Large Scale Identification of Transcription Factor Binding Sites in DNA Sequences

by

Javier Sánchez Galán

School of Computer Science
McGill University
Montréal

January 2010

A thesis submitted to
McGill University
in partial fulfilment of the requirements for the degree of
Master of Science

DEDICATION

This document is dedicated to those who think, to those who hope, to those who work hard and to those who never give up until they fulfill their dreams.

> *"Blessed is the man who finds wisdom, the man who gains understanding, for she is more profitable than silver and yields better returns than gold. She is more precious than rubies; nothing you desire can compare with her".*
>
> Proverbs 3:13-15

Contents

List of Figures

List of Tables

ABSTRACT

To date, gene regulation is still one of the most studied processes in molecular biology. Among its main actors, proteins called transcription factors, play an essential role in controling the rate of expression of genes, by binding to specific sites on the DNA sequence. These sites are short in lenght (5 to 15 basepairs) and are called transcription factor binding sites (TFBSs). These interactions between proteins and DNA have a fundamental role at several stages of cell development and in response to stress conditions. Various computational methods that exploit specific characteristic of TFBS have been developed and tested for the purpose of the identification of TFBSs. Examples include, the identification of TFBSs via phylogenetic footprinting, via cis-regulatory modules and via statistical over-representation.

In this thesis we present a new approach that uses elements of the three identification methods to develop a large-scale approach that assesses the over-representation of TFBS in DNA sequences. Results of application of this new method are presented for five biological datasets: including a set of regions bound by estrogen receptor (ER). We also present new results, yet to be validated experimentally, from two interesting biological datasets. The first is a dataset containing coding regions under non-coding selection (called CRUNCS). The other is a set of genes regulated by proteins called angiopoietins.

Finally, a new public bioinformatic software, used to estimate the over-representation of TFBSs in DNA sequences, that we call the Genome-Wide Analysis of TFBS Over-Representation (GATOR), is introduced.

RÉSUMÉ

À ce jour, la régulation des gènes est encore l'un des processus les plus étudiés en biologie moléculaire. L'une de ses principales categories d'acteurs, des protéines appelées facteurs de transcription, joue un rôle essentiel dans le contrôle du taux d'expression des gènes, en se liant à des sites spécifiques sur la séquence d'ADN. Ces sites sont des séquences courtes (de 5 à 15 paires de bases) et sont communément appelés sites de liaison pour les facteurs de transcription (TFBSs, en anglais). Les interactions entre ces protéines et l'ADN jouent un rôle fondamental à plusieurs stades du développement cellulaire et de la réponse à divers types de stress. Diverses méthodes de calcul qui exploitent les caractéristiques spécifiques des TFBS ont été développées et testées dans le but de l'identifier de tels sites de liaison. Citons par exemple l'identification des TFBS à l'aide des empreintes phylogénétiques, des modules de régulation cis et de la sur-représentation statistique.

Dans cette thèse nous présentons une nouvelle approche qui utilise des éléments des trois méthodes d'identification susmentionnés pour développer une approche à grande échelle qui évalue la sur-représentation des TFBS, dans les séquences d'ADN. Les résultats de l'utilisation de cette nouvelle méthode sont présentés pour cinq ensembles de données biologiques. Parmi eux, un ensemble des régions de sites de liaison liées aux récepteurs d'œstrogène (ER), un ensemble de données qui contient des régions codantes sous sélection non codante (appelé CRUNCS) et finalment, un ensemble de génes régulés par des protéines appelées angiopoietines.

Finalement, nous présentons un nouveau logiciel bioinformatique public qui sert à estimer la sur-représentation des TFBSs dans les séquences d'ADN et que nous avos appelé le Genome-Wide Analysis of TFBS Over-Representation (GATOR).

ACKNOWLEDGMENTS

I would like to thank God for putting the right people in my way to make my dream happen. The government of Panama who has given me full financial support to continue my education through the Professional Excellence Scholarship Program organized by IFARHU and SENACYT.

Thanks to Dr. Mathieu Blanchette for being an amazingly open and patient supervisor. For believing in my potential and for teaching me with passion and dedication about computer science, bioinformatics, statistics and about everyday life.

I also want to thank the rest of the students and professors that are part of the McGill Centre for Bioinformatics and the School of Computer Science of McGill University (special thanks to Dr. Doina Precup), for insightful discussions and for their support during the preparation of this work.

Thanks to Dr. Pierre-Étienne Jacques from Institut de Recherche Cliniques de Montréal (IRCM), as well as Dr. Sabah Hussain from Royal Victoria Hospital, for facilitating datasets for testing our implementation.

Last but not least I want to thanks my friends (too many to list!), my mentors (Dr. Victor Lopez (ABD) and Dr. Oris Sanjur) and my family (Rafael, Osmila and Rafael Jr.), for the full support and inspiration they have given me during all these years.

Chapter 1

Introduction and Thesis Outline

In this first chapter, concepts of molecular biology will be introduced briefly. An special emphasis is placed on the process of DNA transcription in higher eukaryotes (e.g. humans). Transcription factor binding sites computational representation and discovery will be explained. The last part of this chapter is dedicated to describing the existing limitations when searching for TFBS and lastly, the purpose of this project will be described in details.

1.1 Molecular Biology as of Today

Although many new discoveries have been made in molecular biology since the publication of Crick's seminal papers [25, 26] describing the Central Dogma of molecular biology, the statement that DNA sequences can be copied to DNA, by means of DNA replication, DNA information can be copied to mRNA, by means of transcription and proteins can synthesized using the information in mRNA as a template, by means of translation (as seen in Figure 1.1), remains valid. In general terms the central dogma remains a very accurate attempt to describe the process by which the sequence infor-

mation is transferred from nucleic acids (DNA and RNA) into proteins.

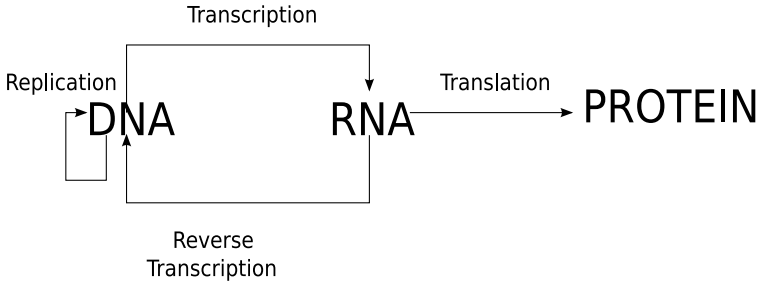

Figure 1.1: Central Dogma of Molecular Biology as explained by Crick

Today, thanks to the complete sequencing of the human genome, we can say that we understand the process of information transfer (from DNA to Proteins) a little better. In fact, two major fields have been founded for the specific analysis and study of all genes and all proteins of an organism, namely *Genomics* and *Proteomics*.

In the case of genomics, which is the main topic of this thesis, we have been able to figure out that only a tiny fraction (around %2) of the DNA in the human genome is functional (that is, it consist of protein coding exons) and that around 50% of the human DNA consists of non protein coding repetitive sequences [88].

Despite all of our current knowledge about the molecular mechanisms of the cell, the inner working of processes like gene expression regulation remains a big question to be answered and DNA transcription remains one of the most widely studied processes in molecular biology [85].

1.2 Gene Expression and Transcription Factors

In human cells, genes are divided in two portions: (1) coding sequences, called *exons*[1], which carry the required information for protein synthesis, or in simpler terms,

[1]Exons are not necessarily coding sequences

sequences that define what the gene produces; and (2) non-coding sequences, called *introns*, which are in part responsible of determining if the gene is activated or not. The process of gene expression can be defined as the process by which the complete structure of a gene (DNA sequence) is transformed in a functional gene product (a protein). This process of transformation of DNA sequences into proteins occurs in three phases: a gene is first copied entirely (*DNA Transcription*), with its coding and non-coding sequences, yielding to a primary transcript. The transcript is then processed to remove the introns (*Splicing*), creating a *mature transcript* or messenger RNA (*mRNA*). This mature transcript is translated into a sequence of amino acids, which defines a protein (*RNA Protein Synthesis or Translation*).

Even though an essential characteristic of genes is the possibility of being expressed, not every gene is expressed at the same rate at the same time or under the same conditions. On the one hand, some are expressed in every cell of the human body all the time, these genes are called *housekeeping genes*, which are essential for many basic cellular functions. On the other hand, other genes are expressed in a particular type of cells or tissue or at particular stages of cell development [3].

Gene expression is a complex process which is controlled by proteins called *Transcription Factors* (TFs). These proteins bind to the surrounding DNA sequences of a gene and become responsible for the activation or repression of the gene.

1.2.1 DNA Binding of Transcription Factors

At the molecular level the main actor of the transcription is an enzyme called RNA polymerase (or RNAP) that using the DNA sequence as a template is the responsible of creating the RNA molecules. In detail, what happens is that the free RNAP molecule binds strongly to a specific DNA sequence called *promoter sequence*, the promoter is usually found upstream of a gene and contains the *transcription start site*

for RNA synthesis or the site where RNA synthesis begins. The DNA double helix is opened by the RNAP, which then progresses along the template strand in the 3' to 5' direction one nucleotide at the time[2]synthesizing a complementary RNA molecule, this process is often called *elongation*. The synthesis ends at a *termination signal* or stop signal where both the polymerase and the new created RNA molecule are released. The typical RNA molecule is between 70 and 10,000 nucleotides long [3].

1.2.2 Impact of Transcription Factors on Gene Expression

Transcription factors and the specific DNA sequences they bind[3]are very important for the gene expression machinery. They have been said to be two of the most important functional elements in any genome [14]. It is very well documented that defects in the process of transcription can lead to the occurence of various diseases, for example, a variety of cancers result from chromosomal rearrangements (translocations) involving either regulatory elements or transcription factors [55].

Transcription factors are protein themselves and must be regulated by other TFs. All genes and proteins are part of a molecular regulatory machinery that starts with the TFs present at the beginning of development (the early TFs present in embryonic state of an organism).

A better understanding of the interaction between TFs and TFBSs will allow a mapping of the regulatory pathways in cells which in turn will provide a clearer interpretation of the role of individual genes in health and disease.

[2]Although RNA polymerase traverses the template strand from 3' to 5', the coding (non-template) strand is usually used as the reference point, so transcription is said to go from 5' to 3'.

[3]This DNA sequences are often called Transcription Factor Binding Sites or TFBS

1.3 Identification of Transcription Factor Binding Sites

The functions of the transcriptional regulatory elements are determined in their majority by the protein/DNA interactions, thus, the identification of all protein binding sites is a major and necesary step in the characterization of functional elements in the human genome [22]. However, the task of identification of TFBSs is a non trivial one and requires a synergistic collaboration between computational and experimental methods [29].

Traditional Methods Traditionally, TFBS have been determined using experimental methodologies such as: footprinting methods [33], gel-shifts [34] and South-western blotting [10]. These methods are generally expensive, time consuming and not scalable to genome-wide analysis [14]. In the last decade a number of high-throughput technologies have been developed for the purpose of identifying TFBS *in vitro*[4]and *in vivo*[5].

One high-throughput technique for finding high-affinity binding sequences *in vitro* is the *Systematic Evolution of Ligands by Exponential Enrichment* or SELEX. In this technique, randomized single stranded DNA sequences (RNA sequences) are generated and paired with a protein of interest. Those sequences that bind with high affinity to the protein or compound of interest are selected and the rest of the sequences are removed by using affinity chromatography [6].

Another very effective technique for measuring the interactions between DNA and

[4]Experiments performed in a controlled environment usually in a test tube, outside of a living organism.

[5]Experiments performed inside an organism or a living tissue.

[6]Affinity chromatography can be defined as separation method applied to biochemical mixtures, which is based in the biological interactions between the solution that is wanted to purify and the molecule that is used to purify it.

TFs *in vivo* is the *genome-wide location analysis* or ChIP-chip (also known as ChIP on chip). This technique combines chromatin immunoprecipitation *ChIP* with DNA microarray technology *chip*. The goal of a ChIP-chip experiment is to isolate and identify the DNA sequence which are bound by TFs and one of its main advangates over other methods is that it can be used to identify binding sites on a genome-wide scale by using special microarrays called tiling-arrays, which are basically a way of looking to specific regions of interest of the genome.

It is important to mention that in the last two years a newer technology called ChIP-sequencing (or ChIP-seq, for short) which combines chromatin immunoprecipitation (ChIP) with massively parallel DNA sequencing was introduced to the market and it is slowly replacing ChIP-chip experiments. Contrary to ChIP-chip, in ChIP-seq instead of needing thousands of runs to cover the complete genome, just a single sequencing run can do genome-wide associations with higher resolution [53].

Computational Methods The rise in favor for more efficient and reliable *in silico* or computational methods for detection of TFBS happens for three main reasons: (1) because experimental methods are costly (still nowadays); (2) because while most experimental methods just report a few number of binding sites per experiment, computational methods can search for new putative motifs exhaustively and (3) because the binding of a TF *In vitro* not always translate to a functional binding *in vivo* [14].

Stormo [80] wisely divides the computational prediction of DNA binding sites in two subproblems: the representation problem and the prediction problem. These problems, the solutions that have been found and their limitations will be explained in the next two sections.

1.4 Computational Representation of Transcription Factor Binding Sites

Defined in a formal way, the problem of computational representation of TFBS can be stated as follows: *Given:* A collection of known binding sites for a given TF. *Find:* A representation of those sites that can be used to search for new sequences and predict where additional sites occur.

One of the most important characteristics of TFs is that for regulatory purposes they accept variations in the sites they bind. Therefore, the only way to describe a binding site correctly and still be true to the real phenomenon is to create a representation that maps more than one DNA sequence to a single TF.

Three distinct but related approaches have been successfully used to represent the alignment of different sequences for a TF: *consensus sequences* (also called sequence patterns or regular expressions), *position weight matrices-PWM* (also called position-specific scoring matrices-PSSM, often pronounced as *possums*), and, more complex *higher order models* [28].

1.4.1 Consensus Sequences

A consensus sequence is a way of representing the results of a multiple sequence alignment. In the context of TF representation, these sequences usually come from experimental results that represent the different DNA binding sites affinities shown by a transcription factor. The consensus sequence shows which nucleotides are most abundant in the alignment at each position.

When building a consensus sequence for a TF, all the experimentally found binding sites DNA sequences for a certain factor are first aligned. Then, the number of

occurrences of every nucleotide in every position of the DNA sequences is counted, thus creating a *profile.* Once the profile is created, a decision about which nucleotide should be the most representative for each position is made. One strategy to do this is to assign each position to the nucleotide (A, C, G or T) that is observed the most often at each position.

In many cases there are more than one nucleotide that are observed with a reasonably high frequency at a given position, when this occurs the position is said to be *degenerated.* For the purpose of allowing a better representation of degenerated positions, the degenerate IUPAC nucleic codes [24] were created. They extend the number of symbols used to represent a given position, therefore allowing for a greater variations. The symbols present in the IUPAC nucleic codes are shown in Table 1.1.

Finally, when the most observed nucleotide or the degenerate symbol has been defined for every position of the DNA sequence, then that string of symbols is said to be a *consensus sequence* that represents a given TFBS.

A simple example that we will use for the remaining of this section are the binding sequences for the LBP-1 transcription factor, described in the TRANSFAC database as M00644. In Table 1.2, five DNA sequences which serve as binding sites for this factor are shown.

Table 1.3 shows the profile for LBP-1, with the respective counts for every nucleotide in every position. Also, the final consensus is shown in the bottom of the table.

Conclusions about Consensus Sequences An advantage of using consensus sequences as a way to represent TFBS is that they are a simple representation that can be understood by humans and by computer programs alike. Specifically, in the case

Table 1.1: IUPAC code for representing degenerate nucleotide sequence patterns

Symbol	Meaning	Origin of Designation
G	G	Guanine
A	A	Adenine
T	T	Thymine
C	C	Cytosine
R	G or A	puRine
Y	T or C	pYrimidine
M	A or C	aMino
K	G or T	Keto
S	G or C	Strong interaction
W	A or T	Weak Interaction
H	A or C or T	not-G, follows G in the alphabet
B	G or T or C	not-A, follows A in the alphabet
V	G or C or A	not-T (not-U), follows U in the alphabet
D	G or A or T	not-C, follows C in the alphabet
N	G or A or T or C	aNy

Table 1.2: Binding sites (DNA sequences) for the LBP-1 transcription factor as described in TRANSFAC profile M00644

Site 1	C	A	G	C	T	G	C
Site 2	C	G	G	C	T	T	G
Site 3	C	C	G	C	T	G	G
Site 4	C	A	G	C	T	G	C
Site 5	C	A	G	C	T	G	C

Table 1.3: Binding site profile and consensus sequences for the LBP-1 transcription factor

Position	1	2	3	4	5	6	7
A	0	3	0	0	0	0	0
C	5	1	0	5	0	0	3
G	0	1	5	0	0	4	2
T	0	0	0	0	5	1	0
Consensus	C	A	G	C	T	G	S

of computer programs consensus sequences can be seen as regular expressions [7], and as regular expressions are manipulated easily by computer programs, they facilate the process of motif scanning (explained in detail in section 1.5).

Despite these facts, consensus sequences do not contain precise information about the relative likelihood of observing the alternate nucleotides at every position of a TFBS [14], thus failing to reflect the quantitative characteristics of TFBSs. For example, in Table 1.3 the second position of the consensus was set to be an A, but someone can argue that the V symbol would have been a better representation, one that captures all the nucleotides in play at the moment of binding. A similar event happens in the last position of the consensus, it is set to S, but someone can easily argue that it should have been a C instead of a S. In general, the decisions about each position of the consensus are subjectively made by the person who builds it.

The lack of objectivity and the fact that by using a consensus sequence as representation for a binding site, information about the original DNA sequences is lost, has motivated the creation of more accurate methods for binding sites representation, for instance the position weight matrices.

[7] Regular expressions provide a flexible way to identifying strings of characters (particular characters, words, or patterns of characters) of text of interest

1.4.2 Position Weight Matrices

When building a Position Weight Matrix (PWM) the same process used to build a consensus is followed, however, it differs in that the profile, with its counts for every nucleotide in every position, is normalized, or in other terms, every position of every column is divided by the total count for that column (see formula 1.1), thus converting the profile into a *position frequency matrix* (PFM).

$$PFM_{b,i} = \frac{P_{b,i}}{N} \tag{1.1}$$

, where $P_{b,i}$ represents position *b, i* of the binding site profile and N is the number of sequences available.

The normalized profile or PFM is just a table in which each cell contains the probability of observing a given nucleotide at a given position of the motif and in which every column sums to a total of one.

Following our example of the LBP-1 transcription factor we can easily convert Table 1.3 into a PFM, as it is shown in Table 1.4.

Table 1.4: Position frequency matrix (PFM) for the LBP-1 transcription factor

Position	1	2	3	4	5	6	7
A	0.0	0.6	0.0	0.0	0.0	0.0	0.0
C	1.0	0.2	0.0	1.0	0.0	0.0	0.6
G	0.0	0.2	1.0	0.0	0.0	0.8	0.4
T	0.0	0.0	0.0	0.0	1.0	0.2	0.0
Σ	1.0	1.0	1.0	1.0	1.0	1.0	1.0

Usually, to avoid zero valued probabilities, small amounts called *pseudocounts* are added to the counts in the PFM before normalizing. This has a twofold benefit: first,

to avoid problems while computing a motif score (for this task the PFM is converted to a log-scale, therefore the PFM values must be non-zero values) and second, is a way to account for unobserved nucleotides (given the characteristics of TFBSs one cannot be certain that a particular nucleotide never occurs in a real binding site [45]).

While there is not a specific formula for the calculation of pseudocounts, Wasserman and Sandelin [85], give a very generic formula (seen in Formula 1.2) to calculate the values of a pseudocount corrected PFM or (CPFM).

$$CPFM_{b,i} = \frac{P_{b,i} + s(b) * Pr(b)}{N + s(b)} \tag{1.2}$$

, where $P_{b,i}$ represent the counts of base b in position i, N is the number of sequences from where the profile was build, $s(b)$ is the pseudocount function (usually set to 1) and $Pr(b)$ is the background model, or the probability of seeing a nucleotide b in the aligned sequences.

The background model can be set to give equal probability of seeing a nucleotide in the sequences as: Pr(A)=Pr(C)=Pr(G)=Pr(T)=0.25. Specifically, in the case of studying the human genome, we can use as probabilities the DNA composition values discovered by Erwin Chargaff's [18]. In his findings he discovered that the four basic nucelotides (A, C, G and T) are present in the human genome in the following percentages: A=30.9% and T=29.4%; G=19.9% and C=19.8%. These and the discoveries made for other organisms were later used to formulate what we know now as the Chargaff's rules[8].

