignment. But two questions arise (1) How can a MSA

be scored? and (2) What method can be used to efficiently find the optimal alignment? Regarding to the second question, the dynamic programming algorithms can be extended to k sequences, but the computational cost increases exponentially with k which means that, in fact, only small k and short sequences can be aligned using dynamic programming.
Many heuristic algorithms that do not guarantee the best alignment but give good approximations have been proposed. They can be divided into different kinds of methods (Notredame, 2007).

1.4.2.1 Progressive methods

Progressive methods construct a phylogenetic tree using the unaligned sequences and the two closest sequences of the tree are first aligned. Then the other sequences are added according to the distances into the phylogenetic tree. One example of an iterative algorithm is CLUSTALW (Thompson et al., 1994).
The main problem of this MSA algorithms is that they have a high dependence on the quality of the first phylogenetic tree. If the sequences are not closely related and the phylogenetic tree can not be trust, then the quality of the final alignment is also poor. This problem can be partially addressed using a library of weighted pairwise alignments to construct the first phylogenetic tree as T-coffee (Notredame et al., 2000). The library is constructed with the pairwise alignment of all the sequences, that is weighted according to the similarity between them.

1.4.2.2 Iterative methods

To avoid the dependency in the construction of the phylogenetic tree, the iterative methods put the previous algorithm into a loop where the tree and the alignment are estimated iteratively until convergence. Different algorithms reconstruct the tree in different ways but the basic idea is that the pairwise scores are recalculated during the construction of the alignment and then the tree is reconstructed which, in turn is used for the new alignment. An example of a iterative algorithm is MUSCLE (Edgar, 2004).

1.4.2.3 Machine learning approaches

Other approaches have been used to solve the problem of MSA. For example genetic algorithms (Notredame and Higgins, 1996) or more frequently Hidden Markov Models (HMM), which study the transition probabilities between sequences and are more general allowing local and global alignment (Eddy, 1998).

1.5 DNA Motif Detection

Even if the experimental detection of binding sites has become very effective, it is still a complex and expensive process. Motif detection algorithms can complement, or in some case substitute, the experimental determination of motifs like binding sites, splicing sites or miRNA.

Motif detection algorithms have also some difficulties to overcome, the most relevant one is that motif sequences can show some variability without loss of function, which makes impossible to look for a specific sequence. Other characteristics such as the shortness of the sequences and the fact that they can be located anywhere in the genome convert the detection of binding sites into a computational challenge. (Sandve, 2008)

Every motif detection algorithm has two main steps. First the construction of the model and then the scoring of a candidate sequence. Some algorithms use known motifs in order to find new instances in some candidate sequence, others try to find over-represented motifs within a set of unaligned sequences from co-regulated genes or using phylogenetic foot-printing. The first ones are the motif finding algorithms and the latter the motif discovery algorithms which are doing the two steps (modelling and scoring) at the same time. Both motif finding and motif discovery algorithms can be classified according to the models that they use for the binding sequences.

The first computational model for a binding site motif appeared at the 70's (Korn et al., 1977), which models the motifs like oligonucleotides. Since then, the increasing amount of data available made possible the appearance of many computational methods modelling binding sites and the first simple consensus models have been evolved to more complex models (Pavesi et al., 2004a; D'haeseleer, 2006; Sandve and Drablos, 2006; Hannenhalli, 2008).

Most of the motifs models are based on Position Specific Scoring Matrices (PSSM) (Stormo, 2000) which are matrices of weights of each nucleotide in each position and

that assume that each position within a binding site is independent. Since some experimental and computational studies suggested that interdependences between positions exist (Bulyk et al., 2002; Zhou and Liu, 2004; O'Flanagan et al., 2005; Tomovic and Oakeley, 2007), new methods appeared which use probabilistic models to model binding sites using interdependences.

1.5.1 Word-enumeration Models

A consensus is a way to describe a set of sequences using the most frequent nucleotide in each position. It can be considered the most perfect form of a binding site, the one that would be most likely bound for the corresponding transcription factor. But since the binding sites have a certain variability, a number of mutations e should be allowed when looking for binding sites using a consensus model, or some distances such as the Hamming distance (Hamming, 1950), the number of positions with different corresponding symbols, should be calculated.

These models are very rigid because they do not account for occurrences of different nucleotides in certain positions. In order to avoid this problem, the IUPAC code, which takes into account that different nucleotides can be present in some positions, is used to model the binding motifs. Figure 1.5 shows the binding sequences of a motif in a), the consensus sequence in b) and the consensus sequence using the IUPAC alphabet in c).

When a nucleotide is known to have some well-conserved positions, the algorithms looking for binding sequences using a consensus model can be improved allowing mutations just in the less-conserved positions. But it is difficult to find a trade-off between the mismatches allowed, the flexibility used to represent the sequence and the precision of the search.

Consensus models were the first models that appeared for binding motifs but, despite their simplicity and their limitations, they are still among the most used methods due to the low computational time and the high performance that they can achieve. Some examples are WEEDER (Pavesi et al., 2004b) and more recently a method using the DNA Gray code (Ichinose et al., 2012).

a)
```
TGACTCA
TTCCTGG
TGAGTCA
TGAGTCA
TGAATCA
AGTGTCA
TGAGTCA
TGAGTAA
TGATTAA
```

b) `TGACTCA`

c) `WKHNTVR`

d)

	1	2	3	4	5	6	7
A	1/9	0	7/9	1	0	3/9	8/9
C	0	0	1/9	2/9	0	5/9	0
G	0	8/9	0	5/9	0	1/9	1/9
T	8/9	1/9	1/9	1/9	1	0	0

e)

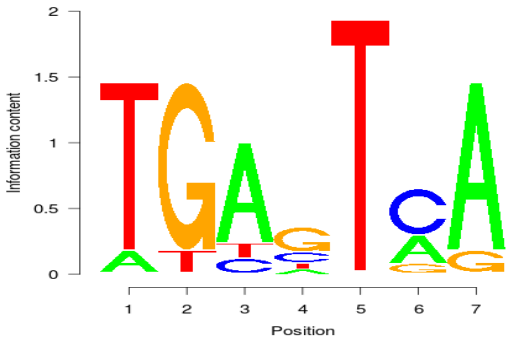


Figure 1.5. Sequences of a binding motif in (a), consensus sequence generated using the DNA alphabet in (b), consensus sequence using the IUPAC code in (c) PSSM matrix in (d) and finally the Logo representation of the sequence in (e).

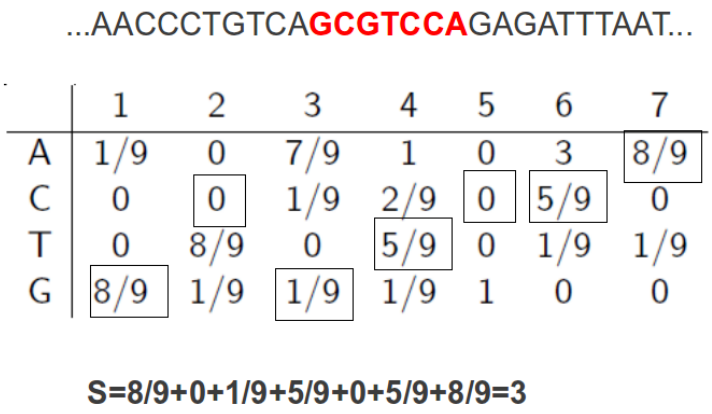


Figure 1.6. Calculation of the Score of a candidate sequence. It is calculated as the sum of the scores in each position of the binding site.

1.5.2 Profile Models

1.5.2.1 Position Specific Scoring Matrices (PSSM)

The most used way to calculate the model of a binding motif is to use PSSM. PSSM are $4 \times M$ matrices of frequencies of each nucleotide at each position, where M is the number of positions. For example, in the figure 1.5 (d) the PSSM for the binding motifs is shown. It can be calculated dividing the count of each nucleotide in each position by the number of sequences of the motif. Each row of the matrix is the frequency of one nucleotide in each position.

Once the PSSM is calculated, in order to calculate the score of a candidate sequence, the frequency of the corresponding nucleotide of the candidate sequence in each one of the positions is summed and the result is the final score of the sequence. One example can be observed in the figure 1.6 where the score of the sequence in red is calculated using the PSSM model of the example in the figure 1.5

A higher score means a high probability of being a binding site. Each one of the methods that use PSSM have a different way to calculate the significance of the scores of the candidate sequence.

Models using PSSM can be improved if the information per position is calculated

instead of the frequency (Osada et al., 2004). The information can be calculated, using also the nucleotide distribution in the background organism with the equation (1.2) (Schneider, 1997).

$$I(i) = \sum_b f_{b,i} log_2 \frac{f_{b,i}}{p_b}, \tag{1.2}$$

where $I(i)$ is the information, $f_{b,i}$ is the frequency of the b nucleotide at the i position and p_b is the genomic probability of the b nucleotide. The PSSMs calculated using information theory can be represented by a sequence logo which indicates the information of each nucleotide in each position as it is show in figure 1.5 e)

If the free energy of binding of a TF to its binding site is calculated as the sum of the free energy of the binding to each position (Berg and von Hippel, 1987), then the information per position can be related to the free energy of the binding, and the score of a sequence is related to the energy of binding of that sequence (Stormo and Fields, 1998). Even if this is not strictly true, in some cases it can be a good approach (Benos et al., 2002). For this reason, some PSSM methods use a biophysical approach to model binding sites (Roider et al., 2007).

The most famous motif discovery algorithms which use a PSSM to model the motif are MEME (Bailey and Elkan, 1994) and Gibbs sampling (Neuwald et al., 1995). Given a set of N unaligned sequences that contain k different motifs, the Gibbs sampling algorithm first divides each sequence into subsequences and randomly assigns each subsequence to one of the $M_0...M_k$ models, where M_0 is the model of the background sequences that do not belong to any motif and $M_1...M_k$ are PSSM models for each one of the k models. Then two steps are repeated until convergence: (1) A sequence s_i is selected and the corresponding model is recalculated (2) A model is sampled taking into account the probability that the selected sequence s_i was derived from that model. MEME uses a Expectation-maximization (EM) algorithm in order to find motifs in co-regulated sequences. First the subsequences of width W are chosen as a starting point to construct the possible models. Then the models are constructed and 1 iteration of the EM algorithm is run, the model with a highest likelihood is chosen and the EM algorithm is used until convergence to find the optimal model. Finally, the subsequences belonging to the found motif are erased and the previous steps are repeated until the k different motifs are found. Both algorithms have been recently updated in order to incorporate prior information or heterogeneous backgrounds (Thompson, 2003; Bailey

et al., 2010).

Also most of the motif finding algorithms are based on PSSM. MAST (Bailey and Gribskov, 1998) which is part of the MEME suite (Bailey et al., 2009), predicts the presence of one or more known motifs within a large genomic sequence. The motifs are modelled as PSSMs where the score of each nucleotide in each position is the logarithm of the frequency of the nucleotide, as in equation (1.3).

$$S_{b,i} = -log\frac{f_{b,i} + B}{p_b + B}, \tag{1.3}$$

where $S_{b,i}$ is the Score of the b nucleotide in the position i. $f_{b,i}$ is the frequency of the b nucleotide in the i position and B is a pseudo-count usually set to $B = 0.1$. p_b is the background probability of the b nucleotide, calculated using a Markov model of the background genome. The final score of a sequence, as it is explained above, is the sum of the scores for the corresponding nucleotide in each position. The p-value for the probability of the sequence being a binding site is calculated and, if more than one motif are studied ,the final result is the product of p-values. The probability that the product of p-values is due to the presence of the different motifs is the output of the algorithm. If only one motif is studied, it is equivalent to calculate the p-value.

MATCH is another PSSM algorithm, available from TRANSFAC (2005) database which uses the information per position. The score of a candidate sequence in MATCH is then calculated as in equation (1.4)

$$S = \sum_{i=1}^{L} I(i) f_{b,i}, \tag{1.4}$$

where S is the score, i the position of the sequence of length L $I(i)$ is the information calculated as in equation (1.2) but assuming that in the background all the nucleotides have the same probability ($p_b = 1/4$). As in the above equations $f_{b,i}$ is the frequency of the b nucleotide in the i position. Instead of using a p-value to determine if a sequence belongs to the modelled motif, MATCH algorithm calculates the similarity score of the sequence and the similarity score of the first five consecutive positions of the matrix, the core of the motif. They are calculated in equation 1.5

$$SS = \frac{Current - Min}{Max - Min} \tag{1.5}$$

Where Max and Min are the maximum and minimum scores given the PSSM matrix. These similarity score go from 0 to 1, and a threshold in the matrix similarity score and the core similarity score are used to decide whether a sequence is a binding site or not. Usually, as the core are the 5 most conserved positions, the cut-off for the core similarity score is set higher than the matrix similarity score.

MatInspector (Cartharius et al., 2005), part of the Biobase company works in a similar way than MATCH, calculating a matrix and a core similarity score in order to distinguish between binding sites and non-binding sequences. But the score of each position and the similarity scores are calculated in a different manner. The score per position is calculated using a coefficient C_i shown in equation (1.6).

$$C_i = \frac{100}{ln5} \sum_b f_{b,i} ln(f_{b,i}) + ln5, \tag{1.6}$$

where, as in previous equations, b is the nucleotide and $f_{b,i}$ is the frequency for each nucleotide in the i position. The similarity score is calculated in the same way than in MATCH.

In 2010 Maynou et al. (2010b) developed an algorithm which uses the Rényi entropy in order to model the binding motifs. The Rényi entropy is a parametric measure of entropy defined in equation 1.7

$$H_q(i) = \frac{1}{1-q} log_2 \sum_b f_{i,b}^q, \tag{1.7}$$

where q is a positive number and i the position within the binding site. When $q = 1$ the Rényi entropy is equivalent to the Shannon's entropy. To normalize the measure of H_q in the interval from 0 to 1 a new variable, the redundancy $R_q(i)$ of each position is calculated for the motif (equation (1.8)).

$$R_q(i) = 1 - \frac{H_q(i)}{H_q^{max}(i)} \tag{1.8}$$

When a candidate sequence is evaluated, the sequence is first added to the motif and the redundancy recalculated. The difference between the two redundancies, with and without the candidate sequence, is used as a discriminant measure. When the sequence does not belong to the modelled motif the redundancy will decrease if adding it to the model. In contrast, if the sequence belongs to the model, the redundancy will remain constant. A p-value is calculated to decide whether a candidate sequence belongs to

the binding motif.
Some binding sites can have two different profiles corresponding with two different types of sites for the same transcription factor. PSSM are not able to model these motifs. In order to avoid it, some methods use a mixture of profiles. In these models a binding site is represented for a weighted set of profiles, a higher weight means that the profile is more specific, and the score of a sequence is calculated using the weighted sum of the scores in each profile (King, 2003).

1.5.2.2 Models with interdependences

Substitutions in binding site positions do not occur independently, a substitution in a given position might imply a substitution in another position. PSSMs can be easily extended to take into account pairwise dependences, but usually this is not enough.
The first generalization of the PSSM were the weight array matrix (WAM) models, which are Markov models of the motifs (Zhang and Marr, 1993). A Markov model of order n is a probabilistic model which describes the probability P of a nucleotide X in a certain position i being b_i, depending on the previous n nucleotides, as it can be seen in equation 1.9.

$$P(X = b_i) = P(X = b_i|X_{i-1}...Xi - n) \tag{1.9}$$

These models have the disadvantage that the number of parameters to adjust increases exponentially with the order of the model. A way to overcome this difficulty is allowing permutations within the positions of the Markov model in order to find long-range interdependences without increasing the degree n of the Markov model. In the algorithm created by Ellrott et al. (2002), for every pair of positions i, j they calculate a dependence score as $G(i,j) = -log(p(i,j))$. They pick the two positions with higher interdependences according to $G(i,j)$, and then the position with higher total interdependences with i and j and this continues until $n+1$ positions are chosen. This positions will be the central positions of the model. After that, a new position is added with a total dependence score with a subset of k chosen positions is maximized. A new position is added at one end to maximize its total dependence with the neighbouring position and a subset of $k-1$ of the first positions. The procedure is repeated until $2k+1$ positions are chosen and finally, the new positions are added at each end using

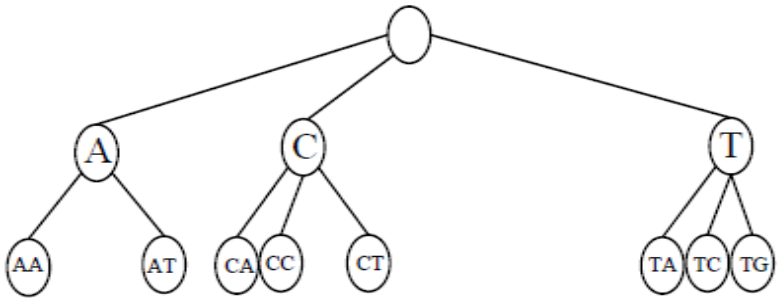


Figure 1.7. Representation of a variable order Markov model. The degree $n = 2$ of the model is pruned depending on the context. For example is the preceding base two bases are TT only the first T matters, and if the preceding nucleotides are GX, the new base is independent (Zhao et al., 2004).

the maximum dependence score as in the first step. In this algorithm the score of a candidate sequence is just the probability that the sequence has been generated using the model.

To reduce even more the parameters of the model, variable length Markov models (VMM) can be used. VMM are Markov models where the order n of the Markov chain can be reduced depending on the context. For example, in the figure 1.7 a 2-order VMM for a motif is shown. In this example, depending on the context, the probability of a nucleotide $P(X = b_i)$ will depend on the first preceding base, on the two preceding bases or will be independent. For example if the two preceding bases are AA the new nucleotide will depend on both, if the preceding bases are AC, it will depend only on A, and in the context of a GX, the new nucleotide will not depend on the previous bases.

The VMM can also be permuted, in order to bring together important dependences keeping a low n Markov order (Zhao et al., 2004). As the number of context trees increases quickly with the order n of the Markov model, a forward selection or a Markov Chain Monte Carlo have to be used to select the best model, according to the Akaike Information Criteria (AIC) or the Bayesian Information Criteria (BIC). The number of permutations increases with the length of the binding motif, and it is not realistic to do an exhaustive search for models that have more than 9 positions. A way to approximate the global optimum is to use simulated annealing.

The likelihood of a sequence to belong to the model is given by equation 1.10

$$Likelihood = log \frac{P(x|model))}{P(x|background)}, \tag{1.10}$$

where $P(x|model)$ is the probability that the sequence belongs to the motif model and $P(x|background)$ is the probability that the sequence belongs to the background model, constructed using a 3rd order Markov chain.

An alternative to the Markov models are the Bayesian networks which can easily incorporate long-range interdependencies without increasing the number of parameters to estimate. Bayesian networks are a graphical representation of probabilistic models where some influencing positions (parents) are connected with and edge to the influenced position or child.

Bayesian trees are a kind of Bayesian networks that allow arbitrary dependences within any two positions in a model. In other words, each position can depend on any other position (but just one). This tree networks have been used to model splicing sites (Cai et al., 2000). In order to construct the dependency tree the mutual information (MI) of every pair of positions is calculated and then used to construct a graph G where each node is the position i and the weight of the edges connecting i and j is the MI between the two positions. This graph G is used to construct a maximum spanning tree which is a tree (acyclic graph) including all the nodes of the graph G and where the maximum sum of the weight of the edges. After that the variable X_0, in the case of splice recognition sites the nucleotide b_0, is set as the root of the tree and the conditional probability between the dependent positions is calculated. The score of a sequence is the probability of that sequence being generated by the model. First order Markov models are only a special case of this algorithm, but as the Bayesian tree looks for solutions in a wider space of models and, even if the number of parameters is the same, this kind of models are more likely to overfit.

Adding a hidden variable T to the structure of the tree and allowing each variable to depend on T and also one of the other variables $x_1...x_L$ allows to capture more complex interdependences only multiplying the number of parameters by a factor C that is the number of different values that can take the T hidden variable (Barash et al., 2003). More complex Bayesian networks can be able to model higher-order dependencies, but the number of possible networks increases exponentially. (Castelo and Guigó, 2004) created an algorithm that can efficiently find the best Bayesian network. Another solution is to create variable order Bayesian networks (VOBN) where the order of the network depends on the context of the parents nucleotides (Ben-Gal et al., 2005). Using

the MI as a dependence measure, first a Bayesian Network of order n where each position depends on other n positions in the sequence is constructed. Then, the network is pruned in all the dependencies that can be removed without an important change in the transition probability from parents to child. To do that a forward algorithm and the Kullback-Leibler divergence are used. The score of a sequence is the log-likelihood that the sequence belongs to the model, compared to that of the sequence belonging to the background.

Naughton et al. (2006) proposed a non-probabilistic graph model that can capture complex interdependences. In this graph model the nucleotides are treated as $k-mer$ and represented as node occurrences in a graph. An edge connect two nodes if the Hamming distance between them is under some threshold. Pairwise dependencies create clusters in the graph and more complex dependencies create other structures. To score a new sequence two heuristic criteria are defined: the Sequence Similarity (SS) and the Identical K-mers (IK). The SS measures how a sequence is close to at least one member to the motif, it is higher when the number of mutations between the candidate and a motif k-mer is low. The IK gives value to the multiple occurrences of a k-mer in the motif: the more occurrences of a k-mer exist in the motif, the more likely than a closely related k-mer is also part of the motif. The score of a candidate sequence is defined in equation (1.11).

$$S = \sum_{m=1}^{N} \Theta_{SS}^{d} \Theta_{NS(b_1,b_2)} \sum_{j=1}^{n_m} \Theta_{IK}^{j}, \tag{1.11}$$

where N is the number of unique k-mers present in the motif and n_m is the number of instances of a single k-mer. Θ_{SS} is the relative score between zero or one mutations and d the hamming distance from the candidate sequence to a motif sequence. $\Theta_{NS(b1,b2)}$ is the transition matrix from the nucleotide $b1$ to the b_2, if there are more than one mutations it is taken as the average of the existing mutations. These substitution rates can be calculated for each database. Finally Θ_{IK} determines how much we value the existence of multiple k-mers. To evaluate the significance of this score, a null distribution of it was calculated for all the studied databases.

While all these models are able to take into account dependences between positions, they usually need more sequences than the currently available for most of the binding sites, and also they usually have high computational times.

1.5.3 Higher order detection

Many effects may alter the functionality of binding sites, from the chromatin structure to the interaction with other transcription factors, this is why most of the motif finding algorithms, even if they work well *in vivo*, they cannot be trusted when *in vitro* situations are studied.

Transcription is not regulated by a single binding site but by means of a combinatorial set of interactions between TF at their binding sites, what is called a cis-regulatory module (CRM). The formation of CRM implies that TFBS are not located randomly through the genome but they usually have specific distances between them that allow their interaction. Another effect, crucial to make a binding site functional,is the chromatin packaging around the binding site, because it determines the availability of the DNA to the binding protein. On top of that, protein expression is a dynamic process and different cells, cell-cycle or development stages need different proteins at different times, which converts a true positive under certain conditions into a false positives if conditions are changed. Many CRM models use known motif finding algorithms to look for different binding sites within a promoter sequence and then, use a combination of the scores for each binding site and the distances between them to assess the significance of the regulatory model. The final score can be calculated for example using a Hidden Markov Model (HMM) (Frith et al., 2001) or a self-organizing map (SOM) (Mahony et al., 2005). Other algorithms calculate the density of binding sites within a promoter to study the functionality of the promoter (Berman et al., 2002). Most of the algorithms use PSSM to detect the binding motifs, but some of them also take into account the interdependences of the motif (Xing et al., 2003).

More recently, some studies show that the incorporation of nucleosome positioning sequences (NSP) can also help to reduce the number of positives that are, in fact, non-functional sites. Stable nucleosomes are found in the surroundings of non-functional sites, while the functional sites usually have a more open chromatin configuration. The study of the nucleosome occupancy can reduce the number of false positives (Daenen et al., 2008).

1.6 DNA signal processing

Genomic information is discrete in the sense that it is encoded in a four-letter alphabet. The conversion from the symbolical signal into a numerical one allows the use of signal processing to the study of DNA sequences, making possible a better visualization of the DNA data and also facilitating the analysis of the sequences.

1.6.1 Numerical Conversions

Many numerical conversions have been proposed (Anastassiou, 2001; Cristea, 2005). The most common one is a 4-D conversion where each nucleotide is assigned to a digital value. The 4-D vector is 1 in the position where the nucleotide is present and 0 otherwise. In this case the nucleotide conversion correspond to: $A = (1, 0, 0, 0)$, $C = (0, 1, 0, 0)$, $T = (0, 0, 1, 0)$ and $G = (0, 0, 0, 1)$. This conversion is symmetric for all the nucleotides, because the distance between two nucleotides is always the same. For all the elements U_k corresponding to the k nucleotide, $U_A + U_C + U_T + U_G = 1$, that means that the dimensionality of the conversion can be reduced to 3 .
This conversion is thus reduced to a 3-dimensional conversion where each nucleotide is placed at the vertex of a regular tetrahedron, as it can be seen in equation (1.12).

$$\begin{aligned} A &\equiv (1, 1, 1) \\ C &\equiv (-1, 1, -1) \\ G &\equiv (-1, -1, 1) \\ T &\equiv (1, -1, -1) \end{aligned} \tag{1.12}$$

The tetrahedron can be changed in order to make the distance D between two nucleotides $D = 1$ and also to make all the vertex of the tetrahedron positive, without losing any generality in the symmetry of the 3-dimensional conversion. The figure 1.8 shows the 3-dimensional representation when the distances between the nucleotides are set to $D = 1$, and was proposed by Silverman and Linsker (1986).

A further reduction of dimensionality looses the symmetry in the conversion, but it can also be useful in cases where some biochemical properties of the nucleotides are important. The most used two dimensional conversion is built projecting the tetrahedron into the complex plane. The way the tetrahedron is projected can be chosen according

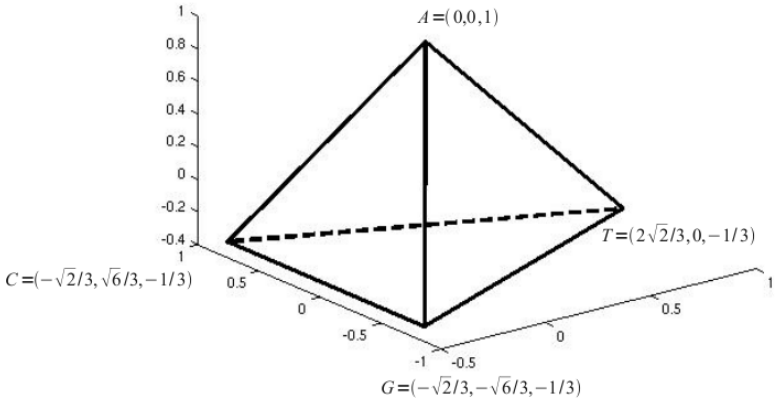


Figure 1.8. Three dimensional conversion of the DNA, where each nucleotide is placed at the vertex of a regular tetrahedron. This conversion is symmetric for all nucleotides and the distances between them is $D = 1$ (Pairó et al., 2012)

to the properties needed to preserve. In equation (1.13), the 2-dimensional conversion is chosen to reflect the complementarity of the bases A-T G-C by the symmetry to the real axis and the chemical similarity (purines and pyrimidines) is expressed by the symmetry with respect to the imaginary axis. The distances A-C and G-T are larger than the others as it can be seen in figure 1.9 where this conversion is represented. Of course, the representation where the complementarity is reflected by the symmetry with the real axis as well as other complex conversions are equally valid.

$$\begin{aligned} A &\equiv 1 + j \\ C &\equiv -1 - j \\ G &\equiv -1 + j \\ T &\equiv 1 - j \end{aligned} \tag{1.13}$$

Similarly, conversions where the different nucleotides are placed at the axis of the x-y plane, have been proposed, one example can be seen in equation (1.14). Obviously these representations are equivalent to the representations in the complex plane.

$$\begin{aligned} A &\equiv (1, 0) \\ C &\equiv (0, 1) \\ G &\equiv (-1, 0) \\ T &\equiv (0, -1) \end{aligned} \tag{1.14}$$

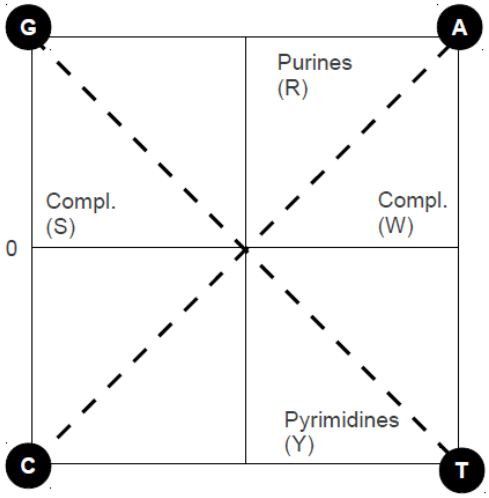


Figure 1.9. Example of a 2-dimensional conversion where each nucleotide is placed in a complex plane. The complementarity of the bases is shown by its symmetry with respect to the real axis, and the chemical similarity is shown by its symmetry to the complex axis.

Finally, a 1-dimensional conversion where each nucleotide is assigned to a real number can also be used. The weight of each nucleotide (indicated by how large is the number) and the distances between nucleotides have to be carefully chosen according to the purposes of the analysis. This simple conversion has been used in some applications such as gene discovery (Akhtar et al., 2008).

1.6.2 Applications in Genomic signal processing

The first applications of Digital signal processing to DNA sequences, appeared more than twenty years ago, when the 3-dimensional tetrahedron conversion was used to find DNA periodicity (Silverman and Linsker, 1986). Later, the 4-dimensional conversion was used to study the short and long-range correlations in DNA signals (Voss, 1992; Arneodo et al., 1995; de Sousa Vieira, 1999).

Another applications are the constructions of DNA spectrograms that allow a better visualization of the DNA data than the symbolical DNA. Discrete Fourier transforms (DFT), wavelet transforms and other methods have been used mostly to find protein coding regions within genomic sequences (Afreixo et al., 2004; Akhtar et al., 2007; Wang, W and Johnson, 2002). Some methods try to convert a whole DNA sequence in a vector, as DNA walks (Peng et al., 1992). In the first DNA walks, the walker

steps up $u_i = +1$ when a pyrimidine appears at i distance and steps down when a purine appears at this distance. Long-range correlations and the presence of introns have been studied using this method. Equivalent and similar concepts have been used to construct 2-D and 3-D graphical representations of large DNA sequences which take advantage of the different properties of coding and non-coding regions to visualize the differences and detect the coding regions (Nandy, 1996; Yuan et al., 2003).
More recently the 4-dimensional conversion has been used to the detection of binding sites using a SVM method, without taking into account the interdependences (Jiang et al., 2007).

1.7 Multivariate methods

The conversion from symbolical to numerical DNA allows the application of signal processing techniques to the DNA. Some examples are shown in the previous section. In this thesis two techniques have been used for motif detections: principal component analysis and parallel factorization.

1.7.1 Principal Component Analysis

Principal component analysis (PCA) is a multivariate technique to reduce the dimensionality of a large set of intercorrelated data while capturing the maximum variance. The data is transformed into a new set of variables, the principal components, which are uncorrelated and ordered in a way that few of them can retain most of the variance. (Jolliffe, 1989). It was first developed by Pearson (1901) who was studying the lines and planes that best fit a set of points in a p-dimensional space. Later, Hotelling (1933) independently developed the technique for the statistical analysis.
PCA can be defined as a bilinear decomposition of the data as it is shown in equation (1.15).

$$X = AB^T + E, \tag{1.15}$$

where X is the original $N \times M$ data matrix with N samples and M variables. A is the projected data or Scores, a $N \times nPCS$ matrix. B corresponds to the loadings which is the $M \times nPCS$ matrix defining the subspace where the data is projected. And E is

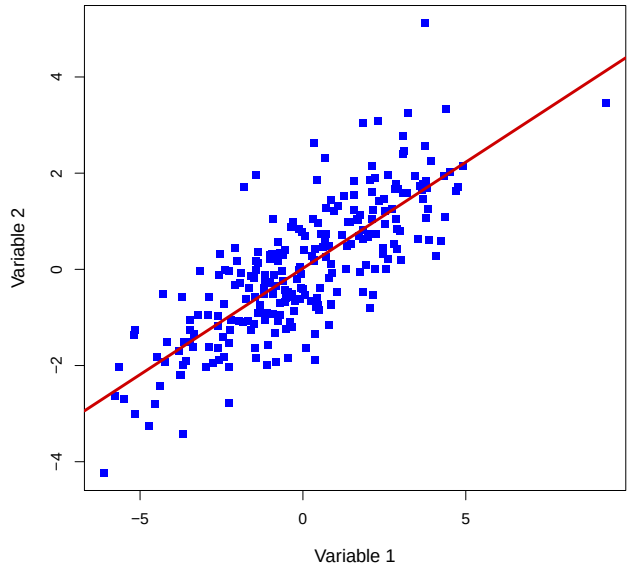


Figure 1.10. Example of a two dimensional correlated data, that can be described by a subspace of reduced dimensionality, the first principal component, in red.

the $N \times M$ error matrix.

The sum of the variance perpendicular to the data is minimized, which is equivalent to find the eigenvectors of the covariance matrix. In order to perform a PCA the covariance matrix is calculated and then diagonalized. The eigenvalues are ordered from higher to lower. The eigenvectors with higher eigenvalues will be the ones explaining most of the variance, and will be the Principal Components.

The scores are the projection of the variables in the new subspace, and they can be used to show the structure of the data. The loadings are the new variables expressed as a linear combination of the old ones, and are useful to interpret the new subspace. One example can be seen in the figure 1.10, showing a two-dimensional data base. This data can be well explained by the subspace defined by the first component (line in red) which retains most of the variability of the data. The loading will be the vector indicating the first principal component (PC) and the scores the projection of the data in the PC.

Some measurements can be used to assess how well a sample can be explained by the principal components: the Hotelling T-square and the Q-residuals. The Hotelling T-

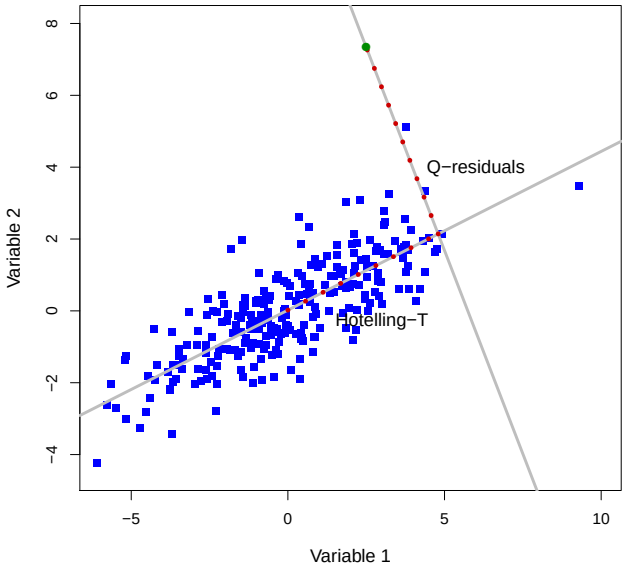


Figure 1.11. Hotelling T-square and Q-residuals for a new sample using the previous PCA model, presented in figure 1.10. In this figure the previous model is shown, with the 1 -component subspace as a line in grey, and the perpendicular distance to the subspace as a line perpendicular to the subspace, also in grey. The new sample is presented as a green dot ffl, and the distance within the subspace, known as Hotelling T-square and the Q residuals, or the distance from the sample to the subspace are shown in red dotted lines. It can be inferred that the new sample can not be explained using the previous model because this distances are large.

square is a measure of the distance of a sample to the center of the subspace, within the subspace and taking into account the variance of each dimension. And the Q-residuals measure the distance perpendicular to the subspace of principal components. In the figure 1.11, the same example as above is presented, but with an added sample, in green. In the figure it can be seen that the new sample has a large distance to the center of the subspace (high Hotelling T-square value shown as a dotted red line), and also a large distance perpendicular to the subspace (also a dotted red line). The new sample is an outlier to the model.

The Hotelling T-square can be calculated using equation (1.16)

$$T^2 = (X - \bar{X})S^{-1}(X - \bar{X}), \tag{1.16}$$

where T^2 is the Hotelling T-square value for a sample, X is the sample vector

projected to the subspace of principal components, $\bar{X}$ is the mean of the projection of the modelled samples and S is the covariance matrix

The Q-residuals are calculated as the square of the euclidean distance of a sample to the subspace of principal components. They can be calculated using equation (1.17)

$$Q = EE^T \tag{1.17}$$

where E is the $3M$ error vector obtained from projecting the sequence into the Principal Components subspace, and Q is the Q-residual of the candidate sequence. The Q-residuals can be converted to follow a Gaussian distribution using the transformation described in equation (1.18), developed by Jackson (2004).

$$\begin{aligned}
\Theta_1 &= \sum_{i=npcs+1}^{p} l_i \\
\Theta_2 &= \sum_{i=npcs+1}^{p} l_i^2 \\
\Theta_3 &= \sum_{i=npcs+1}^{p} l_i^3 \\
h_0 &= 1 - \frac{2\Theta_1\Theta_3}{3\Theta_2^2},
\end{aligned} \tag{1.18}$$

where Θ_1, Θ_2, Θ_3 and h_0 are the new variables, l_i the eigenvalues of the principal component analysis, *npcs* the number of components and p the number of the original dimensions of the X data. The confidence interval C for the new Q-residuals which are normally distributed with $\mu = 0$ mean and $\sigma = 1$ variance can be computed as in equation (1.19).

$$c = \Theta_1 \frac{[(\frac{Q}{\Theta_1})_0^h - \frac{\Theta_2 h_0(h_0-1)}{\Theta_1^2} - 1]}{\sqrt{2\Theta_2 h_0^2}} \tag{1.19}$$

C is the confidence interval for a given value of Q and Θ_1, Θ_2, Θ_3 are the new variables. PCA is one of the most used multivariate techniques. Finding the principal components can be useful to display multidimensional data in order to find a good interpretation and explanation of those data. At the same time it serves the purpose of dimensionality reduction.

In bioinformatics the challenge of high dimensionality of the data is very common. For this reason PCA has been widely used, specially in order to analyse gene expression data (Ma and Dai, 2011). The PCA analysis of gene expression microarrays can be helpful to find the combination of genes that better explain the phenotype (Wall et al., 2003), or to find the transcription factor binding sites in chip analysis. It can also be applied to time-series experiments to find dynamic models of gene expression (Holter et al., 2001) and, although it has been used as a preprocessing step in some cluster analysis of genes, it was shown that the first components do not necessarily lead to meaningful clusters (Yeung and Ruzzo, 2001). In whole-genome analysis PCA has been used to find gene pathways (Ma and Kosorok, 2009).

1.7.2 Multiway Analysis

Multiway data is characterized for a set of variables that are measured in a crossed fashion. This kind of data is very popular in psychology where different measures are taken for different subjects and times (John R., 2003) or chemometrics (Bro, 1999) with the main example of fluorescence data, where the emission spectra is measured for different samples and different excitation wavelengths.
There are many algorithms that can be used to model the N-way data (Kiers, 2001),the most famous ones are PARAFAC which is a trilinear decomposition of the data and Tucker-3 which can be seen as an N-way extension of PCA (Tucker, 1966). PARAFAC, which is described in detail below, decomposes an N-way array into N matrices, while tucker-3 decomposes it in a set of N matrices plus a core N-way tensor.
Multiway models have some common characteristics. The most important one is that they are simpler in a mathematical way than the two-way models because they have less degrees of freedom which actually leads more restricted models and, generally, to poorer fits. PARAFAC, which can be seen as a restricted Tucker-3 model (the core is restricted to be superdiagonal), is the one with a poorest fit. Tucker-3 can at its turn be seen as a restricted PCA model, which also means that it would have a poorer fit than PCA (Kiers, 1991). In general,then, multiway models are used not to find a better fit to the data but to create easily interpretable models, because organising the data into a N-way array allows to maintain all the information. Sometimes, for example if PARAFAC is the adequate model to the studied data, the other models can be just fitting the noise.

In contrast with the two-way models, the three-way algorithms cannot be calculated sequentially, which means that the solutions are not nested. Every time that the number of components changes the model has to be recalculated

1.7.2.1 PARAFAC

PARAFAC is a multilinear model of N-way data. It was independently developed in 1970 as PARAllel FACtor Analysis (PARAFAC) by Harshman (1970) and as CANonical DECOMPosition (CANDECOMP) by Carroll and Chang (1970). Harshman developed PARAFAC using as initial idea the principle of parallel proportional profiles, which tried to solve the problem of the rotational freedom for 2 two-way analysis, and to find a model with a unique solution (Cattell, 1944).
In PARAFAC a N-way array is decomposed as the sum of the elements from its N loading matrices, while the unweighted sum of squares is minimized. The three-way PARAFAC decomposition is described by equation (1.20), but it can be easily extended to N-way arrays.

$$x_{ijk} = \sum_{r=1}^{r=R} a_{i,r}, b_{j,r} c_{k,r} + e_{i,j,k}, \tag{1.20}$$

where x_{ijk} is the original data, $a_{i,r}$, $b_{j,r}$, $c_{k,r}$ are the elements of the A, B, C loading matrices that describe each one of the modes (i, j or k indicate the mode and r the component), R is the number of components of the model and $e_{i,j,k}$ are the elements of the three-way array error. A graphical representation of a PARAFAC decomposition, with F components can be seen in figure 1.12 where the original data x is decomposed in 3 matrices, each one having F components.

As it is said above, PARAFAC can be seen as an extension of the bilinear Principal Component analysis to N-way data. However, there are differences between the PARAFAC and the PCA models. The most important ones are that PARAFAC does not impose orthogonality to its components, and that the PARAFAC models cannot be rotated without any loss of fit. This means that PARAFAC has no rotational freedom, although scaling and permutations can be performed without changing the fit.
Degenerate solutions can also appear in PARAFAC, when the solution is in a swamp or when there is no optimum solution. The conditions in which these solutions appear where studied by Ten Berge and Kiers, and are related to the rank of the the A, B and

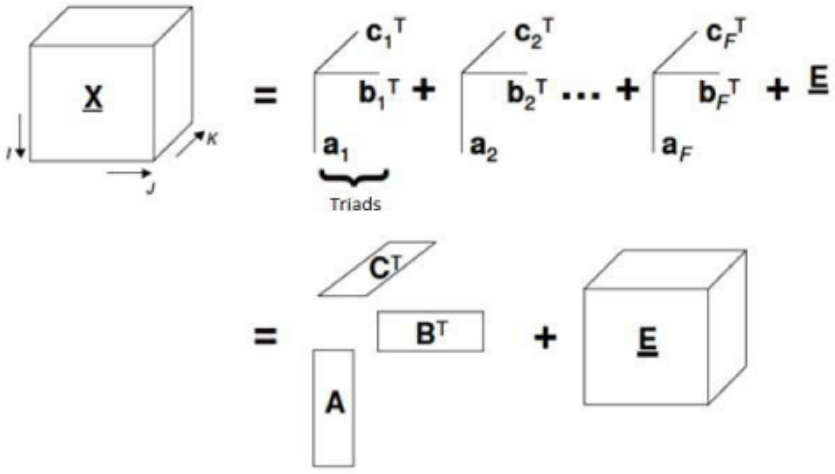


Figure 1.12. Geometric representation of a F components PARAFAC model. The initial X cube is decomposed into the sum of the loadings of the A, B and C matrices plus the error associated to the model. (Luna and Pinto, 2014)

C matrices (Bro, 1998). If K_x is the rank of the A matrix the sufficient condition for the uniqueness of the solution is that $K_A + K_B + K_C \geq 2R + 2$. There R is the number of components.

Many algorithms can be used to fit PARAFAC but the most used one is the alternating least squares(ALS). First A, B, C are taken (randomly or with some estimating algorithm) and then B and C are fixed and A estimated. After that, A and C are fixed an B estimated and so on, until the convergence criteria is reached. The most frequent issue is that the algorithm can reach a local minimum, being unable to find the most appropriate solution. To avoid this, the convergence criteria must be strict and the algorithm must be run several times with different initial conditions (Hopke et al., 1998).

When the data to model is trilinear and the signal-to-noise ratio is appropriate, a PARAFAC model with the appropriate number of components represents the true underlying phenomena, for example in the excitation-emission fluorescence spectra, where some excitation and emission wavelengths are calculated for different samples. But if the data is not trilinear PARAFAC may not be the best option to fit it, leading to unstable or inappropriate solutions. Several steps can be followed to see how good is the PARAFAC model to explain the data (Bro, 1997).

1. PARAFAC does not impose orthogonality to its solutions, as the number of components increases, the new components can be just a linear combination of the old ones, not adding new information to the model and making it more complex. When this happens, the variance explained by the model does not increase as the number of components increases. On the other hand, the difference between the variance explained by one component and the variance that can be explained only by this component increases because many components explain the same part of the variance. In that cases, the number of components of the PARAFAC model should be reduced.

2. The Q-residuals and the Hotelling T-square can be studied in order to find outliers, as in the two-way analysis.

3. The core consistency is a measure of how trilinear the data is. In order to calculate the core consistency, the data is first modelle with PARAFAC which can be seen as a Tucker-3 model with a restricted superdiagonal core. Then the A, B and C loadings of the PARAFAC model are used to calculate the equivalent Tucker-3 core using a regression model. If the PARAFAC model is valid, then the Tucker-3 core should be similar to the PARAFAc core (this means superdiagonal). When the core similarity is 100, then the Tucker-3 core is superdiagonal and the data does not have any non-trilinearity, as the non-linearities in the data appear, the core consistency drops. A core consistency lower than 70 may mean that the data has too many non-triniliearities to be explained with a PARAFAC model, or that the PARAFAC model used has too many components, and a model with less components can explain the data better.

4. PARAFAC allows the use of some constraints such as orthogonality or non-negativity. This can be helpful to find more meaningful models even if the fit would be worse.

5. In order to avoid local minima, the algorithm must be run several times. Also, to study the robustness of the model, a split-half or a k-fold cross-validation can be run. If the models built using the cross-validation are similar, then the model is not sensitive to the samples. This step needs a large amount of data, otherwise a l.o.o. cross validation can also be used.

All the tips mentioned above are just a guide to find the best model, but the most important thing is to have a good knowledge of the system in order to find the appropriate PARAFAC model which can best be useful to interpret the data.
As commented before, most of the PARAFAC applications are on the Psychology and Chemometrics fields, where it has been applied to a large variety of problems. But more recently PARAFAC was also applied to the study of the origin of seizure (Acar et al., 2007), or to the study of the dynamics of stem cells biology because it allows the integration of time-course data (Yener et al., 2008). Other multi-way techniques have been applied to the integration of data from different microarrays (Omberg et al., 2007) and have been shown to have a great potential in the study of systems biology (Conesa et al., 2010).

1.8 Thesis Goal

1.8.1 Definition of the problem

Determining where in DNA each TF can bind is an important issue in biology, because transcriptional regulation is essential to understand a range of cellular processes which go from cell differentiation to specific cell-type regulation. Moreover, mutation on TFBS are likely to underlie several diseases that are responsible for differences in morphology physiology and behaviour (Wray, 2007).
The methods to detect binding sites sequences using previous knowledge of the binding sites can be divided into two main groups (1) PSSM which do not take into account interdependences between positions and (2) Methods that take into account interdependences. While the first group of methods, the most commonly used, have the disadvantage that do not take into account interdependences, the second group needs too many sequences and too high computational times.
A method is needed which can take into account interdependences between positions in the binding sites without needing a high computational time or many sequences to construct a good model.

1.8.2 General Objective

The general objective of the thesis is to use the knowledge of event detection in numerical sequences in order to find binding motifs within large genomic sequences. The constructed detectors will use well-established multiway signal processing techniques and will use covariance, which is a second order statistics, in order to find interdependences between positions. This detectors should be fast and easy to build as PSSM detectors but also able to detect position with interdependences.

1.8.3 Goals of the Project

The specific goals of the project are:

1. Characterization of the binding sites and their relation to the regulated genes: study of the interdependences of the binding sites and study of the gene-TF interaction.

2. Construction of a Q-residuals detector. Converting the DNA matrix into a numerical matrix, a Principal Component Analysis of the numerical matrix is used to model the binding sites and then the Q-residuals are used to distinguish between binding sites and other genomic sequences.

3. Construction of a Quadratic discriminant analysis (QDA) detector. Converting the DNA sequences into a cube a PARAFAC analysis can be applied which has biological information of the sequences. The scores and the Q-residuals of PARAFAC can be combined to construct a QDA detector.

2

Binding Sites Characterization

Even if all transcription factors affect the regulation gene expression, they can not be considered a single group of proteins, as it can be seen with the huge variety of existing transcription factor families, depending on the binding domain. The specific function in gene regulation can also be very different, going from the TF needed for transcription to occur in almost all genes to the ones that are activated after some cell signalling. This is translated in a large variety of motifs. In this chapter I will characterize the TF and its binding sites, studying first the number of genes regulated for each TF and the number of TF needed for the regulation of each gene and after moving to binding sites and looking at the interdependences between binding site positions. The characterisation will end with an study of the interdependences of some binding sites separated by families.

2.1 Study of interactions between genes and transcription factors

Some transcription factors actuate in a cell type specific or tissue specific manner, others like the transcription factors contained in the TATA-box are needed in the expression of most of the genes. CTCF can bind to many sites in the genome, mostly as an insulator but also as an enhancer or repressor of gene expression (Phillips and Corces, 2009). Summarising, transcription factors vary widely in the number of binding sites across the genome (Whitfield et al., 2012).
Using the large amount of data released by the ENCODE project, the statistics about

the genes and its associated TF can be studied in a cell-type manner (Wang et al., 2012). This kind of studies can help to separate functional biding sites from sites that are only positive *in vitro*. A more general way to calculate TF-gene relationships, without doing it in a cell-type basis, is to retrieve the known TF-gene functional relationships from databases. Even if the statistics are not as accurate as the analysis of the experimental biological events in ENCODE, they can proportionate information for a larger number of TF and any cell type.

2.1.1 Data

In order to study the interactions between transcription factors and the corresponding genes, all genes from NCBI genbank database (Benson et al., 2012) for *Homo sapiens* were retrieved, a total of 22812. The interactions between transcription factors and these genes were extracted from STRING and SabioSciences databases, using the StringSabio R-package described in the Appendix B. Extracted in June 2012, the total number of found TF-gene interactions is 193882. From these 103152 were reported from the STRING database and 89986 were reported from the SabioSciences database. The overlap between databases was only 744 interactions.

2.1.2 Results

The number of interactions depends on the studied gene. In figure 2.1 the genes have been classified according to the TF interactions that regulate them. The most numerous group is the one that is regulated by an interval between 5 and 10 transcription factors which is comprised by 7003 genes. Specifically, the group which is regulated by 8 transcription factors is the largest, with 1773 genes. The table 2.1 summarizes the number of transcription factors participating in the regulation of genes.

TF were classified according to the number of genes that they regulate, and the result can be observed in figure 2.2. Most of the studied TF regulate between 1 and 5 genes, and 280 of TF of them regulate only one. The last step is to study the variability of the distribution of genes respect to transcription factors. It can be observed in the figure 2.3 where the change in the number of genes regulated by a number n of TF is shown with a scale where red is maximum and blue is 0, that the number of TF regulating a gene increases quickly until it achieves a stationary value of 10 TF per

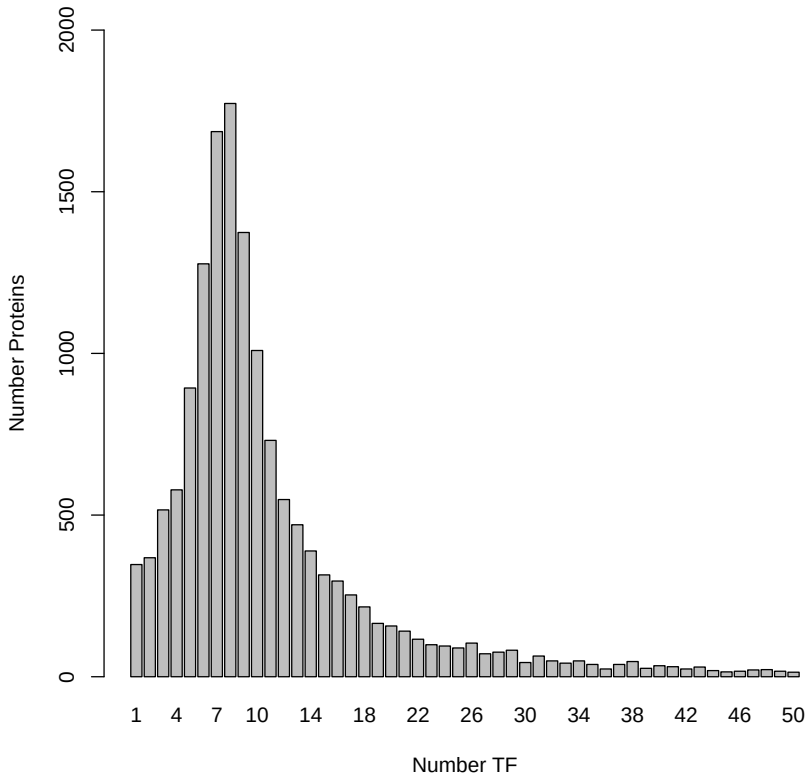


Figure 2.1. Histogram showing the number of TF regulating each gene. The most numerous group is regulated by 8 transcription factors and a peak can be seen between 5 and 10.

Table 2.1. Information about the classification of genes according to the number of TF regulating its summarized.

Number of TF	Number of genes
$1 < N < 5$	1809
$5 < N < 10$	7003
$10 < N < 15$	3147
$15 < N < 20$	1245
$20 < N < 30$	1030
$30 < N < 50$	665
$50 < N < 100$	324
$N > 100$	61

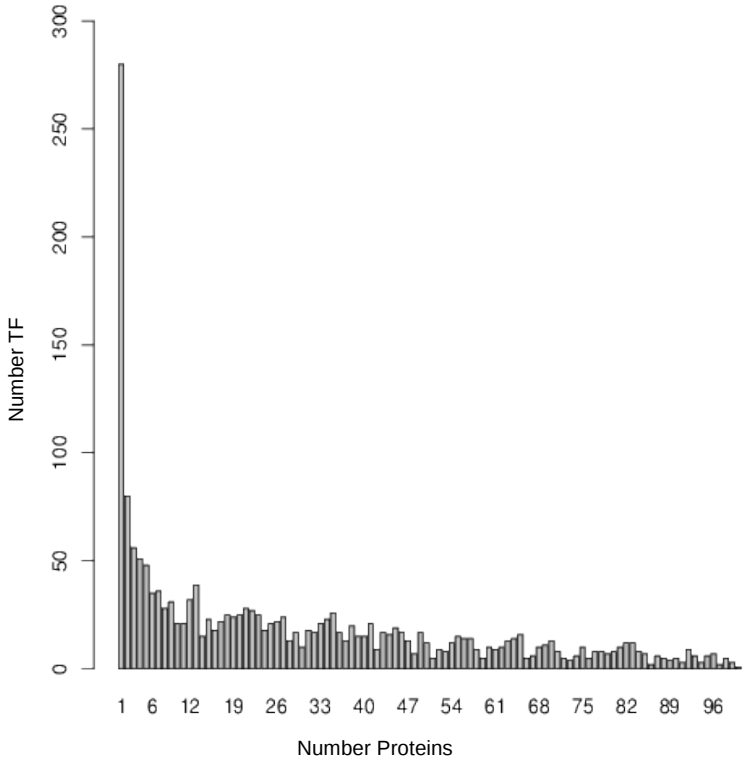


Figure 2.2. Histogram showing the number of genes regulated by each TF. Most TF regulate between 1 and 5 genes, then the number decreases.

gene.

2.2 Study of the interdependences

In 2001, an experimental study revealed for first time evidence of interdependences between neighbouring positions in some binding sites. Since then, many experimental data showed that the interdependences where not only in neighbouring positions, and computational studies tried to calculate these interdependences using binding sites data from transcription factor databases (Tomovic and Oakeley, 2007; Zhou and Liu, 2004)

Table 2.2. Information about the classification of TF according to the number of genes that they regulate. The most numerous group is the TF regulating between 1 and 5 genes.

Number of genes	Number of TF
$1 < N < 5$	467
$5 < N < 10$	178
$10 < N < 15$	128
$15 < N < 20$	112
$20 < N < 30$	220
$30 < N < 50$	331
$50 < N < 100$	404
$N > 100$	404

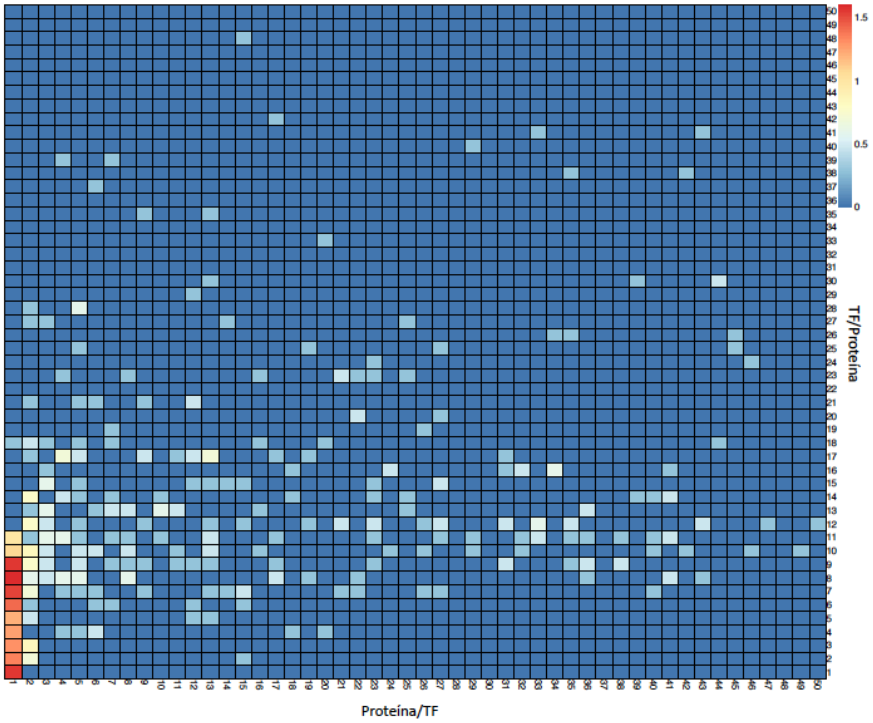


Figure 2.3. Variabilty of the number TF regulating a protein. The maximum increase is represented in red and the 0 (no change) in blue. The number of TF regulating a protein increases until it reaches a stationary value at 10 Transcription factor per protein.

2.2.1 Data

Two JASPAR (2010) collections were used to study the interdependences between positions, the JASPAR core and the JASPAR families. Four organisms were chosen

Table 2.3. Information about the transcription factor families and the motifs included in each family

Family	TF motifs
bHLH	TAL1 TCF3, Hand1 Tcfe2a, Mycn, USF, ARNT, MAX, MYC MAX, Ahr ARNT
bZIP	HLF, NFIL3, bZIP910, bZIP911, CREB1
ETS	GAPBA, ELK4, EIP74EF, ELK1, SPI1 1, SPIB, ETS1
Forkhead	FOXD1, Foxq1, Foxd3
HMG	SOX17, SRY, Sox 5, HMG IY, HMG 1
Homeo	HNF1A, Nkx2 5, Ubx 1, En1
MADS	SRF, SQUA
Nuclear	usp, PPARG, RXRA VDR, RORA 1, RORA 2, NR2F1, PPARgamma RXRA
REL	NFKB1, RELA, REL, dl 1, dl 2
TRP	IRF2, IRF1, GAMYB

from JASPAR core:*Homo sapiens*, *Drosophila melanogaster*, *Mus musculus* and *Rattus norvegicus*, and the motifs having more than 10 binding sites were extracted. In total there are 181 sequences: 43 from humans, 26 from the mouse, 11 from rat and 102 from fly.

From JASPAR families database all sequences corresponding to the families: bHLH, bZip, ETS, forkhead, HMG, Homeo, MADS, nuclear, REL and TRP were extracted. The table with the families and the binding sites included as a representation of each one of these families is presented in table 2.3.

2.2.2 Measurement of the interdependences

A first attempt to calculate the interdependence between two binding site nucleotides, situated in the position i and j of the binding site can be made using mutual information, shown in equation (2.1).

$$MI_{i,j} = \sum_{b_i,b_j} P_{b_i,b_j,i,j} log_2 \frac{P_{b_i,b_j,i,j}}{P_{b_i,i}, P_{b_j,j}} \tag{2.1}$$

where b_i and b_j correspond to the nucleotides in the studied positions i, j and P_{b_i} is the probability of the b_i nucleotide in the position i. The joint probability of having nucleotide b_i in position i and b_j in position j is described by P_{b_i,b_j}. The main problem

of this approach is that it is not straightforward to calculate whether the obtained mutual information value means a significant interdependence or not.
Tomovic and Oakeley (2007) proposed different methods to calculate only the significant interdependences, a χ^2 test, an exact method using Montecarlo simulations and the Bayes Factor. Giving two Hypothesis *H0* and *H1*, the Bayes factor is an alternative to hypothesis testing which gives the posterior probability of the null hypothesis when the prior probability is 0.5 (Kass and Raftery, 1995). The equation 2.2 defines the Bayes factor, giving the posterior distribution pr and the data D.

$$BF = \frac{pr(D|H0)}{pr(D|H1)} \tag{2.2}$$

The less restrictive method is the χ^2, then the Bayes Factor and finally the exact method which is the one finding less significant interdependences but that it has a large computational cost. In order to achieve a compromise between the restrictiveness to consider significant interactions and the computational time, the Bayes Factor was used to calculate interdependences. This method was also chosen by Zhou and Liu (2004). In the study of interdependences, the Bayes Factor (BF) described in equation (2.3) was used to test the Null hypothesis, H_0, of independence between positions i and j against H_1, the alternative hypothesis of dependence, in order to determine the significance of the dependencies found:

$$BF(H_0;H_1) = \frac{\Gamma(\sum_{b_i,b_j}\alpha_{b_i,b_j}}{\Gamma(M+\sum_{b_i,b_j}\alpha_{b_i,b_j}}\prod_{b_i}\frac{\Gamma(N(b_i,i)+\alpha_{b_i})}{\Gamma(\alpha_{b_i})}$$
$$\prod_{b_j}\frac{\Gamma(N(b_j,j)+\alpha_{b_j})}{\Gamma(\alpha_{b_j})}\prod_{b_i,b_j}\frac{\Gamma(\alpha_{b_i,b_j})}{\Gamma(N(b_i,b_j,i,j)+\alpha_{b_i,b_j})} \tag{2.3}$$

where M is the size of the bindings sites sequences, $N(b_i,i)$ is the number of b_i nucleotides in position i, and α refers to the parameter of the Dirichlet prior distribution. When $\alpha_{b_i,b_j}=1$ and $\alpha_{b_i}=\sum_{b_j}\alpha_{b_i,b_j}$ the Bayes Factor is related to the mutual information as shown in equation (2.4) (Minka, 2003).

$$log_2(BF(H_0;H_1)) \approx -MMI_{i,j} \tag{2.4}$$

Formula (2.4), where $MI_{i,j}$ is the mutual information and M the number of sequences in a binding site motif, was used to calculate the Bayes Factor, $BF(H_0;H_1)$. And as in Tomovic and Oakeley (2007), a threshold of $BF < 0.1$ was set to indicate strong

evidence of interdependences between positions. For each motif, the proportion of positions showing interdependences was named Complexity of the factor or *Comp* and calculated as in equation2.5.

$$Complexity = Comp = \frac{NP_{interdep}}{NP_{Total}} \tag{2.5}$$

Where $NP_{interdep}$ is the number of positions that have significant interdependences according to the Bayes Factor calculation and NP_{Total} is the total number of position within the binding site.

2.2.3 Results of interdependences

2.2.3.1 General Results

The interdependences were calculated for all the retrieved binding motifs from JASPAR database. The minimum Complexity of a motif is 0, when all the positions are independent, and the maximum is $Comp = 0.37$, corresponding to the binding sites of $PPAR\gamma$ transcription factor in humans. $PPAR\gamma$ is a transcription factor of the nuclear family that regulates adypocite differentiation and it has been implicated in many diseases including obesity and cancer. Some studies have shown that $PPAR\gamma$ binds to sites composed by the repeat of two hexanucleotides separated by one nucleotide, and also that some upstream nucleotides have influence in the binding specificity. The fact that the hexanucleotides should ideally be equal is a good explanation for the large number of positions with interdependences (Okuno et al., 2001).

The histogram of the complexity of the database is presented in figure 2.4, where it can be observed that most of the motifs have interdependences. There is not a clear peak in the number of interdependences, but it can be noted that most of the motifs have a Complexity *Compl* between 0.2 and 0.3. The percentage on binding sites that do not have interdependences is very low, just a 6.62%. Even if the motifs without interdependences do not have a large number of positions, there is no clear correlation between the number of positions or sequences and the percentage of interdependences of a binding site.