As our PFM for LBP-1 transcription factor (Table 1.4) have a couple of positions with probability of zero, we can use Formula 1.2 to calculate the pseudocounts. If we use s(b)=1 and background probabilities as Pr(A)=Pr(T)=0.32 and

[8]The first rule of Chargaff's states that in DNA the number of guanine bases equals the number of cytosine bases and the number of adenine bases equals the number of thymine bases

Pr(C)=Pr(G)=0.18 we get the values shown in Table 1.5.

Table 1.5: Corrected position frequency matrix (CPFM) for the LBP-1 transcription factor

Position	1	2	3	4	5	6	7
A	0.05	0.55	0.05	0.05	0.05	0.05	0.05
C	0.86	0.20	0.03	0.86	0.03	0.03	0.53
G	0.03	0.20	0.86	0.03	0.03	0.70	0.36
T	0.05	0.05	0.05	0.05	0.89	0.22	0.05

A CPFM can be converted into PWM by determining the *weight* of each nucleotide at each position. For this task many valid approaches exists. The most commonly used approach is to set every weight to the log-likelihood ratio of the nucleotide with respect to the background frequency of the nucleotide in the aligned sequences or in the entire genome (as seen on Formula 1.3).

$$PWM_{b,i} = -log_2(\frac{CPFM_{b,i}}{Pr(b)}) \tag{1.3}$$

, where Pr(b) is the background probability of base b and $CPFM_{b,i}$ is the corrected probability of base b in position i.

If we have the background distribution of nucleotides in the aligned sequences or in the genome we are studying, for example, Pr(A)=Pr(T)=0.32 and Pr(C)=Pr(G)=0.18, we can convert our CPFM for LBP-1 (Table 1.5) to a PWM using Formula 1.3. Results of the conversion are shown in Table 1.6.

Other approaches to convert a CPFM to a PWM are: (1) setting every weight to the negative logarithm of the frequencies of each base at each position [80] (see formula 1.4). (2) setting every weight to the expected (average) self-information (see formula 1.5).

Table 1.6: Position weight matrix (PWM) for the LBP-1 transcription factor

Position	1	2	3	4	5	6	7
A	-2.58	0.79	-2.58	-2.58	-2.58	-2.58	-2.58
C	2.26	0.13	-2.58	2.26	-2.58	-2.58	1.56
G	-2.58	0.13	2.26	-2.58	-2.58	1.95	1.01
T	-2.58	-2.58	-2.58	-2.58	1.47	-0.54	-2.58

$$PWM_{b,i} = -log(PFM_{b,i}) \tag{1.4}$$

$$PWM_{b,i} = -PFM_{b,i} * log(PFM_{b,i}) \tag{1.5}$$

The information content (IC) of a PWM as a whole can be calculated as the sum of the expected self-information of every element by using formula 1.6. In general terms a lower IC indicates higher variability (lower specificity) in the sites.

$$IC(PFM) = \sum_{b=A}^{T} \sum_{i=1}^{l} PFM_{b,i} * log_2(\frac{PFM_{b,i}}{Pr(b)}) \tag{1.6}$$

Sequence logo When weights of a CPFM are calculated via their self-information (by using Formula 1.5), it is most likely that their final representation is not the PWM, but instead a *sequence logo* [76]. A sequence logo is a graphical representation used to display patterns in a set of aligned sequences. Sequence logos are very useful as they show how well nucleotides are conserved at each position and also show the relative frequency of bases and the information content (measured in bits) at every position of a site or sequence.

A sequence logo for the LBP-1 transcription factor was made using the WEBLOGO [27] tool (as seen in Figure 1.2) using as reference the binding sites shown in

Table 1.2.

Figure 1.2: Sequence Logo from five DNA sequences that represent binding sites for LBP-1 transcription factor

Conclusions about PWMs PWMs represent a great advance from consensus sequences. They are a more informative way of representing binding sites, however, there are a number of considerations about PWMs that are worthy of mention:

1. Since PWMs are based on observed data, the greater number of DNA sequences that are observed, the better the matrix representation reflects the real binding preference. Hence, those TFs that are very short in length and non commonly observed will have weak PWM representations.

2. PWMs are somewhat accurate in identifying *in vitro* target sequences but are insufficiently specific in the identification of sites with *in vivo* function to provide a meaningful predictions [14].

3. PWMs, because of their expressiveness, are more suitable for motif scanning than consensus sequences (this will be this discussed in next section).

1.4.3 Higher Order Models

Although both methods, consensus sequences and PWMs, offer a fairly clear and straightforward representation of biologically interesting candidates, they both as-

sume that every nucleotide of the motif is independent from the others, ignoring the inter-nucleotide dependencies.

For example, suppose there are two candidate sequences: $S_1 = AGTTG$ and $S_2 = CGAAG$, and PWM representation P under an independent nucleotide model. Then, the background probability of seeing G as the second nucleotide is the same on both sequences. On contrary, under the higher order model, the background probability of seeing G as the second nucleotide in the sequence also depends on the nucleotide in the first position (AG or CG), thus giving different background probabilities for S_1 and S_2.

Clever models that account for all the interactions left out in the *independent nucleotide models* have been developed and implemented. For instance di-nucleotide matrix models has been suggested [81], in which the alphabet has been extended to 16 letters, as $\{AA, AC, AG...TG, TC, TT\}$ in order to represent interaction between nucleotides. Also, models in which the background is represented as a (j-th order Markov model) in which the probability of finding a nucleotide in a given position of a motif depends on the j nucleotides that are preceding that position, has been also proposed [82]. However, the reality is that these models have achieved modest specificity, when compared to the basic position-independent models [49]. For instance, Marstrand *et al.* [54] present a zero-order PWM outperforming a third-order model both on artificial and experimental data.

1.5 PWM Databases

A great number of experimentally and computationally defined TFBSs have been already assembled in databases for various organisms as: yeast (SCPD) [92], bacteria (DBTBS) [79] and even for human (TRANSFAC) [87] and (JASPAR) [74].

Even though, both, JASPAR and TRANSFAC databases contain PWMs for human TFBS, they have been assembled for different purposes. JASPAR is an open database and contains a non-redundant set of 436 matrices[9], while TRANSFAC is a commercial database[10] and contains a redundant set of 892 matrices[11].

1.5.1 Motif Scanning

Once having TFBSs represented as consensus sequences or PWMs, the next logical step is to scan selected DNA sequences (or even complete genomes) to discover possible TFBSs locations. This problem is often called motif scanning and can be defined in a formal way as: *Given*: A DNA sequence S and a PWM M of length l. *Find:* Binding sites present in S that match M.

For motif scanning we can use as template any of the TFBSs representations that were described in the previous section. However, as consensus sequence are a limited description of binding sites that can only tell us if the scanned sequences match or not the consensus, their predictive power is low. In other hand, a DNA sequence can be compated to a PWM and a score for every position of the sequence can be calculated by adding the scores of every column of the PWM (see formula 1.7[12])

[9]At the moment of writing this thesis JASPAR release 3.0 included the JASPAR CORE, which consists of 138 matrices and the JASPAR Collections: JASPAR PHYLOFACTS, JASPAR FAM, JASPAR CNE, JASPAR POLII and JASPAR SPLICE, which contains 174, 11, 233, 13 and 5 matrices respectively

[10]There is a part of TRANSFAC database that is public for Academic and Non-profit Organizations, It can be accessed in http://www.gene-regulation.com/pub/databases.html

[11]At the moment of writing this thesis TRANSFAC 2009.2 contains 892 matrices

[12]It is needed to explain that formula 1.7 represents a sum of the values at each position of PWM assuming that they represent the logarithm of base 2 of the likelihood ratio, ratio that describes the nucleotide with respect to the background frequency of the nucleotide in the aligned sequences calculated. However, this score can also be computed by using multiplying each individual likelihood, that is, the raw ratio without the logarithm calculation.

and only those segments whose score reaches some predetermined threshold or cut-off value are reported as matches or interesting candidates.

$$Score_S = \sum_{i=1}^{l} PWM_{S_i,i} \tag{1.7}$$

where $Score_S$ is the score of aligning a DNA sequence S to a PWM of length l.

Si represents the nucleotide in position i in an input sequence S.

As example, if we want to score a given DNA sequence $S = \{CCAATTG\}$ against the PWM representation of LBP-1 that we have in Table 1.6, we will just sum the values (weights) that appears in every position of the PWM for every nucleotide of the string. For easier reading these positions have been shaded in Table 1.7.

Table 1.7: Position weight matrix (PWM) for the LBP-1 transcription factor

Position	1	2	3	4	5	6	7
A	-2.58	0.79	-2.58	-2.58	-2.58	-2.58	-2.58
C	2.26	0.13	-2.58	2.26	-2.58	-2.58	1.56
G	-2.58	0.13	2.26	-2.58	-2.58	1.95	1.01
T	-2.58	-2.58	-2.58	-2.58	1.47	-0.54	-2.58

$Score_S = PWM(C,1)+PWM(C,2)+PWM(A,3)+PWM(A,4)+PWM(T,5)+PWM(T,6)+PWM(G,7)$
$= 2.26 + 0.13 + (-2.58) + (-2.58) + 1.47 + (-0.54) + 1.01 = -0.86$

By using Formula 1.7 it is also possible to scan strings S that are longer than the number of columns in our PWM, l. However, a sliding window approach is needed to make substrings of S of length l, to be fed to the equation asynchronously.

1.6 Reducing False Positive Rate of Predictions in Motif Scanning

A lot of progress has been achieved by using Motif Scanning. However, the problem remains a non-trivial one, because long DNA sequences (like the human genome) report a large number of sites that are predicted to be binding sites. Specifically, some motifs with weak representations (very short sequences) will yield too many predicted sites, which will be mostly uninteresting.

Three characteristics of the gene regulation process are frequently used as extra information in predictive models in order to reduce the number of false predictions:

1. Functional regions, like TFBSs, tend to be conserved through evolution of species. Phylogenetic footprinting is a method that takes advantage of this fact to improve the TFBS prediction (see section 1.6.1).

2. TFs tend to act in groups or clusters when regulating the expression of a gene. Methods looking for TFBS clusters are described in section 1.6.2.

3. Functional TFBSs will be over-represented in the regulatory regions of co-regulated genes when compared to a background set of genes. Over-representation approaches are discussed in section 1.6.3.

From these three characteristics, three different approaches for the prediction of TFBS have been proposed: *phylogenetic footprinting*, or the problem of predicting one TFBS given many orthologous sequences [13] and *clustering of TFBSs*, or the problem of predicting many TFBSs given one sequence and *Over-represented TFBSs*.

[13] Two sequences are orthologous if they share a common ancestor and are separated by speciation.

1.6.1 Prediction of Transcription Factor Binding Sites by using Phylogenetic footprinting

There are two common ways to discover regulatory elements in genomic sequences: (1) to try do discover these elements by using sequences from coregulated genes and searching for similar matches in a single genome and (2) to try to discover regulatory regions by using sequences from a single gene and searching for similar matches in multiple genomic sequences. Over-representation and clustered CRM modules methods falls in the first category, while phylogenetic footprinting falls in the second.

With the completion of the sequencing of genomes for different species, the idea of looking for similar DNA regulatory elements among different species has given birth to phylogenetic footprinting, which studies the structural relationship of functional elements between different genomes by comparing orthologous gene sequences [14] [49].

Phylogenetic footprinting approaches are often used to understand the mechanisms of genomic evolution that occurs in different species or simply to find similarities and differences between regulatory regions. Phylogenetic footprinting is based on two basic assumptions. The first is that mutations will be less frequent in functional regions of the genome, than in regions without specific functions. The second is that orthologus genes are usually regulated by the same mechanism in different species.

A phylogenetic footprinting approach will typically follow the following steps [85]:

1. Selection of orthologous sequences: The sequences to be aligned need to be of an appropriate evolutionary distance, in order to show conservation of functional elements.

   On the one hand, aligning sequences from closely related species will align very

[14] Two sequences or genes are orthologous if they share a common ancestor and are separated by speciation events and paralogous if their divergence is caused by duplication events.

well, almost without gaps, which will make very hard to distinguish conserved functional elements from non-functional elements. On the other hand, aligning sequences from very divergent species will be difficult and most likely will show no conservation of functional elements between the two sequences [36].

2. Alignment of the promoter sequences for comparison: Methods for sequence alignments can be divided into three groups: local, global and hybrid (also called glocal). They are considered local, if they align short similar fragments of the input sequences, global if they align entire sequences given as inputs into a single alignment and hybrid if they use a combination of local and global alignment to produce the final result [6]. Also, according to the number of sequences they align they can be divided in pairwise sequence alignment (due to their nature, the pairwise multiple sequence alignment methods also fall in this category) and multiple sequence alignments.

   Examples of implementations that are used for pairwise sequence alignment of promoters regions are: BLASTZ [77], for local alignments and LAGAN [13] and AVID [11] for global alignments. In the case of global multiple sequence alignments, we find the multiple sequence versions of LAGAN and AVID called MLAGAN [13] and MAVID [12], respectively. Also, TBA/MULTIZ [8] and many members of the CLUSTAL family [21] [83]. It is worth mentioning hybrid approaches which are not used for purely phylogenetic purposes as T-Coffee [59] and the DIALIGN [58] family of methods.

3. Visualize identified segments of conservation: as multiple sequence alignment is done in the previous step, what is left to do, is just to find a way to visualize and finally interpret the results. Two known multiple sequence alignment visualization packages are: rVista [51] and PipMaker [78].

Phylogenetic approaches have succesfully clarified conserved regulatory regions between the human genome and genomes several vertebrates [67] and between the

human genomes and several mammals [90]. For instance, two well known implementation of phylogenetic footprinting approaches are: ConSite [75], that use a profile-based phylogenetic footprinting approach to detect conserved regions in mouse and human and FootPrinter [9], in which the phylogenetic footprinting is approached not only from as a mere multiple sequence alignment problem, but also with an evolutionary point of view by using the notion of parsimony between sequences.

1.6.2 Prediction of Cis-Regulatory modules

When regulating the expression of a gene, several TFs can bind to DNA sequences in segments that can cover a few hundreds of base-pair long, forming what is often called a *Cis-regulatory module (CRM)*. Genes can have multiple CRMs in their flanking non-coding sequence. These modules are believed to control transcription regulatory processes in space and time [36].

The methods for prediction of CRM are usually classified in two groups: *supervised* and *unsupervised*. In the first group fall all the methods that employ machine learning techniques that use the characteristics of known regulatory modules in order to discover sequences with similar characteristics. For this task, probabilistic models such as HMMs are used to represent the CRMs. For instance, Frith *et al.* [32] created an HMM model for intra and inter CRM regions from a single sequence, by using two major states: modules and background sequences, as well as transition and emission probabilities among them.

In the second group fall all the *ab initio* approaches that try to predict the optimal subset of DNA motifs that will be present in a given CRM, without prior information. *Ab initio* discovery implementations have given good results for yeast and drosophila [37]. Also, an interesting method is presented by Blanchette *et al.* [7], in which a phylogenetic based CRM discovery approach led to the prediction of more

than 118,000 CRMs in the human genome.

1.6.3 Prediction of Statistically Over-Represented Transcription Factor Binding Sites

One solution found to overcome the great number of false positives predictions that result from motif scanning, is by adding information from the sequences scanned. For instance, scanning the promoter regions of genes that are known (or believed) to be co-regulated by the same TF and then look for the over-represented TF motifs present in those sequences.

Over-represented TFBS prediction works under the assumption that if a functional motif is present in co-regulated sequences, then the number of matches will be greater that will be expected by chance in a similar, but random generated set.

This problem can be defined in a formal way as. *Given:* The promoter sequences S of n co-expressed genes and a set of PWMs $M_1, M_2, \ldots, M_n$ of length $l_1, l_2, \ldots, l_n$. *Find:* PWMs whose number of predicted sites is surprisingly large[15].

A typical approach of this type will involve the following steps [66]:

1. Score the set of sequences of co-expressed genes against the set of PWMs using the formula described in Equation 1.7.

2. Define a threshold T to discriminate interesting scores from uninteresting ones.

3. Keep the PWMs that report a score greater or equal to the threshold T.

[15]Surprisingly large means that the number of occurrences is statistically larger than what will be expected by chance

4. Generate set of random DNA sequences and score them against the PWMs selected in step 2.

5. Compare the scores obtained by the sequences co-expressed genes to the scores obtained by random sequences by using a test of statistical significance. A test of statistical significance is just a procedure used to evaluate if the differences between counts is either *statistically significant*, meaning that the difference is large enough to conclude that the corresponding population values are different or *not statistically significant*, meaning that the difference between a sample value and another value should be attributed to random error or chance [61].

   For the purpose of quantifying the statistical significance the standard score or Z-Score (seen in Formula 1.8) is often used. It is necessary to mention that there are many others test that can also be used, among them the Chi-square test, F-test, Wilcoxon signed-rank test and T-test.

$$Z = \frac{MatchCor - \mu(MatchRand)}{\sigma(MatchRand)} \tag{1.8}$$

   where $MatchCor$, represents the number of matches found in the sequences investigated, $\mu(MatchRand)$, represents the mean number of matches found in random sequences and $\sigma(MatchRand)$ represent the standard deviation of the number of matches found in random sequences[16].

6. Statistical measurements (Z-scores) for each PWM is transformed into P-values and finally each motif is reported as under-represented or over-represented.

   On the one hand, a motif with low p-value suggest that the motif is significantly over-represented in the DNA sequences under investigation, implying that they are present for a real biological reason. On the other hand, a high p-value sug-

[16]The contents of motifs in random sequences are supposed to be normal distributed, hence the use of the mean and standard deviation

gest that the motif is significantly over-represented in random DNA sequences and under-represented in the sequences under investigation.

The limitations of this approach are twofold: the difficulty of selecting a unique threshold to quantify PWMs that are heterogeneous in nature (each one have a different Information Content) and finding an unbiased way to model the background or generating the random sequence. Also another limitation of this type of methods is that even if a TFBS results to be significantly over-represented it does not imply a direct biological function [54], this in part due to the fact of existence of post-transcriptional events such as: alternative splicing, nuclear export, stability, localization and translation [44].

One particular implementation which uses the over-represented statistics and a phylogenetic footprinting is oPOSSUM [41], which uses conserved promoters between human and mouse with statistical significance methods to identify over/under-represented sites in co-expressed genes. Another implementation of this methods is Cis-eLement OVer-representation (CLOVER) [31], which uses PWMs from TRANSFAC and JASPAR database profiles and used different genomes (from mammals, birds and drosophila) as background correction.

1.7 Ab Initio Transcription Factor Binding Sites Motif Discovery

Motif Scanning in its different flavors serves as a good way to understand the gene regulation process. However, it relies on the assumption that we have a representation such as a PWM or a consensus sequence. The question is, what happens if we do not have a representation of binding sites for a given transcription factor?

Defined in a formal way the computational prediction of TFBS problem can be stated as follows: *Given:* A set of sequences known to contain binding sites for a common factor (but not knowing where the sites are and what the TF is). *Find:* The location of the sites in each sequence and a representation for the specificity of the protein.

Searching for recognizable patterns in DNA sequences from scratch (*ab initio)*, without any other information than the target sequence, is a problem that has been addressed since the beginning of computational biology [36]. In essence *ab initio* prediction of TFBS is a very challenging problem because:

1. We do not know the motif sequence
2. We do not know where it is located relative to the genes start sites
3. Motifs can differ slightly from one gene to another
4. We do not know how to distinguish real from non-real (random) motifs

The methods that have been developed for this task fall in two categories: enumerative or exhaustive methods commonly called *Pattern Driven* and alignment-based methods commonly called *Sequence Driven.*

1.7.1 Pattern Driven Methods

In these type of methods significant patterns of length l are identified from a given set of DNA sequences. In essence the problem is solved by first generating all possible patterns of a given length l, then searching for the occurrences of the pattern, counting and scoring them according to a statistical significance method and finally reporting the patterns that achieve the higher scores.

Approximate sequence patterns or degenerate patterns can be identified in the sequences and their similarity can be assessed using the Hamming distance (number of positions in which the two patterns differ) or the Levinstein distance (number of substitutions, insertions or deletions needed to transform one string into another), when multiple of these patterns are close enough (according to their distance) they can be merged into one single approximate pattern.

Pattern driven methods are enumerative in nature and are guaranteed to find optimal solution in a restricted search space, however searching for long patterns is computationally expensive. To circumvent this problem, preprocessing techniques have been used to reduce the search space, thus reducing the cost of searching for patterns. A particularly interesting enhancement has been the use of *suffix trees* [38], which accelerates the search by organizing the input sequence in an indexing structure [65] and by that allowing to search for longer patterns since the search time is linear in the length of the patterns, but exponential in the number of mutations to be tolerated in the sites [36].