While in our study more than 90% of the motifs have some interdependence, previous studies showed smaller percentages of motifs with interdependences, a 25% in TRANSFAC database from Zhou and Liu (2004) and a 62.62% in Tomovic and Oakeley (2007)

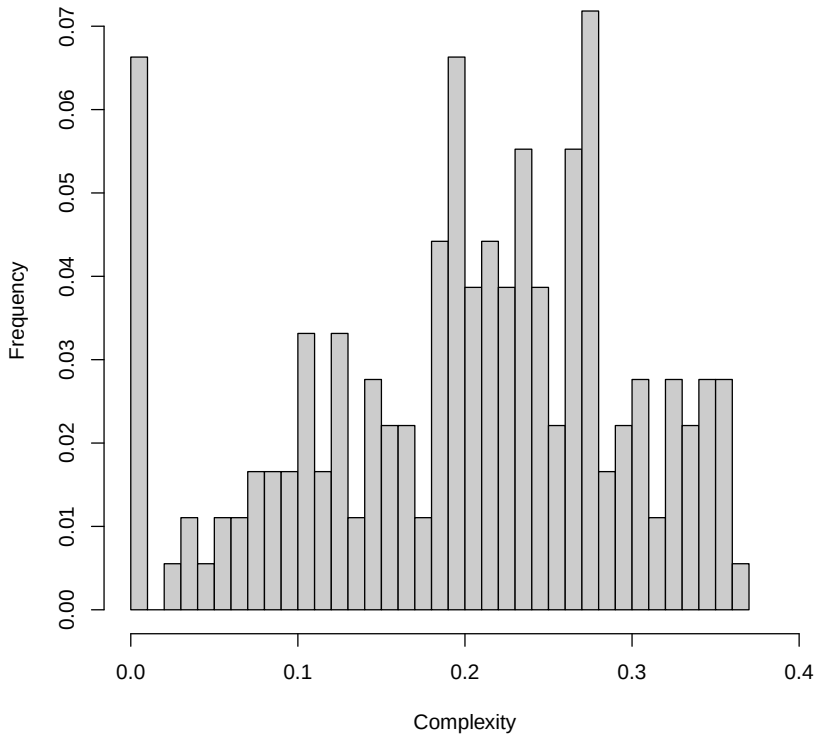


Figure 2.4. Histogram of the Complexity of the JASPAR motifs. The simplest binding sites have a $Compl = 0$ Complexity, meaning that all the positions are independent, and the maximum complexity is $Compl = 0.37$, corresponding to the $PPAR\gamma$ binding sites.

using the same methodology that we applied and the JASPAR (2006) database. The differences can be due to different factors, for example the new data included in the databases (94 sequences in JASPAR 2006 and 181 sequences from four organisms in JASPAR 2010). The other factor is that we did not take into account the motifs that have less than 10 binding sites from the same database and they did, this is supported by the fact that they found no interdependences in the motifs having less sequences.

2.2.3.2 Interdependences for Families

Binding sites are classified in families according to their DNA binding domain, which means that all members of a family have a similar structure that binds to DNA. The sequences from JASPAR (2010) family have been analysed in order to see if the Complexity of a motif can be identified with its family.

The results show that the binding sites cannot be classified by families using the Complexity, because all families have binding sites with different complexities. This is shown in figure 2.5 where the complexity of the different families is presented, each one in a different colour. This database is smaller than the JASPAR core, but the distribution of the Complexity of the sites does not change significantly.

Similar to the families but more general, are the structural classes of TF. Each struc-

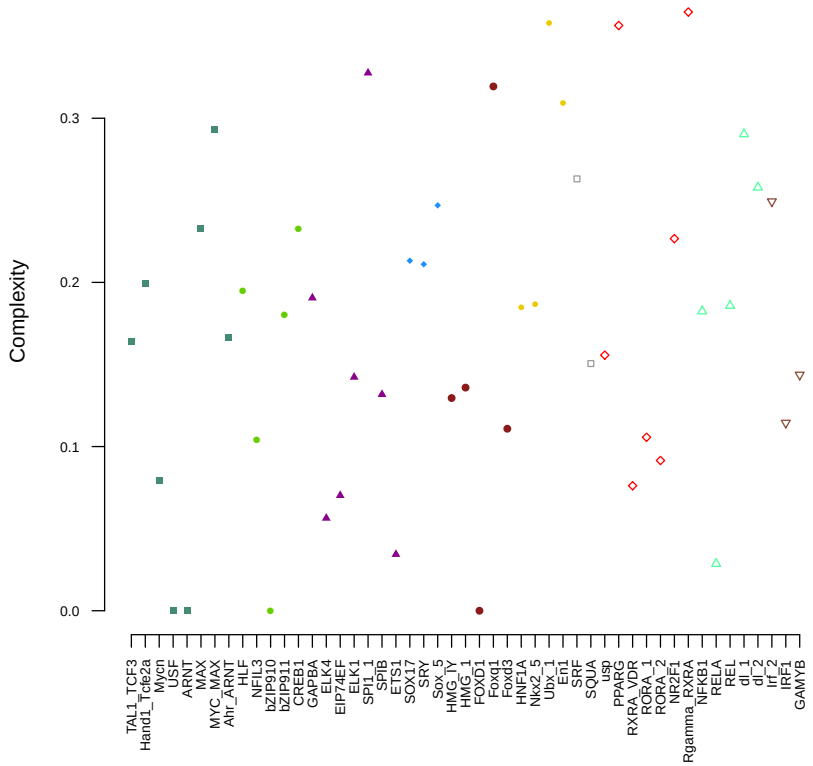


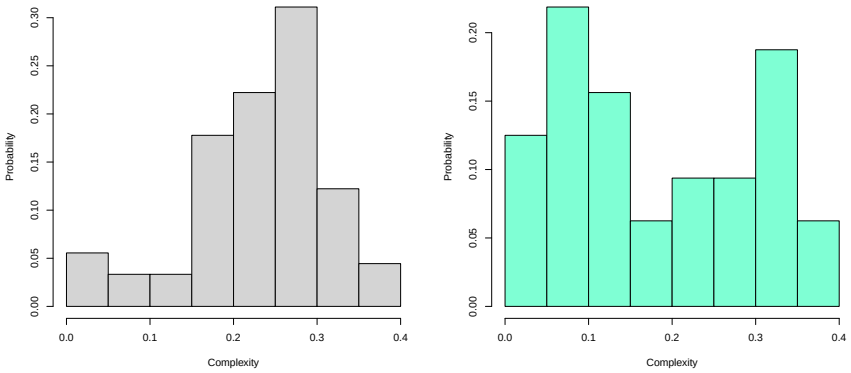
Figure 2.5. Complexity for families. The different families are presented in a different colour. It can be seen that the complexity of the TF in a family has a high variability and that the families cannot be separated using the Complexity value.

tural class comprises some families and it can be characterized by its binding domain. The structural classes included in the JASPAR database are: Beta-hairpin ribbon, beta-sheet, Helix-turn-helix, Ig-fold, Other alpha-helix, Winged Helix-turn-helix, zinc coordinating zipper type and others (which include the binding sites that cannot be classified in any of the mentioned structural classes). The distribution of the structural classes on the studied motifs is summarized in table 2.4

Table 2.4. Summary of the number of TF that belong to each structural class.

Structural Class	Number of TF
Beta-Hairpin-Ribbon	1
Beta-Sheet	0
Helix-turn-Helix	90
Ig-fold	13
Other Alpha-Helix	4
Winged Helix-turn-Helix	16
Zinc-coordinating	32
Zipper-type	19
Other	3

The results show that the structural classes cannot be classified using just the Complexity, but that more complex classifiers are needed (Narlikar et al., 2006). In the figure 2.6 the histograms for the Helix-turn Helix class (a) and the Zinc coordinating class (b) are represented. The histograms show a similar distribution of the Complexity in the different classes. This fact can be explained because the DNA-protein bindings depend also on sequence-based specific conformation or distortion of the structure or water-mediated contacts. Moreover, the binding affinity can be affected by neighbouring amino acids, specially if there are collaborative effects between transcription factors.



(a) Helix-turn-Helix **(b)** Zinc coordinating

Figure 2.6. Histograms of the Helix-turn-Helix and the Zinc coordinating structural classes. Both classes have complexities which go from the range from 0 to 0.4, showing that there is not a clear difference in the complexity of the families.

3

Q-Residuals Detector

The objective of the chapter is to construct a subspace model of the binding sites using the covariance matrix of the aligned DNA sequences. The Q-residuals of this covariance model can be used to construct a binding sites detector which takes into account inter-dependences between positions. In this chapter I first explain the conversion from an aligned DNA binding motif to a numerical matrix and then I explain the TFBS modelling by means of PCA and how the Q-residuals detector can be built. Finally, the Q-residuals detector is compared to state-of-the-art modelling algorithms. The results of this analysis were published by Pairó et al. (2012)

3.1 Methodology

3.1.1 Data sets

3.1.1.1 TFBS data

The transcription factor motifs were extracted from JASPAR 2010 release and TRANSFAC 7.0 (2005) databases (section 1.2.4). From TRANSFAC database, all the motifs with more than 10 binding sequences were chosen. The motifs correspond to different organisms: *Homo sapiens*, *Drosophila melanogaster*, *Mus musculus*, *Rattus norvegicus*, *Gallus gallus* and *Saccharomyces cerevisiae*. After downloading, the sequences were aligned using CLUSTALW (Larkin et al., 2007) and a leave-one-out cross validation method in order to see which sequences have more than five consecutive positions without gaps. Only those sequences fulfilling this latter condition were chosen. The studied sequences from TRANSFAC totalled 23 samples.

Table 3.1. Information about TFBS used for each database, the organisms and the

Organism	JASPAR	TRANSFAC
Saccharomyces cerevisiae	0	6
Drosophila melanogaster	10	3
Mus musculus	25	5
Rattus norvegicus	11	4
Homo sapiens	43	4
Gallus gallus	0	1
TOTAL	89	23

Four organisms were chosen from JASPAR database:*Homo sapiens*, *Drosophila melanogaster*, *Mus musculus* and *Rattus norvegicus*. Following the same criteria, the motifs with more than 10 binding sequences available in JASPAR were extracted. From the organism *Drosophila melanogaster* only 10 of the motifs were used. Table 3.1 shows the summary of the motifs classified by organisms and databases.

3.1.1.2 Background Data

All promoter sequences from the used multicellular organisms were extracted from the EPD database version based on the EMBL release 105 (sept 2010). The sequences from -1000 to +500 relative to the TSS were used to construct a background model for each organism, calculating the probability of each nucleotide in the promoter sequences. Then, two background sequences from each organism were randomly chosen to study the binding site detectors.

For the *Saccharomyces cerevisiae* binding sites, the promoter sequences used belong to positions 44730-46230 in chromosome 1, 678930-680430 in chromosome 16 and a region comprising 11410-12910 in chromosome 1. In table 3.2 the details of the genes whose promoter was used for each organism are explained, except for saccharmoyces cerevisiae.

Table 3.2. Information about the background sequences for each organism. The backgrounds correspond to the positions -1000 bp to +500 bp relative to the TSS from the genes in the table

Organism	gene 1	gene 2
Mus musculus	$Igk'T$	$Igk'MPC11$
Rattus norvegicus	$LC3_fP2$	$PSBPC2$
Homo sapiens	$RPS9P2+$	$PSMA2$
Gallus gallus	$apoVLDLII$	$a'A-globin$

3.1.2 Conversion to Numerical Matrix

The TRANSFAC motifs were aligned using ClustalW, in order to construct a matrix of DNA binding sequences. JASPAR sequences did not need any further preprocessing because the sequences are stored as an aligned matrix in the database.
To convert the aligned DNA motif into a numerical matrix, the 3-dimensional conversion where all the nucleotides are placed at the vertex of a regular tetrahedron with distance $D = 1$ between nucleotides, explained in section 1.6.1 was used. This conversion that can be observed in equation (3.1) where A,C,G,T are the three-dimensional conversion for the a,c,g,t nucleotides respectively was chosen because it is symmetric for all nucleotides and has been extensively used in genomic signal processing (Liew et al., 2005).

$$\begin{aligned} A &\equiv (0,0,1) \\ C &\equiv (-\frac{\sqrt{2}}{3}, \frac{\sqrt{6}}{3}, -\frac{1}{3}) \\ G &\equiv (-\frac{\sqrt{2}}{3}, -\frac{\sqrt{6}}{3}, -\frac{1}{3}) \\ T &\equiv (2\frac{\sqrt{2}}{3}, 0, -\frac{1}{3}) \end{aligned} \tag{3.1}$$

The numerical vectors corresponding to each position are concatenated. Therefore, the result of the numerical conversion is a $N \times 3M$ matrix of numerical sequences. Where N is the number of sequences and M the number of positions per sequence.
Due to the differences in length of the non-aligned binding sites, some gaps at the beginning and the end of the sequences can appear during the alignment process. The

numerical value of these gaps is imputed taking into account the probability of each nucleotide in the promoter model, as it can be seen in equation (3.2).

$$GAP = P(a)A + P(c)C + P(g)G + P(t)T \tag{3.2}$$

where GAP is the position of the gap in the tetrahedron, $P(a)$, $P(c)$, $P(g)$ and $P(t)$ are the background probabilities of each nucleotide and A, C, G, T are the positions of the a,c,g,t nucleotide in the vertex of the tetrahedron. Only when the nucleotide is available at least for 50% of the sequences the gap is imputed, otherwise the position is neglected.

3.2 Subspace Model

3.2.1 Building the model

In order to build the model a PCA is applied to the $N \times 3M$ numerical TFBS matrix, using equation (1.15) (X is, in our case the numerical DNA motif). The A scores represent the $N \times nPCS$ matrix with the projected DNA data and B is the $(3M) \times nPCS$ loadings defining the subspace which captures the maximum of the motif variance. The $N \times (3M)$ error matrix corresponds to the square of the euclidean distance of the TFBS to the subspace defined by the loadings.

To biologically interpret the model, we must look at the $3M \times 3M$ covariance matrix which captures the interdependences between the numerical positions. If the covariance is a diagonal matrix it means that all the positions of the studied motif are not correlated, and the non-zeros out of the diagonal indicate interdependences between the binding sites positions. In the PCA model, this information is explained in the loadings which in this case, due to the 3-dimensional representation of the DNA and its conversion to a matrix, must be interpreted in a per 3 basis. The variance of a position can be seen in the 3 components of the loading representing this position, if the three of them are almost zero, then the position is conserved and if they differ from zero the position varies. To analyse the covariance, between two positions is needed to look at the loadings of these two positions.

In the figure 3.1 the covariance matrix, the first loading and the binding sequences from the *Drosophila melanogaster* DL motif are shown. This motif has some interdependences that can be observed looking at the sequences (e.g interdependences between

positions 4 and 5). The interdependences are reflected into the covariance matrix and then, into the loadings which are large in absolute value, while the loadings of the most conserved positions are smaller.

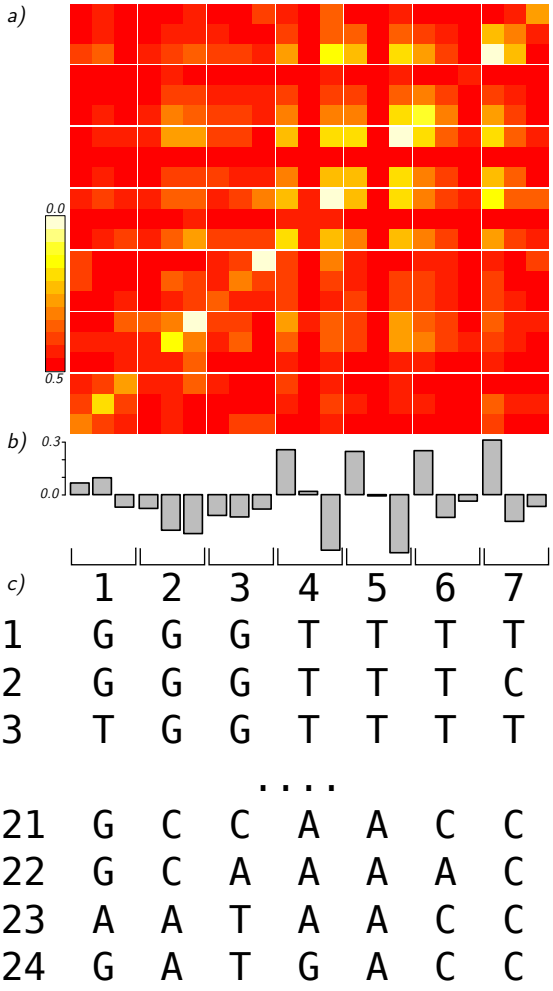


Figure 3.1. Covariance matrix (a), first loading (b) and binding site sequences (c) for the DL motif from the organism *Drosophila melanogaster*. The $3M \times 3M$ covariance matrix shows the interdependences between numerical positions, that can also be observed looking at the aligned motif. The covariance is then explained by the loadings, which are closer to zero when a position is more conserved.

3.2.2 Construction of the Detector

The Q-residuals detector can be built using the Q-residuals statistics of the numerical DNA sequences. When a candidate sequence is projected to the principal components

subspace, the hypothesis done is that the residuals of the binding sites sequences will be smaller than the residuals of the other genomic sequences. The Q-residuals statistics threshold can be directly estimated from the confidence interval calculated in equation (1.19), using the equation (3.3).

$$Q_\alpha = \Theta_1 [\frac{c_\alpha \sqrt{2\Theta_2 h_0^2}}{\Theta_1} + \frac{\Theta_2 h_0 (h_0 - 1)}{\Theta_1^2}]^{1/h_0} \tag{3.3}$$

Where, as in equation (1.19), Θ_1, Θ_2, Θ_3 and h_0 are the new variables used to transform the Q-residuals distribution into a normal distribution, Q_α is the Q-residuals threshold and c_α is the chosen confidence interval.
To show how this detector works, one example of the construction of the detector using promoter sequences and PPARG binding site sequences is represented in the figure 3.2. To plot this example, a 3-components model of the PPARG numerical binding sites matrix was calculated, and then the Q-residuals of these binding sites and promoter sequences projected to the model were represented using a histogram. The Q-residuals of the PPARG sequences are represented in blue and the Q-residuals of the promoter sequences in red; it can be observed that in this example a Q-residuals threshold can be used to detect binding sites within genomic sequences.

3.3 Comparison to Other Algorithms

3.3.1 PSSM Algorithms

To compare our detector to existing PSSM methods the MEET R package, available in the R-forge project `http://r-forge.r-project.org/projects/meet`, was developed (Pairó et al., 2011). This R package allows us to combine several alignment methods with different algorithms to search for TFBS within a large sequence. The package can be configured to call external alignment methods including CLUSTALW2, MUSCLE (Edgar, 2004), and MEME which has as an internal multiple alignment method. The current version of the package, MEET 5.1 is described in the Appendix A.
The proposed Q-residuals method is compared with MAST and an implementation of MATCH algorithm that takes into account the probability distribution of the nucleotides in the promoter sequences of each organism.

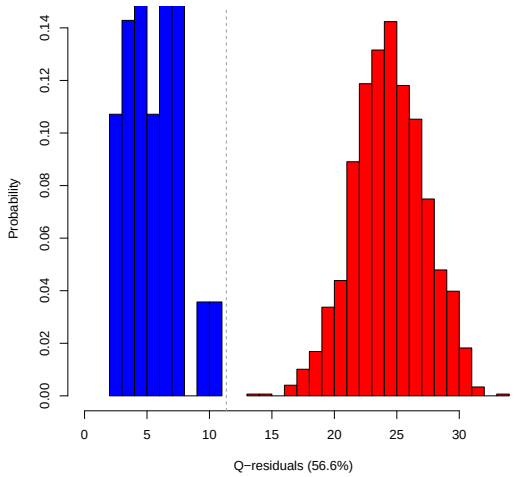


Figure 3.2. Q-residuals for the PPARG model using 3 principal components, in blue, and the Q-residuals of a human promoter in red. Selecting a Q-residuals threshold the binding sites can be easily distinguished from the non-binding sequences.

CLUSTALW2 with the default parameters, $gapextend = 0.2$, $gapopen = 10$ was used to align the sequences in all the compared methods in TRANSFAC.

3.3.1.1 MAST Algorithm

The comparison with MAST algorithm was done using the source available to download at MEME suite, MEME 4.4.0, which allows us to combine different alignment algorithms to construct the PSSM and then use the PSSM as an input to MAST. To calculate the PCA model and the Q-residuals in R, the pcaMethods R package was used (Stacklies et al., 2007).

3.3.1.2 MATCH Algorithm

To implement MATCH, the same algorithm explained in (Kel et al., 2003) was used. The only difference between the implementation and the algorithm described in the paper is the use of the background nucleotide probability distribution specific for each organism as it is described in equation 1.2, instead of a 0.25 background probability

of each nucleotide. The use of a specific background probability usually improves the detection of binding sites using PSSM methods.

3.3.1.3 Validation of the Detector

The measurements chosen to assess the performance of the algorithms are the ROC curves and the Area under the ROC curve (AUC). The ROC curve shows the True Positive Rate (TPR) against the False positive Rate (FPR) and its AUC goes from 0 to 1 being closer to one when the performance of the detector is good.
The validation was done by the MEET R package using a double leave-one-out method. First a sequence A is removed and inserted into the background sequence. Then, the rest $N-1$ sequences of the same motif are used for a standard LOO to construct models with $N-2$ sequences. These $N-2$ sequences are first aligned and the chosen algorithm is applied to build a model. Finally each one of the $N-1$ models of the L.O.O. is used to detect the sequence A within the known position of the background. After that, sequence A is inserted again into the group and a second sequence B is used to repeat the process N times. The methodology allows the calculation of the ROC curves the AUC and also the variance associated to these measurements.
As the location of the true positives is known, the threshold of the detectors can be moved in order to generate the N different ROC curves and their AUC. This threshold varies upon the detector, in the Q-residuals is the residuals statistics of the PCA model, in MATCH is the sequence similarity and in MAST is the p-value. Once the N ROC curves are generated, the standard deviation is used to estimate the variability of the ROC curve points and the AUC.
In the case of the Q-residuals detector, the AUC was calculated for a range from 1 to 10 principal components, and in the case of MATCH, the varying parameter was the Core Similarity, going from 0.5 to 0.95 by 0.05. Only one set of ROC curves and AUCs were calculated in MAST because the length of the sequence (parameter to optimize in MEME) is defined by the number of positions of the PSSM constructed using the aligned sequences.
The mean and the variance of AUC for the studied range of principal components were calculated for each motif. Models built using different numbers of principal components can have an equivalent performance when the AUC mean and the AUC variance are

taken into account. Between these models, the one with a smaller AUC variance averaging between backgrounds 1 and 2 was chosen as the best model. The same criteria was used to choose the threshold of Core Similarity in MATCH algorithm.
As the number of negative examples greatly exceeded the number of positive examples in this study, it was also convenient to compare the algorithms using Precision-Recall (PR) curves (Buckland and Gey, 1994). These curves plot the precision which is the rate of found positives that are actually true positives against the recall which indicates the true positive rate.
There exists a unique correspondence between the PR curves and the ROC curves, and when an algorithm dominates in the ROC spaces it also dominates in the PR space, however optimizing the AUC under the two different methods is not the same thing (Davis and Goadrich, 2006). To show that the PR curves confirm the results obtained with the ROC curves,the curves were calculated for the optimal parameters for each detector. The ROC curves, the AUC and the PR curves were calculated using the ROCR package (Sing et al., 2005).

3.3.1.4 Comparison Results

In this section we first present the results of the comparison between the Q-residuals detector, MATCH and MAST using the 112 motifs presented above and two different backgrounds for each organism. Then we describe in more detail the comparison between MAST and Q-residuals, and we show a study of the interdependences.
One example of detection can be seen in the cMyB motif in figure 3.3, a set of transcription factor binding sites belonging to *Homo sapiens*. The ROC curves show the performance of the three algorithms using the first background for *Homo sapiens*. A significant improvement is observed when the Q-residuals detector is used instead of MAST or MATCH.
Another example, for the FOXO3 motif also from the *Homo sapiens* organism can be observed in the figure 3.4, but in this case the curve represented is the Precision-Recall curve. In the figure the average PR curve is presented for the Q-residuals detector in black, the MAST algorithm in red and MATCH in green. Using the Precision-recall, the Q-residuals detector also performs better than the studied PSSM methods.
In order to quantify the differences in performance among the Q-residuals detector

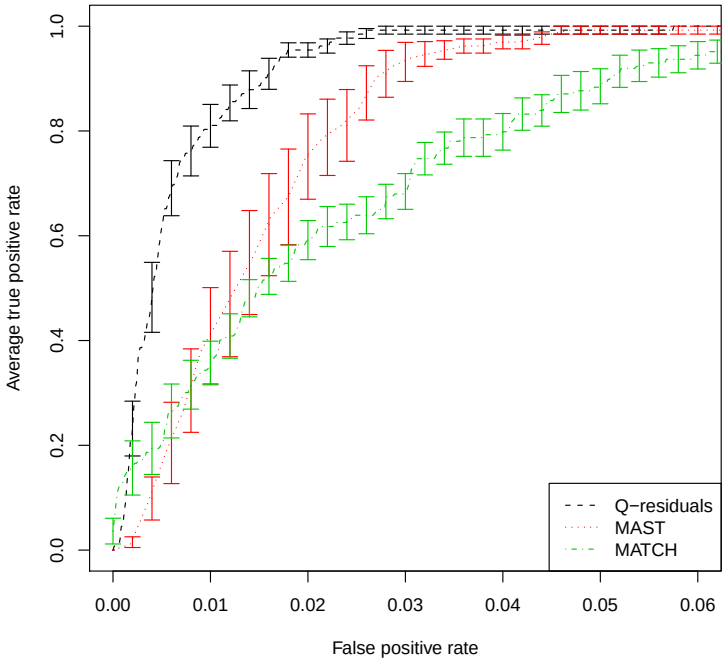


Figure 3.3. ROC curve for Q-residuals in black, MAST in red and MATCH in green using the cMyB transcription factor and the Homo Sapiens background 1. The ideal number of components and the ideal MATCH Core Similarity have been used to compute the ROC curve. The error bars correspond to the variation in detection using the LOO cross validation. The figure shows the improvement of the detection using the Q-residuals algorithm

and the other algorithms a Wilcoxon rank-test (Wilcoxon, 1945) was performed in the AUC distributions, using as a null hypothesis that the two distributions are the same and as an alternative hypothesis that AUC using Q-residuals is closer to 1 than using MAST or MATCH. In the table 3.3 the performance of the three different detectors Q-residuals, MATCH and MAST is shown for the two different backgrounds in each organism and the TRANSFAC motifs. The best number of components, which is usually between 1 and 4 is shown together with the mean AUC for each background and method. The increment in AUC and the p-value of the Wilcoxon-Rank the test are also represented.

The table 3.4 summarizes the results for all the studied TF motifs , showing for each organism, the total number of motifs and in how many of them Q-residuals performs significantly better than MATCH or MAST. It can be seen that Q-residuals performs

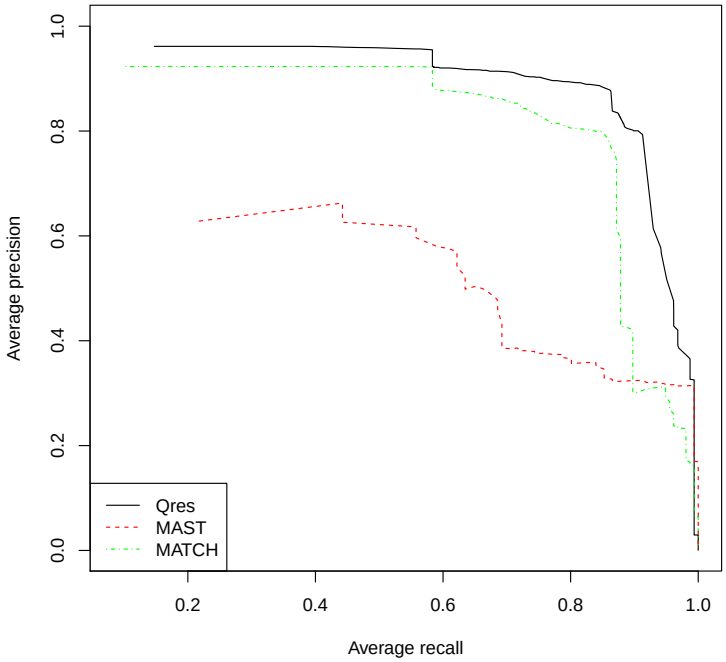


Figure 3.4. Precision-Recall (PR) curve for Q-residuals in black, MAST in red and MATCH in green using the FOXO3 transcription factor and the Homo Sapiens background 1. The ideal number of components and the ideal MATCH Core Similarity have been used to compute the PR curve. The depicted curve is the average for each leave-one-out iteration.

significantly better than Match in 57 of the 112 studied motifs and significantly better than MAST in 63 of them, with $p - value < 0.05$.

For a better visualization of the performance detectors, we represented the AUC box plots in figure 3.5. The box plots represent the AUC and its variation when the leave-one-out cross validation is applied. The figure 3.5 shows the box-plots for the first background and the JASPAR motifs corresponding to *Mus musculus* comparing the Q-residuals detector to MAST. In most cases, not only the mean AUC is closer to one in Q-residuals but also the variance is smaller, which suggests that the Q-residuals detector behaves more robustly.

The proportion of positions showing interdependences calculated using equation (2.3), complexity or *Comp*, varies among the studied binding sites as it can be observed in figure 3.5 (where it is named Idep). A correlation test was performed between the *Comp*

Table 3.3. Results for Q-residuals detector compared to MATCH and MAST algorithms, corresponding to the 2 backgrounds of each organism in TRANSFAC. The AUC shown for each method is the mean of the areas using the cross-validation method and the number of principal components for Q-residuals is chosen as the number of components with less variance in the AUC. The ΔAUC is the mean AUC improvement of Q-residuals vs. MATCH and MAST, respectively. The level of significance corresponds to the p-value calculated when a Wilcoxon-rank test is performed, with the null hypothesis being that the AUC distributions using Q-residuals detector and the other algorithm are the same and the alternative hypothesis being that the AUC distribution calculated with the Q-residuals detector is closer to one. A relation of the 89 JASPAR motifs and 23 TRANSFAC motifs can be found in the supplementary material 2.

TF	nPCs	Q-residuals 1	Q-residuals 2	Match 1	Match 2	ΔAUC Match [1]	MAST 1	MAST 2	ΔAUC MAST [1]
ABF1	4	0.9991	0.9975	0.9902	0.9964	$5 \cdot 10^{-3}$ ***	0.9957	0.9986	$1.14 \cdot 10^{-3}$
BCD	3	0.9961	0.9952	0.9912	0.9884	$5.85 \cdot 10^{-3}$***	0.9913	0.9947	$2.68 \cdot 10^{-3}$*
CAT8	3	0.9998	0.9995	0.9971	0.9978	$2.21 \cdot 10^{-3}$***	0.9999	0.9992	$9.02 \cdot 10^{-5}$
CEBP β 35	3	0.9931	0.9965	0.9863	0.9878	$7.75 \cdot 10^{-3}$ **	0.9936	0.9946	$6.66 \cdot 10^{-4}$
cJun	1	0.9868	0.9915	0.9700	0.9813	$1.35 \cdot 10^{-2}$ **	0.9575	0.9880	$1.64 \cdot 10^{-2}$*
cMyB	1	0.9905	0.9907	0.9714	0.9714	$1.92 \cdot 10^{-2}$***	0.9818	0.9869	$6.21 \cdot 10^{-3}$*
DL	1	0.9982	0.9962	0.9835	0.9864	$1.23 \cdot 10^{-2}$ ***	0.9682	0.9917	$1.73 \cdot 10^{-2}$*
E2F	4	0.9997	0.9998	0.9991	0.9998	$3.00 \cdot 10^{-4}$ *	0.9988	0.9995	$5.26 \cdot 10^{-4}$
GAL4	1	0.9998	0.9999	0.9742	0.9759	$2.48 \cdot 10^{-2}$ ***	0.9875	0.9653	$2.34 \cdot 10^{-2}$*
GCN4	1	0.9988	0.9997	0.9936	0.9937	$5.68 \cdot 10^{-3}$ ***	0.9951	0.9935	$5.06 \cdot 10^{-3}$***
HNF1 α	9	0.9945	0.9940	0.9807	0.9850	$1.14 \cdot 10^{-2}$ *	0.9943	0.9921	$2.1 \cdot 10^{-3}$
HNF4 α	4	0.9957	0.9972	0.9870	0.9938	$6.05 \cdot 10^{-3}$ *	0.9937	0.9957	$1.79 \cdot 10^{-3}$
HNF6 α	1	0.9977	0.9996	0.9961	0.99358	$3.81 \cdot 10^{-3}$***	0.9838	0.9949	$9.37 \cdot 10^{-3}$*
IRF1	2	0.9992	0.9994	0.9727	0.9912	$1.74 \cdot 10^{-2}$**	0.9970	0.9992	$1.22 \cdot 10^{-3}$
IRF8	3	0.9991	0.9981	0.9926	0.9791	$1.28 \cdot 10^{-2}$ ***	0.9928	0.9967	$3.86 \cdot 10^{-3}$**
KR	3	0.9923	0.9965	0.9933	0.9838	$5.85 \cdot 10^{-3}$ *	0.9926	0.9929	$1.69 \cdot 10^{-3}$
LyF1	3	0.9952	0.9958	0.9689	0.9823	$1.99 \cdot 10^{-2}$***	0.9903	0.9853	$7.68 \cdot 10^{-3}$**
MIG1	1	0.9986	0.9954	0.9766	0.9475	$3.49 \cdot 10^{-2}$ ***	0.9895	0.9896	$7.49 \cdot 10^{-3}$*
NF κ B	2	0.9998	0.9999	0.9995	0.9999	$3.08 \cdot 10^{-4}$*	0.9991	0.9998	$4.38 \cdot 10^{-4}$ ***
p50	2	0.9996	0.9999	0.9995	0.9999	$4.86 \cdot 10^{-5}$	0.9994	0.9998	$1.72 \cdot 10^{-4}$ *
RFX1	7	0.9921	0.9969	0.9721	0.9867	$1.51 \cdot 10^{-2}$ ***	0.9871	0.9837	$9.09 \cdot 10^{-3}$*
ROX1	8	0.9998	0.9985	0.9997	0.9993	$-3.5 \cdot 10^{-4}$	0.9996	0.9980	$3.40 \cdot 10^{-3}$*
T3R α	6	0.9923	0.9919	0.9754	0.9852	$1.18 \cdot 10^{-2}$***	0.9854	0.9757	$1.15 \cdot 10^{-2}$**

and the improvement in binding site detection when Q-residuals detector is compared to MAST. The improvement in binding site detection was calculated subtracting the mean AUC for each binding site calculated using each method. Results show a significant correlation between the number of strong interdependent sites within a binding locus and the amount of improvement of the Q-residuals detector over MAST, in terms of AUC. Performing the test in the results for JASPAR database, $p - value = 0.004$, and in TRANSFAC database $p - value = 0.04$.