1.7.2 Sequence Driven Methods

The main goal of sequence driven methods is to predict the location of sites and their PWM representation using the raw sequence data. If the location of sites is known, then building a PWM is trivial. However in the *ab initio* motif discovery problem this information is missing, thus it has to be learned from the data, usually using machine learning techniques. While pattern driven methods yield optimal results, by enumerating exhaustively all possible patterns, in sequence driven methods obtaining globally optimal results is not guaranteed, because they are based on heuristics.

The three main techniques that have been commonly used for *ab initio* motif discovery are: greedy algorithms, expectation-maximization (EM) and Gibbs sampling.

Greedy Algorithms This type of algorithms were the first methods to be used for motif discovery, introduced by Hertz *et al.* [39] and implemented in the software package Consensus [40]. In essence, given a set of sequences and motif length l to be searched, the algorithm will build a set of matrices by comparing all pairs of sequences and progressively including the sequences of the sites which maximize the information content (IC) of the matrix.

One of the flaws of this method is that there is no way to discern which matrices are interesting and which not, causing the risk of storing unnecessary matrices that correspond to random patterns (uninteresting patterns) in the initial steps and discarding the matrices that correspond to occurrences of a real motif [64].

Expectation Maximization (EM) EM simplifies the problem of searching with missing information by iteratively looking for the parameters that maximize the likelihood of the data. In detail this type of algorithms iterate between two steps the E step (expectation) and M-step (maximization).

Initially the algorithm generates an initial PWM (which is either made at random or via prior knowledge of the binding sites), then in the E-step this initial PWM is used to estimate the probability of each subsequence being a bindings site. In the M-step based on these probabilities a new PWM is generated. The two steps are repeated until the method converges to a PWM representation.

One very well known implementation of this approach is Multiple Expectation Maximisation for Motif Elicitation (MEME) [5], which allows for the identification of multiple motifs in the same set of sequences in a single run [65].

Gibbs Sampling This is the most widely used motif discovery method [64] [36]. One of the reasons is that, unlike EM algorithm, being probabilistic [17]in nature avoids local minima easier. As it is common with stochastic methods, this algorithm has to be run many times, in other words, multiple searches starting from different random positions have to be done to confirm that the a motif is really present in the sequences.

Two well known implementation of this method are: AlignACE [42], that is suited to work on DNA regulatory sequences and ANN-Spec [89], which combines Gibbs sampling with neural networks by training a neural network to identify the binding sites, instead of using a PWM to represent and scores subsequences.

1.8 Can We Do Better?

After analyzing the actual situation regarding the prediction of TFBS, three main factors has lead us to think that the current methods for the computational prediction of TFBSs are falling short in predicting *bona fide* sites and that a different method can be useful:

1. The *Futility Theorem* as defined by Wasserman and Sandelin [85], tells us that*essentially all predicted TFBSs will have no functional role.*

2. Tompa *et al.* [84] reviewed thirteen different motif discovety tools and as conclusion they write that *no prediction method should be used alone and usually predictions from different algorithms and approaches result to be complementary.* Therefore, another tool which uses a different approach for TFBS prediction evaluation will be more beneficial than hurtful for the scientific community.

[17]Gibbs Sampling is a type of Markov Chain Monte Carlo (MCMC) algorithm, in essence is just a probabilistic variation of the EM algorithm

3. Computation prediction of TFBS and its consequent statistical evaluation is a very important task that will help us understand gene regulation: knowing more about how TF interact with the DNA will help us elucidate the complicated mechanism of gene regulation and protein production, which will open the new frontiers in many fields of medicine, giving us a better understanding of health and diseases.

In the lights of this reality, a new approach which statistically quantifies and evaluates genome-wide predicted TFBS seems to be an interesting addition. This thesis presents a new way to evaluate the genome-wide over-representation in TFBS predictions, using as working dataset the predictions previously published in Blanchette *et al.* [7].

1.9 Thesis Outline

Chapter 2 will introduce the theoretical foundations of two new methods to discover over-represented TFBSs. Chapter 3 discusses the inner details of the implementation of the approaches presented in Chapter 2. Chapter 4 presents the application of these approaches on various biological datasets, with their corresponding discussion. Finally, in Chapter 5 conclusions and future directions of the presented work are given.

Chapter 2

Quantifying Over-Representation of Transcription Factor Binding Sites Predictions

In the previous chapter, the importance of TFBS and TFBS predictions was introduced. In this chapter, two approaches to identify over-representated TFBS are discussed. Finally, results from the application of these approaches to two biological dataset are analyzed.

2.1 Problem Formulation

Defined in a formal way, the problem of quantifying the number of predicted TFBS in a set of regions of the human genome can be stated as follows: *Given*: A set of regions of the human genome. *Find*: The total number of TFBS that are predicted to be bound in these regions and say if the number of predicted TFBS differs from the number of predictions expected by chance in regions of the same size.

In simpler words the approach hereafter formulated gives answers to the following questions: How to know which and how many TFBS are in a given region of the human genome? Is a given TF bound to this specific set of regions of the human genome? How sure can we be that the observed results are not due to the effect of randomness?

The solution proposed to address these questions is an algorithm that determines the over-representation or under-representation of TFBS in a set of DNA sequences (regions of the genome), by comparing the number of predicted occurrences of a motif described by PWMs from TRANSFAC and JASPAR databases [7], to the number of occurrences that will be expected to be found by chance in random DNA sequences.

The proposed solution is partly based on the over-representation statistics defined in subsection 1.6.3 and works under the following considerations:

- Most of TFBS predictions we work with are false positives (previously expressed in the definition of the futility theorem in chapter 1).

- False positive predictions are distributed randomly in the genome.

- The user provides regions of human genome to explore:

  - If the motif is not present in the investigated regions, then the number of binding sites predicted for that TF follows the background probability.

  - If the motif is present in the the investigated regions, then the number of binding sites predicted will be larger than expected by chance, because there will be more real binding sites in these regions and as many false positives.

2.2 Binomial Over-Representation (BOR) Approach

The first way to approach our problem is a statistically simple one, where the probability of having a predicted TFBS can be described as a binomial variable $X_i = 0$, if there is no prediction at position i and $X_i = 1$, if there is a prediction at position i.

The total count N of predicted TFBSs in the genome can be described as:

$$N = \sum_{i=1}^{GenomeSize} X_i \tag{2.1}$$

where N include the overlaps of different TFBS and $GenomeSize$ is the total human genome size or $2.98X10^9$ basepairs.

If the site density is assumed to be independent of the position i, under our background model we get that the probability of having a prediction on position i is given by:

$$P[X_i = 1] = \rho = \frac{N}{GenomeSize} \tag{2.2}$$

Now consider a set of genomic regions of total length *RegSize.* Under the background model the total number of predictions is a random variable that we will call *S,* that follows a binomial distribution as:

$$P[S = s] = \begin{pmatrix} RegSize \\ s \end{pmatrix} \rho^s (1-\rho)^{RegSize-s} \tag{2.3}$$

The expected number of predictions and variance of that number *S,* are obtained as follows:

$$Exp[S] = RegSize * \rho \tag{2.4}$$

$$Var[S] = RegSize * \rho * (1 - \rho) \tag{2.5}$$

As the Binomial distribution is gaussian-like in shape, but discrete instead of continuous, when the expected number of predictions is *large enough*[1] then the binomial distribution approaches a normal distribution with: mean $\mu = RegSize * \rho$ and variance $\sigma = RegSize * \rho * (1 - \rho)$. This approximation is better explained by the Central Limit Theorem, which states that: *the sum of a sufficiently large number of independent random variables, each with finite mean and variance, will be approximately normally distributed* [61].

Given that the process of counting the number of predicted TFBSs in a genomic region becomes a normally distributed process we can make use of a statistical significance measurement, in this case the Z-score, to determine if the number of predicted sites are over-represented or under-represented given the background model.

As example, if we observe n_0 predicted sites, the Z-scores is given by:

$$Z = \frac{n_0 - Exp[S]}{\sqrt{Var[S]}} \tag{2.6}$$

2.2.1 Results of the Binomial Over-Representation Approach

Genomic regions obtained by Carroll et al. [16] for estrogen receptors (ERs) were used to test the effectivity of the binomial over-representation (BOR) approach. In their study 5,782 regions containing bona fide estrogen receptor and RNA polymerase II binding sites were located on a genome-wide scale via ChIP-chip experiments.

[1] Large enough is usually defined by the parameters of the model being build.

Methods and Motivation Estrogen receptors have an important function as regulators of gene expression[2]. In nature there exist two different forms of the estrogen receptor, usually referred to as α and β, each encoded by a different gene (ESR1 and ESR2 respectively). Each ER form is expressed in a different tissue type. For instance, ER-α is found in breast cancer cells [72], ovarian cells [68] and the hypothalamus [48] and ER-β has been found in the brain [63], heart [4] and lungs [57].

The binding specificity of ER is not very clear since they can interact with DNA directly via genomic sequences encoded by estrogen response elements (EREs) with motif GGTCAnnnTGACC or indirectly by attaching itself to nuclear proteins, such as AP-1 and Sp1 transcription factors [50].

As our binomial approach was formulated to specifically to find over-representation of TFBS in genomic regions and as we knew that the expected results (profile for ER factors, or ER like factors were going to have good Z-scores), this dataset showed to be a suitable candidate for using as a control dataset to test the efficiency of our method.

Results The Z-scores of each TF in TRANSFAC was evaluated for the regions described in Carroll et al. paper. These Z-scores had an unexpectedly wide range of -100 to 100 (see the histogram in Figure 2.1). Interestingly GC rich motifs obtained better Z-scores than their AT rich counterparts. Example of the Z-scores obtained for over-represented and under-represented motifs can be seen in Table 2.1 and Table 2.2, respectively.

Analysis of the Results After analyzing the results on the Carroll dataset doubts about the validity of the method were raised. Although the ER matrix was among

[2]ER-α has been identificated as a cause of breast cancer and as marker and therapeutic target for many other diseases

Table 2.1: Z-scores (greater or equal to 80.00) obtained by the binomial representation approach using the Carroll dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00470	AP-2gamma	111.53
M00469	AP-2alpha	107.45
M00649	MAZ	94.18
M01045	AP-2alphaA	88.73
M00959	ER	85.22
M00932	Sp-1	84.84
M00933	Sp-1	84.14
M00698	HEB	84.11
M01033	HNF4	82.43
M00915	AP-2	81.67

Table 2.2: Z-scores (less or equal to -80.00) obtained by the binomial representation approach using the Carroll dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00138	Oct1	-113.66
M00091	BR-C Z1	-95.82
M00012	CF2-II	-93.67
M00013	CF2-II	-93.31
M00094	BR-C Z4	-85.71
M00092	BR-C Z2	-85.44
M00713	TBP	-80.63
M00081	Evi-1	-80.20

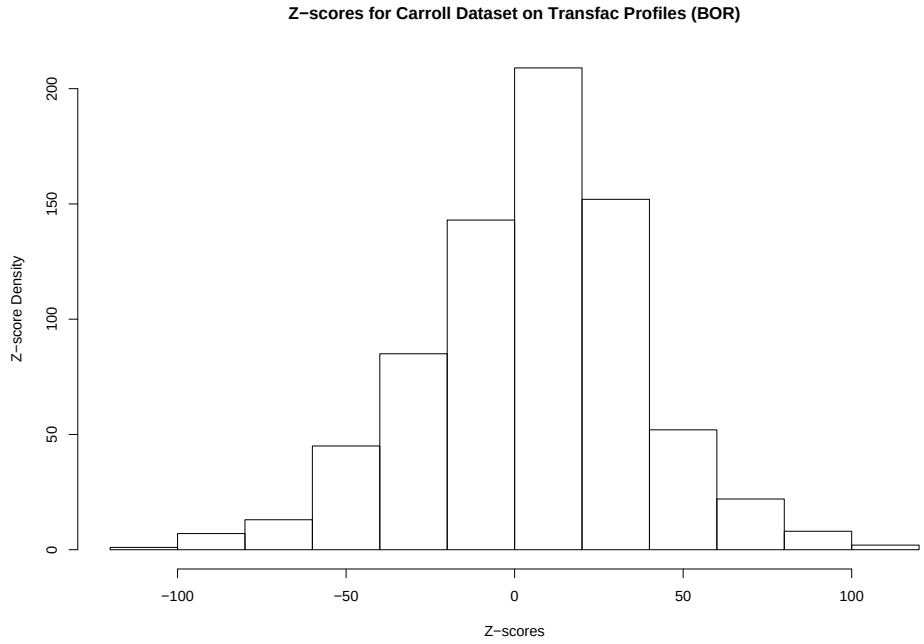

Figure 2.1: Histogram of Z-scores of the Carroll dataset over TRANSFAC predictions using the binomial approach

the top-scoring matrices, our method was giving better scores to matrices having GC rich motifs (a.e. GCCYNNGGS for the AP-2gamma factor) and poorer scores for AT rich and undetermined (N) motifs (a.e. NNNNNNNWATGCAAATNNNWNNA for the Oct1 factor).

Some reasoning about characteristics the human genome helped us realized that our method was being affected by an AT/GC bias. In part because the numerical difference between AT and GC bases in the human genome (41% GC bases and the rest 59% made of AT bases [23]) and in part because GC content of the regions in the dataset analyzed can be different from the GC content used as our background model, which is, the distribution of the GC content of the whole human genome. This later phenomenon can be explained better by using Figure 2.2. The red distribution represents the actual distribution of GC content in the human genome (centered around 41%), the blue distribution represents the distribution of GC content in a given dataset (centered around 60%). By simple inspection it becomes clear that the

two distributions are very different, hence using the whole genome as a background model is just as good as comparing apples and oranges.

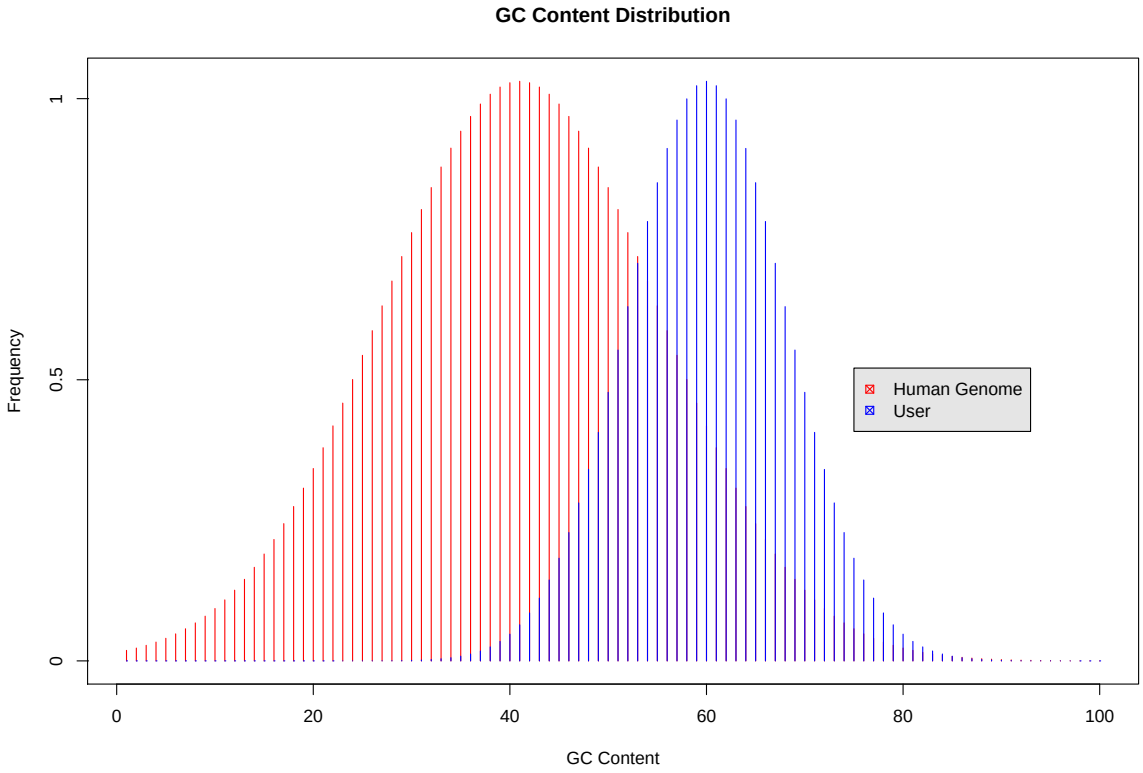

Figure 2.2: GC Content Distribution

Even though the first results obtained were not the expected, they shed light into a second approach that does not use the whole genome as a background model, instead, it tries to make the background model more accurate by comparing the user given dataset to the regions of the human genome which contains the same amount of GC content.

2.3 GC Content Stratified Binomial Over-Representation Approach

2.3.1 GC Content Calculation

The second approach here described is theoretically similar to the first one, however, it differs drastically in the way it creates the background model. As the GC Content of the genome varies considerably chromosome to chromosome, the only way to create a fair background model is to stratify (classify) the genome in groups according to their GC Content for each chromosome and then sample from each of these *stratums* independently.

In general to calculate the GC Content of a given region the following procedure is used.

1. Define non-overlapping substrings of the genome of size k (which in our approach we call *GC Windows*).
2. For every GC Window the number of G's and C's in them are quantified
3. Use the counts for every GC window and convert them into a single percentage that will represents the total GC content for that GC Window.

For example, suppose we want to quantify the GC content of a string $S = \{ACGGTNNGNNAATTN\}$ of length 15 and we let the GC window size be of $k = 5$. The GC content will be calculated by: First, creating three GC windows of size k=5 (as seen in the second column of Table 2.3). Then the number of G's or C's in each window is counted, multiplied by 100 and divided by the window length, resulting in the percentage[3] seen in the third column of Table 2.3.

[3]As there are many undetermined bases (N-bases) in the human genome and we are only control-

Table 2.3: GC Window Calculation - Substring Creation

Window	Substring	GC Content (%)
Window #1	ACGGT	(3*100)/5 = 60%
Window #2	NNGNN	(1*100)/1 = 100%
Window #3	AATNN	(0*100)/3 = 0%

Finally, the calculated GC content for string S is :

$$GC_S = [60, 60, 60, 60, 60, 100, 100, 100, 100, 100, 0, 0, 0, 0, 0]$$

It is important to mention that the window size of 100 was selected because of two main factors:

1. It was selected as a tradeoff between computation and practicality. By having windows of 100 the GC content of the regions can be somehow less representative that if we used a bigger window size, however we realized that the performance of the implementation (see Chapter 3) was similar to the BOR approach with this window size.

2. It was selected taking in account the size of TFBSs. As TFBSs are usually of a lenght between 5 and 20 basepairs, then we descarted the use of windows of small size like 10 or 50 basepairs.

2.3.2 Problem Re-formulation

This second approach takes in account the GC content of region analyzed, therefore, conceptually it answers a different question. This new question can be defined in

ling for GC content, then the length of Window #2, becomes 1 and length of Window #3, becomes 3. We can not penalize undetermined bases as having no GC content.

a formal way as: *Given*: A set of regions of the human genome. *Find*: The total number of TFBS that are predicted to be bound in these regions and whether the number of predicted TFBS differs from the number of predictions expected by chance in random regions with the same GC Content.

As in our first approach, the presence of a predicted binding site at position i can be described as a bernoulli variable $X_i = 0$, if there is no prediction at position i and $X_i = 1$, if there is a prediction at position i. Contrary to before, thought, we will not assume that the probability $P[X_i = 1]$ is independent of i, but rather it depends on the GC content of the region centered at i.

Let $gc(i)$ be the GC percentage rounded to the nearest percentage point, of the k basepairs window centered at i, let $GenomeSize(g)$ be the total number of bases of the human genome with GC content g, and let N_g be the total count of predicted TFBS in all regions of the human genome with GC Content g, then:

$$N_g = \sum_{i=1,\ gc(i)=g}^{GenomeSize} X_i \quad (\text{for } 0 \leq g \leq 100) \tag{2.7}$$

The background probability is given by the following formula:

$$P[X_i = 1 | gc(i) = g] = \rho_g = \frac{N_g}{GenomeSize(g)} \quad (\text{for } 0 \leq g \leq 100) \tag{2.8}$$

Under the background model s_g, the number of predictions in regions with GC content g follows a binomial process with sample size $RegSize(g)$ and probability ρ_g is given by:

$$P[S_g = s] = \begin{pmatrix} RegSize(g) \\ s \end{pmatrix} \rho_g^s * (1 - \rho_g)^{RegSize(g)-s} \tag{2.9}$$

where $RegSize(g)$ is the number of bases that have GC content g in the regions under consideration.

We then obtain the total expected number of predictions and total variance as follows:

$$Exp[S] = \sum_{g=0}^{100} Exp[S_g] = \sum_{g=0}^{100} RegSize(g) * \rho_g \tag{2.10}$$

$$Var[S] = \sum_{g=0}^{100} Var[S_g] = \sum_{g=0}^{100} RegSize(g) * \rho_g * (1 - \rho_g) \tag{2.11}$$

Finally the Z-score is then assigned as described in equation 2.6.

2.3.3 Results of the GC Content Stratified Binomial Over-Representation Approach

As with the first method the Z-scores for each TF profile in TRANSFAC were obtained and evaluated for the regions described in Carroll et al. paper. The newer Z-scores show a narrower range of -40 to 40 (see the histogram in Figure 2.3). GC rich matrices obtained positive scores, but their scores are not as positives as the ones from the first approach. The same happened to AT profiles, they obtained negative scores, but not as big as the ones from first approach. The Z-scores for over-represented and under-represented matrices can be seen in Table 2.4 and Table 2.5, respectively.

Table 2.4: Z-scores (greater or equal to 25.00) obtained by the GC content stratified approach using the Carroll dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00959	ER	43.89
M00926	AP-1	42.18
M00925	AP-1	40.45
M00174	AP-1	39.56
M00515	PPARG	38.04
M00470	AP-2gamma	37.06
M00469	AP-2alpha	34.72
M01045	AP-2alphaA	33.23
M00724	HNF-3alpha	32.74
M01033	HNF4	32.61
M00292	Freac-4	32.40
M00517	AP-1	32.26
M00191	ER	32.11
M00199	AP-1	31.11
M00289	HFH-3	31.05
M00156	RORalpha1	30.66
M00924	AP-1	30.17
M00727	SF-1	28.26
M00765	COUP direct repeat 1	26.65
M00035	v-Maf	25.27

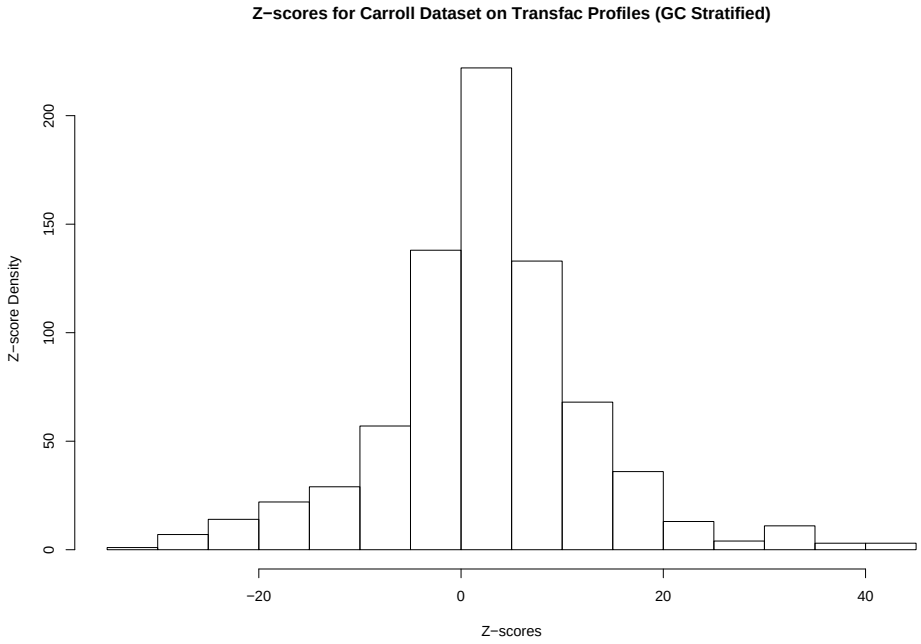

Figure 2.3: Histogram of Z-scores of the Carroll dataset over TRANSFAC predictions using the GC stratified approach

Analysis of the Results The results obtained look like what was initially expected. By changing the background model and correcting the GC Content ER matrices (M00959 and M00191) are present among the top scores. It is relevant to mention that AT/GC bias was successfully corrected and a non GC rich motif resulted as the top score (NAGGTCANNNY for factor ER) and a non AT rich motif resulted as a minimum score (WNTAATCCCAR for factor PITX2).

It is interesting to notice the fact that the results also give good scores to members of the Activator Protein transcription factor family (AP) and the Hepatocyte Nuclear Factors/Forkhead transcription factor family (HNF/Fox). AP-1 is known to interact with ER to promote transcription [16] and FoxA1/HNF-3Alpha plays a central role in ER signaling [52]. Even more important is to notice that other factors among the top scores are COUP, RoRAlpha1, Freac-4, HNF4 and PPARG, which are nuclear factors with similar binding affinity as ER.

Table 2.5: Z-scores (less or equal to -25.00) obtained by the GC content stratified approach using the Carroll dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00482	PITX2	-32.73
M00012	CF2-II	-29.89
M00092	BR-C Z2	-29.79
M00013	CF2-II	-29.32
M01048	Tra-1	-27.01
M00022	Hb	-26.67
M00305	HAP1	-25.80
M00706	TFII-I	-25.79

2.3.4 Analysis of Another Estrogen Receptor Dataset

Following the successful results on the Carroll *et al.* dataset, a side test was done on another ER dataset with the hopes of recovering the same ER factors resulted with top scores with the Carroll dataset or finding new results which showed other factors related to the ER factor.

Lin *et al.* [50] mapped Estrogen receptor alpha binding sites in MCF-7 breast cancer cells by using a ChIP-PET technique[4] and were able to identify 1,234 novel regions that contain ER-α binding sites.

This paper shed light into some important features of ER, for instance, that ER-α can influence the expression of a gene in distances of up to 100 kilobases or more.

[4]ChIP-PET is a combination of Chromatin immunoprecipitation (ChIP) and a newer specialized cloning techniques and vectors, called paired-end diTags (PETs)

Analysis of the Results for the ER on MCF-7 Dataset The Z-scores for over-represented (positive Z-scores over 12.00) of the Lin dataset on TRANSFAC predictions can be seen in Table 2.6.

Table 2.6: Z-scores (greater or equal to 12.00) obtained by the GC content stratified approach using the Lin dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00515	PPARG	39.31
M00191	ER	30.41
M00959	ER	26.79
M00926	AP-1	19.63
M00925	AP-1	19.14
M00174	AP-1	18.84
M01045	AP-2alphaA	18.27
M00469	AP-2alpha	18.02
M00156	RORalpha1	17.31
M01032	HNF4	14.00
M00204	GCN4	13.84
M00727	SF-1	13.50
M00199	AP-1	13.18
M00517	AP-1	13.03
M00292	AP-1	12.92
M00511	ERR alpha	12.57
M00724	HNF-3alpha	12.00

The results obtained were as expected, ER matrices (M00959 and M00191) are present among the top 3 scores. Also, similar to the Carroll results, the Activator Protein transcription and the Hepatocyte Nuclear Factors/Forkhead transcription factor families are recovered (HNF/FoxA), as well as some nuclear receptors as RAR-related orphan receptor alpha (ROR-Alpha), PPARG (peroxisome proliferator-activated re-

ceptor gamma) and Estrogren related receptor (ERR).

2.4 Summary of the Chapter

The decision of using the Carroll et al. dataset as a control dataset to test the efficiency of our method was crucial. A GC/AT bias was discovered and forced us to correct the method. The second approach (GC stratified approach) gave the expected results, recovering profiles related to ER with high Z-scores on two different datasets.

In the next Chapter details of the implementation of this method in the C language are presented and various other datasets analyzed using the implementation are discussed in chapter 4.

Chapter 3

Computational Challenges and Method Implementation

In the previous chapter, two approaches to identify over-represented TFBS were presented and results when applied to two sets of genomic regions were shown. In this chapter, inner details of the implementation of these methods are given.

3.1 Computational Challenges

Implementing the Binomial Over-Representation approach and the GC Stratified approach were challenging tasks because there is a large amount of data that has to be processed by the program in order to compute a Z-score. The program has to read the set of genomic regions to be analyzed and the compare to the set of genome-wide *predicted TFBS* selected by the user. The set of genomic regions can be read and validated easily, but, reading and processing the genome-wide TFBS predictions is non-trivial.

Predictions for every single TFBS in TRANSFAC and JASPAR (around 1000 TFBS) exist as 48 textfiles (for every chromosome there exist predictions on each DNA strand: forward and backward) of more then 80,000 lines. In the worst case (if the user decides to investigate all TFBS in TRANSFAC and JASPAR) the program has to be able to read $24 \times 2 \times 80,000 \times 1,000 = 3,840,000,000$ lines and then calculate the Z-score in the fastest possible way, that is, trying to have a processing time of less than a minute.

3.2 Method Implementation

As in any computer system, the implementation was divided logically in three main parts: the program inputs (described in Section 3.2), the processing of the inputs (described in Section 3.3) and the outputs (described in Section 3.3). Figure 3.1 give a complete view of the system clearly identifying the three principal parts of the implementation: the inputs (in green), the Z-score calculations (in yellow), the resulting outputs (cyan) and the supporting data structures that were created to make the process fast (in red and orange).

3.2.1 Program Inputs

The program receives as input paths to three directories. The first one is a path to the directory where the file containing the regions of the human genome to be analyzed is located. The second is a list of paths to TFBS predictions for every TF the user decides to investigate in the genomic regions to be analyzed. The third is a path to the GC files, which are the files that contain the GC Content for every chromosome of the human genome calculated via the GC Window calculation method introduced in section 2.3 in windows of size 100 basepairs.

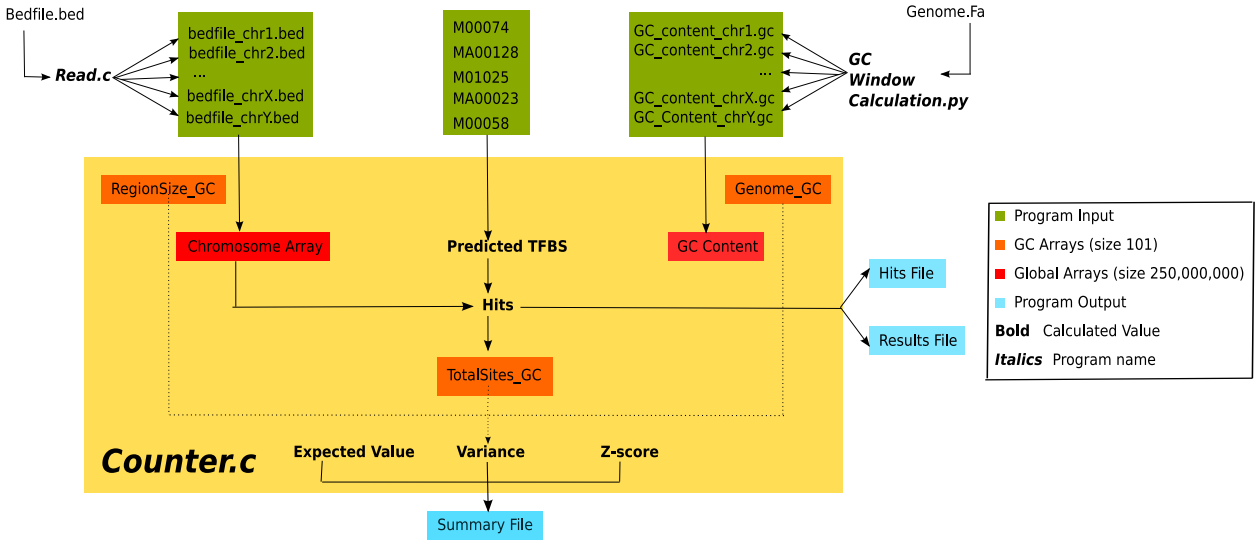

Figure 3.1: Diagram of the main parts of the implementation (functions, variables and files)

File containing genomic regions The file containing the regions to be investigated should be formated in bedfile format (with extension .bed), in order to be processed. A bedfile is just a tab separated text file formatted in three colums, in which the first column defines the chromosome to which the regions to be analyzed belongs, the second and third define the position where this region start and ends, respectively. Hereafter bedfile and set of genomic regions will be used indistinctively.

As the number of regions specified in the bedfile that can vary from dataset to dataset and this number can be sometimes very big, a pre-processing program was implemented in order to save the time. This program is called *Read.c*, it reads the befile once and divides it in 24 smaller files (one for every chromosome), which are read individually in the main program. In this way, the time consumed in opening and reading a large file line by line for each chromosome, is transformed in the task of reading 24 smaller files, thus avoiding a linear search. This process is depicted in the top left corner of Figure 3.1.

Path to predicted TFBS As explained in section 3.1, there exist 48 prediction files for every TFBS, therfore, for every TF that the user wants to investigate the presence in the genomic regions for interest, it is necesary to read each file once for every, this process is detailed later in section 3.2.2.3. The number of profiles to be compared can also vary from dataset to dataset. To solve this, each profile is passed to the main program as a different argument. More about this solution will be explained in next section *Running the Implementation in Parallel*.

Path to GC Content files The files containing the GC content of the human genome were calculated for every chromosome in windows of 100 bases. These files are just text files with one column representing the percentage of GC content in a size 100 basepairs region. More on how this files are used to calculate the background model is discussed in section 3.2.2.1.

3.2.2 Z-Score Calculation (Processing of the Inputs)

The processing of the inputs is the central part of the implementation. It is where the set of genomic regions is processed and Z-scores are calculated. This process can be divided in four subtasks (or procedures if we want to refer to the program organization):

1. Loading the GC Content file for every chromosome into memory to be able to calculate the background model.

2. Reading the regions described in the bedfile and loading them into memory.

3. Counting the number of predicted TFBS that are in the regions provided by in the bedfile.

4. Statistically assess the counts obtained from reading the predictions files and calculating the Z-Score.

3.2.2.1 Loading the GC Content

As the background model is essential for the calculation of the expected value and variance, in our approach, the GC content of every chromosome was precomputed (following the formulas specified in section 2.3) in windows of size k=100 and saved to files that we call *GC files*.

To speed up the calculations these GC files are loaded into memory to an array we call the *GC content array*. The GC content array is a data structure in which every position represent a nucleotide of a chromosome, thus, a data structure large enough to accommodate the GC content of every nucleotide of the largerst chromosome, namely, chromosome 1 with its 247 million nucleotide base pairs (this translate to have in memory an array of 250mb). When loading the GC content to the GC content array, every position of the array is assigned the calculated GC content in a similar procedure to the one shown in Algorithm 1.

3.2.2.2 Loading the Bedfile Into Memory

Each one of the genomic regions specified in the bedfile (which are separated for each chromosome created by the *Read.c* program) is read and marked into an array similar in size and purpose to the GC Content array, that we call *Chromosome Array* (See Algorithm 2).

In detail, when loading the regions to the Chromosome Array each region is first given an integer identifier (as seen in Algorithm 2 line 5). This region identifier is used to distinguish nucleotides belonging to different regions. Secondly each region is

Algorithm 1 Load the GC Content to the GC content array

Require: chr, GC_file

Output: an array containing the GC content for chromosome chr

1: $GC\_Content\_Array[max\_chr\_size] \leftarrow 0$ /*clears the contents of the array*/
2: $start \leftarrow 0$
3: $end \leftarrow 100$
4: **for** $line = 1$ to EOF (for every line in the GC_file) **do**
5: $content \leftarrow GC\_chr[line]$
6: $Genome\_GC \leftarrow content$ /*Genome_GC is a counter that contains the total number of positions of the genome with GC content g, where g can be 0-100*/
7: **for** $position = start$ to end **do**
8: $GC\_Content\_Array[position] \leftarrow content$
9: **end for**
10: $start \leftarrow end$
11: $end \leftarrow end + 100$
12: **end for**
13: **return** $GC\_Content\_Array, Genome\_GC$

loaded into the chromosome array. In other words, every base in the regions described in the bedfile are marked in the Chromosome Array by its region identifier (as seen in Algorithm 2 line 10).

Algorithm 2 Load the regions specified on the bedfile to the Chromosome array

Require: $bedfile$

Output: an array containing the regions of chromosome chr to be explored

Output: total number of regions to be explored

1: $Chromosome\_Array[max\_chr\_size] \leftarrow 0$ /*clears the contents of the array*/
2: **for** $line = 1$ to EOF (every line in bedfile) **do**
3: **if** $valid(bed[line])$ **then**
4: /*every line of the file is validated to be correctly formated, which means that it has three columns and that the region_end is bigger than region_start*/
5: $region\_counter \leftarrow region\_counter + 1$
6: $region\_start \leftarrow bed[line][2]$
7: $region\_end \leftarrow bed[line][3]$
8: $content \leftarrow line$
9: **for** $position = region\_start$ to $region\_end$ **do**
10: $Chromosome\_Array[position] \leftarrow region\_counter$
11: $content \leftarrow GC\_Content\_Array[position]$
12: $Region\_Size\_GC[content] \leftarrow Region\_Size\_GC[content] + 1$ /*Region_Size_GC is a counter that contains the total number of positions of the genome with GC content g, where g can be 0-100*/
13: **end for**
14: **end if**
15: **end for**
16: **return** $Region\_Size\_GC, Chromosome\_Array, region\_counter$

3.2.2.3 Counting the Number of Predicted TFBS

Once the genomic regions to be investigated have been marked into the Chromosome Array, the program proceeds to read the TFBS predictions for the selected profile(s) from TRANSFAC or JASPAR. If a predicted binding sites is found in one of the regions specified in the bedfile, a counter that contains the total number of observed *hits* for that region is increased by one (see Algorithm 3).

3.2.2.4 Calculation of the Expected Number of Hits, Variance and Z-score

One way to estimate the significance (over/under-representation) of the number of observed *hits* in the genomic regions being investigated is to compare it to a similar -but theoretical- model using the formulas described throughly in Sections 2.2 and 2.3.2.

In order to calculate the expected value and variance of the theoretical model is necessary to have an estimate of size, GC Content and number of hits of the genomic regions investigated. In order to have this estimates we make use of three data structures of size 101 (0 to 100) which we call *GC arrays*, namely they are the they are the Genome_GC, the Region_Size_GC and the Total_Sites_GC arrays which introduced earlier in sections 3.2.2.1, 3.2.2.2 and 3.2.2.3 respectively.

Expected Number of Hits and Variance After the predicted TFBS are read and all the GC Arrays contains the total counts for a given chromosome, the expected value and variance are calculated by summing over the different GC contents (0-100) using the formulas described in Equations 2.10 and 2.11 (see Algorithm 4).

Algorithm 3 Count the number of predictions found in regions specified by the user

Require: list of matrix files to read, chromosome

Require: total number of regions for chromosome chr

Output: the number of predicted sites in regions specified in the bedfile

1: **while** $M \leq total\ number\ of\ matrix\ to\ read$ **do**
2: /*this while cycle is executed twice, one for predictions in the forward strand and one for predictions on the backward strand*/
3: **for** $line = 1$ to EOF (every line in the prediction file) **do**
4: $site = mat[line]$
5: $position \leftarrow Chromosome\_Array[site]$
6: $content \leftarrow GC\_Content\_Array[site]$
7: $Total\_Sites\_GC[content] \leftarrow Total\_Sites\_GC[content] + 1$ /* The Total_Sites_GC array is a counter that contains the the total number of predicted sites in the genome with GC content g, where g can be 0-100*/
8: **if** $(position \neq 0)\ and\ (position \leq region\_counter)$ **then**
9: $hits\_for\_region[position] \leftarrow hits\_for\_region[position] + 1$
10: $total\_hits\_GC[content] \leftarrow total\_hits\_GC[content] + 1$
11: **end if**
12: **end for**
13: $M = M + 1$
14: **end while**
15: **for** $x = 1$ to $region\_counter[chr]$ **do**
16: $hits\_counter \leftarrow hits\_counter + hits\_for\_region[x]$
17: **end for**
18: $total\_hits\_for\_chr[chr] \leftarrow hit\_counter$
19: **return** $total\_hits\_for\_chr[chr], Total\_Sites\_GC$

Z-score After all chromosomes have been read and every expected value and variance have been computed, they are added in to one global expected value and variance value, which is used along with the total number of hits for the bedfile to calculate the final Z-score of the whole set of genomic regions investigated.

Algorithm 4 Calculate the expected value, variance and Z-score

Require: $chromosome, windowsize k$

Output: Z-score, expected value and variance for chromosome chr

1: **for** $g = 0$ to 100 **do**
2: $exp\_GC \leftarrow \frac{total\_sites\_GC[g]}{genome\_GC[g]} * region\_Size\_GC[g]$
3: $var\_GC \leftarrow exp\_GC * (1 - \frac{total\_sites\_GC[g]}{genome\_GC[g]*k})$
4: $exp\_GC\_chr[chr] \leftarrow exp\_GC\_chr[chr] + exp\_GC$
5: $var\_GC\_chr[chr] \leftarrow var\_GC\_chr[chr] + var\_GC$
6: **end for**
7: $Z[chr] \leftarrow \frac{total\_hits\_for\_chr[chr] - exp\_GC\_chr[chr]}{var\_GC\_chr[chr]}$
8: **return** $exp\_GC\_chr[chr], var\_GC\_chr[chr], Z[chr]$

3.2.3 Main Program

All the four procedures above described are connected via a main program. In this program all the global variables (for instance, the GC Arrays and the GC Content and Chromosome arrays) are declared and instructions for the input and output are processed. The Algorithm 5 shows a high level structure of the main program which is depicted in Figure 3.1 as a yellow rectangle (*counter.c*).