The computational time of the Q-residuals detector MAST and our R implementation of MATCH have been compared when they are used to detect TFBS within promoter

Table 3.4. Summary of the results of the Q-residuals detector compared to MAST and MATCH, classified by organisms. The table shows in how many binding motifs for each organism the Q-residuals detector performs better than MAST or MATCH, and the total number of motifs for each organism.

Organism	motifs	Comp MAST	Comp MATCH
Saccharomyces cerevisiae	7	4	6
Drosophila melanogaster	13	10	8
Mus musculus	30	17	12
Rattus norvegicus	15	9	10
Homo sapiens	47	20	18
Gallus gallus	1	1	1
TOTAL	112	63	57

sequences. To compare the three algorithms the MAST algorithm (MEME version 4.4.0) installed in the computer, the C code for Q-residuals using the ideal number of components, and the implementation of MATCH algorithm in R with the ideal Core Similarity have been used. The background corresponds to the background 1 for each organism, which consists in 1500 nucleotides, and the threshold for each method was set in a way that the number of positives is similar. In the case of MAST a p-value of p=0.001 was chosen, in Q-residuals a confidence interval of C=0.95 and in MATCH the Similarity was set to S=0.85. The time was calculated in 100 iterations of the program. The averages of the computational times in detection for the TRANSFAC database motifs are $0.003 \pm 0.001s$ using the Q-residuals detector, $0.0191 \pm 0.001s$ using MAST and $0.33 \pm 0.03s$ for the R implementation of MATCH. The results show that Q-residuals detector is faster than MAST and the R implementation of MATCH in all the studied binding sites. The table 3.5 shows the mean computational time for the 23 transcription factors of the TRANSFAC database.

3.3.2 Graph-based algorithm

3.3.2.1 Motifscan Algorithm

PSSM can be easily extended in order to model pairwise dependencies between positions, but transcription factor binding sites can have more complex dependencies. To

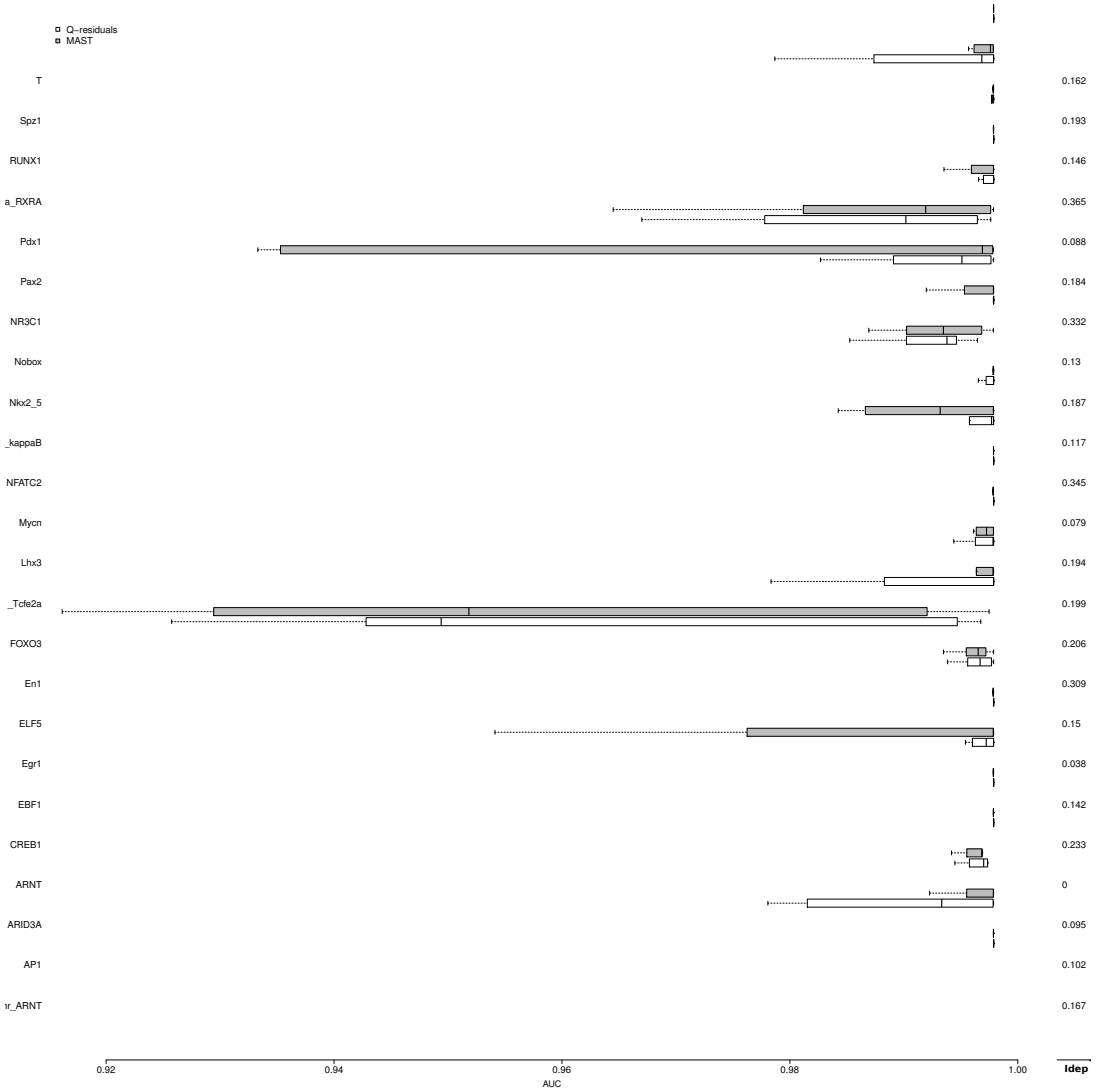


Figure 3.5. Box plot of the AUC and its variation for the studied transcription factors, comparing Q-residuals detector with the chosen number of components in white to MAST in grey. The results correspond to the background 1 of *Mus musculus*. Comp corresponds to the rate of positions within a binding site that have significant interdependences

model these dependencies Naughton et al. (2006) developed Motifscan, a graph-based algorithm which is similar to a k-nearest neighbours applied to binding motifs. The evaluation of a k-mer is based on its Hamming distance to the nearest k-mers of the motif instead of being based on the distance to a centroid as in the PSSM models.
They used 94 JASPAR (2006) motifs to compare Motifscan to PSSM methods. To do the comparison, they calculated the ROC_N curves, which are equivalent to the ROC curves but taking into account just the first N false positives, where N is the number of

Table 3.5. Computing time comparison between Q-residuals detector implemented in C, MAST downloaded from MEME suite (MEME 4.4.0) and MATCH implementation in R. The p-value for MAST was chosen $p = 0.001$, the threshold in Q-residuals as $c = 0.95$ and the Similarity in MATCH as $S = 0.85$, to have the similar numbers of TFBS detected. The background was chosen as the Background 1 of each organism and the parameters for Q-residuals and MATCH correspond to the ideal number of PC and ideal Core Similarity for each motif. All results have been computed used a AMD Athlon(tm) 64 X2 Dual Core Processor.

TF	Q-residuals (s)	Mast (s)	MATCH (s)
ABF1	0.0037	0.0196	0.3396
BCD	0.0034	0.0191	0.3293
CAT8	0.0024	0.0191	0.3372
CEBP β 35	0.0041	0.0188	0.3564
cJun	0.0043	0.0186	0.4259
cMyB	0.0034	0.0188	0.3474
DL	0.0026	0.0190	0.3199
E2F	0.0025	0.0193	0.3067
GAL4	0.0034	0.0208	0.3739
GCN4	0.0041	0.0196	0.2652
HNF1 α	0.0044	0.0194	0.3631
HNF4 α	0.044	0.0183	0.3391
HNF6 α	0.0036	0.0185	0.3519
IRF1	0.0036	0.0190	0.3456
IRF8	0.0038	0.0191	0.3413
KR	0.0033	0.0204	0.2977
LyF1	0.0040	0.0191	0.3095
MIG1	0.0024	0.0197	0.3520
NF κ B	0.0035	0.0182	0.3047
p50	0.0035	0.0182	0.3061
RFX1	0.0045	0.0185	0.3061
ROX1	0.0102	0.0186	0.3394
T3R α	0.0036	0.0193	0.3395

sequences available for the selected motif. A significant improvement of one algorithm over anoother is considered when a 5% increase in the ROC_N AUC is achieved.

Using the same methodology and 93 of the 94 motifs of the old JASPAR version (the old version of the remaining one was not available), the AUCs of the ROC_N curves were

calculated for the Q-residuals detector, and the results used to compare the detectors.

3.3.2.2 Comparison Results

Using the same criteria as Naughton et al. (2006), a 5% increase in the ROC_N AUC is required to consider a significant improvement. The results show that in 34 of the 93 studied motifs Motifscan performs better than the Q-residuals detector and the PSSMs methods, Q-residuals is the best detector in 25 of the 93 motifs and PSSM just in 1 of them. The three detectors perform equally good in 16 motifs, Q-residuals and Motifscan equally good but better than PSSM in 16 motifs, Q-residuals and PSSM better than Motifscan in 3 motifs and Motifscan and PSSM are better than Q-residuals in 9 of the 93. A visualization of the results in figure 3.6 shows that the performance of Q-residuals is more sensitive to the number of positions. When the sequences are short, the number of false positives using the Q-residuals detector increases leading to a smaller AUC. Motifscan performs better in this situation but, on the other hand, it needs more training sequences, and when the number of sequences is small, Q-residuals performs better than Motifscan. Focusing on the 37 motifs which have less than 20 sequences available, in the 43.24% of the cases the AUC of Q-residuals is significantly the best algorithm, while motifscan is the best just in a the 27.02% of this instances. In most cases, even if Motifscan is significantly better than Q-residuals, the Q-residuals algorithm performs better than PSSM methods also for this comparison.

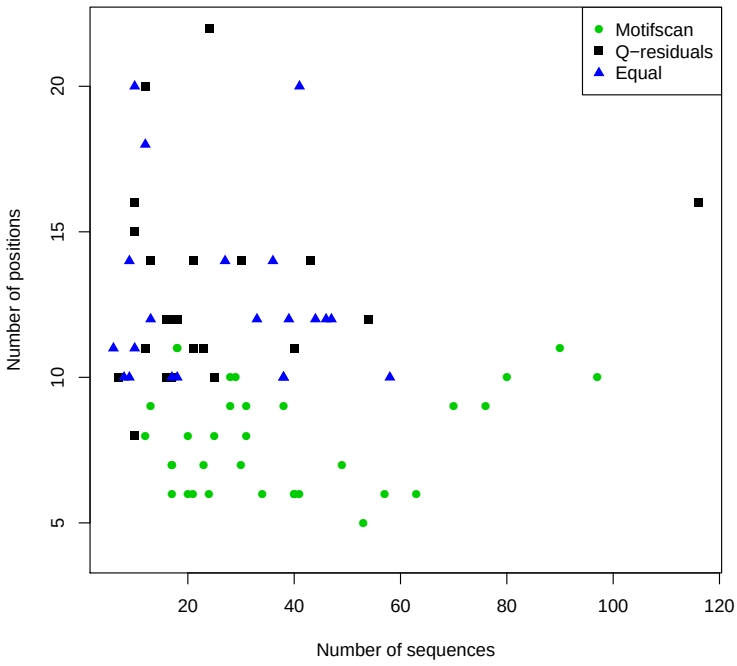


Figure 3.6. Number of position and number of sequences of the motifs where Motifscan was the best algorithm, (●) green or Q-residuals was the best algorithm, in (■) black or both perform equally (less than 5% difference in AUC) in (▲) blue. Q-residuals performs better for small number of sequences, but performs worse when the number of position per sequence is small.

4

Three way detectors

From the results in the previous chapter, it is demonstrated that a covariance analysis of numerical DNA sequences can be used to model and predict binding sites and also to find correlations between different positions. The information in each position is difficult to recover due to the arrangement of the numerical data into a 2-way matrix (e.g. loadings should be grouped into length-3 vectors and the correlation matrix into 3×3 submatrices in order to study the original sequence). The DNA binding matrix can also be arranged in a 3-way array, where the first dimension is related to the sequences, the second to the positions of the nucleotides within the binding site and the third represents the numerical DNA conversion for a particular nucleotide. Multiway techniques can then be applied to the DNA 3-way array to model the binding sites and find interdependences between positions. Thanks to the characteristics of these techniques, the resulting models could be more interpretable than PCA models. Detectors can also be built using 3-way techniques, in an analogous way to the PCA Q-residuals detector. Because of the uniqueness of PARAFAC models, the scores can be used to construct a combined QDA detector which produces similar results than the Q-residuals detectors.

4.1 Methodology

4.1.1 Datasets

A preliminary study of the application of PARAFAC analysis to model DNA motifs was done using the 5 motifs from the *homo sapiens* organism and the JASPAR database

which have more than a 30% of positions with interdependences (equation (2.3)). These 5 motifs are: ESR1, INSM1, NFATC2. NR3C1 and PPARG. Also the motif with highest dependences from TRANSFAC database, the DL binding motif, was chosen for this first analysis. DL binding sites were the ones used to show the loadings, the correlation matrix and the kind of information that we could extract from a PCA model. The idea of this study was to see how PARAFAC models can fit to DNA 3-way data and how interdependences are captured in 3-way models, choosing manually the model that could explain more features for each binding motif.
To compare the 3-way detectors to the other detectors, the 93 motifs from the JASPAR (2006) database have been used. Using this database, the 3-way detectors can be directly compared to algorithms which take into account interdependences and also to the Q-residuals detector.

4.2 PARAFAC models

4.2.1 3-way Conversion

The same numerical conversion, defined in equation (3.1) can be used to transform the symbolical DNA matrix into a numerical cube. This conversion produces a three-way data set. The data reflects the numerical conversion of the different positions of a motif for all the binding site sequences.
Given a motif with N binding sequences, each one having M positions the dimensionality of the cube is $N \times M \times d$ where d is the dimensionality of the numerical conversion which in this study is $d = 3$. An scheme of the cube with an example of the conversion of a single sequence with 6 positions is represented in the figure 4.1. Each one of the 6 nucleotides (second mode) is converted into its numerical representation (third mode), and the process is applied to each one of the sequences of the motif (first mode).

4.2.2 PARAFAC Analysis

The PARAFAC analysis of the numerical DNA cubes was done following the equation (1.20) where R refers to the number of components. In this case, $x_{i,j,k}$ are the elements of the numerical $N \times M \times d$ cube X of DNA sequences. $a_{i,r}$ are the elements of the matrix A, a $N \times R$ matrix of loadings corresponding to the first mode, the sequences. B is a $M \times R$ matrix of loadings with elements in the equation (1.20) $b_{j,r}$ corresponding to

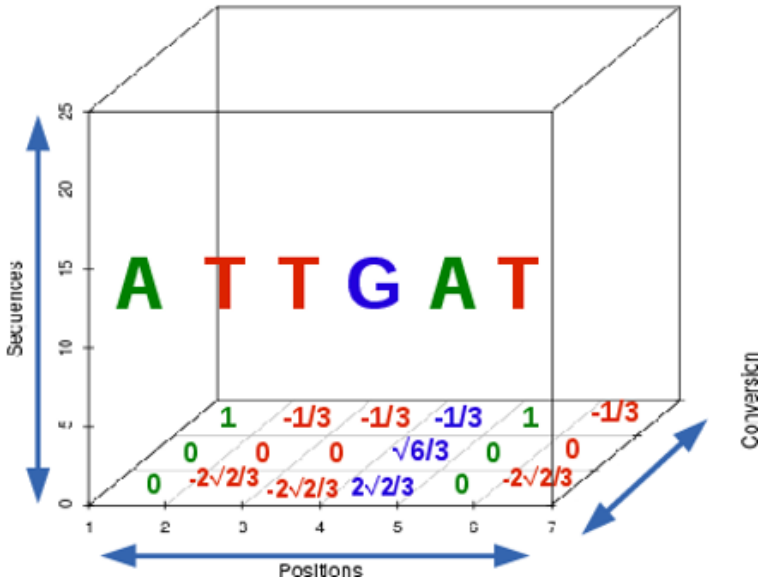


Figure 4.1. Scheme of the numerical conversion of sequences using the cube. The first mode represents the number of sequences, the second the position within the motif and the third mode represents the numerical conversion. An example of numerical conversion of a sequence is shown.

the positions of each sequence (second mode) and $c_{k,r}$ refer to the elements of C which is a $d \times R$ matrix of loadings corresponding to the third mode, the different nucleotides. $e_{i,j,k}$ are each one of the elements of the error cube.

If the PARAFAC models are interpretable, the A matrix will have information about the different sequences of the model and the B matrix will have information about the nucleotides in each position of the motif. To recover the motif information the best PARAFAC model with the ideal number of components needs to be chosen manually. The models from 1 to 5 principal components were run for all the chosen motifs. In order to choose the ideal model several steps need to be followed.

1. Construct the model for several components
2. Study of the stability of the solutions
3. Interpretation of the models

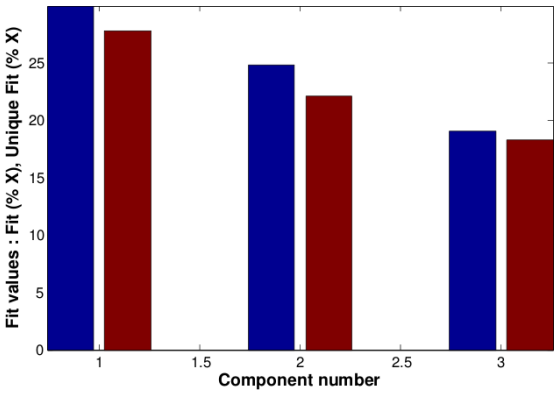
4.2.3 Building the model

In the first step the criteria to choose the number of components was the variance explained for each component and the variance that can be explained just using that

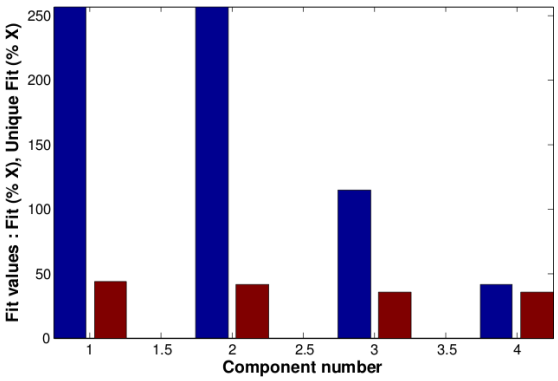
component. When a component is just a linear combination of the others, the difference between the variance explained by that component and the variance that can be explained just using that component is large, and adding more components is not translated in a better fit of the model to our data.
In the figure 4.2 the differences of the variance explained per component, in blue, and the variance explained just using that component, in red, are shown for a valid and an invalid model. In the valid model, the figure 4.2 (a) corresponding to the 3-components model of the ESR1 binding sites, the differences between the variance per component and the unique variance are small. That means that the components are almost uncorrelated to each other and that the total explained variance can be calculated adding the variance explained for each component. In the figure 4.2 (b), which represents the variance of a ESR1 4-components model, it can be observed that the differences between variances are large. Some components are linear combinations of the others, and a increase in the number of components does not translate into an increase of the explained variance nor into a more interpretable model.

Once the maximum number of components was chosen, using the difference of variance criteria, two main issues with PARAFAC model needed to be addressed: avoiding the local minima, and the robustness of the model. To avoid the local minima, the PARAFAC algorithm was run 100 times using different initial values, and then the residues were plotted to show the differences. If the model is reaching a global minimum then the residuals should be the same in all runs, but, as it can be seen in the figure 4.3 where the residuals of 30 runs of an INSM1 unstable model are represented, if the PARAFAC algorithm is reaching a local minimum the residuals change when different initialisations are used.

The second issue was studied looking at the stability of the model when sequences are removed from the training data. Because the number of sequences is typically small, a l.o.o. cross-validation was used, and the scores of the different models were compared to show the robustness. After this validation, 2 or 3 models which satisfy the main conditions (independent components, no local minima and robustness) were chosen as valid models for each motif. In the table 4.1 the number of components of the valid models for the 5 different motifs are shown.



(a) Correct number of components



(b) Too many components

Figure 4.2. The variance captured per component (in blue) and the variance captured using a simple component (in red) are presented for the 3 and 4 components PARAFAC models of the *Homo sapiens* ESR1 motif. In the example (a) a 3-components PARAFAC model is fitted to the data, and the differences between the variance per component and the variance explained using just one component are small. As the number of components increases to four, as in the example (b), the differences between the variances increase, showing that the four components are just linear combinations of each other. This means that too many components are being used.

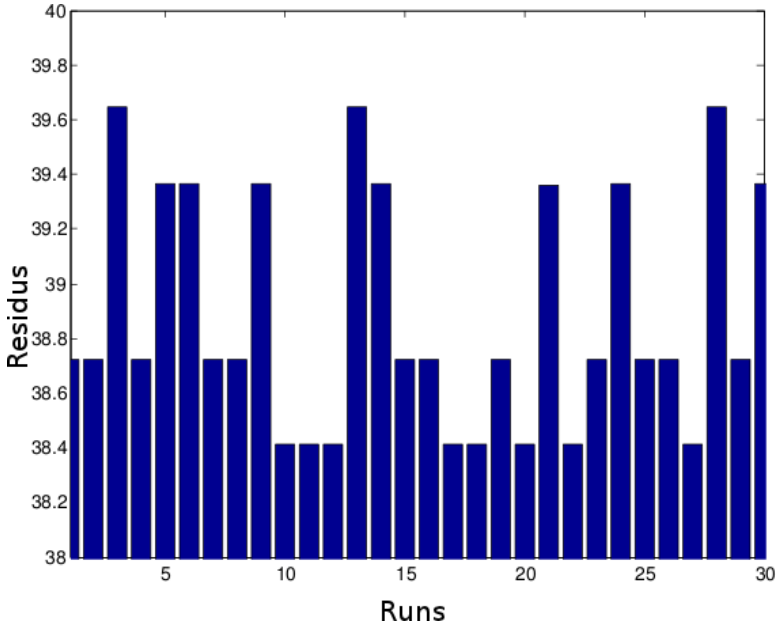


Figure 4.3. Q-residuals of 30 different runs of INSM1 motif 4-components model. The residuals vary over the different runs, indicating that the model is stuck in local minima.

Table 4.1. Table showing for each of the studied motifs which PARAFAC models are valid from a mathematical point of view and which one has the best reproduction of the sequence Logo. The one with the best reproduction of the sequence Logo was chosen as best model.

TF	Stable models	Best model
ESR1	1,2,3	3
INSM1	1,2,3	2
NFATC2	1,2,3,4	3
NR3C1	1	1
PPARG	1,2,3	3
DL	1,2	2

4.2.4 Model Interpretability

The final step on the decision of the validity of the models is to study the biological information that the model contains. In order find the model with more relevant information, we tried to identify the Logo of the sequence using the PARAFAC mode-2 loadings which are the ones referring to the positions. If the PARAFAC model has a biological meaning, the second mode should have information about the nucleotides

present in each position in a similar way than the PSSM models have.
The process to recover this information was as follows: first a position which all the nucleotides are the same (e.g. vector of A in all the N sequences), was projected into the model. In a PARAFAC model, this is equivalent to let the two matrices A and C fixed and use a least-squares to fit the second mode matrix B. This process was performed for all four nucleotides, and then the distance between the model in each of the M positions and the projection of each one of the nucleotides was calculated. Figure 4.4 shows the projection of the four nucleotides in the PPARG 2-mode using a 2-components model. The consensus sequence of the different positions is presented in different symbols and colours. Positions with an A as a consensus are presented in blue, positions with a C in magenta, the ones with a T in red and positions with a majority of G are presented in green. As it can be seen each position is closer to the nucleotide represented in the consensus sequence and more distant to the nucleotides less represented in that position. Calculating the distances of each position to each of the projected nucleotides, the Logo sequence can be recovered.

If the Logo could be recovered from the PARAFAC 2-mode scores, then the model was considered to have a meaning and therefore was considered the best model. The best model for each motif is represented in table 4.1.
The first mode of the PARAFAC model consists in a $N \times R$ matrix, being N the number of sequences of the motif and R the number of components of the PARAFAC model. The interpretation of the first mode is related to the conservation of different positions among the different sequences of the motif. Each one of the components of the PARAFAC first mode models a group of conserved positions within the binding site.
If a motif has a well defined consensus sequence, with some well conserved groups of positions, the projection of the consensus sequence in each component has a extreme value (maximum or minimum). The difference between the score of a single sequence and the score of the consensus is related to how this sequence differs from the consensus. The sequences are modelled in a similar way than with the PSSM models. One example of these motifs is the PPARG motif from *Homo sapiens*, whose different components (1,2 and 3) of the first mode in a 3-components PARAFAC model are represented in the figure 4.5. Figure 4.5 (a) presents the scores of the different motif sequences and the consensus sequence in the first and second mode, and (b) shows the same scores for the

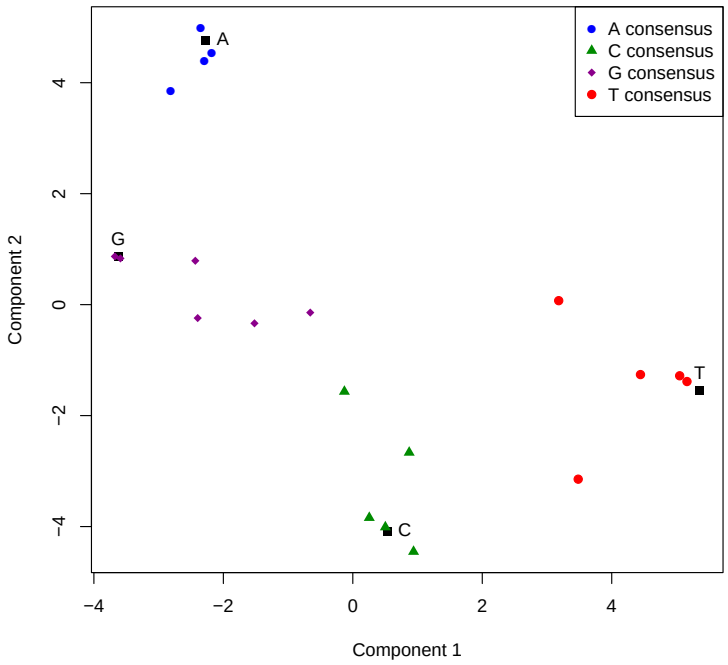


Figure 4.4. First and second components of the second mode of the PPARG 2-components model. In black the projection of each one of the nucleotides is shown, and the consensus nucleotide for each of the positions is presented in a different colour (A in blue, G in magenta, C in green and T in red). As it can be seen each position is closer to the nucleotide corresponding to its consensus sequence.

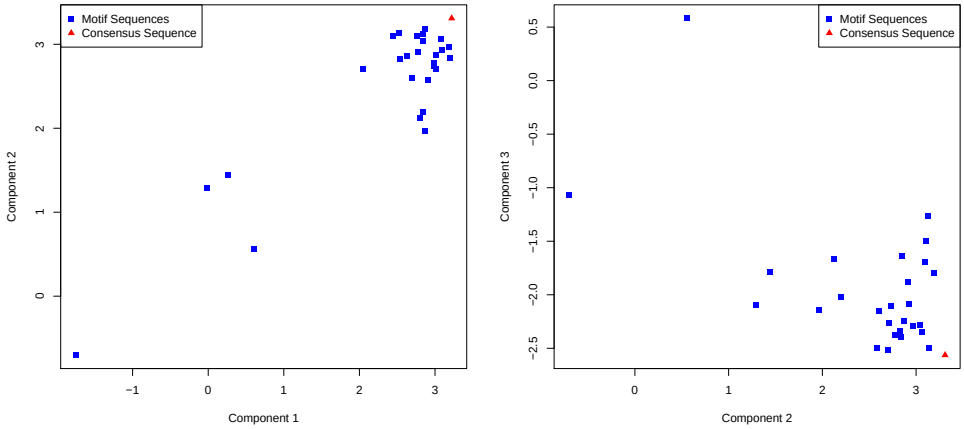
second and third mode. In both the motif sequences are represented using a blue point and the consensus sequence is represented with a red triangle. In figure 4.5 (c), the different sequences are shown together with the Logo. Looking at figure 4.5 (a) and (b), it can be seen that the consensus sequence has the highest score (in absolute value) for the three components, and sequences with differences in the most conserved positions, as the ones highlighted in yellow in the figure 4.5 (c) where the T in the second position has been substituted by an A among other changes, have low scores in absolute value. Some other motifs cannot be well described using a simple consensus. The example used to show the covariances in chapter 3, the *DL* motif from the organism *Drosophila melanogaster* is one example. In this case, there are two groups of conserved positions: some sequences have a group of A nucleotides at the beginning of the sequence and some others have a group of G nucleotides at the end. In this example, shown in the figure 4.6 a 2-components PARAFAC model can be used to describe the sequences. The first component is related to the group of G-conserved positions, and the second group to the A-nucleotide. A sequence having G-conserved positions has a high score in the first component, and a sequence with the A-conserved group has higher second component score. The consensus is not a extreme value in this case, and it can be found among the other sequences. PSSM models are not useful to model this kind of motifs.

4.3 PARAFAC detectors

4.3.1 Residuals Detector

4.3.1.1 Construction of the Detector

Following the Q-residuals detector constructed using PCA, we can also use the residuals of the PARAFAC model in order to detect transcription factor binding sites within a DNA promoter sequence. The Q-residuals follow, in a PARAFAC analysis, the same distribution as in PCA. They can be converted in the same way to follow a Gaussian distribution (Durante et al., 2011).



(a) First and second components **(b)** Second and third components

(c) Sequences and logo

Figure 4.5. Scores for the PPARG 3-components model in a blue circle and the consensus PPARG sequence in a red triangle. Figure (a) represents the first and second components and (b) the second and the third and (c) shows the sequences and the logo, and some of the most diverging sequences from the consensus have been highlighted in yellow. As PPARG has a clear consensus sequence with highly conserved positions, the consensus sequence is an extreme value and as the difference between a sequence and the PPARG consensus increase,the score of the sequence differs more from the consensus sequence.

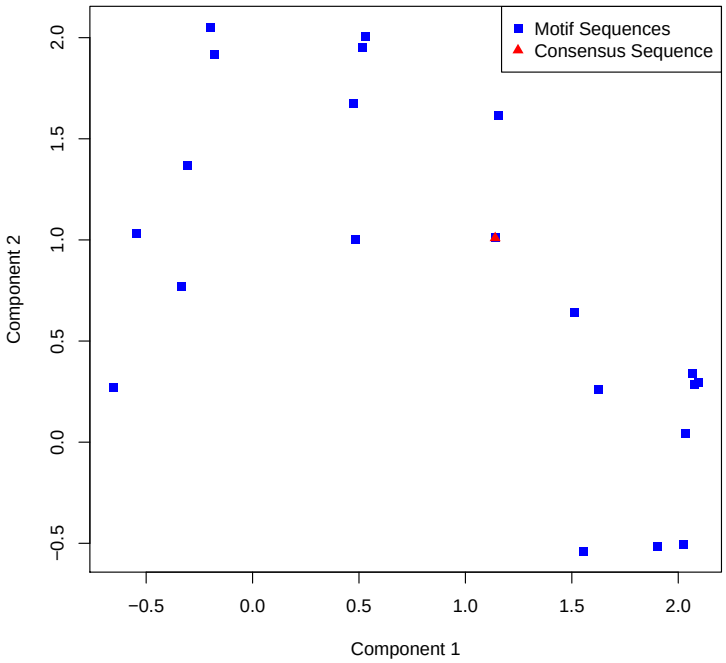


Figure 4.6. First and second components of the DL 2-components model in blue and the DL consensus scores in red. The DL motif has different kind of sequences and it has not a clear consensus, the consensus is not an extreme and the different components are representative of the different sequences.

Table 4.2. Summary of the performances of the different algorithms: PARAFAC, Q-residuals PCA and Motifscan using the JASPAR (2006) database.

Method	Best
Motifscan	32
PARAFAC	15
Q-residuals	9
PSSM	0
None	39

4.3.1.2 Detection Results

To compare the PARAFAC Q-residuals detector with the one constructed using a PCA, and at the same time to the other detectors, we used the database of Motifscan and we compared the ROC_N curves for the 93 TF.