Algorithm 5 Main Program

Require: $bedfile, prediction\_file, GC\_files$

1: $max\_chr\_size \leftarrow 250,000,000$
2: $Genome\_GC[100], Region\_Size\_GC[100], Total\_Sites\_GC[100]$
3: $Chromosome\_Array[max\_chr\_size], GC\_Content\_Array[max\_chr\_size]$
4: $GC\_window\_size \leftarrow 100$
5: **for** $chr = 1$ to 24 **do**
6: $load\_GC(chr)$
7: $load\_Bedfile(bedfile\_chr)$
8: $count\_Predictions(prediction\_file)$
9: $write\_Hits\_file()$
10: $write\_Results\_file()$
11: $calculate\_Z{-}score(total\_hits\_for\_chr[chr], total\_exp\_GC[chr], total\_var\_GC[chr])$
12: $total\_hits \leftarrow total\_hits + total\_hits\_for\_chr[chr]$
13: $total\_exp \leftarrow total\_exp + total\_exp\_GC[chr]$
14: $total\_var \leftarrow total\_var + total\_var\_GC[chr]$
15: **end for**
16: $total_$Z-score $= \frac{total\_hits - total\_exp}{sqrt(total\_var)}$
17: $write\_Summary\_file()$

3.3 Program Output

Three files contain the results computed at various step of the Z-score calculation described in the previous section. First, a file called *Results file* contains the total number of hits for each region of the bedfile for a given TFBS. Secondly, a complementary file to the Results file is the *Hits file*, which contains in which position and strand every hit is found. The third file, that we call the *Summary file* contains a summary of the statistical significance calculation (calculation of expected values, variance and Z-score) chromosome by chromosome and globally for the entire genome.

Once the results files are created, a set of scripts are executed in order to post-process the output files into a human readable file that is easier to open as a spreadsheet or matrix in any technical computing software. For instance, the result file is formated into a tab separated file, a (.res) file, which contains regions as rows, selected matrices as columns and number of hits as the intersections. Also, the summary file is formated into a tab separated file, a (.sum) file, in which the information of the summary file (observed hits, expected number of hits, variance and Z-score) appear as columns, the selected matrices appear as rows and the intersection of both shows the consequent information for each TRANSFAC or JASPAR profile specified for comparison.

3.4 Running the Implementation in Parallel

As explained in section 3.2.1, the program receives as input a list of paths to TFBS predictions for every TF the user decides to investigate in the genomic regions to be analyzed, therefore the program have to be executed as many times as selected TFBS are in the list of paths. In order to speed up these calculations, we decided to make use of a simple parallelization technique by using a computer cluster, passing every

path to the TFBS predictions as an argument the counter.c program, thus making each *run* an independent job (one job do not interact with the others) and queueing them to the cluster as jobs.

The cluster is in charge of queuing, distributing the task among its nodes, processing and then collecting all the resulted files (described in the previous section) without human intervention and transparently to the execution of the main program.

3.4.1 Running Time

For calculating an estimate of the running time of our implementation a simple script which counts the number of jobs in queue in the cluster at given intervals was developed. A data set containing 41,582 regions of interest identified by Robertson *et al.* [70] for the STAT1 TF using ChIP-sequencing (ChIP-seq) and massively parallel sequencing was used for this calculation. Using these regions and comparing them to predicted binding sites for 60 Factors from TRANSFAC took about 3 minutes, which is considerably fast considering the quantity of regions of this data set.

It is important to mention that the performance of our implementation varies greatly with the number of genomic regions being investigated, the number of TFs to be compared and with the number of jobs already in the cluster at the moment of queuing the jobs, so for some data sets it is no wonder to have results under 45 seconds.

3.5 The Web Based Front End

Since the start of the project one of its goals was to make the implementation available for public use. Once the implementation was completed and preliminary results

showed that the tool was working correctly, the next step was to make the tool available to the rest of the scientific community. Given that the tool was implemented in C language, which is not a portable language, we decided to make a web based front end written in the PHP language and called it **GATOR**, which is an acronym for: **G**enome-wide **A**nalysis of **T**FBS **O**ver-**R**epresentation.

3.5.1 Architecture of the Front End

Even though at first the implementation of the front end seemed pretty trivial, the fact that the whole implementation relied on the cluster architecture to produce faster results made the development of the front end difficult. Part of the problem is that the webserver (where the front end runs) and the cluster main node (were the program runs) are different computers, therefore, to be able to activate the execution in the master node, a network connection (secure shell) had to be made from the PHP script (the front end) to the master node.

Figure 3.2 give a complete view of the system, including both actors, the Webserver (front end) and the master node of the cluster. In this picture the four main parts of the front end are identified, including the connection to the cluster and the creation of results.

The architecture of the front end is quite simple. It consist of the four main parts (scripts):

1. Homepage (home.php): the homepage is the first page the user see when he enters the GATOR website. It has important information about the tool, how to use it and how to contact the webmaster for references and help. More important, it has a fillable form [1] that the user has to fill in order to use the

[1] Validations of this fillable form are made in the javascript language

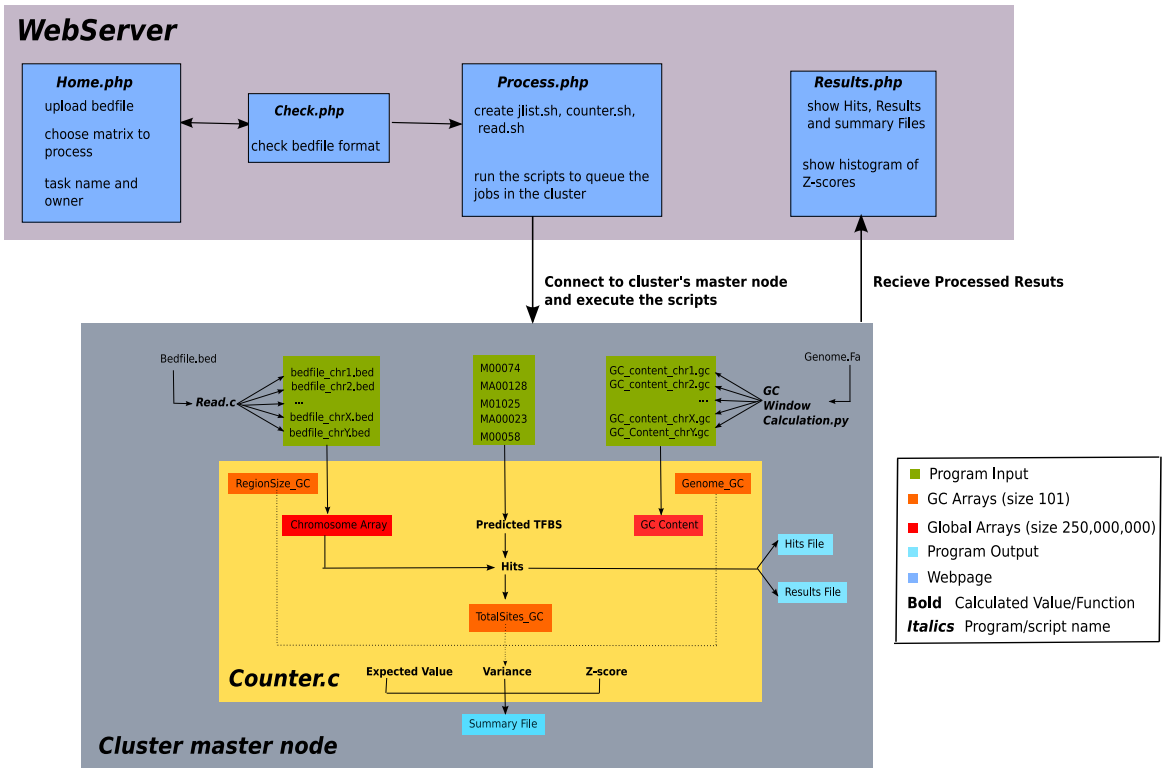

Figure 3.2: Diagram of the main parts (programs, functions, variables and files) of the web implementation

tool. This form consist in the following fields: the upload of genomic regions or genes to be analyzed, matrix selection (where the user chooses which profiles he wants its dataset to be compared to) and identification field (where the user gives a name to the task and leaves a valid email address to which the results will be send). Once the fields are validated the data is sent to the next script.

2. Validation (check.php): In this step the bedfile is validated, it is checked to be in the correct format. If the file is invalid a message will appear on the screen advising the user to upload a valid file formated in the three column format. If the file is valid the data is sent to the next script.

3. Process the input (process.php): in this step the user input from home.php is

used to create the job scripts which contains the instructions to be executed in the main node. For instance they have information about the job, which TFBS are to be compared and where the uploaded bedfiles resides.

Once the scripts files are produced a SSH call is made by the process.php script to connect to the master node of the cluster in order to execute the activator scripts. In this moment the webserver give the control to the cluster, which takes as much time as needed to process the data. Once the cluster is finished a flag is set and the .sum and .res files (which where introduced in Section 3.4) are created.

4. Results (results.php): this is the final step of the process in which the user is notified via email that his bedfile has been analyzed and his results are ready to be downloaded in a specific URL location.

3.6 Summary of the Chapter

The decision of implementing this tool in the C language was in part forced by the fact that other languages, such as Python and Java, were not as flexible in terms of memory management as C is. The memory issue became critical with the introduction of the GC Arrays, which, combined with the Genome Wide arrays, give the program a memory footprint of more than 1Gb of memory.

The access to the cluster has been an important element of the implementation, giving the flexibility to have results in minutes for simple datasets and moreover, extending the capacity of the tool to be shared with the scientific community via a webserver.

In Chapter 4, three datasets are presented and the results of their execution in our C implementation are described and analyzed in detail.

Chapter 4

Analysis of Over-Represented TFBS in Different Biological Contexts

The previous chapter presented details how the methods described in chapter 2 were implemented. In this chapter, results of the application of the approach to three different datasets are discussed in details. In detail, we discuss results from tests of applying our method to one validation set, which is a randomly generated dataset and to two biological relevant sets of genomic regions, the Ang-1 and CRUNCS datasets.

4.1 Analysis of the CRUNCS Dataset

Approaches that use comparative genomics are frequently used to identify conserved regions among different species, these methods can be extended to identify conserved regulatory regions on non-coding regions (introns) and on coding regions (exons). Interestingly enough, there are functional elements as transcription factor bindings sites, that can be located within exons, for instance: some transcription factor binding sites, exonic splicing enhancers and RNA secondary structure elements affecting

mRNA stability, localization, or translation [20]. Chen and Blanchette [20] identified 8785 of these regions within coding regions (exons) in the human genome and named them **C**oding **R**egions **U**nder **N**on-**C**oding **S**election or CRUNCS for short. Table 4.1 describe the number of CRUNCS regions found by Chen and Blanchette for each chromosome.

Table 4.1: Number of Regions per Chromosome for the CRUNCS Dataset

Chromosome	Number of Regions	Chromosome	Number of Regions
1	937	13	149
2	851	14	389
3	546	15	324
4	438	16	251
5	470	17	569
6	493	18	169
7	333	19	82
8	333	20	143
9	297	21	60
10	511	22	124
11	385	X	439
12	489	Y	3

4.1.1 Motivation

Little is known about CRUNCS but work from Mayhew and Blanchette [56] suggest important characteristics of these regions for instance: that CRUNCS bases are more often found near coding exon edges than in middle coding exons; that CRUNCS-containing genes are significantly enriched for regulation of transcription and translation, protein ubiquitination, mRNA processing and gene splicing regulation and that CRUNCS are significantly enriched for RNA secondary structure elements.

As the previous results above mentioned suggested that TFBSs can be found in CRUNCS regions and because our approach was formulated specifically to find over-representation TFBS in genomic regions, we decided to test the CRUNCS dataset in order to find interesting TF over-represented in these regions.

4.1.2 Methods

Results of scoring the CRUNCS dataset against predicted TFBS from TRANSFAC using our implementation can be seen in the form of an histogram in Figure 4.1.

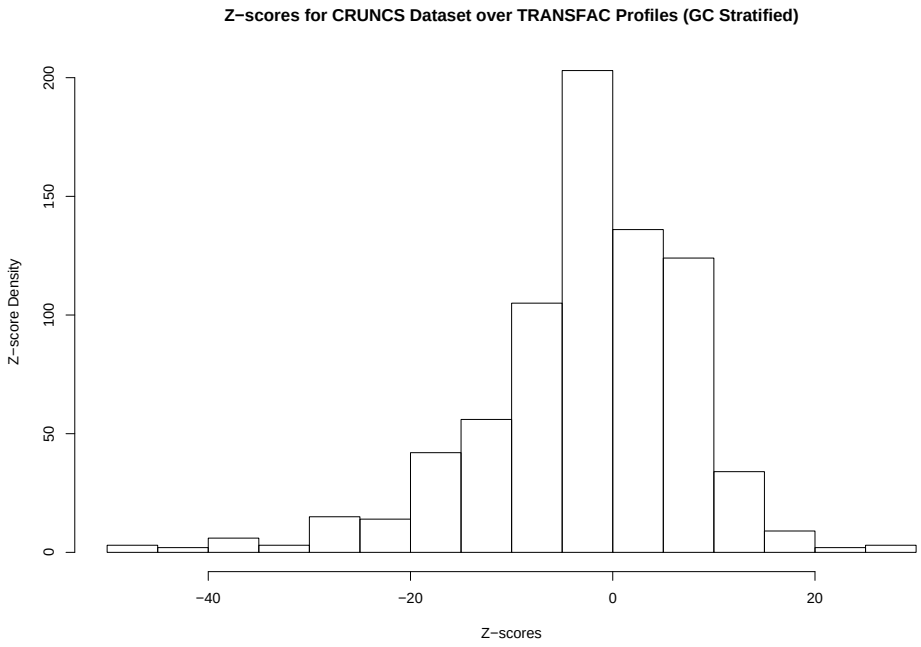

Figure 4.1: Histogram of Z-scores of the CRUNCS dataset over TRANSFAC predictions using the GC stratified approach

Some TFs as LBP-1, Adf-1, MATa1 and HEB, showed an interesting positive enrichment, the complete list of over-represented factors (Z-scores for factors that obtained a score above 15.00) can be seen in Table 4.2. Also, a list of under-represented factors can be seen in Table 4.3.

Table 4.2: Z-scores (greater or equal to 15.00) obtained by the GC content stratified binomial over-representation approach using the CRUNCS dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00644	LBP-1	26.84
M00171	Adf-1	26.33
M00923	Adf-1	26.15
M00030	MATa1	24.36
M00698	HEB	22.02
M00927	AP-4	19.85
M01057	ERF2	19.29
M00801	CREB	17.63
M00374	D-Type LTRs	17.10
M00226	P	16.95
M00106	CDP CR3+HD	16.51
M00993	TAL1	16.44
M00017	ATF	15.69
M00683	XBP1	15.26

Table 4.3: Z-scores (less or equal to -30.00) obtained by the GC content stratified binomial over-representation approach using the CRUNCS dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00130	FOXD3	-48.77
M00022	Hb	-48.19
M00091	BR-C Z1	-48.16
M00791	HNF-3	-43.30
M00092	BR-C Z2	-40.72
M00081	Evi-1	-38.98
M00972	IRF	-36.93
M00456	FAC1	-36.90
M01021	ID1	-36.18
M01010	HMGIY	-35.28
M01012	HNF3	-35.24
M00094	BR-C Z4	-32.82
M00422	FOXJ2	-31.90
M00809	FOX	-30.35

Analysis of the Preliminary Results Preliminary results showed significant over-representation of four transcription factors: Lipid Binding Protein (LBP-1), Adh transcription factor 1 (Adf-1), methionine adenosyltransferase I, alpha (MATa1) and transcription factor 12 (HEB). A closer look at the factors revealed interesting characteristics. For instance, other than the two Adf-1 factors, no factors have common motifs (as seen Table 4.4). Also a search in literature did not showed any significant relationship between the factors.

Table 4.4: Top five over-represented TF found in CRUNCS regions (preliminary results)

Transcription factor	Consensus sequence	Organism	Z-score
LBP-1	CAGCTGS	Human	26.84
Adf-1	VCGCYGCMGYCGCTGMCNGCG	Drosophila	26.33
Adf-1	CCGCYGC	Drosophila	26.15
MATa1	TGATGTANNT	Human	24.36
HEB	RCCWGCTG	Human	22.02

4.1.3 GC Window Correction for CRUNCS Dataset

As the preliminary results of applying our methods to CRUNCS regions were inconclusive and as we were motivated to find interesting results which gave us inside knowledge of the nature of CRUNCS, a variation of our method done specially to analyze CRUNCS regions was conceived.

The main idea behind this new approach is to take in account an essential characteristic about the CRUNCS, they are found in *coding regions*. With this fact in mind, a variation in the background calculation was introduced. Instead of taking the GC content of the whole genome (stratified in windows of 100 bp) to calculated our background model, we only take in account the GC content of regions that are known as coding regions. To this end, a file containing the coding regions of the

human genome was downloaded from the UCSC Genome Browser [46] and a program was implemented to write the GC content of coding regions.

Calculating the GC Content of Coding Regions The procedure to create the background model for coding region is similar to the one described in Section 2.3. In fact, as we had calculated the genome-wide GC Content on increments of $k = 100$ bases and saved them to the GC files (Section 3.3.1), we thought that it was just a matter of deciding to which regions we assigning the previously calculated GC content and to which regions we set to a content of 0.00. However, since the regions specified in the coding regions file does not follow the same 100 bases incremental structure and instead followed a different pattern, a new GC Content calculation had to be formulated.

For instance, assume that a coding region starts in position 175 and spans for 400 bases until position 575, then there are five GC Content windows that are covered by this region: 100 to 200, 200 to 300, 300 to 400, 400 to 500 and 500 to 600; then the question becomes how to define which regions should be in the background without losing precision? We tried two ways:

1. To keep only the GC Content of coding regions which were completely covered by a window, which in our example will be the following windows: 200 to 300, 300 to 400 and 400 to 500. However, this resulted in a poor background model in which most the windows had content of 0.00.

2. To consider all the windows that had at least 1 base of coding region, which in our example will be all the windows between 100 and 600 bases. In other words, we mapped the coding region start to the start of the window it falls in and the end of the coding region to the end of the window it falls in and created what we define as *mapped coding region start* and *mapped coding region end*.

These two calculations can be done mathematically by the use of the ceiling and flooring functions as shown in Equations 4.1 and 4.2.

$$Mapped\_Region\_Start = \lfloor \frac{Coding\_Region\_Start}{k} \rfloor * k \tag{4.1}$$

$$Mapped\_Region\_End = \lceil \frac{Coding\_Region\_End}{k} \rceil * k \tag{4.2}$$

Analysis of the GC Content Corrected Results for CRUNCS Regions Interestingly, if we compare the preliminary results (seen in Figure 4.1) with the results obtained with the newer background model (seen in Figure 4.2), it is easy to see that the majority of the Z-scores are between -10 to 10 or basically became more centered around zero. Another interesting difference found in the results with the newer background model is that the magnitude of Z-scores is smaller, with a range of -30 to 20, instead of -60 to 20. A list of over-represented Z-scores over 8.00 can be seen can be seen on Table 4.5 and a list of under-represented factors with Z-scores less than -20.00 can be seen in Table 4.6.

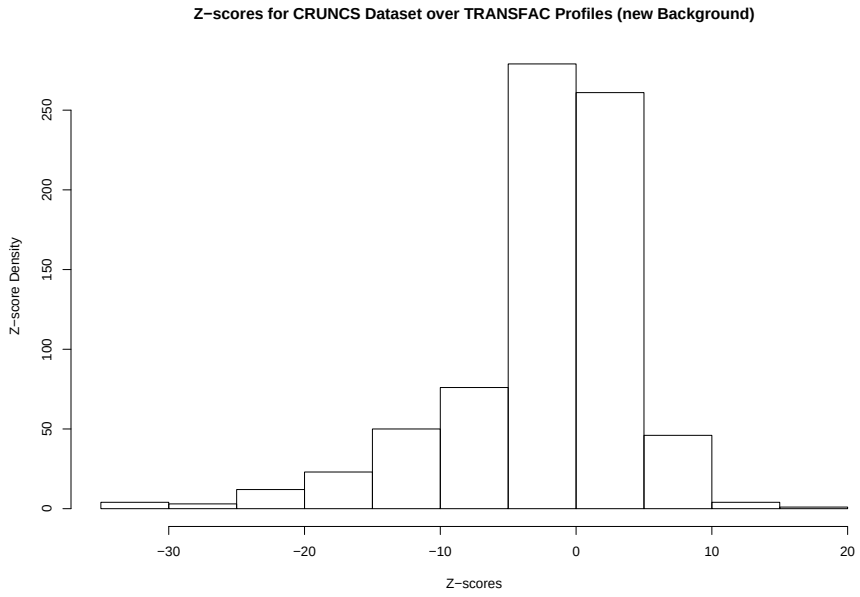

Figure 4.2: Histogram of Z-scores of the CRUNCS dataset over TRANSFAC predictions using the GC stratified approach with a corrected background (only coding regions)

Table 4.5: Z-scores (greater or equal to 8.00) obtained by the GC stratified approach with a corrected background using the CRUNCS dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00171	Adf-1	15.62
M00923	Adf-1	12.45
M00698	HEB	11.43
M00030	MATa1	10.75
M00644	LBP-1	10.72
M00374	D-Type LTRs	9.84
M00106	CDP CR3+HD	9.03
M00683	XBP1	8.73
M00927	AP-4	8.69
M01017	PBX1	8.55
M00993	TAL1	8.47

Table 4.6: Z-scores (below or equal to -20.00) obtained by the GC stratified approach with a corrected background using the CRUNCS dataset over TRANSFAC profiles

Matrix ID	Factor	Z-score
M00091	BR-C Z1	-31.43
M00094	BR-C Z4	-30.68
M00022	Hb	-30.24
M00972	IRF	-28.09
M00081	Evi-1	-28.04
M00791	HNF-3	-26.26
M01021	ID1	-23.91
M00092	BR-C Z2	-23.62
M00422	FOXJ2	-23.51
M01010	HMGIY	-22.93
M01012	HNF3	-22.06
M00268	XFD-2	-21.88
M00713	TBP	-21.75
M00809	FOX	-21.73
M00131	HNF-3beta	-21.52
M00456	FAC1	-21.17
M00138	Oct-1	-20.89
M01011	HNF1	-20.59

4.1.4 Analysis of Results for the CRUNCS Dataset

Even after changing the background, the factors for Lipid Binding Protein (LBP-1), Adf-1, MATa1 and HEB, obtained the top scores for over-represented factors. As for under-represented factors, there is also a correspondence between the newer and older results. There is a noticeable trend as the same factors (or factor families) appear as top score. For instance, Broad complex factors (Z1, Z2 and Z4), the Hepatocyte Nuclear Factors/Forkhead transcription (HNF/FoxA), Evi-1, Hb and IRF appear to be under-represented in both cases.

In conclusion, there is not a clear relation between the resulting over-represented and under-represented factors and with properties of CRUNCS. Therefore, a deeper biological investigation of these factors is suggested.

4.2 Randomly Generated Dataset

A set of 8785 randomly generated regions were created to test the theoretical validity of the approach. These 8785 random regions use the CRUNCS regions as model in the sense that they have the same distribution of per chromosome as the CRUNCS (this distribution is shown in Table 4.1) and their average region size was calculated using the CRUNCS as model.

4.2.1 Motivation

Regardless of the number or the average regions size, one thing is theoretically expected if one pick random regions of the genome as regions of study. Theoretically, the Z-scores of this dataset scored against TFBS predictions will follow a Normal (0,1) distribution.

4.2.2 Methods

To generate the regions of the random dataset a C program was developed. This program takes as arguments the desired number of regions to be generated, the size of the chromosome that we are generating regions from and the desired average region size.

Is it important to explain that as our implementation does not generate DNA strings, it only generates random positions on the chromosome, that become the region start. The region end is calculated by adding the desired region size to the regions start.

4.2.3 Analysis of the Results for the Randomly Generated Dataset

As it can be seen in Figure 4.3 the Z-scores obtained by the randomly generated regions against the TFBS predictions found in TRANSFAC are quasi normally distributed [1] with a little skew to the negative side of the axis, if compared to a gaussian distribution.

A Q-Q plot or a Quantile-Quantile plots which is a probability plot used for comparing two probability distributions, by plotting their quantiles against each other was used to compare the results obtained with our dataset to what was expected from a normal distribution.

The theory behind a Q-Q plot explains that when comparing two distributions or comparing an observed distribution to a theoretical distribution (usually normal distribution). If the observed distribution matches the theoretical, the plot will appear

[1] The mean of this data is -0.45 and its variance 1.23

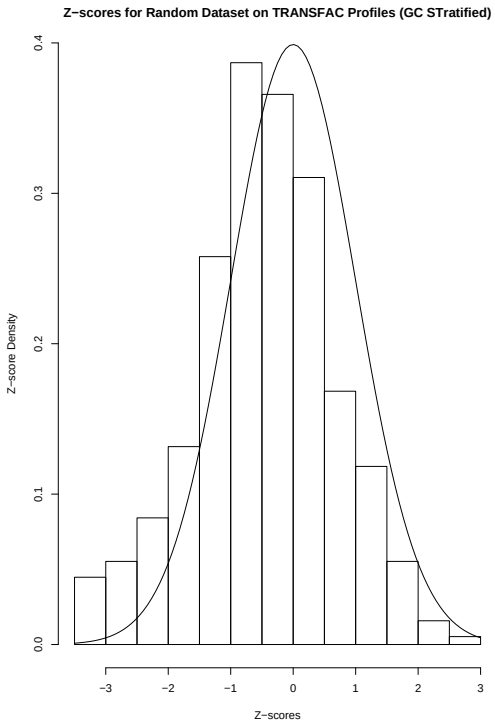

Figure 4.3: Distribution of Z-scores for Random Dataset over TRANSFAC Predictions

as a straight line. If the distribution does not agree then it will appear as a non linear function and the model is said to be a poor fit.

As it can be seen in Figure 4.4, the Q-Q plot comparing the observed Z-scores to those expected from a normal distribution reveal a good correlation (almost a straight line) with a little variation in the values between 0 and -1, which seems to be due to the fact that the 0 to -1 range, is the one that holds most of the values.

There is no accurate explanation for the greater number of Z-scores in the 0 to -1 range. Even after trying with different random datasets (varying in number of regions and different average region sizes) and checking for overlaps in the generated regions, there was no clear answer.

One of the answers that may be given to explain this phenomenon is that there

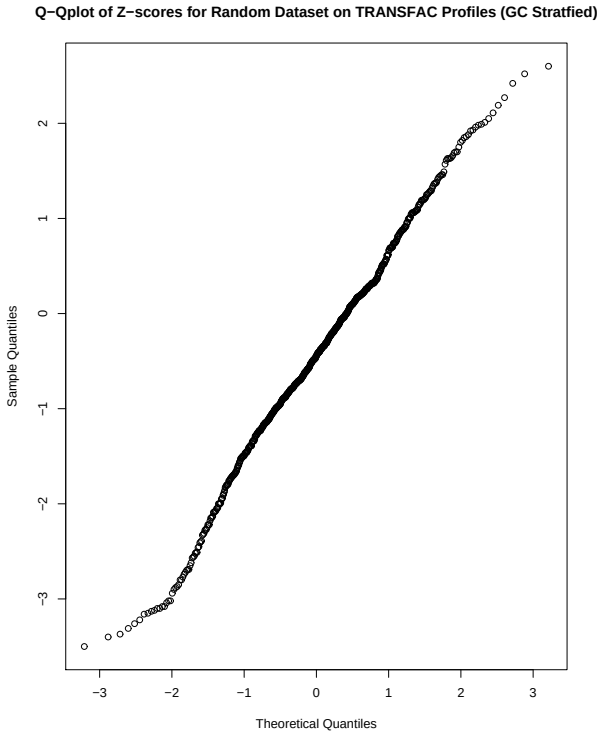

Figure 4.4: Q-Q Plot of distribution of Z-scores for the Random Dataset over TRANSFAC Predictions

are more AT rich factors than their GC counterparts in the TRANSFAC database. Fogel et al. [30] studied 292 of this matrices and concluded that adenine was the most common nucleotide found in full motifs and core motif regions, which in fact reflects the distribution of A's, C's, G's and T's in the human genome. But, any of these assumptions have yet to be confirmed.

4.3 Angiopoietin-1 Dataset

This set of regions was provided by Dr Sabbah Hussain from the Microbiology and Immunology department of McGill University. It consist of a list genes significantly up-regulated and down-regulated by angiopoietin-1, for the purpose of identifying

transcription factors activated by this growth factor.

4.3.1 Motivation

Blood vessels are an important part of the circulatory system, which is the system that is in charge of transporting blood throughout the body. There exists three major types of blood vessels: (1) arteries, which carry the blood away from the heart; (2) capillaries, which enable the actual exchange of water and chemicals between the blood and the tissues; and (3) veins, which carry blood from the capillaries back towards the heart.

The growth of arterial (blood) vessels can be happen in various different ways [15]:

1. Vasculogenesis: refers to the formation of vascular structures from circulating or tissue-resident endothelial progenitors. This form is particularly related to the development of the vascular system in embryos.

2. Angiogenesis/Arteriogenesis: refers to the sprouting of thin-walled endothelium-lined structures and its stabilization. This form plays is particularly related to the repair mechanism of damaged tissues in adults.

3. Collateral Growth: refers to the expansive growth of pre-existing vessels, forming collateral bridges between arterial networks.

All of these forms of vessel growth are often summarized as angiogenesis. Angiogenesis is a normal process in growth and development, and has a very important role in wound healing. On the one hand, when vessel growth is dysregulated this can contribute to the development of several malignant inflammatory, infectious and immune disorders, for example: cancer, psoriasis, arthritis, blindness, obesity, asthma atherosclerosis [62]. On the other hand, insufficient vessel growth and vessel

regression can cause: heart and brain ischemia [2], neurodegeneration, hypertension, pre-eclampsia, respiratory distress, and osteoporosis [15].

The biological mechanisms that stimulate the process of angiogenisis are diverse. There exists various angiogenic proteins, including several growth factors that are involved in this process, among them, some called the angiopoietins.

Angiopoietins are protein growth factors that promote angiogenesis. Four different angiopoietins proteins have been identified: Ang1, Ang2, Ang3, Ang4. Of those four, Ang-1 and Ang-2 are the best characterized angiopoietins, while Ang-3 and Ang-4, are less characterized. Angiopoietins function by binding and activating their physiologic receptors Tie-1 and Tie-2. Also, transcription factors Egr-1 [1] and KLF-2 [73] have been identified as important TF downstream from Tie-2 receptors.

4.3.2 Methods

The list of genes described in the begining of the section consisted of 58 up-regulated genes and 43 down-regulated genes identified in Dr. Hussain's lab by exposing human umbilical vein endothelial cells to phosphate buffer saline (PBS) as control and to 300 ng/ml of angiopoietin-1 (ligand for Tie-2 receptors) for a period of four hours and later hybridized as RNA to Affymatrix oligo microarrays.

As the list provided indicated the names of up-regulated and the down-regulated genes, but not regions of interest to be investigated, a conversion to BED format had to be done. This formating can be divided into three steps:

1. For every gene on the list transcription start sites (TSS) were localized and used as references to identify promoters regions upstream and downstream.

[2] ischemia can be defined as an inadequate blood supply to an organ or part of the body

2. Flanking DNA regions of size 1kb, 10kb and 100kb were selected around the TSS [3].

3. The flanking regions were converted to BED format (indicating the chromosome, region start and region end) and written to text files, six of them in total. Three for the down-regulated genes (down1kb.bed, down10kb.bed and down100kb.bed) and three for the up-regulated genes (up1kb.bed, up10kb.bed and up100kb.bed).

After the files were in a suitable format each bed file was analyzed using our program to identify over-represented TF binding sites.

Preliminary Results for the Angiopoietin-1 Dataset Surprising Z-scores (scores greater than 6.00) for sixty-two PWMs in either one of the up-regulated or down-regulated regions were found. Some of these results are shown for the activator protein family (AP) in Table 4.7, for signal transducer and activator of transcription (STAT) in Table 4.8, finally for the ETS family in Table 4.9. In the three tables Z-scores greater than or equal to 6 are shaded in gray.

It is important to notice that in Table 4.7, Table 4.8 and Table 4.9, most of the surprising Z-scores are in the flanking regions of size 100kb. However, there are a few large Z-scores in the flanking regions of size 10kb.

These preliminary results raised some intersting questions, for instance: can we know exactly which specific genes generated those surprising Z-scores?

To answer this question a more detailed analysis was done, as described below.

[3]Flanking size of 1kb, means 1kb upstream and 1kb downstream, for a region of total size of 2kb. The same applies for flanking size of 10kb and 100kb.

Table 4.7: Z-scores obtained by the program using the Ang-1 dataset over TRANSFAC profiles (AP Family)

		Upregulated			Downregulated		
Matrix ID	Factor	1kb	10kb	100kb	1kb	10Kb	100Kb
M00172	AP -1	0.08	1.95	6.05	-2.53	-0.73	2.13
M00173	AP -1	0.65	3.49	6.08	-0.96	0.61	2.75
M00174	AP -1	1.28	6.03	11.64	-0.92	-1.63	4.87
M00175	AP -4	-2.97	1.57	9.71	-1.10	5.27	11.97
M00176	AP -4	-2.09	1.59	9.30	0.34	6.36	11.33
M00199	AP -1	-1.56	8.87	14.19	-2.39	-0.71	6.21
M00924	AP -1	0.27	4.65	6.17	-3.09	-0.74	3.06
M00925	AP -1	0.41	5.07	12.81	-2.38	-0.94	5.95
M00926	AP -1	-0.02	5.10	13.64	-1.99	-1.21	6.10
M00927	AP -4	-2.87	3.18	12.15	-2.76	5.61	11.14

Table 4.8: Z-scores obtained by the program using the Ang-1 dataset over TRANSFAC profiles (STAT Family)

		Upregulated			Downregulated		
Matrix ID	Factor	1kb	10kb	100kb	1kb	10Kb	100Kb
M00223	STATx	2.24	5.92	9.15	0.67	0.34	5.06
M00259	STAT	5.26	4.22	6.21	2.70	3.66	7.12
M00457	STAT5A	5.76	9.24	9.91	2.60	1.50	5.64
M00459	STAT5B	2.73	6.68	7.26	1.41	2.84	5.70
M00460	STAT5A	6.42	5.29	5.73	2.95	1.78	2.45
M00493	STAT5A	3.56	5.06	10.06	3.02	2.42	5.88
M00494	STAT6	4.53	5.17	10.32	3.26	2.86	6.70
M00496	STAT1	5.01	7.45	13.07	4.63	5.49	8.15
M00497	STAT3	2.42	1.94	0.96	0.14	-0.64	2.07
M00498	STAT4	3.96	4.97	9.70	4.40	3.78	9.02
M00499	STAT5A	3.05	1.70	3.81	6.42	2.56	3.11
M00500	STAT6	2.68	5.07	11.99	3.88	2.09	7.05

Table 4.9: Z-scores obtained by the program using the Ang-1 dataset over TRANSFAC profiles (ETS-type Family)

		Upregulated			Downregulated		
Matrix ID	Factor	1kb	10kb	100kb	1kb	10Kb	100Kb
M00025	Elk-1	5.57	6.97	12.51	2.36	2.59	7.50
M00074	c-ETS-1(p54)	3.37	3.63	9.54	0.54	2.30	4.54
M00339	c-ETS-1	2.40	5.71	9.04	-0.15	-0.15	4.65
M00340	c-ETS-2	3.03	4.90	9.96	0.67	4.62	7.34
M00341	GABP	4.54	4.96	7.94	3.75	3.44	4.36
M00771	ETS	3.76	7.43	13.22	1.90	2.01	7.22
M00971	ETS	4.81	6.15	13.03	1.72	2.49	6.46

4.3.3 Gene-by-Gene Scoring

For the purpose of figuring out which genes were responsible for generating the resulting Z-scores, individual bedfiles were first created for every gene, comprising regions of flanking size 1kb, 10kb and 100kb and then analyzed individually to obtain Z-score for each of the 62 PWMs of interest selected from the preliminary results.

The analysis resulted in a two-dimensional matrix of Z-scores with 62 columns (the PWMs) and 43 and 58 rows for down-regulated and up-regulated, respectively. This matrix was further analyzed using the *heatmap.2* function of the *gplots* library of **R** [69] Statistical Tool and heatmaps were created.

The heatmap.2 function works by first finding out patterns between rows and columns by using similarity distance measurements (hierarchical clustering). In other words, the rows and columns of the matrix are re-ordered to group together rows and columns with similar values. Once all the similar values are together, the function creates a visual representation (a heatmap) of this new ordering, which uses a color scale to represent the value range (lowest to highest) and which uses a dendrogram to represent the associations (clusters) formed in the data.

Rationale Behind the Heatmaps Associations in the heatmaps are numerically not interesting, because the algorithm is just clustering together positions with similar values, however, there is a deeper biological meaning for these associations, a cluster in the heatmaps reflects how groups of genes are regulated by a transcription factors, group of different TFs or families of TFs.

In the next two subsection the resulting clustered heatmaps for up-regulated and down-regulated are presented. It is important to say that for some clusters in these heatmaps we found biological evidence for the association, these clusters are explained in detail. For many other clusters the association remains unclear, these clusters are

just mentioned and associations are to be further investigated.

The heaptmaps shown in the next two subsections were done using the default values for the heatmap.2 function, that is, we used *euclidean distance* as similarity distance and *complete linkage* for the hierarchical clustering. It is important to mention that other settings were also tried on the datasets, for instance the *average* and *single* linkage for the hierarchical clustering and even pearson correlation coefficient as a similarity distance. However not a significant difference was noted, that is, the same relations between genes and TFs appeared, but in different locations.

4.3.4 Resulting Heatmaps for Up-Regulated Genes

The resulting heatmaps for the Ang-1 dataset for up-regulated genes can be seen in Figure 4.5 for 1kb flanking regions, in Figure 4.6 for 10kb flanking regions and in Figure 4.7 for 100kb.

There is noticeable cluster in Figure 4.5 comprised of the following genes: DUSP4, C8FW, BHLHB2, CCND1, HMGA2, FBXW2, DIPA, STC1,VEGFC and the following factors: Adf-1, STAT, E2F, STAT5A, STAT5B. This is an interesting cluster because it pairs the *vascular endothelial growth factor C* gene, which encodes the VEGFC protein which is active in angiogenesis and endothelial cell growth and survival [60] and the *Signal Transducers and Activator of Transcription* (STAT) factor which is activated by the Janus Kinase (JAK) and dysregulation of this pathway is frequently observed in primary tumors and leads to increased angiogenesis [47]. Moreover, Chen *et al.* [19] found evidence of over-expressions of VEGF and STAT3, STAT5 activation in ovarian carcinoma cells.

In the heatmap in Figure 4.6 there exist at least four noticeable clusters:

1. The first cluster is located in the left upper corner and is comprised of the follow-

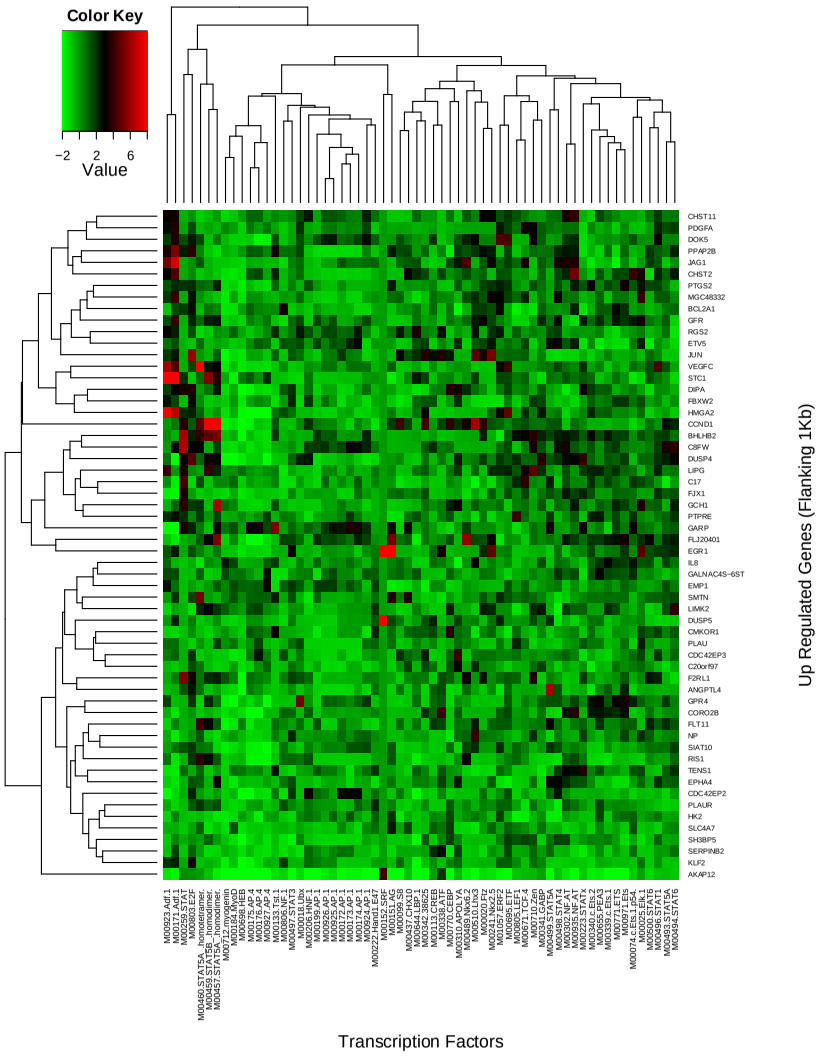

Figure 4.5: ANG-1 upregulated flanking of 1,000 bases

ing genes: FBXW2, CHST2, GFR, PPAP2B, JAG1 and the Adf-1 transcription factor.