The results using the PARAFAC detector are comparable to those using the other detectors, even if PARAFAC only captures the trilinearities.

When compared only to the Q-residuals detector, the results show that PARAFAC and Q-residuals performances are similar. In 34 motifs Q-Residuals is better with more than a 5% increase in the AUC_N, in 35 motifs Parafac is better with more than a 5% and in 28 motifs there is no significant difference between the methods. As it can be seen in the figure 4.7 no differences in performing are related to the length of the sequences or the number of sequences available for modelling.

Motifscan and PSSM results where also added to the comparison with the Q-residuals PARAFAC. The results do not change very much respect to the previous comparison with the PCA Q-residuals detector. The Motifscan algorithm performs better in 32 of the 93 studied TF binding motifs, and it was better in 34 of them when it was compared to the PCA Q-residuals. The two that changed perform better with PARAFAC. The PCA Q-residuals performs better in 9 of the 93 motifs, and PARAFAC in 15. A summary of the comparison results can be seen in the table 4.2

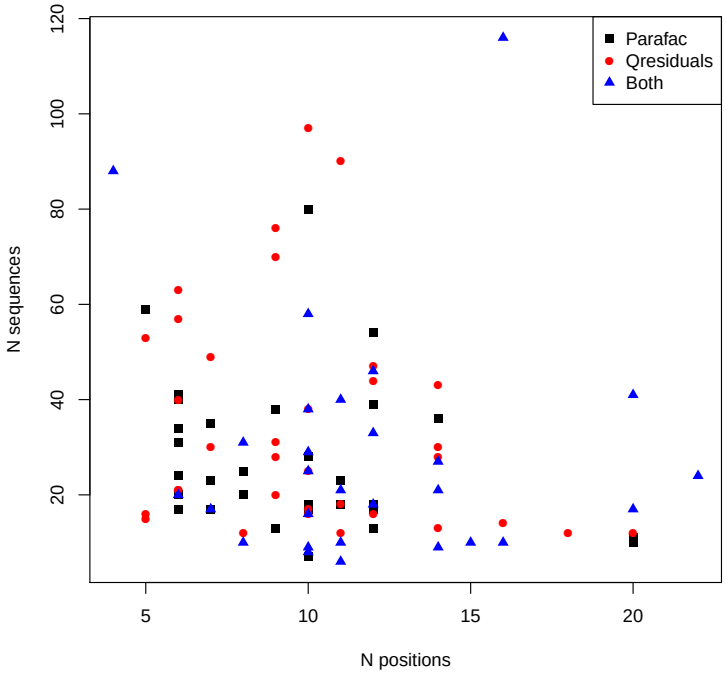


Figure 4.7. PARAFAC Q-residuals detector Compared to PCA Q-residuals Detector. The motifs where PARAFAC performs better are represented in black, the ones where Q-residuals performs better in red and the motifs where both algorithms perform in a similar way in blue. The x-axis represents the Number of positions of each motif, the y-axis the number of sequences. As it can be observed, the numbers of positions or sequences do not have a clear influence on which detector performs better, unlike in the comparison between Motifscan and PCA Q-residuals.

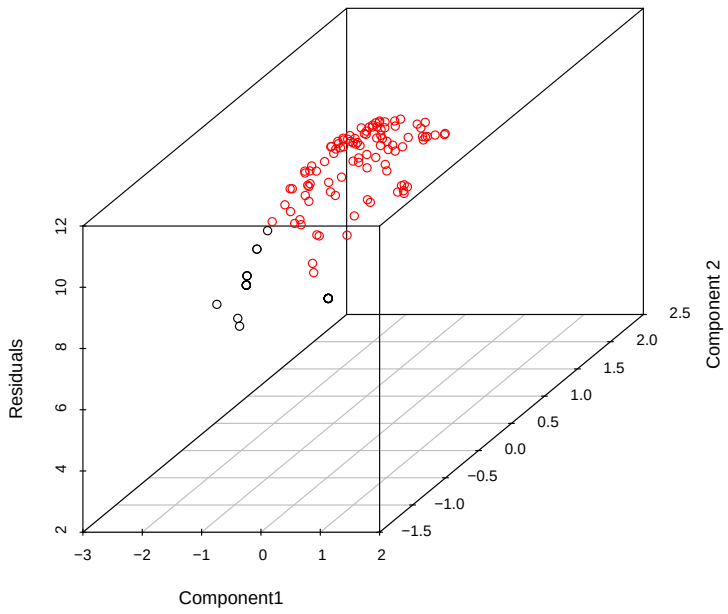


Figure 4.8. Scores and Q-residuals for the PARAFAC model of a 2-components model of the INSM1 binding sites in black and the scores and Q-residuals of 100 random sequences projected into this model in red. The Q-residuals and the scores can be combined to produce a binding sites detector.

4.3.2 QDA Detector

4.3.2.1 Construction of the Detector

The advantage of the PARAFAC model is that the scores represent properties of the sequence, meaning that the scores from the binding sites sequences should be different than the scores from random or other genomic sequences. The figure 4.8 shows the scores and the Q-residuals of the INSM1 2-components models in black and the scores and the residuals predicted for 100 random sequences in red. Both the Q-residuals and the scores can be used to differentiate the binding sites from the genomic sequences and thus, a combined detector can be constructed to improve the PARAFAC Q-residuals detector results.

To incorporate multiple measurements in a discrimination problem Ronald A. Fisher (1936) developed the linear discriminant analysis (LDA). LDA is a discrimination tech-

nique which uses a linear combination of features in order to separate classes of objects or events. It assumes that the classes are normally distributed and also that the covariance is the same for all the different classes. The Quadratic discriminant analysis (QDA) is closely related to the LDA, but the covariance of the classes is not assumed to be identical (McLachlan, 1992). As the covariance of the Q-residuals and the scores of the binding sites and the other genomic sequences should not be the same, QDA seems more appropriate than LDA in order to construct a binding site detector.The Quadratic discriminant detector, when there are two classes $K = 0, 1$ uses the log-likelihood ratio as a measure for discrimination, as it is shown in equation (4.1)

$$\frac{\sqrt{2\pi \mid \Sigma_{k=1} \mid}^{-1} exp(-\frac{1}{2}(x-\mu_{k=1})^T \Sigma_{k=1}^{-1}(x-\mu_{k=1}))}{\sqrt{2\pi \mid \Sigma_{k=0} \mid}^{-1} exp(-\frac{1}{2}(x-\mu_{k=0})^T \Sigma_{k=0}^{-1}(x-\mu_{k=0}))} < t, \tag{4.1}$$

where Σ_k is the variance of the class k, and μ_k is the mean. x is the vector of values used for discrimination and t is the threshold that makes the separation of the classes maximum. In our case the two classes represent the binding sites and the genomic sequences and x is the vector joining the scores and Q-residual for the studied sentence The main difference between the QDA and the LDA is that with QDA, instead of having a linear separation between classes, there is a quadratic surface of separation. In bioinformatics QDA has been widely used, for example, to identify protein coding regions (Zhang, 1997), or to detect splice sites (Zhang, 2003).

The methodology that was used to compare the QDA detector is as follows: Using a l.o.o cross-validation, $N-1$ the binding motif and 1000 random sequences were used to create a training set and to construct a detector. The left-out sequence was then detected within a promoter, and with the N points the ROC_N curve was computed and used to calculate the AUC_N. The procedure was followed for a range between 1 and 3 PARAFAC components. The best number of components was used to compare the QDA to the other detectors.

4.3.2.2 Detection Results

In the comparison between the QDA detector and the PARAFAC Q-residuals detector, the results show that the QDA detector performs at least as well as the PARAFAC Q-residuals detector in most of the motifs.

Only in 7 of the motifs, the Q-residuals PARAFAC performs significantly better than

Table 4.3. Summary of the performances of the different algorithms when QDA is compared to PARAFAC, Motifscan, Q-residuals and PSSM.

Method	Best
Motifscan	28
PARAFAC	1
Q-residuals	5
QDA	11
PSSM	0
None	48

the QDA. Most of them coincide with the motifs that have a QDA training matrix with singular covariance. In 28 of the 93 motifs QDA performs better than Q-residuals PARAFAC with a 5% increase in the AUC_N, and in all the others they perform similar. Doing a global comparison, the Motifscan detector is still the best detector in 28 of the 93 motifs, the Q-residuals detector is the best in 5 motifs, the Q-residuals PARAFAC just in 1 and the QDA detector in 11. The table 4.3, summarizes the results when all the methods are taken into account

If we look at the Motifscan detector, it was the best detector in 34 of the 93 motifs when it was compared only to the PCA Q-residuals detector, and it was the best in 28 when the two other numerical detectors are included in the comparison. The results of the comparison do not change very much as we include numerical detectors. On the other hand, when the numerical detectors are included in the comparison, the results of these detectors change. It can be inferred that there are some motifs that are best detected with numerical detectors and others that are best detected with Motifscan. And also that both motifscan and the numerical detectors perform significantly better than the PSSM algorithms.

The comparison between the numerical and non-numerical detectors is shown in figure 4.9. In the figure the number of sequences and the number of positions of the motifs are shown, the motifs where the numerical detectors perform better are depicted in blue, and the ones where the non-numerical detectors are better are depicted in red. If the two detectors perform similar, the motifs are depicted in black. As it happened in the comparison with the Q-residuals detector, it can be seen that when the number of

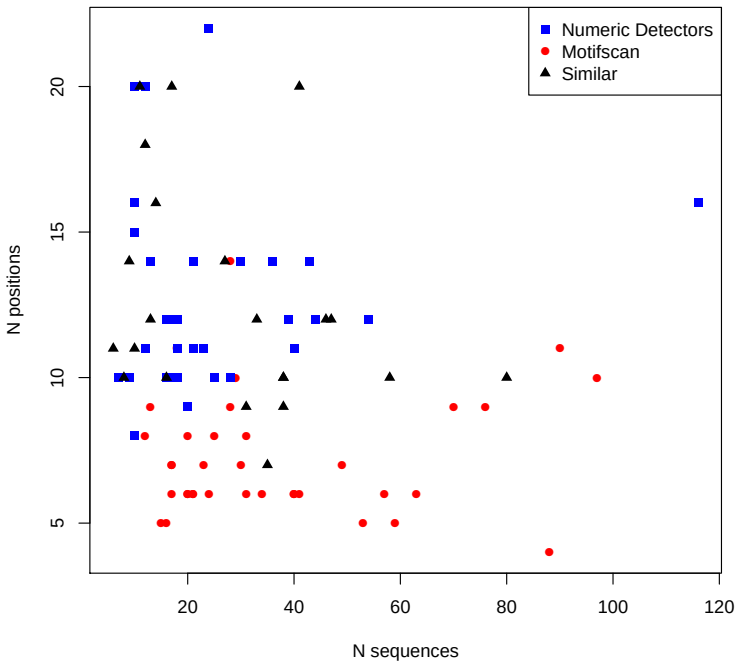


Figure 4.9. Comparison between numerical and Non-numerical detectors. The number of sequences and the number of positions per sequence of each of the 93 JASPAR(2006) motifs are depicted. If the best detector is Motifscan the motifs are depicted in blue, if the best detector is a numerical detector the motifs are depicted in red, and if the performance is similar they are depicted in black. The results show again that the numerical detectors need less sequences to perform better, but in the other hand they need more positions per sequence.

available sequences is small, the numerical detectors perform better than the Motifscan, but that they need longest sequences.

5

Conclusions

In this thesis a new methodology to detect binding sites using a set of known binding sequences was studied. The new methods use multivariate signal processing techniques, PCA and PARAFAC, in order to model the binding motifs. The detectors built using these techniques outperform the PSSM methods in all the studied datasets and need less sequences than the methods that take into account interdependences.

- Some TF participate in the transcription of almost all genes, as the TATA-box, and others are only involved in the transcription of genes in response to some signal or associated to some tissue. The relationship between the number of genes regulated by a TF and the number of TF that are involved in the regulation of a gene was studied for *Homo sapiens*. Most TF are involved in the regulation from 5 to 10 genes, and most genes are regulated by a range varying from 1 to 10 TF.

- A characterization of the interdependences between positions can be performed using the Bayes factor. Almost all binding motifs have significant interdependences, but a simple study of the percentage of interdependences between positions is not enough to separate binding sites within families or classes, because of the complexity of the binding (e.g. TF cooperation to start gene expression, binding to small molecules).

- Converting DNA into numerical sequences can be used to apply known signal processing techniques to the study of binding sequences. The variance of the

numerical sequences, in spite of being a second order statistics, is able to capture interdependences between the different positions of the binding sites.

- A PCA model was applied to the numerical binding site motifs and the Q-residuals of this PCA model were used to distinguish between binding sites and genomic sequences. When there are no interdependences the Q-residuals detector performs as good as the studied PSSM models, MATCH and MAST, and there is a correlation between the improvement in AUC and the percentage of positions showing interdependences into a TFBS motif. This result proves that a covariance-based model can be useful to detect TFBS within large databases.

- The average computational time of the Q-residuals detector, for a background sequence of 1500 bp is $0.0191 \pm 0.001s$, compared to the $0.03 \pm 0.001s$ of the MAST algorithm, also implemented in C, or the $0.33 \pm 0.01s$ of an R implementation of MATCH . The constructed Q-residuals is faster than PSSM based methods in contrast with other methods that take into account interdependences which usually have a high computational cost.

- Compared to a method that takes into account interdependences, the Q-residuals detector shows a significant improvement on the performance when the number of sequences is small, but it shows a larger sensitivity to the number of positions. It needs more positions than Motifscan or PSSM-based methods to decrease the number of false positives.

- Converting the aligned motifs to 3-way numerical data allows the use of N-way methods as PARAFAC which can provide an interpretation of the models. PARAFAC captures some of the features of the binding motif, e.g. the consensus sequence if there is one, or different sequences when the motif can bind to two consensus. PSSM models are unable to model the second kind of motifs.

- Binding sites can be detected using the Q-residuals of the PARAFAC model, analogously than using the Q-residuals PCA model. The PARAFAC detector performs similar than the PCA detector. The scores of the PARAFAC model can also be used to construct a quadratic detector that performs better than the PARAFAC Q-residuals detector.

- When PARAFAC and PCA detectors are compared to MotifScan, which takes into account interdependences, it can be seen that the numerical detectors usually need less sequences in order to construct a reliable model (which means a reliable detector) of the binding motifs. On the other hand, they are more sensitive to the number of positions.

The future work can go into different directions: the first one is to apply the current detectors and the constructed models in order to find binding sites whose mutations can be related to some diseases, and the second one is to incorporate some external information in order to find functional binding sites and reduce the number of *in vivo* false positives. This could be done taking into account the presence of other binding sites and the absence of nucleosomes or compacted chromatin structure.

6

Resum en català: Detecció de punts d'unió de factors de transcripció mitjançant tècniques de processament de senyal

6.1 Introducció

Els organismes tenen la seva informació genètica codificada en els quatre nucleòtids de l'àcid desoxirribonucleic o ADN. La mínima unitat d'informació són els gens, curtes cadenes d'ADN que contenen la informació necessària per crear una proteïna.

El dogma general de la biologia molecular diu que la informació codificada en l'ADN és primer transcrita cap a l'àcid ribonucleic (ARN) i després traduïda a proteïnes. I tot i que, com a idea general es considera encara vàlida, en realitat el procés d'expressió genètica és molt més complex. De fet, més del 90% de l'ADN forma part de seqüències no codificants la majoria de les quals tenen la funció de regular l'expressió dels diferents gens.

A la figura 6.1, on les regions no codificants estan representades en blau, es poden observar els diferents passos que porten des de l'ADN fins la proteïna. La representació de l'ADN abans de començar el procés d'expressió ens mostra com el gen es troba precedit d'una seqüencia no codificant, anomenada seqüència promotora o simplement promotor, on molts dels elements que controlen la transcripció del gen s'uneixen. Dins

6. RESUM EN CATALÀ: DETECCIÓ DE PUNTS D'UNIÓ DE FACTORS DE TRANSCRIPCIÓ MITJANÇANT TÈCNIQUES DE PROCESSAMENT DE SENYAL

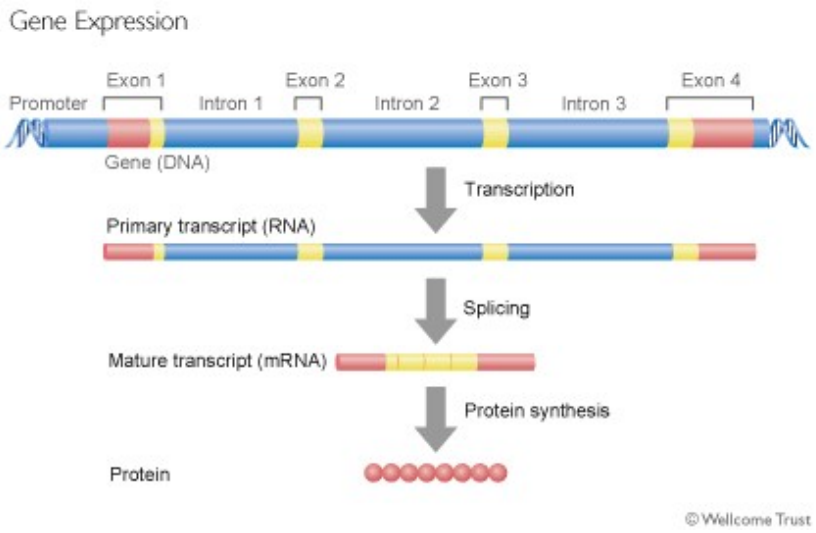


Figure 6.1. Descripció del procés d'expressió genètica, que dóna lloc a les proteïnes. El gen es troba precedit per una seqüència regulatòria, el promotor del gen, on unes proteïnes s'uniran per tal de començar la seva expressió. Dins el gen hi ha també altres seqüencies, anomenades introns, que no formaran part de la proteïna final. En un primer pas, el gen és transcrit a ARN, incloent exons i introns, i després, mitjancant el splicing alternatiu els introns són tallats donant pas l'ARN final que després de la traducció dóna lloc a una proteïna.

el gen es poden obervar també diferents regions codificants o exons i no codificants o introns. En el procés de transcripció el DNA és convertit a la seva cadena de ARN complementària, i el resultat és el ARN missatger (mRNA) que conté tots els introns i exons. Un altre procés anomenat splicing alternatiu comença en aquest moment, i els introns es tallen donant pas al ARN missatger madur, format només per seqüencies codificants i que, després del procés de traducció donarà lloc a la proteïna.

Els genomes dels primers organismes van ser descoberts a mitjans dels 90, des de llavors i sobretot des que el genoma humà va ser descobert, hi ha hagut el què s'anomena la revolució genòmica, i cada vegada tenim accés a més bases de dades que ens permeten estudiar més a fons el procés d'expressió. En aquesta tesis ens centrarem en la regulació de la transcripció i, sobretot, en la regió promotora on unes proteïnes anomenades factors de transcripció s'uneixen per tal de començar-lo i regular-lo.

6.1.1 Regulació gènica

Tal i com hem comentat abans, la regulació de l'expressió dels diferents gens és molt complexa, i té lloc a tots els diferents processos que van des de l'ADN fins a la proteïna final.

6.1.1.1 Regulació de la transcripció

El primer mecanisme regulatori és el control de l'accés a l'ADN a la maquinària que el transcriu mitjançant l'estructura de la cromatina a l'entorn del gen.
La cromatina és la estructura formada per l'ADN i les histones. La seva unitat bàsica és el nucleosoma que consisteix en l'ADN enrotllat 2 vegades en 8 histones. Després aquesta estructura pot estar més compacta, formant un solenoid quan es vol evitar la transcripció dels gens, o més oberta si es vol permetre.
A través del que s'anomenen factors de remodelació de la proteïna, i també a través de mecanismes més complexes, l'estructura de la cromatina es modifica al voltant d'un gen per tal de permetre el següent pas en la seva regulació que és la unió de factors de transcripció a la seqüència promotora. Els mecanismes d'actuació són normalment tres: alterar l'associació de les histones al cromosoma, moure les histones a una altra regió de l'ADN i posar el nucleosoma a una altra molècula.
El segon pas en la regulació de la transcripció ve donada pels factors de transcripció que són proteïnes que s'uneixen a seqüències específiques a l'ADN, i donen la senyal per iniciar o impedir l'expressió del gen regulat. Aquestes seqüencies, anomenades punts d'unió dels factors de transcripció, són seqüencies curtes i degenerades, és a dir, que poden canviar alguns nucleòtids sense perdre la funció i, tot i que generalment es troben a la zona promotora del gen hi ha molta varietat. Hi ha factors de transcripció que s'uneixen a seqüències més distants i actuen com a *enhancers* (o augmentadors), augmentant la quantitat de mRNA transcrit, o com a *insulators* (aïlladors) que aïllen el gen de l'acció d'altres factors de transcripció. Per tant, tot i que el coneixement d'aquestes seqüències ens donaria moltes claus per entendre l'expressió genètica, no podem buscar una sola seqüència en un lloc específic de l'ADN, sinó que hem de desenvolupar mecanismes més complexes, siguin experimentals o computacionals (tal i com explicarem a la secció de mètodes per a descobrir punts d'unió).
En general, la unió d'un sol factor de transcripció no és suficient per a la transcripció

d'un gen, sinó que diferents factors i altres molècules cooperen per fer-la possible.

6.1.1.2 Regulació post-transcripcional

El resultat de la transcripció s'anomena pre-mRNA i, des del mateix moment en què es comença a crear, comença la segona part de la regulació: la regulació post-transcripcional.
El primer pas consisteix simplement en alterar l'extrem 5' de l'ADN per tal d'impedir la seva degradació. Un segon pas té lloc encara dins el nucli, l'alternative splicing, que consisteix en la separació d'introns i exons que permet la combinació dels diferents exons de formes diferents de forma que donin lloc a diferents mRNA madurs que seran després expressats en proteïnes. Després el mRNA madur és transportat fora del nucli i altra vegada regulat pel que s'anomena micro RNA (miRNA) que actua evitant la traducció a proteïnes. Finalment, les proteïnes també poden ser regulades mitjançant processos com la degradació.

6.1.2 Bases de dades de punts d'unió de factors de transcripció

Trobar els punts d'unió per als diferents factors de transcripció és un procés molt important de cara a poder entendre la regulació genètica. Per tant, hi ha bases de dades que emmagatzemen els punts d'unió per a factors de transcripció que s'han pogut verificar experimentalment. Les més famoses són TRANSFAC que té una versió pública que data del 2005 i JASPAR que continua sent pública.

6.1.2.1 TRANSFAC

TRANSFAC és una base de dades que conté punts d'unió que han estat anotats manualment i també experimentalment validats. La seva última versió pública TRANSFAC 7.0 és del 2005 i conté informació de 2397 gens i 6133 factors de transcripció.
Des de llavors TRANSFAC forma part de la companyia BioBase i no té més versions públiques. A la versió actual s'inclou també informació de miRNA, i el número de factors de transcripció estudiat puja fins a 18211.

6.1.2.2 JASPAR

La base de dades JASPAR és, a hores d'ara, la base de dades de punts d'unió de factors de transcripció d'accés públic més gran que existeix. La majoria de punts d'unió han estat verificats *in vivo* tot i que a les noves versions s'hi han començat a afegir alguns factors de transcripció validats *in vitro*.
La versió actual disposa de punts d'unió per a 590 factors de transcripció per a diferents organismes, des de vertebrats fins a fongs o plantes. Els punts d'unió estan organitzats de forma matricial, de manera que és fàcil modelar els diferents factors de transcipció.
A part dels punts d'unió inclosos a la base de dades principal, el que s'anomena JASPAR core, hi ha altres punts d'unió que no reuneixen els criteris de qualitat per a pertànyer a la base de dades, però que poden ser punts d'unió reals i es troben també catalogats. El número de factors de transcripcó final és 840.
A la última versió s'ha inclòs un paquet de Python i un de R que permeten treballar fàcilment amb JASPAR.

6.1.2.3 Altres bases de dades

També hi ha altres bases de dades més petites de factors de transcripció, algunes com ABS i Mapper contenen dades de diferents organismes, altres com VISTA es centren en *enhancers*, i finalment algunes altres com Redfly només tenen dades d'un organisme en concret (en aquest cas la *Drosophila melanogaster*).

6.1.3 Alíniament de seqüències

El concepte d'alíniament de seqüències va aparèixer en biologia per respondre la pregunta de quines seqüències eren homòlogues, ja que es pensa que aquest tipus de seqüències provinents d'un ancestre comú són similars i a més, tenen funcions similars. En el nostre contexte és important ja que els punts d'unió són seqüències similars però no idèntiques i per a modelar-los cal saber quines són les posicions corresponents a cada seqüència.
Per tal d'aliniar dues seqüències les podem escriure una sota l'altre, posant els nucleòtids iguals a la mateixa posició i considerant els diferents una mutació, una inserció

o una deleció. Si volem saber si aquest alíniament és bo o no, podem crear un marcador que ens indiqui quants nucleòtids són iguals i quants difereixen entre les dues seqüències, els més simple dels quals seria posar un $+1$ a cada nucleòtid coincident i -1 a cada mutació, inserció o deleció. Per a cada parell de seqüencies, aquell alíniament amb un marcador més alt és el que considerarem millor.
Aquest concepte d'alíniament es pot extendre fàcilment a l'estudi de N seqüències, per poder crear per exemple, un arbre filogenètic que ens doni les relacions entre totes elles. Hi ha dues preguntes però que necessiten resposta: (1) Com podem evaluar l'alíniament múltiple i (2) Quin mètode es pot fer servir per trobar l'alíniament ideal? Tot i que no hi ha una resposta exacta per aquestes dues preguntes, hi ha molts algoritmes que han trobat bones aproximacions, i es poden dividir en dos grans grups: mètodes progressius i mètodes iteratius.
En els mètodes progressius primer es construeix un arbre filogenètic per trobar les dues seqüències més similars, aquestes dues s'alinien i després les altres seqüències es van afegint a l'alíniament. El principal problema que tenen aquests mètodes és la gran dependència en l'arbre inicial, ja que si és de mala qualitat, l'alíniament no serà bo. Un exemple és el CLUSTALW. En els mètodes iteratius l'arbre inicial i l'alíniament es van construint iterativament fins que l'algoritme convergeix, un exemple seria el MUSCLE. Finalment, en els darrers any han aparegut mètodes d'alíniament més complexes, com algoritmes genètics o cadenes de Markov.

6.1.4 Detecció de punts d'unió

Tot i que els mètodes experimentals per detectar punts d'unió han millorat molt, continuen sent cars i complexes. La detecció de punts d'unió mitjançant mètodes computacionals és doncs un bon complement, o fins i tot una forma de substituir aquests mètodes experimentals. Les tres grans dificultats que els algoritmes de detecció de punts d'unió han de superar i que coverteixen la seva detecció en un repte són:

1. Els punts d'unió són degenerats, és a dir, canvis en la seqüència poden no tenir cap efecte en la seva funció.

2. Són seqüències curtes (uns 20 bp)

3. Es poden trobar en qualsevol lloc del genoma, tot i que principalment es troben a la regió promotora dels gens.

Tots els algoritmes tenen dos principals passos: la construcció del model i després la puntuació dels punts d'unió. Alguns algoritmes utilitzen seqüències conegudes d'un punt d'unió per crear un model i detectar-ne d'altres dins el genoma, altres algoritmes intenten descobrir nous punts d'unió a partir de seqüències no aliniades o relacions filogenètiques, aquests s'anomenen algoritmes de descobriment de punts d'unió.
Els primers algoritmes que es van crear, són algoritmes que representen els punts d'unió com oligunucleòtids, A cada posició li correspon el nucleòtid més comú creant aix'i la seqüència consensus, que seria la que millor s'uniria al punt d'unió. Evidentment per detectar els punts d'unió cal una certa flexibilitat en aquesta seqüència consensus, per exemple permetent un número de mutacions, o representant-la en el codi IUPAC on diferents lletres simbolitzen que hi ha més d'un nucleòtid possible en una posició. Tot i la seva antiguitat i simplicitat, aquests mètodes són encara utilitzats degut a la seva eficàcia, un exemple seria el WEEDER.
Aquestes primeres representacions van evolucionar en el que es coneix com Position Specific Scoring Matrices (PSSM) o matrius de pesos. Tenint les seqüències d'un motiu aliniades, la PSSM consisteix en una matriu de $4 \times M$ dimensions, on M són les posicions del punt d'unió, que conté les freqüències de cadascun dels nucleòtids en cada posició. Per a evaluar si una seqüència forma part d'un punt d'unió, es sumen les freqüències dels nucleòtids de la seqüència a cada posició, obtenint així un valor final que indica la probabilitat de la seqüència de ser un punt d'unió. Noves versions d'aquestes matrius calculen la informació per posició enlloc de la freqüència, i assumeixen que les seqüències de punts d'unió tenen més posicions conservades i, per tant, un valor basat en la informació per posició es pot fer servir per a calcular la probabilitat que una seqüència sigui un punt d'unió o no. Fent servir la informació per posició es pot construir el que s'anomena un Logo que indica per a cada posició quina és la informació. A la figura 6.2 es veuen les diferents seqüències corresponents a un motiu, la construcció de la consensus (code IUPAC), la matriu de pesos i el Logo.
Des de l'any 2000 han aparegut molts estudis experimentals i computacionals suggerint que les diferents posicions ens els punts d'unió tenen dependències i, per tant, uns models com les PSSM, que només tenen en compte la freqüència en cada posició,no

6. RESUM EN CATALÀ: DETECCIÓ DE PUNTS D'UNIÓ DE FACTORS DE TRANSCRIPCIÓ MITJANÇANT TÈCNIQUES DE PROCESSAMENT DE SENYAL

a)
```
TGACTCA
TTCCTGG
TGAGTCA
TGAGTCA
TGAATCA
AGTGTCA
TGAGTCA
TGAGTAA
TGATTAA
```

b) `TGACTCA`

c) `WKHNTVR`

d)

	1	2	3	4	5	6	7
A	1/9	0	7/9	1	0	3/9	8/9
C	0	0	1/9	2/9	0	5/9	0
G	0	8/9	0	5/9	0	1/9	1/9
T	8/9	1/9	1/9	1/9	1	0	0

e)

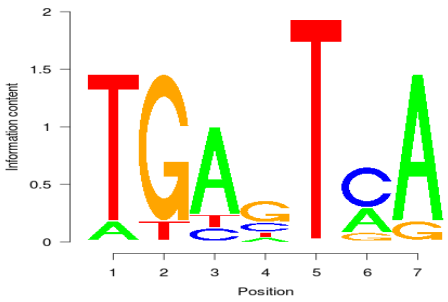


Figure 6.2. Exemple de punts d'unió per a un factor de transcripció, construcció de la seqüència consensus, de la matriu de pesos i del Logo que indica la informació per posició.

són prou adequats per modelar els punts d'unió. Des de llavors alguns algoritmes han començat a afegir les dependències entre posicions, la forma més simple és també afegir di-nucleòtids a les PSSM però això normalment no és suficient i per tant nous models han aflorat.

Les cadenes de Markov d'ordre n són una bona forma de modelar les interdependències, però tenen l'inconvenient que el número de paràmetres creix exponencialment. Les cadenes de Markov d'ordre variable, o les xarxes bayesianes redueixen els paràmetres a estimar, tot i que computacionalment són models costosos. Una altra bona alternativa, i la que es va fer servir per comparar els nostres algoritmes és un model que utilitza grafs.

6.1.5 Processament de senyal per l'ADN

L'ADN està codificat en un alfabet de 4 lletres i, per tant, es pot considerar com a informació digital. La seva conversió a seqüències numèriques permet l'aplicació de tècniques de processament de senyal clàssiques a l'anàlisi genòmic.

La transformació més comú de l'ADN és aquella en què cada base es transforma en un vector de 4 dimensions. A=(1,0,0,0), C=(0,1,0,0), G=(0,0,1,0), T=(0,0,0,1). Ja que la suma de les 4 dimensions sempre serà 1, podem reduir la dimensionalitat sense perdre cap generalitat. Així doncs, podem fer servir una conversió tridimensional on cada nucleòtid es troba al vèrtex d'un tetraèdre regular, tal i com es pot veure a la figura 6.3

Aquesta conversió és simètrica, ja que les distàncies entre els diferents nucleòtids són iguals (D=1). Una nova reducció de dimensions, on cada nucleòtid es troba a a l'extrem d'un quadrat perd aquesta simetria, tot i que pot ser útil si es vol donar més similaritat entre diferents nucleòtids. Finalment, també existeixen representacions unidimensionals de l'ADN on cada nucleòtid es pot representar per un simple número.

Les aplicacions d'aquestes conversions numèriques es troben sobretot en l'àmbit de la detecció de gens, ja que les seqüències codificants tenen una periodicitat, però també hi ha altres aplicacions com trobar correlacions a llargues distàncies a l'ADN, o simplement la visualització de llargues seqüències.