2. The second cluster is located near the right upper corner and is comprised of the following genes: EMP1, ANGPTL4, PLAU, C8FW, AKAP12, SLC4A7 and the AP-1 transcription factor.

3. The third cluster is located near the left bottom corner and is comprised of the following genes: C20orf97, CCND1, CORO2B and the following transcription

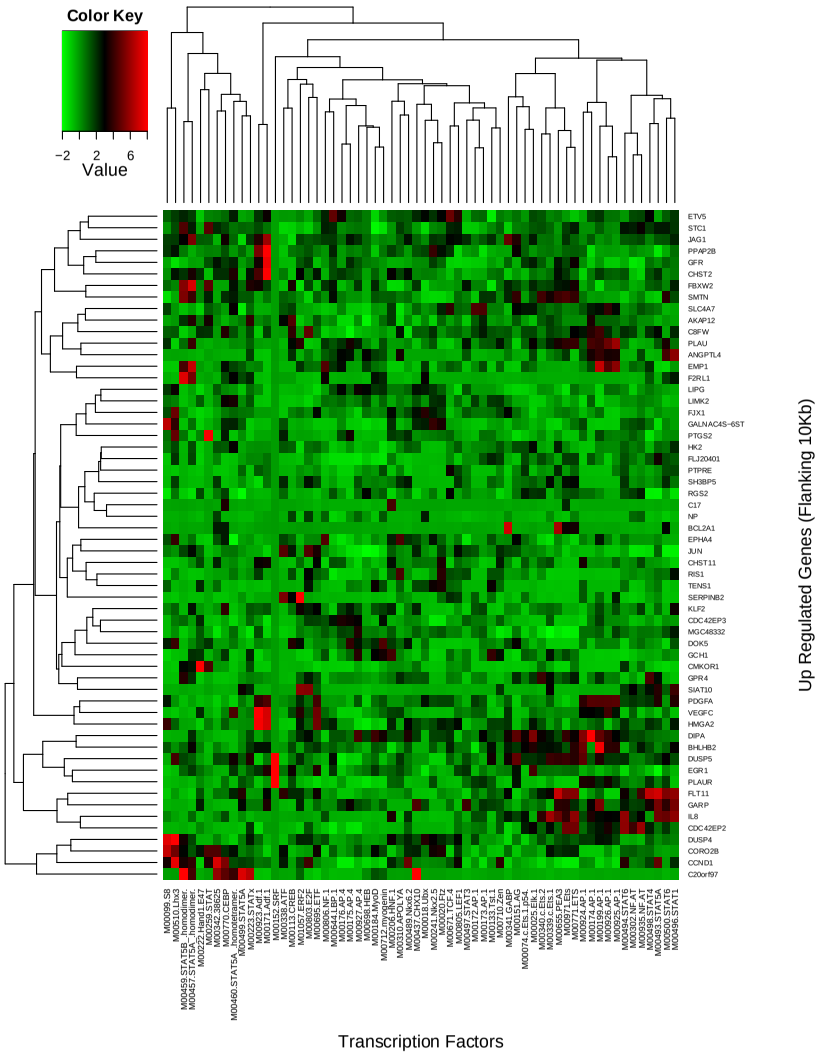

Figure 4.6: ANG-1 upregulated flanking of 10,000 bases

factors: STAT, E2F, STAT5A, STAT5B, S8, LXH3, Hand1:E47, Oct-1 and CEBP.

4. The fourth and biggest cluster is located in the right bottom corner and is comprised of the following genes: CDC42EP2, IL8, GARP, FLT11, DUSP5, BHLHB2, DIPA, HMGA2, VEGFC, PDGFA and some members of the following transcription factors families: STAT, AP and ETS. This cluster shows an interesting relation between the AP-1 transcription factor and the Interleukin-8 (IL-8) gene, which was previously found by Abdel-Malak et al [2], who showed

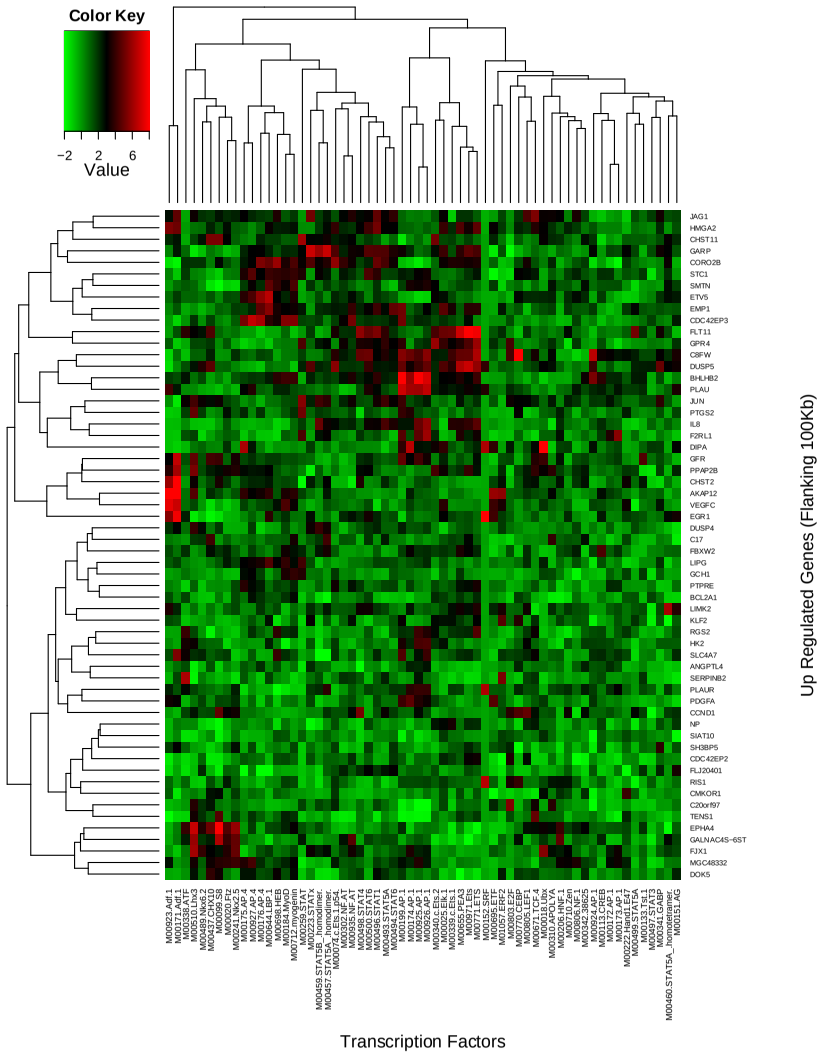

Figure 4.7: ANG-1 upregulated flanking of 100,000 bases

that up-regulation of IL-8 by Ang-1 is mediated through AP-1 binding. Also it shows a relation between Serum Response Factor (SRF) transcription factor and the Early growth response-1 (EGR-1) gene, interestingly SRF is a very important element in the signaling of Vascular endothelial growth factor (VEGF) transcription factor [17] and it is known that VEGF and Ang-1 have and additive effect in the expresion of EGR-1 [1].

In the heatmap in Figure 4.7 there exist at least three noticeable clusters:

1. The first cluster is a very big one, located in the left upper part and is comprised of the following genes: DIPA, F2RL1, IL8, PTGS2, JUN, PLAU, BHLHB2, DUSP5, C8FW, GPR4, FLT11 CDC42EP3, EMP1, ETV5, SMTN, STC1, CORO2B, GARP, CHST11 and some members of the AP and ETS transcription factors families.

2. The second cluster is located in the right part of the heatmap and is comprised of the following genes: EGR1, VEGFC, AKAP12, CHST2, PPAP2B, GFR and Adf-1 transcription factor.

3. The third cluster is located in the bottom of the heatmap and is comprised of the following genes: MGC48332, FJX1, GALNAC4S-6ST, EPHA4 and the following transcription factors: Lhx3, Nkx6.2, CHX10, S8, FTZ, Nkx2.5.

4.3.5 Resulting Heatmaps for Down-Regulated Genes

The resulting heatmaps for the Ang-1 dataset for down-regulated genes can be seen in Figure 4.8 for 1kb flanking regions, in Figure 4.9 for 10kb flanking regions and in Figure 4.10 for 1kb flanking regions.

In the heatmap on Figure 4.8 it can be seen three major clusters (strips):

1. The first strip is comprised of the following genes: SNAPC1, CDKN2C, NMT2, DACH, PALMD, EZH2, ID2 and the E2F transcription factor.

2. The second strip is comprised of the FLJ23056 gene and the following transcription factors: Adf-1, ERF2 and some members of the STAT family.

3. It is also recognizable a cluster near the middle to the right comprised of the following genes: ZNF323, P450RAI-2, PBF, BRD8 and the following transcription factors: NF-AT and some members of the STAT family.

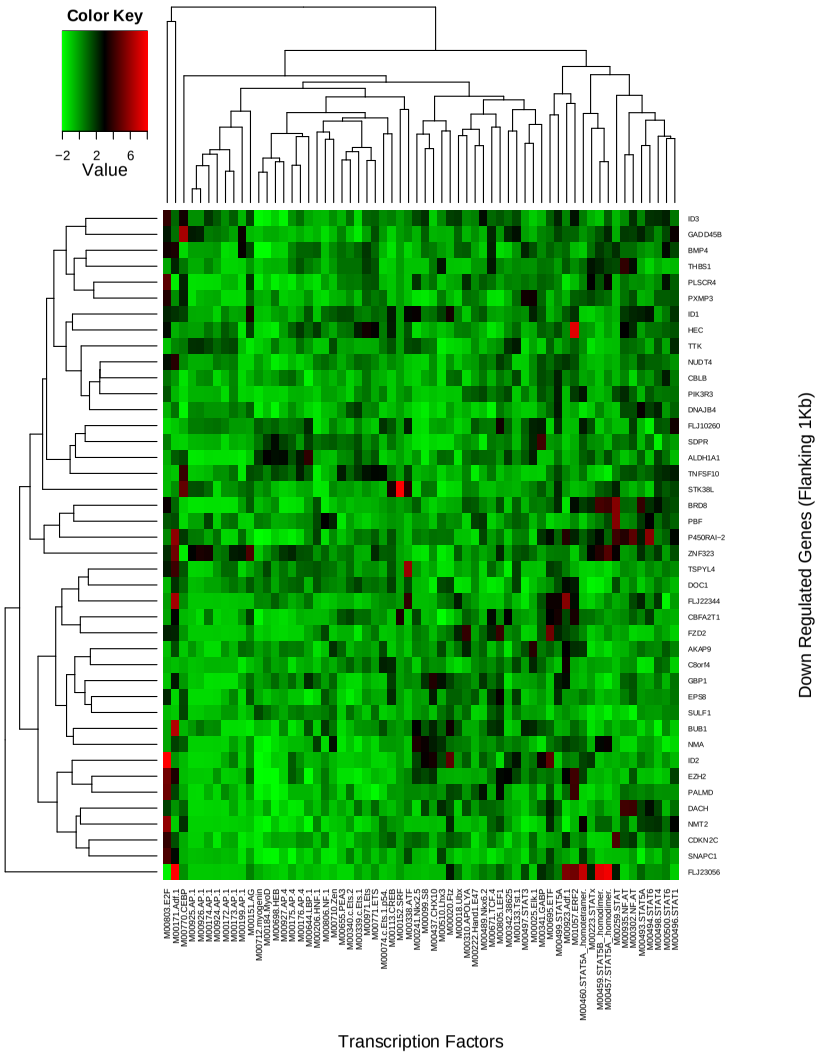

Figure 4.8: ANG-1 upregulated flanking of 1,000 bases

In the heatmap on Figure 4.9 there exist at least two noticeable clusters.

1. The first cluster is located in the left bottom corner and is comprised of the following genes: NUDT4, FLJ22344, BUB1, CDKN2C, P450RAI-2, ID1 and the Adf-1 transcription factor.

2. The second cluster is located in the bottom left corner and is comprised of the following genes: AKAP9, PALMD, CBFA2T1 and the following transcription factors: AP-4, Adf-1, ERF2, myoD, HEB, myogenin and LBP-1.

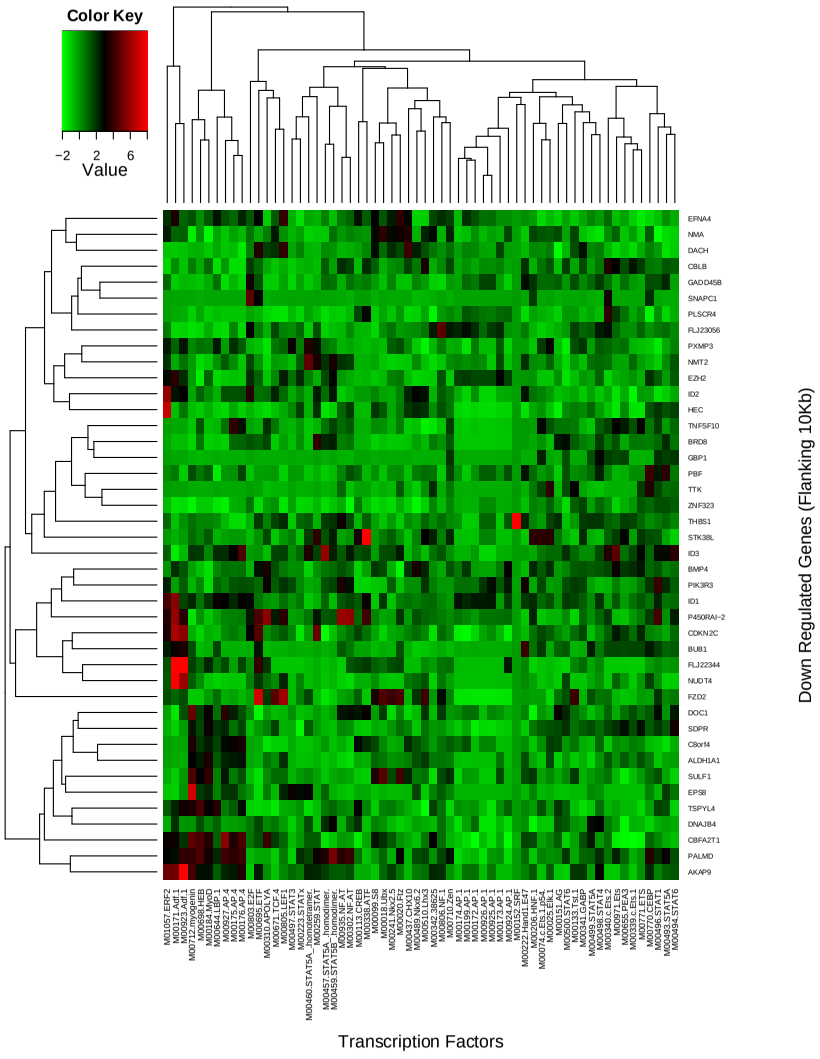

Figure 4.9: ANG-1 upregulated flanking of 10,000 bases

In the heatmap on Figure 4.10 there exist at least four noticeable clusters.

1. The first cluster is located in the left part of the heatmap and is comprised of the following genes: P450RAI-2, PIK3R3, FLJ22344, CDKN2C, BUB1, ZNF323, NUDT4 and the Adf-1 transcription factor.

2. The second cluster is located in the near the first cluster in the bottom left corner and is comprised of the following genes: ID3, ID, PALMD, DOC1, CBFA2T1, NMA, DACH, PBF, CBLB, SULF1, SDPR, BMP4 and the following transcrip-

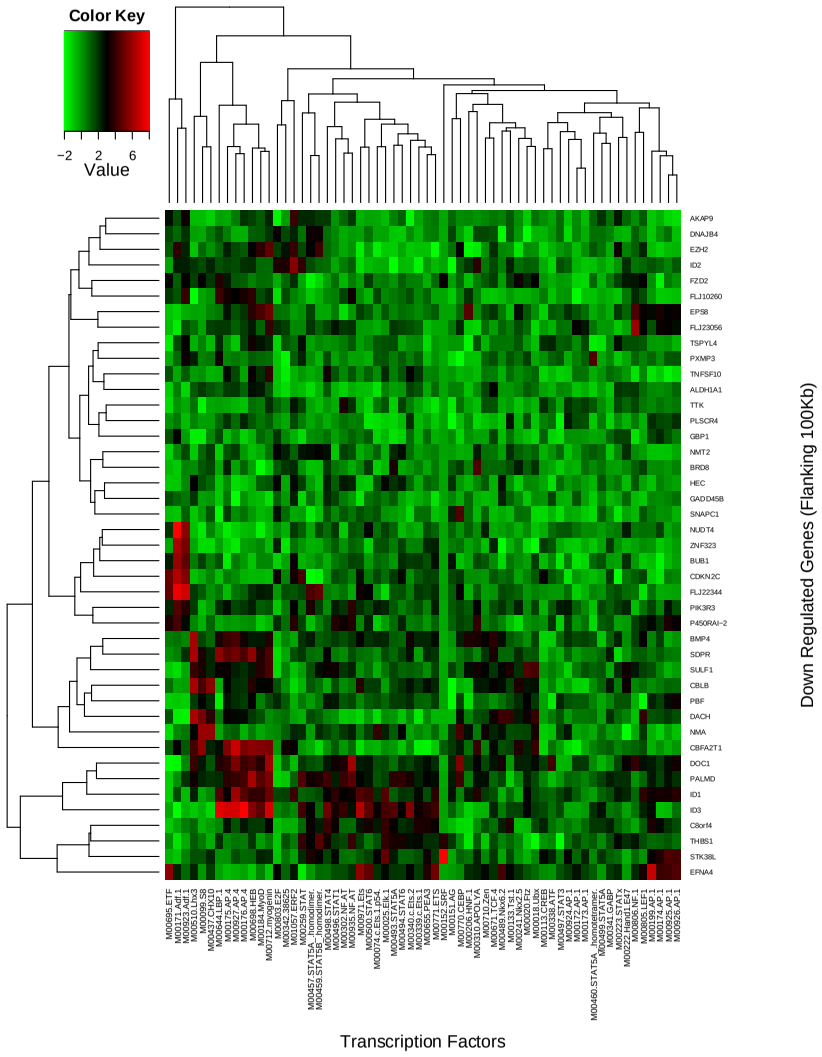

Figure 4.10: ANG-1 upregulated flanking of 100,000 bases

tion factors: Lhx3, S8, CHX10, AP-4, Adf-1, ERF2, myoD, HEB, myogenin and LBP-1.

3. The third cluster is not visible as the first two, it is located in the bottom of the heatmap, it is comprised of the following genes: STK38L, THBS1, C8orf4, ID3, ID1, PALMD, DOC1 and the following transcription factors: NF-AT and some members of the STAT and ETS family.

4. The fourth cluster is located in the bottom of the heatmap, it is comprised of

the following genes: DOC1, CBFA2T1, NMA, DACH, PBF, CBLB, SULF1 and the following transcription factors: CEBP, HNF1, APOLYA, Zen, CF4, Nkx6.2, Nkx2.5, Tst1, Ftz, Ubx.

4.3.6 Analysis of the Results for Ang-1 Dataset

Some conclusions obtained from the analysis of the heatmaps are:

1. In both datasets (up-regulated and down-regulated) the number and size of the visible clusters in the heatmaps increase as the size of flanking the regions increases. Basically, we get more significant motifs with larger flanking regions because we get a higher number of matches. This can also be explained using the notion of *Statistical Power*, in this case we have a typical example case of *High Power*, when there is too much data but no significant effect can be detected.

2. As expected from the initial results (Tables 4.7, 4.8, 4.9), various clusters in the up-regulated genes heatmaps were formed around the STAT, ETS and AP transcription factor families.