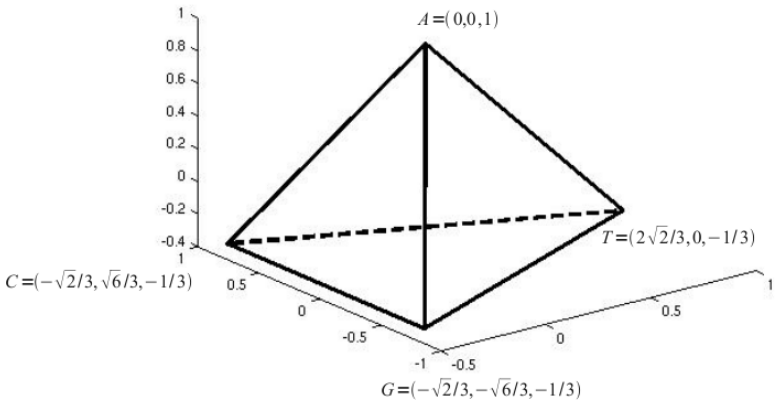


Figure 6.3. Conversió tridimensional de l'ADN simbòlic a ADN numèric. Cada nucleòtid es troba situat a un vèrtex d'un tetraèdre regular, amb distància entre nucleòtids D=1.

6.1.6 Mètodes d'anàlisi multivariant

La conversió d'ADN simbòlic en ADN numèric permet l'aplicació de tècniques d'anàlisi multivariant a les seqüències de factors de transcripció. En aquesta tesis s'han fet servir dues tècniques: anàlisis de components principals i PARAFAC

6.1.6.1 Anàlisis de components principals

L'anàlisis de components principals o PCA és una tècnica d'anàlisis multivariant que consisteix en la reducció de la dimensionalitat d'unes dades intercorrelades tot i mantenint la màxima variança.

L'anàlisis de components principals es pot descriure com una descomposició bilinial de les dades, on la variança perpendicular a l'espai de components principals és minimitzada, cosa que és equivalent a trobar la matriu de valors propis de la covariança. Es pot descriure amb l'equació (6.1)

$$X = AB^T + E \tag{6.1}$$

on X és la matriu de dades originals, amb N mostres i M variables, A és la matriu de dades projectadad al nou espai o *scores* que consisteix en N mostres i nPCS (components principals) columnes i B és la matriu dels *loadings* amb dimensions $M \times nPCs$ que representa el canvi de base. E és l'error associat al model.

Per calcular si una mostra s'ajusta bé al model de components principals es fan servir dues mesures. El Hotelling T-square que consisteix en la distància de la mostra al centre

del subespai, dins el subespai i els Q-residus que són la distància de la mostra perpendicular al subespai de components principals i que es poden calcular amb l'equació (6.2)

$$Q = EE^T \tag{6.2}$$

Fent servir els Q-residus es pot definir un interval de confiança, que ens indica quina és la probabilitat que una mostra no pertanyi al model de components principals.

6.1.6.2 PARAFAC

PARAFAC és un model multilinial que serveix per descriure *N-way* data, com per exemple diverses mesures de diversos subjectes al llarg d'un interval de temps, cosa molt comú en psicologia.
En un model PARAFAC, el cub inicial de dades es decomposa en una suma de matrius, tal i com s'explica a l'equació (6.3).

$$x_{i,j,k} = \sum_{r=1}^{r=R} a_{i,r} b_{j,r} c_{k,r} + e_{i,j,k} \tag{6.3}$$

on $x_{i,j,k}$ és el cub de dades originals, $a_{i,r}$, $b_{j,r}$ i $c_{j,r}$ són els elements de les matrius de *loadings* A,B,C i $e_{i,j,k}$ és l'error del model. Es pot veure com una extensió del model bilinial de PCA, tot i que hi ha algunes diferències com que no s'imposa ortogonalitat i que els models de PARAFAC no es poden rotar sense canviar el model.
Els principal problema de PARAFAC és que els algoritmes poden convergir cap a una solució no òptima, trobant un mínim local. Tot i això, quan les dades son trilinials, PARAFAC ens pot proporcionar un model més fàcil d'interpretar que PCA.

6.1.7 Objectiu

Determinar a quins llocs de l'ADN es troben els punts d'unió dels factors de transcripció és clau per entendre diferents processos, com la diferenciació cel•lular o la regulació depenent del tipus de cèl•lula.
Els mètodes de detecció de punts d'unió es poden dividir entre aquells que no tenen en compte les dependències entre posicions, i aquells que les tenen en compte, que normalment tenen un cost computacional molt elevat.
L'objectiu general d'aquesta tesis és la construcció d'un detector de punts d'unió capaç

d'identificar-los enmig de seqüències genòmiques. Aquest detector farà servir tècniques d'estadística multivariant per modelar les seqüències i també la covariança, una estadística de segon ordre, per modelar les dependències entre posicions. Els objectius més específics són:

1. Caracterització dels punts d'unió i la seva relació amb els gens regulats.
2. Construcció d'un detector de Q-residus. Convertint l'ADN en una seqüència numèrica, es pot fer servir un anàlisis de components principals per tal de modelar el punt d'unió i els Q-residus de les seqüències per crear un detector.
3. Construcció d'un detector mitjançant un Quadratic discriminant analysis (QDA). En aquest cas les seqüències del punt d'unió es converteixen en un cub numèric i es modelen fent servir PARAFAC. Els Q-residus i el model es poden combinar per donar lloc a un detector quadratic, QDA.

6.2 Caracterització dels punts d'unió

Els factors de transcripció són un grup molt heterogeni de proteïnes, que es classifiquen en diferents famílies depenent del domini d'unió a l'ADN. Les seves funcions també són molt variades, des d'aquells factors de transcripció necessàris per a l'expressió de la majoria de gens, fins aquells que només s'activen sota alguns estímuls i en alguns teixits específics.
És interessant, doncs, intentar fer una caracterització dels factors de transcripció mirant primer el nombre de proteïnes regulades per cada factor i el nombre de factors que regulen cada proteïna i també fent un estudi de les interdependències que es poden trobar entre els diferents punts d'unió.

6.2.1 Relació entre gens i factors de transcripció

Per tal d'estudiar la relació entre factors de transcripció i gens, ens vam baixar els gens de *Homo sapiens* de la base de dades NCBI, i vam fer servir dues bases de dades STRING, que indica relacions funcionals entre proteïnes, i Sabiosciences, una base de dades privada, per a trobar factors de transcripció que afectessin els diferents gens. Per a fer l'estudi es va construir el paquet en R StringSabio.
Com era d'esperar, el número de gens regulats per a cada factor de transcripció varia

molt, però té un màxim al voltant de 5. De forma similar, en un estudi del nombre de TF que regulen cada gen, podem veure que el màxim és entre 5 i 10.

6.2.2 Estudi de les interdependències

Per tal d'estudiar les interdependències entre posicions, vam fer servir la base de dades de JASPAR (2010), la JASPAR core, amb informació sobre tots els punts d'unió i també aquella part de la base de dades que té informació sobre les famílies. Per a calcular les interdependències vam fer servir el factor de Bayes (Bayes Factor), que dóna la probabilitat de la hipòtesis nul •la quan la probabilitat a priori és 0.5. En el nostre cas es pot calcular com una constant multiplicada per a la informació mútua tal i com es veu en l'equació (6.4)

$$log_2(BF(H_0, H_1)) = -MM_{i,j} \tag{6.4}$$

on $M_{i,j}$ és la informació entre les posicions i i j, M el número de seqüències del punt d'unió i BF el factor de Bayes. Un llindar de $BF < 0.1$ es va fer servir per a considerar les interdependències significants. La proporció de posicions amb interdependències en un motiu es va anomenar Complexitat del motiu o $Comp$ i és el valor que es va fer servir per a anàlisis posteriors.
Els resultats generals indiquen que la majoria de punts d'unió tenen interdependències,tot i que en una proporció no molt alta, entre el 0.2 i el 0.3 de posicions essent el màxim valor $Comp = 0.37$ per a $PPARG\gamma$, un factor de transcripció amb dos hexòmers molt similars, separats per dues bases.
Un estudi de les interdependències per famílies indica que un valor simple, com la complexitat, no pot classificar els punts d'unió, ja que, a part d'unir-se a l'ADN hi ha molts altres factors involucrats, com molècules petites, o altres punts d'unió propers.

6.3 Detector mitjançant els Q-residus

En aquesta secció es descriu primer la conversió a ADN numèric, i després la construcció del model fent servir la covariança de les seqüencies numèriques aliniades i la detecció dels punts d'unió.

6.3.1 Dades

6.3.1.1 Bases de dades de punts d'unió

Les seqüencies dels punts d'unió es van extreure de les bases de dades TRANSFAC public (2005) i JASPAR (versió del 2010. De la base de dades de TRANSFAC vam triar totes aquells motius que contenien més de 10 seqüencies, i les vam aliniar fent servir CLUSTALW i una validació de leave-one-out. Aquells motius sense més de 5 posicions consecutives aliniades en totes les seqüències no van ser considerats.
Per extreure motius de la base de dades JASPAR vam considerar 4 organismes, i vam triar aquells motius que tenien més de 10 seqüencies. En aquest cas, com que les seqüencies ja estan aliniades a la base de dades no vam necessitar cap procediment extra. En total vam fer servir 89 motius de JASPAR i 23 de TRANSFAC.

6.3.1.2 ADN de les seqüencies promotores

Les seqüències promotores fetes servir per la detecció dels punts d'unió provenen de la Eukaryotic promoter database (EPD) excepte pel *Saccharomyces Cerevisiae* on vam utilitzar seqüències extretes del genoma de l'organisme. Per a cada organisme es van fer servir dues regions promotores (des de $-0.5Kb$ fins a $1Kb$) escollides a l'atzar. Per tal de calcular la probabilitat de cada nucleòtid en les regions promotores dels diferents organismes, es van fer servir totes les dades obtingudes de les diferents regions promotores.

6.3.2 Model del subespai

El primer pas per poder calcular el model és convertir les seqüències aliniades dels punts d'unió en matrius de seqüències numèriques. Els punts d'unió provinents de TRANSFAC cal aliniar-los, es va fer fent servir l'algoritme CLUSTALW, mentres que els punts d'unió de JASPAR ja vénen aliniats.
La transformació feta servir, és la que posa cada nucleòtid al vèrtex d'un tetraèdre regular, ja que és simètrica per a tots els nucleòtids. Els vectors numèrics corresponents a cada nucleòtid o posició es van concatenar i les diferents seqüències aliniades es van posar una sota l'altra donant lloc a una matriu numèrica de dimensions $M \times 3N$ on M és el número de seqüències i N el número de posicions de cada seqüència. El model de components principals es va construir aplicant l'equació (6.1). En aquest cas els

scores A representen la matriu del DNA projectat al nou espai, els loadings B són el nou subespai que captura la màxima covariança i E és l'error. Per tal d'interpretar aquest model hauríem de mirar la $3M \times 3M$ matriu de covariança, que és diagonal si les posicions no estan correlades i té elements fora de la diagonal indicant correlacions (cal recordar que la matriu s'ha de dividir en submatrius de 3×3 ja que cada nucleòtid correspon a un vector de 3 components). En el model PCA la informació de les correlacions es troba en els loadings, però degut a la compressió de les dades a una matriu enlloc d'un cub són difícils d'interpretar, tot i que es poden observar diferències entre els loadings de posicions més conservades i de posicions variables.

6.3.3 Construint el detector

Fent servir la estadística dels Q-residus del nostre model de PCA podem construir un detector de punts d'unió. La hipòtesis que fem servir és que quan una seqüència candidata és projectada al subespai de components principals, tindrà uns Q-residus menors si és un punt d'unió (s'ajusta al model) que si és una seqüència genòmica que no s'assembla a les seqüències dels putns d'unió i per tant no es pot modelar mitjançant el nostre PCA. Un exemple es pot veure a la figura 6.4 on els Q-residus per als punts d'unió del factor de transcripció fent servir un model PCA de 3 components es mostren juntament amb els Q-residus de 1000 seqüències corresponents a una regió promotora. A la figura es pot observar que definint un llindar es pot crear un detector de punts d'unió.

6.3.4 Comparació amb altres algoritmes

6.3.4.1 Algoritmes de matrius de pesos o PSSM

Per comparar el detector de Q-residus amb altres detectors es va construir el paquet de R MEET, que es pot trobar a R-forge i al CRAN i que permet combinar diferents mètodes d'alíniament amb diferents algoritmes de detecció. Dos detectors basats en PSSM es poden utilitzar des de MEET, el MAST que forma part del conjunt de programes de MEME i una implementació del MATCH que utilitza la un score per a la informació per posició de la matriu i un altre per a les 5 posicions consecutives més conservades.

6. RESUM EN CATALÀ: DETECCIÓ DE PUNTS D'UNIÓ DE FACTORS DE TRANSCRIPCIÓ MITJANÇANT TÈCNIQUES DE PROCESSAMENT DE SENYAL

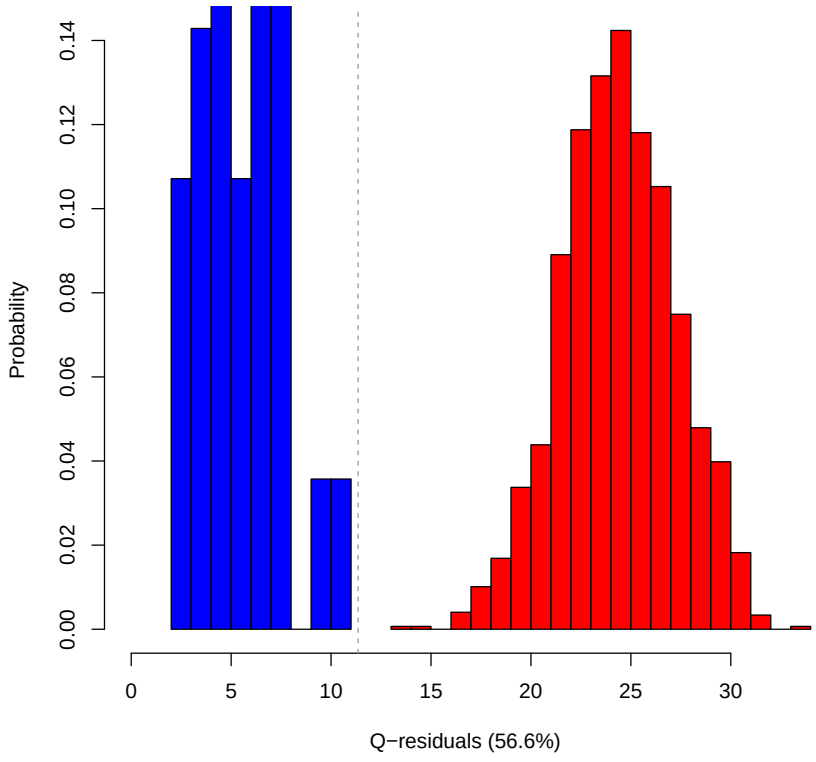


Figure 6.4. Exemple del càlcul dels Q-residus per a punts d'unió en blau i 1000 seqüències promotores en vermell fent servir el model PCA de 3 components dels punts d'unió del factor de transcripció PPARG. Definint un llindar, els punts d'unió es poden separar fàcilment de les seqüències promotores.

Per evaluar els detectors es van fer servir les corbes ROC que mostren la proporció de verdaders positius contra la proporció de falsos i l'àrea sota la corba (auc). El paquet MEET, per tal d'evaluar si les diferències entre detectors són significants, fa servir una doble validació. Primer una seqüència A es treu de la matriu i s'inserta dins el background, després amb les resta de $N-1$ seqüències es fa una validació de leave-one-out (deixa un fora), i es construeixen $N-2$ models que es fan servir per detectar la seqüència A. Després la seqüència A s'inserta altra vegada a la matriu i es fa el mateix amb una segona seqüència B, i així amb les N seqüències del punt d'unió. Així es poden construir N ROC curves, i N auc, de forma que es pot estimar no només el valor mitjà de l'auc sinó també quina és la seva variança. Les corbes ROC i la seva AUC es van calcular per un rang de 1 a 10 components principals, i també per diferents valors dels paràmetres de MATCH (MAST no té paràmetres a optimitzar). El paràmetre òptim es va calcular fent servir l'auc.Per tal de quantificar les diferències es va calcular un Wilcoxon-rank test, que indica la diferència entre dues distribucions. L'algoritme de Q-residus obté millores significants ($p-valor <= 0.05$) en 57 dels 112 punts d'unió estudiats si el compares amb MATCH i en 63 si el compares amb MAST. Una altra característica és que el detector de Q-residus és també més robust.
Per tal de comprobar que el nostre detector pot capturar les interdependències entre els diferents nucleòtids vam calcular la correlació entre la millora en l'auc i el número de interdependències entre posicions, trobant una correlació significativa amb p-valor 0.004 en JASPAR i p-valor 0.4 en TRANSFAC.
Finalment vam mesurar els temps computacionals necessaris per a la detecció de factors de transcripció en seqüències promotores. Per a fer aquesta comparació vam instal•lar l'algoritme MAST a l'ordinador i vam fer servir el codi en C pel detector de Q-residus i la nostra implementació de MATCH en R. També vam ajustar els paràmetres per tal que el número de seqüències detectades fos similar, i vam fer 100 iteracions per a la detecció de cada motiu en un background de 1500 bases. Els temps computacionals per a TRANSFAC són $0.003 \pm 0.001s$ en el detector de Q-residus, $0.0191 \pm 0.001s$ en MAST i $0.033 \pm 0.003s$ en la implementació de MATCH.

6.3.4.2 Algoritmes amb interdependències

Per a fer la comparació amb algoritmes que tenen en compte interdependències es va triar Motifscan, un algoritme que fa servir grafs per a modelar els punts d'unió. En un

article del 2006, es va fer servir la base de dades de JASPAR (2006) per a comparar Motifscan amb algoritmes de PSSM. Per evaluar els detectors es van fer servir les corbes ROC_N (les corbes ROC quan només es tenen en compte els primers N falsos positius), i es va escollir N com el número de seqüències del punt d'unió. Finalment es va considerar que una millora significativa en la detecció era un augment del 5% en l'auc de la ROC_N

Fent servir la mateixa metodologia i 93 seqüències de JASPAR (2006), es pot veure que Motifscan millora el detector de Q-residus i els algoritmes PSSM en 34/93 motius, Q-residus és el millor en 25 i els algoritmes PSSM només en 1.

Un estudi més profund dels punts d'unió on un algoritme és millor que l'altre permet identificar que el detector de Q-residus necessita seqüències amb més posicions per tal de crear un model bo, mentres que Motifscan és més sensible al número de seqüències que hi ha per a cada punt d'unió.

6.4 Detectors de "three-way"

La conversió de les seqüencies d'ADN a matrius numèriques requereix la compressió de la informació. Una forma més natural de fer la conversió és utilitzant cubs on la primera dimensió es refereix al nombre de seqüències, la segona al nombre de posicions en cada seqüència i la tercera a la conversió numèrica de cada nucleòtid. Algunes tècniques de processament de senyal, com PARAFAC, es poden utilitzar en aquests cubs donant lloc a nou detectors que són més fàcilment interpretables.

6.4.1 bases de dades

Per a un estudi preliminar, sobre la utilitat i interpretació dels models PARAFAC per a la detecció de punts d'unió vam fer servir les seqüències de 5 punts d'unió de la base de dades JASPAR i l'organisme *Homo sapiens*. Vam agafar aquelles amb més interdependències, ja que és el que volem modelar. També vam repetir l'estudi amb les seqüències del factor de transcripció DL, que vam agafar com a exemple per a intentar explicar el model PCA.

Per a fer la comparació amb altres mètodes es van fer servir les seqüències de JASPAR (2006), utilitzades en la secció anterior.

6.4.2 Models PARAFAC

Per a convertir les seqüències d'ADN en seqüències numèriques, es va fer servir la mateixa conversió que en el model PCA. Aquest cop, però, les seqüències es van arranjar en un cun de $N \times M \times d$ on N és el número de seqüències, M el de posicions i d és la dimensionalitat de la conversió (3 en el nostre cas).

Els models PARAFAC es van construir fent servir l'equació 6.3, on $x_{i,j,k}$ són els elements del cub d'ADN. Si els models PARAFAC són interpretables, els elements $a_{i,r}$ tindran informació sobre les diferents seqüències d'un punt d'unió i els elements $b_{j,r}$ sobre les diferents posicions. Per aquells punts d'unió triats per a l'estudi d'interpretació de model es van seguir tres passos: (1) construir models per diferents número de components, (2) Estudiar l'estabilitat de les solucions i (3) estudiar la interpretabilitat dels models.

Un criteri per tal de saber quants components són masses per a un model PARAFAC és mirar quina és la variança explicada per cada component i quina és la variança explicada només per aquell component. Un cop la variança explicada arriba a un cert llindar, afegir components només fa que crear components que són combinacions linials d'altres, i per tant no afegeixen valor al model final. Això es pot veure quan la variança explicada per un component és alta però al treure aquell component la variança explicada pel model no varia. Seguint aquest criteri vam triar un número màxim de components (entre 1 i 5), i vam estudiar l'estabilitat dels models amb menys components. L'estabilitat es pot calcular simplement calculant el model diverses vegades i comparant l'error, si el model es troba en mínims locals, l'error variarà. Un cop obtinguts els models estables, vam decidir estudiar la informació biològica que els models PARAFAC poden tenir. El primer pas, va ser intentar recuperar la seqüència consensus del motiu, i la distància de cada seqüència a aquesta consensus. Per a fer-ho primer vam projectar cadascun dels nucleòtids a la nostra matriu de posicions (mode 2 del model PARAFAC, o matriu B), i vam calcular la distància de cadascuna de les posicions als 4 nucleòtids (per exemple, si en una posició tots els nucleòtids són A, la distància de la posició a la projecció de A, hauria de ser 0). La distància mínima de la posició a un nucleòtid, es considera el nucleòtid per a la seqüència consensus, i calculant les diferents distàncies es pot recuperar el Logo de la seqüència.

El primer mode, o la matriu A, del model PARAFAC conté informació sobre les

seqüències i la seva similitud amb la seqüència consensus. Per a estudiar-ho vam projectar la seqüència consensus i vam mirar on es trobava de l'espai i on eren les altres seqüències. Si un motiu té una seqüència consensus ben definida, llavors aquesta té un valor extrem i la distància de cada seqüència a la consensus es pot entendre com a diferències en els nucleòtids (sobretot els més conservats). Altres motius, com el del factor DL, no tenen una consensus definida, i al projectar la consensus al model, aquesta no té un valor extrem sinó mig, i les seqüències dels punts d'unió simplement s'agrupen per similaritat.

6.4.3 Detectors fent servir PARAFAC

6.4.3.1 Detector de Q-residus

El detector basat ens els Q-residus creat pels models PCA es pot fàcilment generalitzar per a un model de PARAFAC. Al comparar els dos detectors fent servir JASPAR (2006), es pot veure que tenen resultats similars, sent el Q-residus PARAFAC millor en 35 i el Q-residus PCA millor en 34. les diferències en aquest cas no es poden relacionar ni amb el número de posicions ni amb el número de seqüències disponibles. Quan es compara amb Motifscan, aquest continua essent encara millor en 32 dels punts d'unió (comparat amb els 34 d'abans).

6.4.3.2 Detector quadràtic QDA

Ja que els scores corresponents a una seqüència poden representar propietats d'aquesta, un detector que combini la matriu A de model PARAFAC i els residus pot donar millors resultats que un detector que simplement faci servir els residus. Per tant, es va construir un detector de *Quadratic Discriminant analysis*, que fa servir una combinació linial de característiques per a classificar entre diferents classes (en aquest cas dues) creant una superfífice de separació entre les classes quadràtica.
Per a entrenar el detector es feien servir a cada pas $N-1$ seqüències del motiu i 1000 seqüències aleatòries. Després aquest es feia servir per detectar la seqüència que faltava, i així construir les corbes ROC_N.
La comparació entre aquest detector i els altres mostra que detecta almenys tan bé com

els detectors de Q-residus (PCA i PARAFAC) en la majoria dels motius. Afegint tots els altres detectors a la comparació podem veure que, si bé QDA és el millor detector en 11 dels motius, motifscan encara ho és en 28. Es pot concloure, doncs, que el número de motius pels quals motifscan és millor no varia massa encara que anem afegint detectors numèrics millorats. Això és perquè els detectors numèrics só més sensibles al número de posicions dels punts d'unió dels factors de transcripció, mentres que motifscan és més sensible al número de seqüències disponibles per a crear el model.

6.5 Conclusions

- Alguns factors de transcripció participen en la transcripció d'un gran nombre de gens, mentres que altres responen a senyals específiques. La majoria de factors de transcripció regulen uns 5 gens, i els gens estan regulats per un nombre proper a 10 factors de transcripció.

- Les interdependències entre posicions es poden calcular mitjançant el Bayes factor. La majoria de factors de transcripció tenen interdependències, però aquest simple número no permet fer una classificació entre famílies.

- Convertir l'ADN en seqüències numèriques permet aplicar tècniques de processament de senyal a l'estudi dels punts d'unió dels factors de transcripció.

- Els Q-residus d'un model PCA de matrius numèriques representant els punts d'unió es poden fer servir per a distingir-los de seqüències promotores. Si no hi ha interdependències aquest model funciona tan bé com els models PSSM, però si n'hi ha, els millora.

- Si es compara amb un mètode que té en compte interdependències, el detector de Q-residus el pot millorar quan el número de seqüències disponible és petit, però és molt sensible al número de posicions.

- Convertir les seqüencies dels punts d'unió en un cub, permet l'aplicació de models PARAFAC, que poden contenir informació sobre les seqüències com la distància a la consensus, o el Logo.

- De forma anàloga als Q-residus es pot construir un detector amb PARAFAC. Els scores també es poden utilitzar i així construir un detector quadràtic que millora els detectors de Q-residus.
- Quan tots els detectors es comparen junts, es pot veure que normalment els detectors numèrics són menys sensibles al número de seqüències però més al número de posicions.

Appendix A

MEET

MEET 5.1 is a modular R package that integrates a set of tools for the detection of cys-regulatory sequences. Besides from allowing the user to create a new motif model to look for binding sites with the available tools, MEET also incorporates a library of models for 181 TFBS which can be directly used to find TFBS.

A.1 Motivation and Background

Most of the computational methods to detect transcripiton factors binding sites have been benchmarked employing different datasets and resulting different models of input and output parameters. This fact makes difficult a systematic comparison between different detection algorithms.

Some studies address the question of which motif discovery algorithm is better optimized (D'haeseleer, 2006; Osada et al., 2004). In these studies the parameters used for the comparison of the different algorithms have to be manually chosen (Tompa et al., 2005) or to be restricted to a few ones, even if there is a large dependence of the performance of the algorithms on the input parameters (Hu et al., 2005).

Most of the developed algorithms have only an on-line version of the algorithm, e.g. VOMBAT (Grau et al., 2006), and sometimes also a package to download the algorithms as the MEME suite (Bailey et al., 2009). Some other web tools allow to choose between different algorithms, but they have the inconvenience that cannot be automated. The comparison between algorithms cannot be done systematically, some

examples are SCOPE (Carlson et al., 2007) or CREDO (Hindemitt and Mayer, 2005). The first problem can be addressed using packages such as BEST (Che et al., 2005) or RSAT (Thomas-Chollier et al., 2011) which integrate a wide collection of tools to analyze DNA sequences looking for binding sites, but the problem with the systematic comparison remains. Focusing on R packages, Rtfbs in the CRAN repository allows to search for binding sites, but only a PSSM method is implemented.

MEET 5.1 is an R-package that includes a TF models library and a set of tools for motif search and discovery algorithms. The different models were built using multiple sequence alignments of binding sites compiled from the JASPAR (Bryne et al., 2008) database. MEET 5.1 can be used to optimize the parameters of the included detectors, allowing a systematic comparison between the different algorithms. Once the parameters of the detector are chosen, MEET 5.1 can be also used to detect possible binding sites within large DNA sequences. MEET allows not only to compare detectors, but also returns the best model for each motif giving the possibility to directly run the detection without worrying about the parameters. MEET 5.1 also incorporates calculated models for 181 JASPAR (Bryne et al., 2008) motifs which can be directly used to detect these binding motifs within chromosomic sequences.

MEET 5.1 includes a set of developed algorithms, ITEME (Maynou et al., 2010a) and Q-residuals (Pairó et al., 2012). Both algorithms capture the information among binding site positions. ITEME uses non-linear models based on information theory to evaluate the information gain and Q-residuals constructs a subspace based on the covariance of the numerical DNA sequences. External algorithms can be used when they are installed in the computer, such as MEME (Bailey and Elkan, 1994), a motif discovery tool which uses expectation maximization, MAST (Bailey and Gribskov, 1998) that is part of the MEME suite and uses a Q-FAST algorithm for motif finding and MDscan (Liu et al., 2002) which is an algorithm that mixes the enumeration of combined words with the Bayesian inference. Finally, MEET 5.1-package also includes an *R* custom implementation of MATCH algorithm (Kel et al., 2003) which is a tool based on the information content per site.

Some external alignment algorithms that are also supported by MEET 5.1 software when they are installed in the computer. These algorithms are MUSCLE (Multiple Sequence Comparison by Log-Expectation) (Edgar, 2004), and ClustalW (Thompson et al., 1994). MEME can also be used as a motif discovery method with the aligned

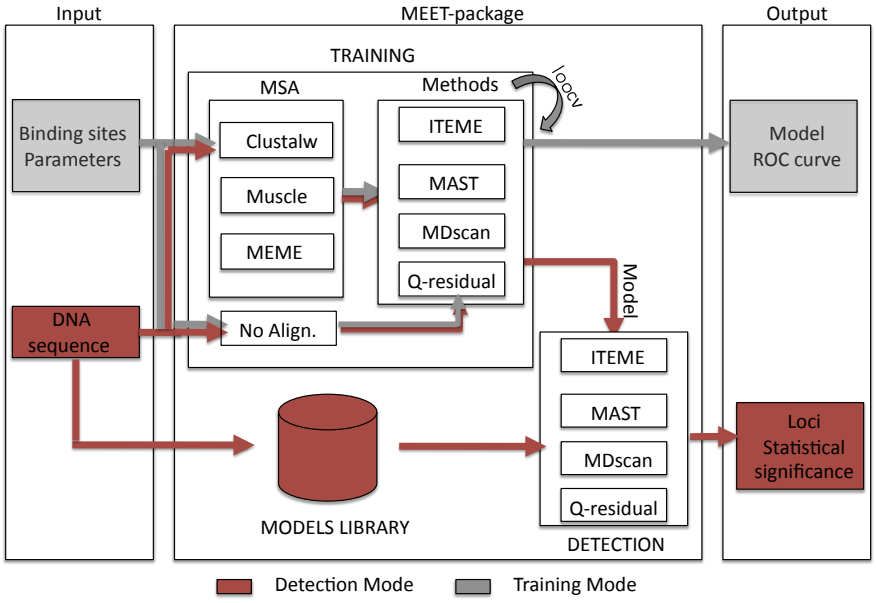


Figure A.1. Description of the MEET architecture including the internal and the external programs (in grey).

motifs as an output. In order to carry out the detection, the user can insert an aligned matrix as input parameter or to use the external alignment algorithms. Using MEET 5.1 R-package the alignment and detection algorithms can be combined in order to choose the combination that better satisfies the user needs.

A.2 Architecture of MEET

MEET has two main functionalities, training and detection. The training mode can be used to study the performance of different detectors, and to choose the best parameters of any of the included detectors, the output includes the chosen model and the chosen parameters. The detection mode can use the constructed model, some inputed parameters or any model in the library to detect binding sites. The Architecture of MEET can be seen in the next figure A.1

A.2.1 Training mode

The main purpose of the training mode is to output the best model for a given TF and a given algorithm. This mode uses a double l.o.o cross-validation to calculate the ROC curves and the error associated to them. The magnitude used to assess the performance of the algorithms is the area under the ROC curve (AUC). In order to find a model with the highest AUC but also to consider the stability of the detector MEET uses a heuristic formula to choose the best model (A.1)

$$C = \mu(AUC)(1 - \sigma(AUC)), \tag{A.1}$$

where μ is the mean and σ the variance of the AUC.
A tree diagram with the functions of the training mode is presented in the figure A.2. The function Construct model calls one of the algorithms to perform the double l.o.o. Then the ROC curve and the AUC are computed and these results are used to create the best model. The output is the best model, the AUC and the ROC curve corresponding to the best parameters.

An example to run the training mode is:

```
library(MEET)
pathMEET<-system.file("exdata", package=MEET)

TrainingResult <- MEET( TF=paste(pathMEET),"AP1.fa", sep="/")
                        seqin=paste(pathMEET),"DNAhomo.fa", sep="/")  ,
                        alg="NONE",
                        mode="training",
                        vector=c(1:8),
                        org="Homo sapiens",
                        method="Qresiduals")
```

The output is a list that can be divided in three parts: two generic parts which have the consensus sequence of the motif and the input parameters of the MEET function (organism, algorithm, etc.) and the third part that has the results. This results part is also a list which incorporates the chosen model, the AUC for the range of parameters studied and the ROC curve of the chosen model. The AUC and the ROC curve can be used to compare the performance of different detectors, and also to compare the AUC of the studied detector in the range of studied parameters. This allows the user to have

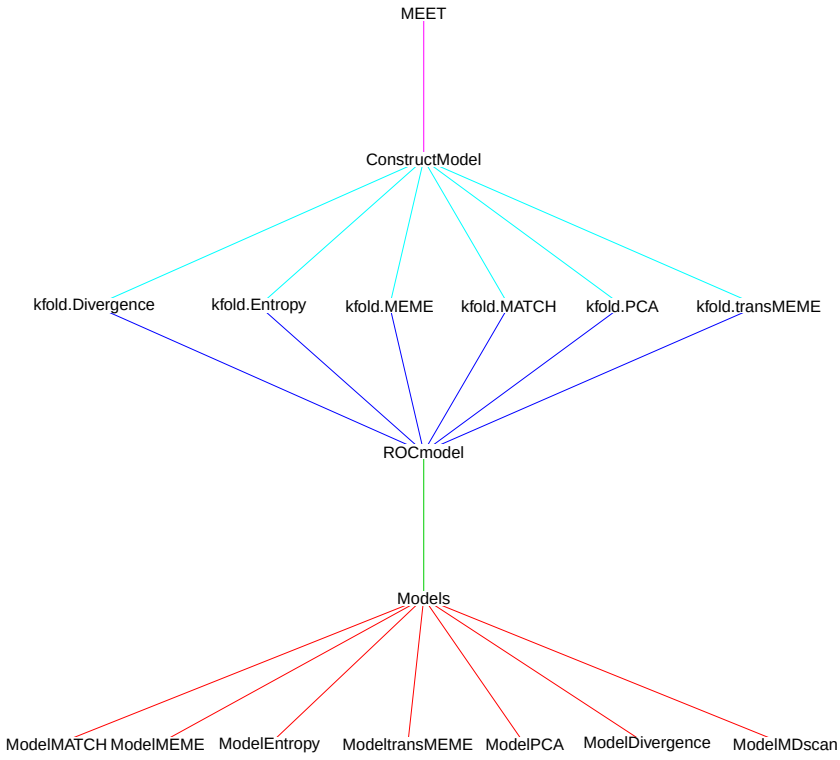


Figure A.2. Diagram of the training mode of the MEET R-package. The main function Construct model calls one of the k-fold functions, corresponding to the chosen algorithm. After the validation, the ROC curves and their AUC are computed, and with that the best model is chosen. The chosen model is constructed with a specific function for each algorithm.

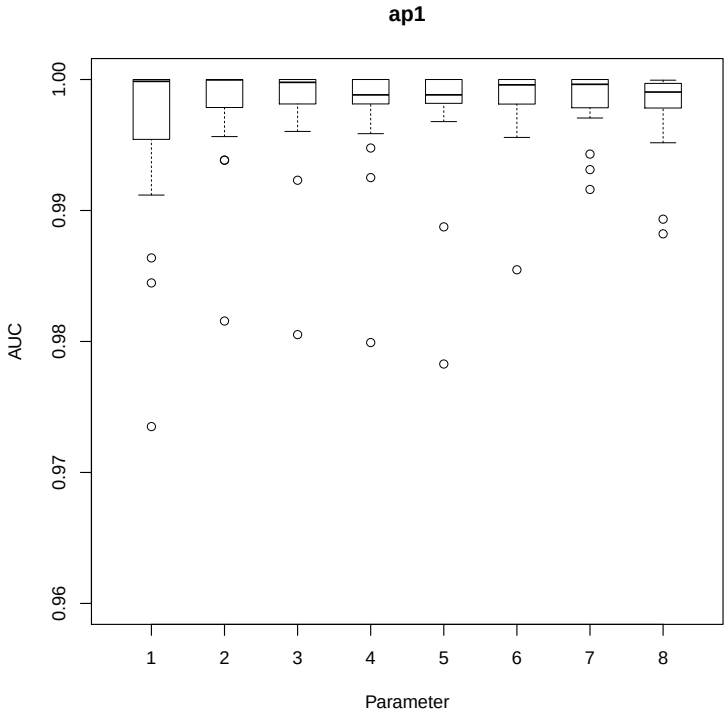


Figure A.3. Boxplot of the AUC of the AP1 binding sites and the Q-residuals detector changing the number of components from 1 to 8. The boxplot can be directly plotted from the MEET output.

another criteria to choose the optimal model and to build a custom motif detector. The chosen model can be easily recovered from the MEET results. If the user prefers to visualize how the performance of the detector changes as the main parameter is changed, a simple boxplot of the AUC can be helpful to visualize the mean and the variance of the AUC using each one of the parameters. In the example above, with the Q-residuals detector and the AP1 motif from *Homo sapiens*, the following text will recover the model and plot the AUC for the number of principal components going from 1 to 10 as it can be seen in the figure A.3.

```
FinalModel <- TrainingResult$Results$Model
boxplot(TrainingResult$Results$Area, xlab="Parameter", ylab="AUC", outline=TRUE)
```

A.2.2 Detection mode

The detection mode of the MEET R-package can be used to look for binding sites within genomic sequences. The input can be (1)one of the models included in the library (2) one model constructed using the training mode (3) the parameters needed to construct one model. As in the case of the training mode, the generic function Detection calls a specific function for one of the algorithms. It can be directly a prediction function which looks for binding sites or, in case the inputed values are the parameters, first a model function. When the prediction function has looked for binding sites within the inputed problem sequence, the ouput given is: the sequences of binding sites found, its p-value and its position within the larger sequence. The summary of this architecture can be seen in the figure A.4. If the searched binding sites belong to the models included in the MEET library the found sequences can also be visualized with a generated HTML file, using the function writeResultsHTML. In the next example the FinalModel obtained with the training method and the Qresiduals algorithm shown above is used for the detection of the AP1 binding sites in a *Homo sapiens* promoter. As the output of the training mode is directly used as a model for the detection mode there is no need to include the parameters of the algorithm. In the example, *seqin* is a DNA sequence with unknown binding sites, *mode* is detection, *model* refers to the built model usin the trainig mode in the example above, *threshold* is the desired p-value threshold and *method* is the used algorithm, in this case Q-residuals

```
library(MEET)
pathMEET<-system.file("exdata", package=MEET)
Detection <- MEET(seqin=paste(pathMEET, "DNAmeet.fa", sep="/"),
                  mode="detection",
                  model=FinalModel,
                  threshold=0.01,
                  method="Qresiduals")
```

To make use of one of the models included in the library, instead of the model, the parameter needed is *nameTF*. The other input parameters should be the same. The example of the R code to run the detection mode using the a1 *Drosophila melanogaster* model built using the Divergence algorithm and a p-value threshold $threshold = 0.001$ is as follows:

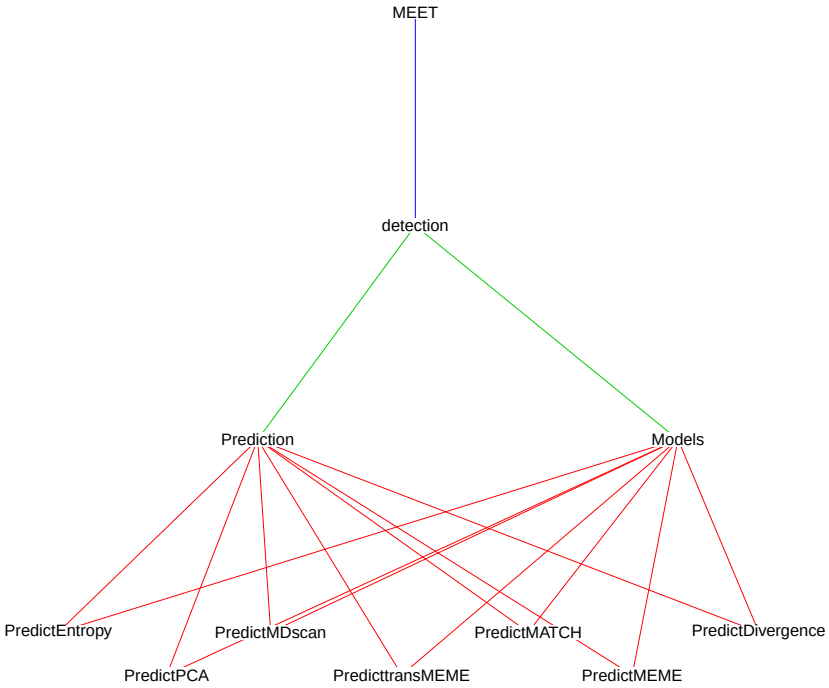


Figure A.4. Tree dependencies of the detection mode of the MEET R-package. Using this mode, the input can be a calculated model or the parameters to calculate a new model. If the input are the parameters, first a model with the chosen parameters is constructed and then the model is used to run the prediction function specific for each algorithm. If the input is a model, then the prediction function is run directly

```
library(MEET)
pathMEET<-system.file("exdata", package=MEET)
Detection <- MEET(nameTF="a1",
                  seqin=paste(pathMEET, "DNAmeet.fa", sep="/"),
                  system="detection",
                  threshold=0.01,
  organism="Drosophila melanogaster",
                  method="Entropy")
```

The output of the detection mode is also a three items list. The first two items, summary and consensus, coincide with the output of the training mode. The third item, the Results, is different. As is is said above, in the detection mode, the Results item of the detection mode consist on a list of found binding sites with its position and its p-value.

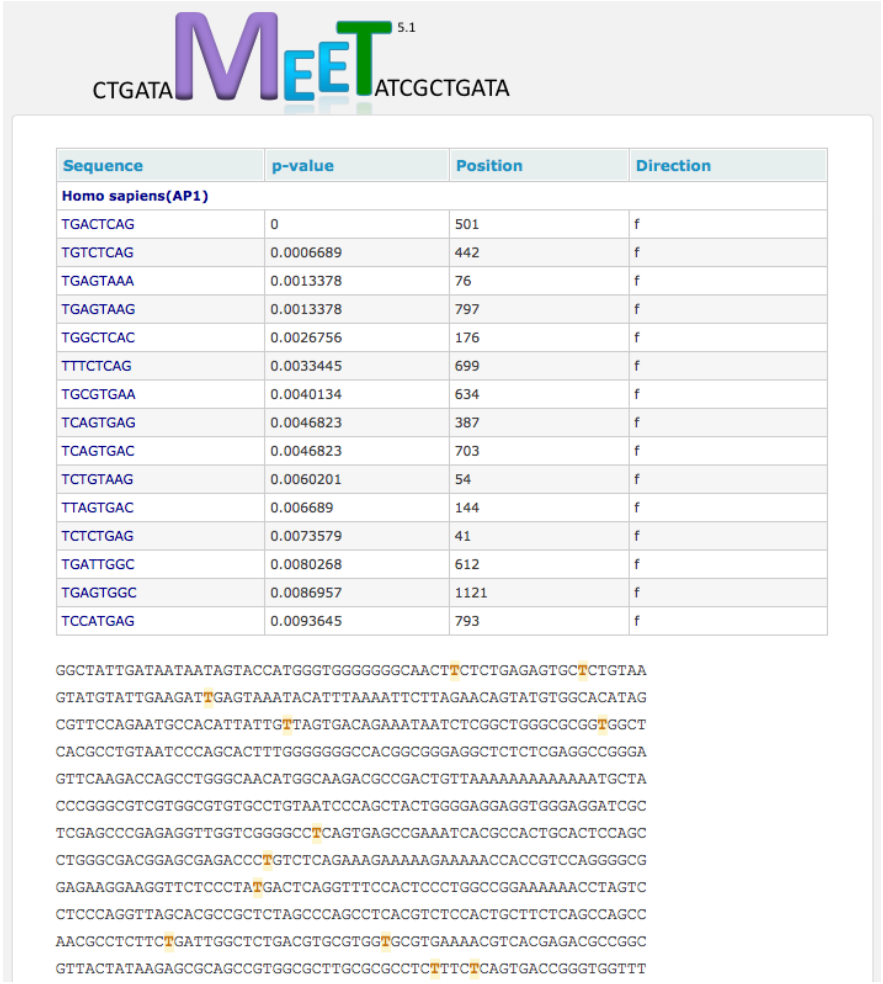
```
##   Position Value       Direction Sequence
## 1 "66"     "0"         "f"       "TATTGAAG"
## 2 "279"    "0.0006689" "f"       "TGTTAAAA"
```

The MEET 5.1 R-package includes a function that allows to show the detection results in HTML format when the library of models is used in the detection mode. As it can be seen in the next example the function arguments are the output obtained from running the detection mode and, optionally, the name of the HTML file that will be generated – index.html is the default name.

```
writeResultsHTML(Detection$Results)
```

The output is an HTML file – index.html in this example case – that will be stored in the R working directory. This HTML file can be seen in a browser and its content is similar to the one shown in Figure A.5. Basically it consists in a table with the found binding

The web service of the detection mode is publicly available through `http://sisbio.recerca.upc.edu/webtools/MEET/`. This platform is mainly based on the Python platform and is developed using a web framework named *web.py* (*http://webpy.org/*). In order to access R from Python in a simple and robust way it is used the RPy^2



Sequence	p-value	Position	Direction
Homo sapiens(AP1)			
TGACTCAG	0	501	f
TGTCTCAG	0.0006689	442	f
TGAGTAAA	0.0013378	76	f
TGAGTAAG	0.0013378	797	f
TGGCTCAC	0.0026756	176	f
TTTCTCAG	0.0033445	699	f
TGCGTGAA	0.0040134	634	f
TCAGTGAG	0.0046823	387	f
TCAGTGAC	0.0046823	703	f
TCTGTAAG	0.0060201	54	f
TTAGTGAC	0.006689	144	f
TCTCTGAG	0.0073579	41	f
TGATTGGC	0.0080268	612	f
TGAGTGGC	0.0086957	1121	f
TCCATGAG	0.0093645	793	f

GGCTATTGATAATAATAGTACCATGGGTGGGGGGGCAACTTCTCTGAGAGTGCTCTGTAA
GTATGTATTGAAGATTGAGTAAATACATTTAAAATTCTTAGAACAGTATGTGGCACATAG
CGTTCCAGAATGCCACATTATTGTTAGTGACAGAAATAATCTCGGCTGGGCGCGGTGGCT
CACGCCTGTAATCCCAGCACTTTGGGGGGGCCACGGCGGGAGGCTCTCTCGAGGCCGGGA
GTTCAAGACCAGCCTGGGCAACATGGCAAGACGCCGACTGTTAAAAAAAAAAAAAATGCTA
CCCGGGCGTCGTGGCGTGTGCCTGTAATCCCAGCTACTGGGGAGGAGGTGGGAGGATCGC
TCGAGCCCGAGAGGTTGGTCGGGGCCTCAGTGAGCCGAAATCACGCCACTGCACTCCAGC
CTGGGCGACGGAGCGAGACCCTGTCTCAGAAAGAAAAAGAAAAACCACCGTCCAGGGGCG
GAGAAGGAAGGTTCTCCCTATGACTCAGGTTTCCACTCCCTGGCCGGAAAAAACCTAGTC
CTCCCAGGTTAGCACGCCGCTCTAGCCCAGCCTCACGTCTCCACTGCTTCTCAGCCAGCC
AACGCCTCTTCTGATTGGCTCTGACGTGCGTGGTGCGTGAAAACGTCACGAGACGCCGGC
GTTACTATAAGAGCGCAGCCGTGGCGCTTGCGCGCCTCTTTCTCAGTGACCGGGTGGTTT

Figure A.5. Output of the Detection mode using the HTML file

Table A.1. Summary of the models included for each organism and method to the models library of the MEET R-package

Organism	Entropy	Divergence	Qresiduals	TOTAL
Drosophila melanogaster	92	92	102	286
Homo sapiens	43	43	43	129
Rattus norvegicus	11	11	11	33
Mus musculus	25	25	25	75
TOTAL	171	171	181	523

package. The web pages are created in *HyperText Markup Language* (*HTML*) and, to make the user interface dynamic and user friendly, it is used *JavaScript*, *Asynchronous JavaScript And XML* (*AJAX*) and *JQuery* (*http://jquery.com/*), is employed to make the result similar to a dynamic online application rather than a static Web site. The Figure A.6 shows the configuration step where the user needs to upload or paste a DNA sequence in FASTA format, select one or more models provided by the application (*Transcription Factors*), select the detection algorithm (*Method*) and select the p-value used as the threshold in detection (*Threshold*). The models provided by the application are grouped by organism and each organism contains a set of TF that can be selected.

A.2.3 Library of TF models

The MEET R-package includes a library of 523 models from 181 motifs extracted from the JASPAR (2010) database. This library consists on the Q-residuals, the Divergence and the Entropy models of the TFBS that have more than 10 available sequences in the JASPAR (2010) database and correspond to the organisms: *Drosophila melanogaster*, *Rattus norvegicus*, *Mus musculus* and *Homo sapiens*.

In order to construct the models, the training mode of MEET has been used. The model chosen by MEET for each motif, according to equation (A.1) has been included in the library. A relation with the number of models for each organism and algorithm can be seen in table A.1

Figure A.6. Initial view of the web of the MEET R-package. The user can choose several motifs for each organism, paste or upload a sequence in .fasta format and then then the package will look for binding sites within the sequence.

A.3 Implementation of MEET

A.3.1 Alignment algorithms

Two alignment algorithms can be used when they are installed in the computer, MUSCLE version $\leq$ 3.8, CLUSTALW version $\leq$ 2.1. The parameters, as gap penalty or gap extension, can be modified directly from the MEET 5.1 input options.

Muscle is an iterative alignment tool. It first aligns two sequences and then the other sequences are added progressively while it realigns the pair of sequences established at the beginning. On the other hand, ClustalW is based on a progressive model, when sequences are added sequentially, the first pair of sequences is not aligned again.

The MEME version 4.4.0 can also be used to construct a motif model from unaligned sequences. MEME is based on expectation-maximization algorithm. The number of motifs and the width of the motifs can also be controlled from MEET 5.1 input parameters.

A.3.2 Detection algorithms

A.3.2.1 ITEME and Q-residuals

The package includes three algorithms, ITEME (Entropy and Divergence) (Maynou et al., 2010a) and Q-residuals. ITEME calculates the information of an aligned set of binding sites, and then the variation of this information when a candidate sequence is added to the model. The assumption made is that, when the new sequence is a binding site, the information gain will be near zero, because the sequence will be similar to the previous ones, but when the sequence is not a binding site the information added will be larger. To calculate the information, two approaches can be taken: to consider that the position within the binding sites are independent, as in equation (A.2) where the Rényi entropy is calculated (Renyi, 1961), or to take into account position interdependences using the divergence (Kullback and Leibler, 1951) as it is described in equation (A.3).

$$H_q = \frac{1}{1-q} log_2 \sum_{i=1}^{N} p_i^q(x) \tag{A.2}$$

$$D_q(X;Y) = \frac{1}{q-1} log_2 \sum_{i=1}^{N} \sum_{j=1}^{N} P(x,y)_{i,j}^q Q_{i,j}^{1-q}, \tag{A.3}$$

where H_q and D_q are the entropy and divergence, respectively, q is a positive number different from 1, N is the number of nucleotides, x and y are two positions in the binding site, $P(x, y)$ and $Q(x, y)$ are probability distribution and $p(x)_i$ the probability of having the nucleotide $i \in \{A, C, G, T\}$ in the position x. Specifically, $P(x, y) = p(x, y)_{i,j}$ is the joint probability of having a nucleotide i in the position x and another nucleotide j in the position y and $Q(x, y) = p(x)_i \cdot p(y)_j$. The Rnyi entropy and divergence are nonnegative measurements for all $q \geq 0$. When q tends to 1, the Rnyi entropy converges to Shannon entropy (Shannon, 1948) and Rényi divergence converges to Kullback-Leibler divergence (Kullback and Leibler, 1951).

The Q-residuals (Pairó et al., 2012) detector is explained in chapter 3.

A.3.2.2 External algorithms

The package allows the use of MATCH, MDscan and MEME/MAST (Bailey and Elkan, 1994; Bailey and Gribskov, 1998) if these programs are detected as available on the installation system. The package also includes a custom implementation of the MATCH algorithm in R, also explained in the chapter 3.

MEME/MAST can be downloaded from the MEME suite (Bailey and Elkan, 2006) and MDscan from the MDscan web page (Liu et al., 2002). The current version of MEET 5.1 is prepared to work with MEME version 4.4.0. and MDscan (2004).

A.4 Examples

A.4.1 Alignment

A.4.1.1 Data

The detection results depend on several factors. One of this factors is the alignment quality. TRANSFAC 7.0 (2005) database (Wingender et al., 2000) and chromosome 12 from *Saccharomyces cerevesiae* have been used to test the influence of the alignment in the detection process.

A.4.1.2 Parameters effect

MEET 5.1 can also overcome the difficulty in choosing the best alignment parameters, making possible an automated comparison between the external alignment algorithms,

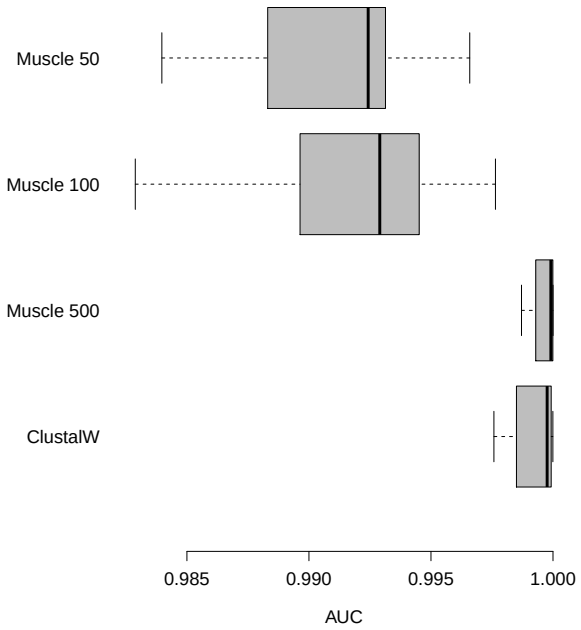


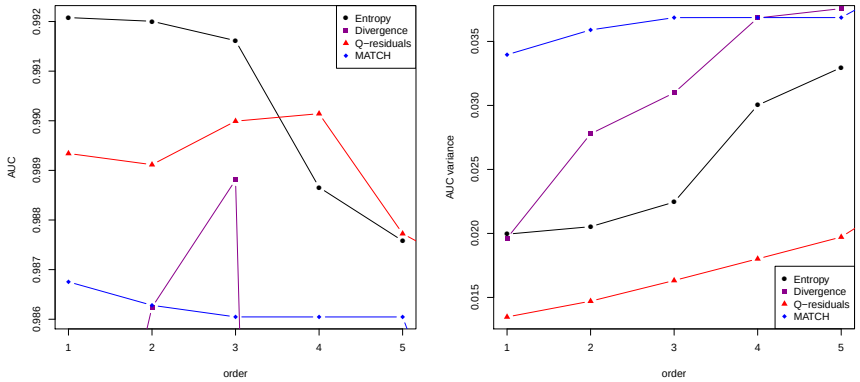
Figure A.7. Comparison of the detection using different alignment algorithms. The results are shown for the Q-residuals detector, and the ABF1 binding sites from *Saccharomyces cerevesiae*. The figure represents the AUC for different alignments, Clustalw with the default *gapopen* = 10, and Muscle with different values of the *gapopen*, 500, 100, 50. There is a decrease in the AUC as the value of the gapopen is decreased.

as it is show in Figure A.7. The figure shows the AUC of the ABF1 binding sites, calculated using the MEET 5.1 validation mode, for different alignments, the Q-residuals algorithm and the ABF1 motif. It can be observed that the quality of the detection changes depending on the alignment algorithms and the alignment parameters. Changing the *gapopen* parameter in the alignment using muscle from 500 to 50 produces a decrease of the AUC, while the comparison between the default parameter of ClustalW (*gapopen* = 10.0) and the muscle with *gapopen* = 500 shows that the two alignments produce similar results for ABF1 binding sites.

A.4.2 Comparison

A.4.2.1 Parameters effect

The use of MEET 5.1 allows to explore directly the parameters space, as it can be seen in figure A.8 where a range of parameters has been studied for the included detectors. The



(a) Mean **(b)** Variance

Figure A.8. The mean and the variance for the AUC are represented for MATCH, Q-residuals and ITEME, both divergence and Entropy. The parameters are ordered from best to worst, and in the x axis the first best parameters for each algorithm are represented. It can be seen that, in general, mean decreases while the variance increases, making the algorithm less sensitive and less robust. Choosing the ideal parameter is crucial to compare the performance of different algorithms.

parameters for each detector are: number of principal components in Q-residuals, the q in ITEME, and the Core similarity in MATCH. MAST does not have any parameters to choose, because the width of the motif is determined by the Position Specific Scoring Matrix (PSSM) used as an input.

Different parameter values have been studied for each detector and then the mean and variance of the AUC have been plotted for this values, ordered from best to worst in each case. The motif studied is the AP1 motif, from JASPAR database Bryne et al. (2008) The changes show a decrease in the mean and an increase in the variance, that means that, when the parameter of the detector changes, the detectors become less sensitive and less robust. MEET 5.1 directly choses the best detector according to equation A.1.

Another functionality of the MEET 5.1 training mode is that its output can be used to directly compare the different detectors. Using the data described above and the best parameter outputted by MEET 5.1, the package has been used to compare the different detectors. The results can be observed in table A.2 where the mean AUC for all the algorithms is shown.

Table A.2. Table with the comparison of the performance of the detectors included in MEET 5.1 using 10 sets of transcription factor binding sites in JASPAR and TRANSFAC database and backgrounds corresponding to promoters of each organism (human, mouse and yeast). The result shown is the mean of the AUC for each TFBS and each method. The best method depends on the binding sites.

TF	Qresiduals	Entropy	Divergence	MATCH	MAST
AP1	0.9893	0.9921	0.979	0.9868	**0.9925**
E2F1	0.9998	0.9979	0.9992	0.9995	**0.9999**
ETS1	0.9965	0.9956	**0.9972**	0.9922	0.9931
HLF	**0.9985**	0.9974	0.9965	0.9953	0.9688
NFLI3	0.9993	0.9992	0.9997	0.9980	**0.9999**
ARNT	0.9998	0.9998	0.9998	**1**	0.9999
FOXO3	0.9914	0.9747	0.9663	0.9765	**0.9947**
NFκB	**0.9998**	0.9747	0.9663	0.9765	0.9865
SPZ1	0.9944	0.9931	**0.9960**	0.9910	0.9913
ROX1	**0.9999**	0.9992	0.9941	0.9997	0.9937

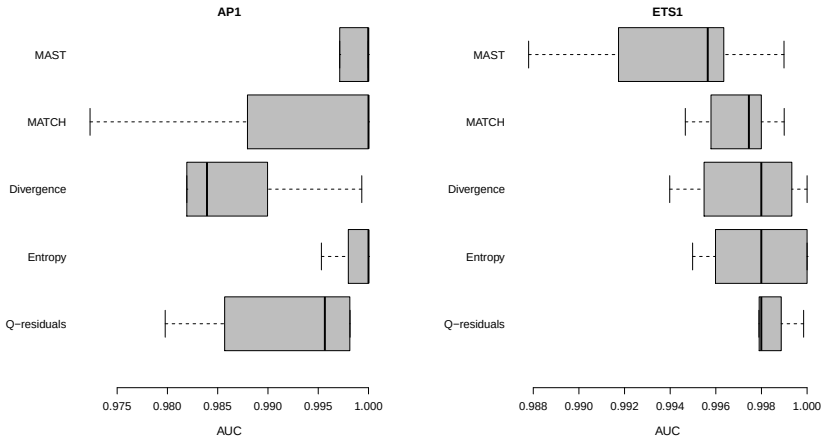
The output of the MEET 5.1 R package can also be used to plot the AUC boxplots, as it can be seen in figure A.9. In this figure the boxplot of the AUC for AP1, ETS1, FOXO3 and SPZ1 are shown using the ideal model chosen by the package. The figure A.9 shows that the mean and the variance depend on the method and the transcription factor.

A.4.3 Detection

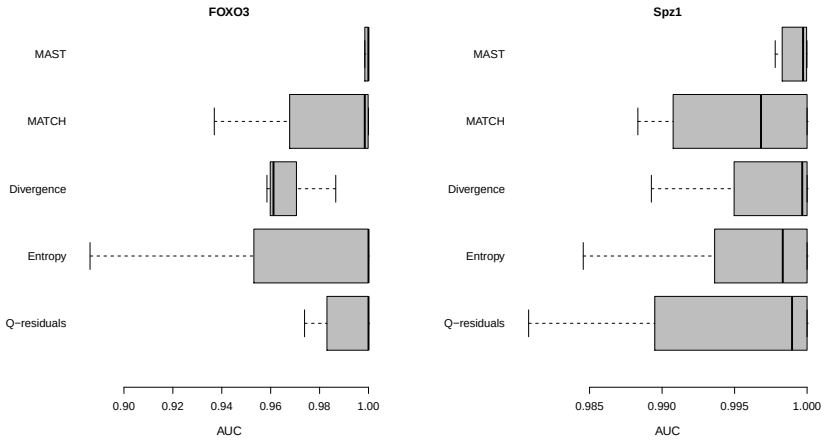
The detection mode of the MEET 5.1 R-package can also be used to detect binding sites within a large genomic sequence. The input of the detection can be the parameters of the algorithm or the model built using the training mode. As MEET 5.1 includes the optimal models for many JASPAR (Bryne et al., 2008) binding motifs, a genomic sequence can be explored in order to find binding sites of these motifs using the model included in the package.

In the table A.3, the detection is performed using all the algorithms available in MEET 5.1. The motif searched is the AP1 in humans, and the background used is the same background used in the training data, with a AP1 binding sequence inserted. The table shows the algorithm used, the p-value or Score of the AP1 sequence and how

h



(a) AP1 **(b)** ETS1



(c) FOXO3 **(d)** SPZ1

Figure A.9. The comparison between five detectors: MATCH, MAST, ITEME (Entropy and Divergence) and Q-residuals are shown in four different studied TFBS: AP1, ETS1, FOXO3 and SPZ1. The results show the robustness of the detectors, and that a single detector cannot be chosen as the best one for all TFBS

Table A.3. Table with the results in detection using all the algorithms available in the MEET 5.1 R package. The sequence models used are the best ones obtained with AP1 binding sites and the training mode, the background is a *mus musculus* promoter sequence with an AP1 binding site inserted in a certain position. The table shows the sequence with highest score, the position and the score corresponding to this sequence.

Algorithm	Order	Score
Entropy	1	0
Divergence	1	0
Q-residuals	1	2.9×10^{-6}
MAST	8	0.01
MATCH	1	0.97
MDscan	14	1.71

many sequences have a higher score.

A.5 Availability and requirements

Project name: MEET 5.1

Project home page: http://r-forge.r-project.org/projects/meet

Operating system(s): Platform independent

Programming language: : R (>= 2.11.0)

Library: seqinr, fields, pcaMethods, Matrix, ROCR, Hmisc, KernSmooth

License: GNU GPL

Any restrictions to use by non-academics: none

Appendix B

StringSabio

StringSabio is an R library which extracts the interaction between TF and genes from the String and SabioSciences databases. The R package is available from `http://sisbio.recerca.upc.edu/R/StringSabio.1.0.tar.gz`.

B.1 Motivation and Background

Data about the interaction between proteins is stored in large databases. Some of the databases take into account just physical proein-protein interactions such as MINT database (Chatr-aryamontri et al., 2007), while others take into account more interactions as co-ocurrence, and also include predicted data, as STRING (Szklarczyk et al., 2011).

These databases have been largely analyzed to study the interactome, usually using graph theory . Each protein can be represented by a vertex and each interaction is represented as an edge. But the current algorithms only study the direct interactions between proteins.

Adding information about the regulatory interactions would allow to obtain more useful information about the interactome. In this study an R package was constructed that can extract the information about the regulatory interactions from the STRING and Sabiosciences databases. The information is obtained in an automatic and non-redundant way.

Table B.1. Summary of the functions included into the stringsabio package. The name of the function is included together with the database used in the function and also a short description .

Function	Origin	Description
idString	String	ID extraction
intSring	String	PPI extraction
bioString	String	Choosing regulatory interactions
intSabioSciences	SabioSciences	Extracting regulatory interactions
SabiotoString	String/Sabio	Homogeneizing ID
StringSabio	String/Sabio	Eliminating redundant information

B.2 Architecture of the package

The StringSabio package contains different functions for a non-redundant extraction of the interaction data. The interactions can be obtained independently from each one of the databases, but also the search can be done combining both databases in order to retrieve as much information as possible.
The functions can be thus divided in three main groups, the String functions, the SabioSciences functions, and those functions that homgeneize the results and avoid redundant interactions. A summary of the functions included in the StringSabio R-package is shown in table B.1. A function called AllStringSabio allows to run all the functions and obtain all data just using one R command.
The basic functionality of the R-package is that, given a protein, interaction between transcription factors and this protein are output.

B.3 Description of the databases

STRING database includes experimental and predicted interactions between proteins. It has a score which gives the confidence of a given interaction, a higher score means a more reliable interaction. The STRING 9.0 version, which was used in the construction of the package contains the interactions between near 5 milion of proteins from 1133 organisms. The total amount of interactions exceeds the 100 milions.
SabioSciences contains the TFBS of each transcription factor. It combines a data

Table B.2. Regulatory interactions between F7 and transcription factors binding sites resulting from the extraction of the StringSabio package. The results include the query protein, the interacting transcription factors and the database where the interaction has been found. .

Protein A	Protein B	Interaction	Origin
FVII	HNF4G	Regulation	String
FVII	SP1	Regulation	String
FVII	HNF4A	Regulation	String/Sabio
FVII	BATF	Regulation	SabioSciences

mining algorithm which extracts regulations from published articles with the UCSC genome browser annotation of the TFBS.

B.4 Example

In the next example, the interactions for the $FVII$ protein in humans are retrieved from the databases.
First we load the needed packages, and then we run the string and sabio functions in order to retrieve the interactions. The easiest way to run the programs is to use the AllStringSabio function which also returns an error when the protein cannot be found the databases.

```
library(Rcurl)
library(XML)
library(string)
interactions<-AllStringSabio("F7", 0)
```

In the example above, 0 is the Sabiosciences taxonomy code for the organism which is: 0 for *Homo sapiens*, 1 for *Mus musculus* and 2 for *Rattus norvegicus*. The output of this example is shown in table B.2, where the interactions between the $FVII$ ptotein and the transcription factors binding sites are shown.The results include the query protein, the interacting transcription factors and the database where the interaction has been found.

References

E. Acar, C. Aykut-Bingol, H. Bingol, R. Bro, and B. Yener. Multiway analysis of epilepsy tensors. *Bioinformatics (Oxford, England)*, 23(13):i10–8, July 2007. ISSN 1367-4811. 47

V. Afreixo, P. J. Ferreira, and D. Santos. Fourier analysis of symbolic data: A brief review. *Digital Signal Processing*, 14(6):523–530, November 2004. ISSN 10512004. 38

M. Akhtar, J. Epps, and E. Ambikairajah. On dna numerical representations for period-3 based exon prediction. In *Genomic Signal Processing and Statistics*. GENSIPS 2007. IEEE International Workshop on, 2007. 38

M. Akhtar, J. Epps, and E. Ambikairajah. Signal processing in sequence analysis: Advances in eukaryotic gene prediction. *Selected Topics in Signal Processing, IEEE Journal of*, 2(3):310–321, June 2008. ISSN 1932-4553. 38

S. Altschul, W. Gish, and W. Miller. Basic local alignment search tool. *Journal of molecular . . .*, 215:403–410, 1990. 22

D. Anastassiou. Genomic signal processing. *Signal Processing Magazine, IEEE*, 18(4):8–20, 2001. 36

B. Aranda, H. Blankenburg, S. Kerrien, F. S. Brinkman, et al. PSICQUIC and PSISCORE: accessing and scoring molecular interactions. *Nature methods*, 8(7):528–529, July 2011. ISSN 1548-7105. 17

A. Arneodo, E. Bacry, P. V. Graves, and J. F. Muzy. Characterizing long-range correlations in dna sequences from wavelet analysis. *Phys. Rev. Lett.*, 74:3293–3296, Apr 1995. 38

T. L. Bailey, M. Boden, F. a. Buske, M. Frith, et al. MEME SUITE: tools for motif discovery and searching. *Nucleic acids research*, 37(Web Server issue):W202–8, July 2009. ISSN 1362-4962. 29, 123

T. L. Bailey, M. Bodén, T. Whitington, and P. Machanick. The value of position-specific priors in motif discovery using MEME. *BMC bioinformatics*, 11(1):179, January 2010. ISSN 1471-2105. 28

T. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 28–36. AAAI Press, August 1994. 28, 124, 136

T. Bailey and C. Elkan. Meme:discovering and analizing dna and protein sequence motifs. *Nucleic acids research*, 34: W369–W373, 2006. 136

T. Bailey and M. Gribskov. Combining evidence using p-values: Application to sequence homology searches. *Bioinformatics*, 14:48–54, 1998. 29, 124, 136

Y. Barash, G. Elidan, N. Friedman, and T. Kaplan. Modeling dependencies in protein-dna binding sites. In *Proceedings of the seventh annual international conference on Research in computational molecular biology*, RECOMB '03, pages 28–37, New York, NY, USA, 2003. ACM. ISBN 1-58113-635-8. 33

I. Ben-Gal, A. Shani, A. Gohr, J. Grau, et al. Identification of transcription factor binding sites with variable-order Bayesian networks. *Bioinformatics*, 21(11):2657–2666, 2005. 33

P. V. Benos, M. L. Bulyk, and G. D. Stormo. Additivity in proteindna interactions: how good an approximation is it? *Nucleic Acids Research*, 30(20):4442–4451, 2002. 28

D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, and S. E. Ostell. GenBank. *Nucleic acids research*, 40(Database issue): D48–53, January 2012. 3, 50

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. *Journal of Molecular Biology*, 193(4):723 – 743, 1987. ISSN 0022-2836. 28

B. P. Berman, Y. Nibu, B. D. Pfeiffer, P. Tomancak, et al. Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the Drosophila genome. *Proceedings of the National Academy of Sciences*, 99(2), 2002. 35

E. Blanco, D. Farré, M. M. Albà, X. Messeguer, and R. Guigó. ABS: a database of Annotated regulatory Binding Sites from orthologous promoters. *Nucleic acids research*, 34 (Database issue):D63–7, January 2006. ISSN 1362-4962. 15

R. Bro. PARAFAC. Tutorial and applications. *Chemometrics and Intelligent Laboratory Systems*, 38(2):149–171, October 1997. ISSN 01697439. 45

R. Bro. *Multi-way analysis in the food industry: Models, algorithms and applications*. PhD thesis, University of Amsterdam, The Netherlands, 1998. 45

R. Bro. Exploratory study of sugar production using fluorescence spectroscopy and multi-way analysis. *Chemometrics and Intelligent Laboratory Systems*, 46(2):133 – 147, 1999. ISSN 0169-7439. 43

J. C. Bryne, E. Valen, M.-H. E. Tang, T. Marstrand, et al. JASPAR, the open access database of transcription factor-binding profiles: new content and tools in the 2008 update. *Nucleic acids research*, 36(Database issue):D102–6, January 2008. ISSN 1362-4962. 124, 138, 139

M. Buckland and F. Gey. The relationship between Recall and Precision. *Journal of the American Society for Information Science*, 45(1):12–19, 1994. ISSN 1097-4571. 69

REFERENCES

M. L. Bulyk, P. L. F. Johnson, and G. M. Church. Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nucl. Acids Res.*, 30(5):1255–1261, 2002. 25

D. Cai, a. Delcher, B. Kao, and S. Kasif. Modeling splice sites with Bayes networks. *Bioinformatics (Oxford, England)*, 16 (2):152–8, February 2000. ISSN 1367-4803. 33

M. F. Carey, C. L. Peterson, and S. T. Smale. Chromatin immunoprecipitation (ChIP). *Cold Spring Harbor protocols*, 2009(9):pdb.prot5279, September 2009. ISSN 1559-6095. 11

J. M. Carlson, A. Chakravarty, C. E. DeZiel, and R. H. Gross. SCOPE: a web server for practical de novo motif discovery. *Nucleic acids research*, 35(Web Server issue):W259–64, July 2007. ISSN 1362-4962. 124

J. Carroll and J.-J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of Eckart-Young decomposition. *Psychometrika*, 35(3):283–319, 1970. ISSN 0033-3123. 44

K. Cartharius, K. Frech, K. Grote, B. Klocke, et al. MatInspector and beyond: promoter analysis based on transcription factor binding sites. *Bioinformatics (Oxford, England)*, 21(13):2933–42, July 2005. ISSN 1367-4803. 30

R. Castelo and R. Guigó. Splice site identification by idlbns. *Bioinformatics*, 20(suppl 1):i69–i76, 2004. 33

R. Cattell. parallel proportional profiles and other principles for determining the choice of factors by rotation. *Psychometrika*, 9:267–283, 1944. ISSN 0033-3123. 44

A. Chatr-aryamontri, A. Ceol, L. M. Palazzi, G. Nardelli, et al. MINT: the Molecular INTeraction database. *Nucleic acids research*, 35(Database issue):D572–4, January 2007. ISSN 1362-4962. 16, 143

D. Che, S. Jensen, L. Cai, and J. S. Liu. BEST: binding-site estimation suite of tools. *Bioinformatics (Oxford, England)*, 21(12):2909–11, June 2005. ISSN 1367-4803. 124

P. Collas and J. A. Dahl. Chop it, chip it, check it: the current status of chromatin immunoprecipitation. *Frontiers in Bioscience*, 13:929–943, January 2008. xiii, 12

A. Conesa, J. M. Prats-Montalbán, S. Tarazona, M. J. Nueda, and A. Ferrer. A multiway approach to data integration in systems biology based on Tucker3 and N-PLS. *Chemometrics and Intelligent Laboratory Systems*, 104(1):101–111, November 2010. ISSN 01697439. 47

G. E. Crawford, I. E. Holt, J. C. Mullikin, D. Tai, et al. Identifying gene regulatory elements by genome-wide recovery of DNase hypersensitive sites. *Proceedings of the National Academy of Sciences of the United States of America*, 101 (4):992–7, January 2004. ISSN 0027-8424. 10

P. Cristea. *Genomic Signal processing and statistics*, chapter Representation and analysis of DNA sequences. Hindawi Publishing Corporation, 2005. 36

F. Daenen, F. van Roy, and P. J. De Bleser. Low nucleosome occupancy is encoded around functional human transcription factor binding sites. *BMC genomics*, 9:332, January 2008. ISSN 1471-2164. 35

J. Davis and M. Goadrich. The Relationship Between Precision-Recall and ROC Curves. In *Proceedings of the 23rd International Conference on Machine learning*, Pittsburg, PA, 2006. 69

M. de Sousa Vieira. Statistics of dna sequences: A low-frequency analysis. *Phys. Rev. E*, 60:5932–5937, Nov 1999. 38

P. D'haeseleer. What are DNA sequence motifs? *Nat Biotech*, 24(4):423–425, 2006. 24, 123

M. O. Dorschner, M. Hawrylycz, R. Humbert, J. C. Wallace, et al. High-throughput localization of functional elements by quantitative chromatin profiling. *Nature methods*, 1(3): 219–225, 2004. 10

I. Dunham, A. Kundaje, S. F. Aldred, P. J. Collins, et al. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74, September 2012. ISSN 0028-0836. 18, 19

C. Durante, R. Bro, and M. Cocchi. A classification tool for n-way array based on simca methodology. *Chemometrics and Intelligent Laboratory Systems*, 106(1):73 – 85, 2011. ISSN 0169-7439. 87

R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. *Biological sequence analysis: Probabilistic models of Protein and Nucleic acids*, chapter 2. Cambridge University Press, 1998. 21

S. Eddy. Profile hidden Markov models. *Bioinformatics*, pages 755–763, 1998. 24

R. Edgar. Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res*, 32(5): 1792–1797, 2004. 23, 66, 124

K. Ellrott, C. Yang, F. M. Sladek, and T. Jiang. Identifying transcription factor binding sites through Markov chain optimization. *Bioinformatics (Oxford, England)*, 18 Suppl 2(1):S100–9, January 2002. ISSN 1367-4803. 31

L. Elnitski, V. X. Jin, P. J. Farnham, and S. J. Jones. Locating mammalian transcription factor binding sites: A survey of computational and experimental techniques. *Genome Research*, 16(12):1455–1464, 2006. 9

R. Fleischmann, M. Adams, O. White, R. Clayton, et al. Whole-genome random sequencing and assembly of haemophilus influenzae rd. *Science*, 269(5223):496–512, 1995. 3

P. Flicek, M. R. Amode, D. Barrell, K. Beal, et al. Ensembl 2012. *Nucleic Acids Research*, 2011. 3

M. C. Frith, U. Hansen, and Z. Weng. Detection of cis-element clusters in higher eukaryotic DNA. *Bioinformatics (Oxford, England)*, 17(10):878–889, 2001. 35

S. M. Gallo, D. T. Gerrard, D. Miner, M. Simich, et al. REDfly v3.0: toward a comprehensive database of transcriptional regulatory elements in Drosophila. *Nucleic acids research*, 39(Database issue):D118–23, January 2011. ISSN 1362-4962. 15

J. Grau, I. Ben-Gal, S. Posch, and I. Grosse. VOMBAT: prediction of transcription factor binding sites using variable order Bayesian trees. *Nucleic acids research*, 34(Web Server issue):W529–33, July 2006. ISSN 1362-4962. 123

REFERENCES

R. E. Green, J. Krause, A. W. Briggs, T. Maricic, et al. A draft sequence of the Neandertal genome. *Science (New York, N.Y.)*, 328(5979):710–22, May 2010. ISSN 1095-9203. 4

O. Hallikas, K. Palin, N. Sinjushina, R. Rautiainen, et al. Genome-wide prediction of mammalian enhancers based on analysis of transcription-factor binding affinity. *Cell*, 124(1):47–59, January 2006. ISSN 0092-8674. 10

M. Hammell. Computational methods to identify miRNA targets. *Seminars in cell & developmental biology*, 21(7):738–44, September 2010. ISSN 1096-3634. 9

R. Hamming. Error detecting and error correcting codes. *Bell System Technical Journal, The*, 29(2):147–160, April 1950. ISSN 0005-8580. 25

S. Hannenhalli. Eukaryotic transcription factor binding sites–modeling and integrative search methods. *Bioinformatics*, 24(11):1325–1331, 2008. 24

R. A. Harshman. Foundations of the PARAFAC procedure: Models and conditions for an" explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16(1): 84, 1970. 44

L. Hellman and M. Fried. Electrophoretic mobility shift assay (EMSA) for detecting proteinnucleic acid interactions. *Nature protocols*, 2(8):1849–1861, 2007. 10

T. Hindemitt and K. F. X. Mayer. CREDO: a web-based tool for computational detection of conserved sequence motifs in noncoding sequences. *Bioinformatics (Oxford, England)*, 21(23):4304–6, December 2005. ISSN 1367-4803. 124

N. S. Holter, A. Maritan, M. Cieplak, N. V. Fedoroff, and J. R. Banavar. Dynamic modeling of gene expression data. *Proceedings of the National Academy of Sciences of the United States of America*, 98(4), 2001. 43

P. K. Hopke, P. Paatero, H. Jia, R. A. Harshman, and R. T. Ross. Three-way PARAFAC / factor analysis: examination and comparison of alternative computational methods as applied to ill-conditioned data. *Chemometrics and Intelligent Laboratory Systems*, 43:25–42, 1998. 45

H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, September 1933. 39

J. Hu, B. Li, and D. Kihara. Limitations and potentials of current motif discovery algorithms. *Nucleic acids research*, 33(15):4899–913, January 2005. ISSN 1362-4962. 123

N. Ichinose, T. Yada, and O. Gotoh. Large-scale motif discovery using DNA Gray code and equiprobable oligomers. *Bioinformatics (Oxford, England)*, 28(1):25–31, January 2012. ISSN 1367-4811. 25

J. E. Jackson. *A user's guide to Principal Components*, chapter 2, pages 36–40. John Wiley & Sons, Inc., 2004. ISBN 9780471725336. 42

B. Jiang, M. Q. Zhang, and X. Zhang. OSCAR: one-class SVM for accurate recognition of cis-elements. *Bioinformatics (Oxford, England)*, 23(21):2823–8, November 2007. ISSN 1367-4811. 39

V. John R. Multiway frequency analysis for experimental psychologists. *Canadian Journal of Experimental Psychology/Revue canadienne de psychologie expérimentale*, 57(3):257–264, 2003. ISSN 1878-7290(Electronic);1196-1961(Print). 43

I. Jolliffe. *Principal Component Analysis*. Springer series in Statistics. Springer-Verlag, New York, 1989. 39

C. Kanz, P. Aldebert, N. Althorpe, W. Baker, et al. The EMBL Nucleotide Sequence Database. *Nucleic acids research*, 33(Database issue):D29–33, January 2005. ISSN 1362-4962. 3

R. E. Kass and A. E. Raftery. Bayes Factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995. 55

A. Kel, E. Gossling, I. Reuter, E. Cheremushkin, et al. MATCHTM: a tool for searching transcription factor binding sites in DNA sequences. *Nucl. Acids Res.*, 31(13):3576–3579, 2003. 67, 124

H. Kiers. Hierarchical relations among three-way methods. *Psychometrika*, 56(3):449–470, 1991. ISSN 0033-3123. 43

I. V. Kiers, Henk A. L.and Mechelen. Three-way component analysis: Principles and illustrative application. *Psychological Methods*, 6(1):12–19, 2001. 43

O. D. King. A non-parametric model for transcription factor binding sites. *Nucleic Acids Research*, 31(19):116e–116, October 2003. ISSN 1362-4962. 31

L. J. Korn, C. L. Queen, and M. N. Wegman. Computer analysis of nucleic acid regulatory sequences. *Proceedings of the National Academy of Sciences*, 74(10):4401–4405, 1977. 24

S. Kullback and R. A. Leibler. On information and sufficiency. *Ann.Math. Stat*, 22:79–86, 1951. 135, 136

E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, et al. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, February 2001. ISSN 0028-0836. 3

M. Larkin, G. Blackshields, N. Brown, R. Chenna, et al. Clustal W and Clustal X version 2.0. *Bioinformatics*, 23 (21):2947–2948, 2007. 61

D. S. Latchman. *Eukaryotic Transcription Factors*. Elsevier, 2008. 6

A. W.-C. Liew, H. Yan, and M. Yang. Pattern recognition techniques for the emerging field of bioinformatics: A review. *Pattern Recognition*, 38(11):2055 – 2073, 2005. ISSN 0031-3203. 63

D. Lipman and W. Pearson. Rapid and sensitive protein similarity searches. *Science*, 227(4693):1435–1441, 1985. 22

X. S. Liu, D. L. Brutlag, and J. S. Liu. An algorithm for finding protein-dna binding sites with applications to chromatin-immunoprecipitation microarray experiments. *Nature biotechnology*, 20(8):835–9, August 2002. ISSN 1087-0156. 124, 136

A. Luna and J. Pinto. Determination of Paracetamol and Ibuprofen in Tablets and Urine Using Spectrofluorimetric Determination Coupled with Chemometric Tools. *Austin Journal of Anal Pharm Chem*, 1(1):7, 2014. xv, 45

REFERENCES

S. Ma and Y. Dai. Principal component analysis based methods in bioinformatics studies. *Briefings in Bioinformatics*, 12(6):714–722, 2011. 43

S. Ma and M. R. Kosorok. Identification of differential gene pathways with principal component analysis. *Bioinformatics (Oxford, England)*, 25(7):882–9, April 2009. ISSN 1367-4811. 43

S. Mahony, D. Hendrix, A. Golden, T. J. Smith, and D. S. Rokhsar. Transcription factor binding site identification using the self-organizing map. *Bioinformatics (Oxford, England)*, 21(9):1807–14, May 2005. ISSN 1367-4803. 35

V. D. Marinescu, I. S. Kohane, and A. Riva. MAPPER: a search engine for the computational identification of putative transcription factor binding sites in multiple genomes. *BMC bioinformatics*, 6:79, January 2005. ISSN 1471-2105. 15

J. Mata, S. Marguerat, and J. Bähler. Post-transcriptional control of gene expression: a genome-wide perspective. *Trends in biochemical sciences*, 30(9):506–14, September 2005. ISSN 0968-0004. 9

S. O. Material, S. Web, H. Press, N. York, and A. Nw. The ENCODE (ENCyclopedia Of DNA Elements) Project. *Science (New York, N.Y.)*, 306(5696):636–40, October 2004. ISSN 1095-9203. 18

A. Mathelier, X. Zhao, A. W. Zhang, F. Parcy, et al. Jaspar 2014: an extensively expanded and updated open-access database of transcription factor binding profiles. *Nucleic Acids Research*, 42(D1):D142–D147, 2014. 14

J. Maynou, J.-J. Gallardo-Chacon, M. Vallverdu, P. Caminal, and A. Perera. Computational detection of transcription factor binding sites through differential rnyi entropy. *Information Theory, IEEE Transactions on*, 56(2):734 –741, feb. 2010a. ISSN 0018-9448. 124, 135

J. Maynou, J.-J. Gallardo-Chacon, M. Vallverdu, P. Caminal, and A. Perera. Computational Detection of Transcription Factor Binding Sites Through Differential Renyi Entropy. *IEEE transactions on information theory*, 56(2): 734–741, 2010b. ISSN 0018-9448. 30

G. McLachlan. *Discriminant analysis and statistical pattern recognition*. Wiley, 1992. 93

T. P. Minka. Bayesian inference, entropy and the multinomial distribution. Technical report, Microsoft Research, 2003. 55

S. Miyazaki, H. Sugawara, K. Ikeo, T. Gojobori, and Y. Tateno. DDBJ in the stream of various biological data. *Nucleic acids research*, 32(Database issue):D31–4, January 2004. ISSN 1362-4962. 3

A. Mohd-Sarip and C. P. Verrijzer. A higher order of silence. *Science*, 306(5701):1484–1485, 2004. 5

a. Nandy. Two-dimensional graphical representation of DNA sequences and intron-exon discrimination in intron-rich sequences. *Computer applications in the biosciences : CABIOS*, 12(1):55–62, February 1996. ISSN 0266-7061. 39

L. Narlikar, R. Gordn, U. Ohler, and A. J. Hartemink. Informative priors based on transcription factor structural class improve de novo motif discovery. *Bioinformatics*, 22 (14):e384–e392, 2006. 59

B. T. Naughton, E. Fratkin, S. Batzoglou, and D. L. Brutlag. A graph-based motif detection algorithm models complex nucleotide dependencies in transcription factor binding sites. *Nucleic Acids Research*, 34(20):5730–5739, 2006. 34, 74, 76

S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48 (3):443 – 453, 1970. ISSN 0022-2836. 21

A. Neuwald, J. Liu, and C. Lawrence. Gibbs motif sampling: Detection of bacterial outer membrane protein repeats. *Protein Sci.*, 4:1618–1632, 1995. 28

C. Notredame and D. G. Higgins. SAGA: sequence alignment by genetic algorithm. *Nucleic acids research*, 24(8):1515–24, April 1996. ISSN 0305-1048. 24

C. Notredame, D. G. Higgins, and J. Heringa. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of molecular biology*, 302(1):205–17, September 2000. ISSN 0022-2836. 23

C. Notredame. Recent evolutions of multiple sequence alignment algorithms. *PLoS computational biology*, 3(8):e123, August 2007. ISSN 1553-7358. 23

R. a. O'Flanagan, G. Paillard, R. Lavery, and a. M. Sengupta. Non-additivity in protein-DNA binding. *Bioinformatics (Oxford, England)*, 21(10):2254–63, May 2005. ISSN 1367-4803. 25

M. Okuno, E. Arimoto, Y. Ikenobu, T. Nishihara, and M. Imagawa. Dual dna-binding specificity of peroxisome-proliferator-activated receptor gamma controlled by heterodimer formation with retinoid x receptor alpha. *Biochem. J.*, 353(2):193–198, 2001. 56

L. Omberg, G. H. Golub, and O. Alter. A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies. *Proceedings of the National Academy of Sciences of the United States of America*, 104(47):18371–6, November 2007. ISSN 1091-6490. 47

R. Osada, E. Zaslavsky, and M. Singh. Comparative analysis of methods for representing and searching for transcription factor binding sites. *Bioinformatics (Oxford, England)*, 20 (18):3516–25, December 2004. ISSN 1367-4803. 28, 123

E. Pairó, J. Maynou, M. Vallverdu, P. Caminal, and A. Perera. Meet: motif elements estimation toolkit. In *EMBC 2011*, september 2011. 66

E. Pairó, J. Maynou, S. Marco, and A. Perera. A subspace method for the detection of transcription factor binding sites. *Bioinformatics*, 2012. xiv, 37, 61, 124, 136

G. Pavesi, G. Mauri, and G. Pesole. In silico representation and discovery of transcription factor binding sites. *Briefings in bioinformatics*, 5(3):217–36, September 2004a. ISSN 1467-5463. 24

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. *Nucl. Acids Res.*, 32(suppl 2):W199–203, 2004b. 25

K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901. 39

C. K. Peng, S. V. Buldyrev, A. L. Goldberger, S. Havlin, et al. Long-range correlations in nucleotide sequences. *Nature*, 356(6365):168–170, 1992. 38

R. C. Périer, V. Praz, T. Junier, C. Bonnard, and P. Bucher. The eukaryotic promoter database (EPD). *Nucleic acids research*, 28(1):302–3, January 2000. ISSN 0305-1048. 4

J. E. Phillips and V. G. Corces. CTCF: Master Weaver of the Genome. *Cell*, 137(7):1194–1211, 2009. ISSN 0092-8674. 49

N. J. Proudfoot, A. Furger, and M. J. Dye. Integrating mRNA processing with transcription. *Cell*, 108(4):501–12, February 2002. ISSN 0092-8674. 8

B. J. Raney, M. S. Cline, K. R. Rosenbloom, T. R. Dreszer, et al. ENCODE whole-genome data in the UCSC genome browser (2011 update). *Nucleic acids research*, 39 (Database issue):D871–5, January 2011. ISSN 1362-4962. 4, 19

B. Ren, F. Robert, J. J. Wyrick, O. Aparicio, et al. Genome-wide location and function of DNA binding proteins. *Science (New York, N.Y.)*, 290(5500):2306–9, December 2000. ISSN 0036-8075. 11

A. Renyi. On measures of entropy and information. In *Proc. 4th Berkeley Simposium on Mathematics, Statistics and Probability*, pages 547–561, 1961. 135

P. Rice, I. Longden, and A. Bleasby. EMBOSS: the European molecular biology open software suite. *Trends in genetics*, 16(6):2–3, 2000. 3

H. G. Roider, A. Kanhere, T. Manke, and M. Vingron. Predicting transcription factor affinities to DNA from a biophysical model. *Bioinformatics (Oxford, England)*, 23(2): 134–41, January 2007. ISSN 1367-4811. 28

Ronald A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–186, 1936. 92

G. Sandve and F. Drablos. A survey of motif discovery methods in an integrated framework. *Biology Direct*, 1(1):11, 2006. 24

G. K. F. Sandve. *Potentials and limitations of motif-based binding site prediction in DNA Geir Kjetil Ferkingstad Sandve.* PhD thesis, Norwegian University of Science and Technology, 2008. xiii, 7, 24

T. Schneider. Information content of individual genetic sequences. *J. Theor. Biol.*, 189:427–441, 1997. 28

C. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948. 136

B. Silverman and R. Linsker. A measure of dna periodicity. *Journal of Theoretical Biology*, 118:295–300, 1986. 36, 38

T. Sing, O. Sander, N. Beerenwinkel, and T. Lengauer. ROCR: visualizing classifier performance in R. *Bioinformatics (Oxford, England)*, 21(20):3940–1, October 2005. ISSN 1367-4803. 69

T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *Journal of molecular biology*, 147 (1):195–197, March 1981. ISSN 0022-2836. 22

W. Stacklies, H. Redestig, M. Scholz, D. Walther, and J. Selbig. pcaMethods a bioconductor package providing PCA methods for incomplete data. *Bioinformatics*, 23(9):1164–1167, 2007. 67

C. Stark, B.-J. Breitkreutz, A. Chatr-Aryamontri, L. Boucher, et al. The BioGRID Interaction Database: 2011 update. *Nucleic acids research*, 39(Database issue):D698–704, January 2011. ISSN 1362-4962. 17

G. Stormo. Dna binding sites: Representation and discovery. *Bioinformatics*, 16:16–23, 2000. 24

G. D. Stormo and D. S. Fields. Specificity, free energy and information content in protein-DNA interactions. *Trends in biochemical sciences*, 23(3):109–13, March 1998. ISSN 0968-0004. 28

D. Szklarczyk, A. Franceschini, M. Kuhn, M. Simonovic, et al. The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic acids research*, 39(Database issue):D561–8, January 2011. ISSN 1362-4962. 16, 143

M. Thomas-Chollier, M. Defrance, a. Medina-Rivera, O. Sand, et al. RSAT 2011: regulatory sequence analysis tools. *Nucleic Acids Research*, 39(Web Server):W86–W91, June 2011. ISSN 0305-1048. 124

J. D. Thompson, D. G. Higgins, and T. J. Gibson. CLUSTALW: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucl. Acids Res.*, 22(22):4673–4680, 1994. 23, 124

W. Thompson. Gibbs Recursive Sampler: finding transcription factor binding sites. *Nucleic Acids Research*, 31(13): 3580–3585, July 2003. ISSN 1362-4962. 28

R. E. Thurman, E. Rynes, R. Humbert, J. Vierstra, et al. The accessible chromatin landscape of the human genome. *Nature*, 489(7414):75–82, September 2012. ISSN 0028-0836. 10

A. Tomovic and E. J. Oakeley. Position dependencies in transcription factor binding sites. *Bioinformatics*, 23(8):933–941, 2007. 25, 52, 55, 56

M. Tompa, N. Li, T. L. Bailey, G. M. Church, et al. Assessing computational tools for the discovery of transcription factor binding sites. *Nature biotechnology*, 23(1):137–44, January 2005. ISSN 1087-0156. 123

L. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966. ISSN 0033-3123. 43

A. Visel, S. Minovitsky, I. Dubchak, and L. a. Pennacchio. VISTA Enhancer Browser–a database of tissue-specific human enhancers. *Nucleic acids research*, 35(Database issue): D88–92, January 2007. ISSN 1362-4962. 15

REFERENCES

R. F. Voss. Evolution of long-range fractal correlations and 1/ f noise in dna base sequences. *Phys. Rev. Lett.*, 68: 3805–3808, Jun 1992. 38

M. Wall, A. Rechtsteiner, and L. Rocha. Singular value decomposition and principal component analysis. In D. P. Berrar, W. Dubitzky, and M. Granzow, editors, *A Practical Approach to Microarray Data Analysis*, pages 91–109. Springer US, 2003. ISBN 978-0-306-47815-4. 43

J. Wang, J. Zhuang, S. Iyer, X. Lin, et al. Sequence features and chromatin structure around the genomic regions bound by 119 human transcription factors. *Genome Research*, 22(9):1798–1812, 2012. 50

Z. Wang and C. Burge. Splicing regulation: From a parts list of regulatory elements to an integral splicing code. *Rna*, 14(617):802–813, 2008. 8

D. Wang, W and Johnson. Computing linear transforms of symbolic signals. *IEEE Transactions on Signal Processing*, 50(3):628–634, March 2002. ISSN 1053587X. 38

T. Whitfield, J. Wang, P. Collins, E. C. Partridge, et al. Functional analysis of transcription factor binding sites in human promoters. *Genome Biology*, 13(9):R50, 2012. ISSN 1465-6906. 49

F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):pp. 80–83, 1945. ISSN 00994987. 70

E. Wingender, X. Chen, R. Hehl, H. Karas, et al. TRANSFAC: an integrated system for gene expression regulation. *Nucl. Acids Res.*, 28(1):316–319, 2000. 136

E. Wingender. The TRANSFAC project as an example of framework technology that supports the analysis of genomic regulation. *Briefings in bioinformatics*, 9(4):326–32, July 2008. ISSN 1477-4054. 13

D. W.Mount. *Bioinformatics: Sequence and genome analysis*, chapter 3. Cold Spring Harbor laboratory Press, 1998. 20

G. a. Wray. The evolutionary significance of cis-regulatory mutations. *Nature reviews. Genetics*, 8(3):206–16, March 2007. ISSN 1471-0056. 47

E. P. Xing, W. Wu, M. I. Jordan, and R. M. Karp. LOGOS: a modular Bayesian model for de novo motif detection. *Proceedings / IEEE Computer Society Bioinformatics Conference. IEEE Computer Society Bioinformatics Conference*, 2:266–76, January 2003. ISSN 1555-3930. 35

B. Yener, E. Acar, P. Aguis, K. Bennett, et al. Multiway modeling and analysis in stem cell systems biology. *BMC systems biology*, 2:63, January 2008. ISSN 1752-0509. 47

K. Y. Yeung and W. L. Ruzzo. Principal component analysis for clustering gene expression data. *Bioinformatics (Oxford, England)*, 17(9):763–774, 2001. 43

K. Y. Yip, C. Cheng, N. Bhardwaj, J. B. Brown, et al. Classification of human genomic regions based on experimentally determined binding sites of more than 100 transcription-related factors. *Genome biology*, 13(9):R48, September 2012. ISSN 1465-6914. 19

C. Yuan, B. Liao, and T.-m. Wang. New 3D graphical representation of DNA sequences and their numerical characterization. *Chemical Physics Letters*, 379(5-6):412–417, October 2003. ISSN 00092614. 39

L. Zhang. Splice site prediction with quadratic discriminant analysis using diversity measure. *Nucleic Acids Research*, 31(21):6214–6220, November 2003. ISSN 1362-4962. 93

M. Q. Zhang. Identification of protein coding regions in the human genome by quadratic discriminant analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 94(2):565–8, January 1997. ISSN 0027-8424. 93

M. Zhang and T. Marr. A weight array method for splicing signal analysis. *Comput. Appl. Biosci.*, 9(5):499–509, 1993. 31

X. Zhao, H. Huang, and T. P. Speed. Finding short DNA motifs using permuted Markov models. *Journal of Computational Biology*, 12(6):894–906, 2004. xiv, 32

Q. Zhou and J. S. Liu. Modeling within-motif dependence for transcription factor binding site predictions. *Bioinformatics*, 20(6):909–916, 2004. 25, 52, 55, 56