3. The up-regulated 10,000 kb heatmap results of great interest for two reasons: (1) it validates known results for SRF, EGR-1 and VEGF factors and (2) show interesting results that can be determined easily with the current wet lab technologies via biological experiments.

Dr. Hussain expressed interest in confirming our in-silico results for up-regulated genes with a flanking region of 10kb, specially for AP-1 and ETS transcription factors and DUSP (DUSP4 and DUSP5) and EGR-1 genes. To date we still wait for the results.

4.4 Summary of the Chapter

In this chapter, results from the application of our method on three different datasets were described and analyzed. We validated our approach theoretically by generating a dataset composed of random regions in the human genome. Also, we tailored a specific model for the study of the TF found in CRUNCS regions. Finally, we presented results on the application of our method to a specific biological dataset for the Ang-1 protein, with results that are being currently investigated in Dr. Hussain's lab.

Chapter 5

Conclusions and Future Work

The inner details of the complex process of gene expression still remain a big question to be answered. With our current knowledge we have been able to figure out, among other things, the important role that the process of transcription of genes plays in the determination of diseases or health. The discovery of the internal workings of this intrincate system lies tightly interlaced to the advances of technology used to recover the protein-DNA interactions and protein-protein interactions.

Experimental technologies that are used to discover TFBS *in vivo* are not exact, and even proven methods have drawbacks and limitations. For instance, ChIP-chip, one of the most frequently used experimental method for the discovery of TFBSs, suffers from problems due to the limited resolution of probes and low signal-to-noise ratios, which often yield inconclusive results. In this panorama, newer and more efficient methods that do not rely on DNA microarrays, as ChIP-sequencing, have proven to be succesful to elucidate the genome-wide location of TFBS for human [43] [70] and mouse [86] and are expected the become the norm in the near future.

Computational methods for the identification of TFBS have become an aid and to certain point a replacement for traditional experimental discovery methods, but,

newer approach are still needed to overcome the limitations imposed by the *futility theorem*.

The task of predicting TFBS *insilico* is a non-trivial one, when thinking about it three reasons that I call *the unfairness* of the human genome come to mind:

- TFBS are short in nature, usually 5-20 base pairs, and the genome is repetitive and long, with a size of almost 3 giga bases, thus making the process of TFBS discovery prone to error and in some sense comparable to find a needle in a haystack.
- The binding variability of TFs force us to use probabilistic ways to represent this dynamicity, hence, forcing us to aproximate answers and make use of heuristic methods.
- As our actual understanding of the processes of gene expression and gene regulation is reduced, a lot of our own human uncertainty is introduced to the models we build, therefore, we can only hope and daydream that as technologies advances our understanding become clearer and clearer.

Even though the TFBS discovery, certainly, we can say that not everything is lost, and, with the day to day increase of genomic data from different species, we hope that in the near future methods like phylogenetic footprinting combined with binding site over-representation and CRM clustering will surprise us by clarifying the relationship between regulatory regions of different species.

5.1 Contributions

In this thesis two statistical over-representation approaches for the discovery of TFBS in genomic sequences were described from their theoretical formulations to the bio-

logical interpretation of the results obtained over five datasets. Our first approach, which was based in simple binomial over-representation statistics, failed due to the GC imbalance existant in the human genome. A second more clever approach was formulated. The main advantage of this new method is that it creates a background model that takes in account the GC Content of the regions studied. For the calculation of the GC content, a way to stratify the genomic regions by percentage of their GC Content was formulated and implemented succesfully.

For the validation of our method three datasets were used. First, two estrogen receptor (ER) datasets coming from regions described in Carroll *et al.* [16] and Lin *et al.* [50], were analyzed using our GC content stratified approach and succesful results were obtained, In both cases the ER profiles and related nuclear factors were found among the top scores. Furthermore, a third validation dataset, consisting of ramdomly generated regions, was created for the purpose of corroborating the theoretical value of our approach. The results obtained from this dataset verified our theoretical assumption and scores were nearly normally distributed as expected.

In addition to the analysis of three validation datasets, two biological datasets were studied. These datasets have interesting characteristics, which on the one hand, made the analysis process difficult and on the other hand, making us extend our method in ways we were not aware were possible. For instance, for the CRUNCS dataset, we had to change the way the background model was constructed and tailor it to only take in consideritions the GC content of coding regions of the human genome. As for the Ang-1 dataset, a complete novel pipeline for the analysis of transcription factors involved in the up-regulation and down-regulation of genes by angiopoietin was created. In gene-by-gene fashion individual over-representation scores were obtained and some biological relations were uncovered by analysing the clustered heatmaps created for the analysis of this specific dataset.

Besides the novel results found on two biological datasets, the main contribu-

tion of this project is GATOR, a public webserver capable of computing the over-representation of TFBS in selected regions of the human genome. We think that the main feature of this implementation is that it will remain useful in years to come due to the following characteristics:

1. The structure of the implementation is engineered to be extended to other genomes. For now it uses predicted TFBS for the human genome, however as long as predictions are available they can be passed as arguments to the tool.

2. The possibility of changing the background model. Thanks to the experience gained by analysing the CRUNCS dataset, we now know that we are able to change the background and tailor it for the necessities of different datasets.

3. The potential of extending the way regions are analyzed and results are presented. Thanks to the experience gained by analysing the Ang-1 dataset, we formulated a pipeline that has a gene-by-gene resolution which included clustered heatmaps for further biological interpretations of the Z-scores.

One tangible contribution of this project (that is not reflected in this thesis, but that we feel is worth mentioning) is the analysis made on a set of genomic regions related to Estrogen Receptor (ER) for our colaborator Dr. Pierre-Étienne Jacques from Institut de Recherche Cliniques de Montréal (IRCM). This results were partially used in their analysis of estrogen receptor signaling, which was later published in a paper by Gévry *et al.* [35], paper in which our colaboration was acknoweledged.

5.1.1 Other Existing Methodologies

Methods for finding over-represented TFBSs in a set of co-regulated or co-expressed genes follow a similar approach as described in Section 1.6.3, but differ in the way

they calculate the background model, the statistical tools to assess the over/under-representation, the way used to decide what defines an observation, if it relies on phylogenic approaches to minimize the False Discovery Rate (FDR) and in the specific case of analyzing gene promoters, the flanking region they look for the presence of TFBSs.

Our method is quite similar and different at the same time to one recently presented by Zambelli *et al.* in [91] called Pscan. It differs in that this method was designed for calculating the over/under-representation of TFBSs co-regulated genes *only* by scanning the promoters sequences of these genes for a in regions of fixed lenght (-450 to +50, -200 to +50 and -950 to +200), does not rely on comparison with orthologous sequences for FDR, it that it can be used to locate TFBSs in genes of four species: human, mouse, rat, fruitfly and Arabidopsis thaliana (plant) and finally the way the background model is defined is very different from our method. They resemble in that they both use TRANSFAC and JASPAR predictions and that the final results are described as heatmaps using the final Z-scores (P-values) ordered as genes vs. transcription factors.

It is also important to mention another method that was recently published called PASTAA [71], which also has as main goal the association of genes and their regulating TFs by scanning the gene promoters, but in this case specifically looking to tissue-specific genes. It differs from Pscan and from our method in that the list of genes to be investigated do not have to be co-regulated or co-expressed, therefore can be used to scan a large set of genes of different categories (tissues or expression patterns) or even a full genome [91]. It is similar to both Pscan and our method in that it uses TRANSFAC predictions as input to their program.

5.2 Future Work

As the methods described in this thesis have a solid theoretical foundation that was proven to work, we think that the future modifications to be done are not method itself, but in the implementation. There are at five main changes that are foreseen for the GATOR implementation:

1. Connect the GATOR program to a TFBS predictor, for instance the one described in Blanchette *et al.* [7], and let the user upload their own profiles, compute their predictions and calculate their Z-scores.

2. Although the GATOR program runs in parallel, we think that it can be made faster by dividing the jobs in the cluster chromosome by chromosome, instead of just only matrix by matrix, which is the way it is done now.

3. The loading of the GC content to the COUNTER program can be made faster if instead of text files the pre-computed GC files are in a binary format (serialized format) and loaded into memory as it.

4. Add necessary connections from the GATOR to R to post-process the resulting Z-scores in order to return the results in a visual representation as an histogram and/or heatmap, as well as a spreadsheet.

5. Add the option to allow the user to upload, not only a list of interesting regions, but a list of genes (as it was the case for the Ang-1 dataset) for it analysis in a gene-by-gene fashion.

The research effort exposed in this thesis is far from being exhaustive, our method is based on a rather *simple* statistics that are used to assess the over-representation of transcription factor binding sites in DNA sequences, thus we do not claim to have a ground-breaking method, but, we feel proud to present to the scientific community

a method that has proven to be valid and even more important, that has proven to be a helpful tool to answer real questions in various domains of biology and medecine.

Bibliography

[1] Nelly A. Abdel-Malak, Mahroo Mofarrahi, Dominique Mayaki, Levon M. Khachigian, and Sabah N.A. Hussain. Early Growth Response-1 Regulates Angiopoietin-1-Induced Endothelial Cell Proliferation, Migration, and Differentiation. *Arterioscler Thromb Vasc Biol*, 29(2):209–216, 2009.

[2] Nelly A. Abdel-Malak, Coimbatore B. Srikant, Arnold S. Kristof, Sheldon A. Magder, John A. Di Battista, and Sabah N. A. Hussain. Angiopoietin-1 promotes endothelial cell proliferation and migration through AP-1-dependent autocrine production of interleukin-8. *Blood*, 111(8):4145–4154, 2008.

[3] Bruce Alberts, Dennis Bray, Julian Lewis, Martin Raff, Keith Roberts, and James D. Watson. *Molecular Biology of The Cell*. Garland Publishing, Inc, 1989.

[4] Fawzi A Babiker, Leon J De Windt, Martin van Eickels, Christian Grohe, Rainer Meyer, and Pieter A Doevendans. Estrogenic hormone action in the heart: regulatory network and function. *Cardiovasc Res*, 53(3):709–719, 2002.

[5] Timothy L. Bailey, Nadya Williams, Chris Misleh, and Wilfred W. Li. MEME: discovering and analyzing DNA and protein sequence motifs. *Nucl. Acids Res.*, 34(suppl2):W369–373, 2006.

[6] Mathieu Blanchette. Computation and analysis of genomic multi-sequence alignments. *Annual Review of Genomics and Human Genetics*, 8(1):193–213, 2007.

[7] Mathieu Blanchette, Alain Bataille, Xiaoyu Chen, Christian Poitras, Jose Laganire, Cline Lefbvre, Genevive Deblois, Vincent Gigure, Vincent Ferretti, Dominique Bergeron, Benoit Coulombe, and Francois Robert. Genome-wide computational prediction of transcriptional regulatory modules reveals new insights into human gene expression. *Genome Research*, 227:656–667, 2006.

[8] Mathieu Blanchette, W. James Kent, Cathy Riemer, Laura Elnitski, Arian F.A. Smit, Krishna M. Roskin, Robert Baertsch, Kate Rosenbloom, Hiram Clawson, Eric D. Green, David Haussler, and Webb Miller. Aligning Multiple Genomic Sequences With the Threaded Blockset Aligner. *Genome Research*, 14(4):708–715, 2004.

[9] Mathieu Blanchette and Martin Tompa. FootPrinter: a program designed for phylogenetic footprinting. *Nucl. Acids Res.*, 31(13):3840–3842, 2003.

[10] Brian Bowen, Jay Steinberg, U.K. Laemmli, and Harold Weintraub. The detection of DNA-binding proteins by protein blotting. *Nucl. Acids Res.*, 8(1):1–20, 1980.

[11] Nick Bray, Inna Dubchak, and Lior Pachter. AVID: A Global Alignment Program. *Genome Research*, 13(1):97–102, 2003.

[12] Nicolas Bray and Lior Pachter. MAVID: Constrained Ancestral Alignment of Multiple Sequences. *Genome Research*, 14(4):693–699, 2004.

[13] Michael Brudno, Chuong B. Do, Gregory M. Cooper, Michael F. Kim, Eugene Davydov, NISC Comparative Sequencing Program, Eric D. Green, Arend Sidow, and Serafim Batzoglou. LAGAN and Multi-LAGAN: Efficient Tools for Large-Scale Multiple Alignment of Genomic DNA. *Genome Research*, 13(4):721–731, 2003.

[14] Martha Bulyk. Computational prediction of transcription-factor binding site locations. *Genome Biology*, 5(1):201, 2003.

[15] Peter Carmeliet. Angiogenesis in health and disease. *Nat Med*, 9(6):653–660, 2003.

[16] Jason S Carroll, Clifford A Meyer, Jun Song, Wei Li, Timothy R Geistlinger, Jerome Eeckhoute, Alexander S Brodsky, Erika Krasnickas Keeton, Kirsten C Fertuck, Giles F Hall, Qianben Wang, Stefan Bekiranov, Victor Sementchenko, Edward A Fox, Pamela A Silver, Thomas R Gingeras, X Shirley Liu, and Myles Brown. Genome-wide analysis of estrogen receptor binding sites. *Nat Genet*, 38(11):1289–1297, 2006.

[17] Jianyuan Chai, Michael K. Jones, and Andrzej S. Tarnawski. Serum response factor is a critical requirement for VEGF signaling in endothelial cells and VEGF-induced angiogenesis. *FASEB J.*, pages 03–1232fje, 2004.

[18] Erwin Chargaff. Some recent studies on the composition and structure of nucleic acids. *Journal of Cellular and Comparative Physiology*, 38(S1):41–59, 1951.

[19] Huaizeng Chen, Dafeng Ye, Xing Xie, Bingya Chen, and Weiguo Lu. Vegf, vegfrs expressions and activated stats in ovarian epithelial carcinoma. *Gynecologic Oncology*, 94(3):630 – 635, 2004.

[20] Hui Chen and Mathieu Blanchette. Detecting non-coding selective pressure in coding regions. *BMC Evolutionary Biology*, 7(Suppl1):S9, 2007.

[21] Ramu Chenna, Hideaki Sugawara, Tadashi Koike, Rodrigo Lopez, Toby J. Gibson, Desmond G. Higgins, and Julie D. Thompson. Multiple sequence alignment with the Clustal series of programs. *Nucl. Acids Res.*, 31(13):3497–3500, 2003.

[22] F Collins, E Green, A Guttmacher, M Guyer, and US National Human Genome Institute. A vision for the future of genomics research. *Nature*, 422:835–847, 2003.

[23] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001.

[24] Athel Cornish-Bowden. Nomenclature for incompletely specified bases in nucleic acid sequences: rcommendations 1984. *Nucl. Acids Res.*, 13(9):3021–3030, 1985.

[25] Francis Crick. Ideas on protein synthesis. *Self Published*, pages 1–2, 1956.

[26] Francis Crick. Central dogma of molecular biology. *Nature*, 227:561–563, 1970.

[27] Gavin E. Crooks, Gary Hon, John-Marc Chandonia, and Steven E. Brenner. WebLogo: A Sequence Logo Generator. *Genome Research*, 14(6):1188–1190, 2004.

[28] Kyle Ellrott, Chuhu Yang, Frances M. Sladek, and Tao Jiang. Identifying transcription factor binding sites through Markov chain optimization. *Bioinformatics*, 18(2):S100–109, 2002.

[29] Laura Elnitski, Victor X Jin, Peggy J Farnham, and Steven J M Jones. Locating mammalian transcription factor binding sites: a survey of computational and experimental techniques. *Genome Research*, 16(12):1455–64, Dec 2006.

[30] Gary B. Fogel, Dana G. Weekes, Gabor Varga, Ernst R. Dow, Andrew M. Craven, Harry B. Harlow, Eric W. Su, Jude E. Onyia, and Chen Su. A statistical analysis of the transfac database. *Biosystems*, 81(2):137 – 154, 2005.

[31] Martin C. Frith, Yutao Fu, Liqun Yu, Jiang-Fan Chen, Ulla Hansen, and Zhiping Weng. Detection of functional DNA motifs via statistical over-representation. *Nucl. Acids Res.*, 32(4):1372–1381, 2004.

[32] Martin C. Frith, Ulla Hansen, and Zhiping Weng. Detection of cis -element clusters in higher eukaryotic DNA. *Bioinformatics*, 17(10):878–889, 2001.

[33] David J. Galas and Albert Schmitz. DNAase footprinting a simple method for the detection of protein-DNA binding specificity. *Nucl. Acids Res.*, 5(9):3157–3170, 1978.

[34] Mark M. Garner and Arnold Revzin. A gel electrophoresis method for quantifying the binding of proteins to specific DNA regions: application to components of the Escherichia coli lactose operon regulatory system. *Nucl. Acids Res.*, 9(13):3047–3060, 1981.

[35] Nicolas Gévry, Sara Hardy, Pierre-Étienne Jacques, Liette Laflamme, Amy Svotelis, François Robert, and Luc Gaudreau. Histone h2a.z is essential for estrogen receptor signaling. *Genes & Development*, 23(13):1522–1533, July 2009.

[36] Debraj GuhaThakurta. Computational identification of transcriptional regulatory elements in DNA sequence. *Nucl. Acids Res.*, 34(12):3585–3598, 2006.

[37] Mayetri Gupta and Jun S. Liu. De novo cis-regulatory module elicitation for eukaryotic genomes. *Proceedings of the National Academy of Sciences of the United States of America*, 102(20):7079–7084, 2005.

[38] Dan Gusfield. *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*. Cambridge University Press, 1997.

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

[40] GZ Hertz and GD Stormo. Identifying DNA and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*, 15(7):563–577, 1999.

[41] Shannan J. Ho Sui, James R. Mortimer, David J. Arenillas, Jochen Brumm, Christopher J. Walsh, Brian P. Kennedy, and Wyeth W. Wasserman. oPOSSUM: identification of over-represented transcription factor binding sites in co-expressed genes. *Nucl. Acids Res.*, 33(10):3154–3164, 2005.

[42] Jason D. Hughes, Preston W. Estep, Saeed Tavazoie, and George M. Church. Computational identification of cis-regulatory elements associated with groups of functionally related genes in saccharomyces cerevisiae. *Journal of Molecular Biology*, 296(5):1205 – 1214, 2000.

[43] Raja Jothi, Suresh Cuddapah, Artem Barski, Kairong Cui, and Keji Zhao. Genome-wide identification of in vivo protein-DNA binding sites from ChIP-Seq data. *Nucl. Acids Res.*, 36(16):5221–5231, 2008.

[44] Jack D. Keene. Rna regulons: coordination of post-transcriptional events. *Nat Rev Genet*, 8(7):533–543, 2007.

[45] Jonathan M Keith. *Bioinformatics: Volume I: Data, Sequence Analysis and Evolution*. Humana Press, 2008.

[46] W. James Kent, Charles W. Sugnet, Terrence S. Furey, Krishna M. Roskin, Tom H. Pringle, Alan M. Zahler, and D. Haussler. The human genome browser at ucsc. *Genome Research*, 12(6):996–1006, June 2002.

[47] T. Kisseleva, S. Bhattacharya, J. Braunstein, and C. W. Schindler. Signaling through the jak/stat pathway, recent advances and future challenges. *Gene*, 285(1-2):1 – 24, 2002.

[48] Frank P.M. Kruijver, Rawien Balesar, Ana M. Espila, Unga A. Unmehopa, and Dick F. Swaab. Estrogen receptor-alpha distribution in the human hypothalamus in relation to sex and endocrine status. *The Journal of Comparative Neurology*, 454(2):115–139, 2002.

[49] Boris Lenhard, Albin Sandelin, Luis Mendoza, Par Engstrom, Niclas Jareborg, and Wyeth Wasserman. Identification of conserved regulatory elements by comparative genome analysis. *Journal of Biology*, 2(2):13, 2003.

[50] Chin-Yo Lin, Vinsensius B Vega, Jane S Thomsen, Tao Zhang, Say Li Kong, Min Xie, Kuo Ping Chiu, Leonard Lipovich, Daniel H Barnett, Fabio Stossi, Ailing Yeo, Joshy George, Vladimir A Kuznetsov, Yew Kok Lee, Tze Howe Charn, Nallasivam Palanisamy, Lance D Miller, Edwin Cheung, Benita S Katzenellenbogen, Yijun Ruan, Guillaume Bourque, Chia-Lin Wei, and Edison T Liu. Whole-genome cartography of estrogen receptor î±binding sites. *PLoS Genet*, 3(6):e87–, 2007.

[51] Gabriela G. Loots, Ivan Ovcharenko, Lior Pachter, Inna Dubchak, and Edward M. Rubin. rVista for Comparative Sequence-Based Discovery of Functional Transcription Factor Binding Sites. *Genome Research*, 12(5):832–839, 2002.

[52] Mathieu Lupien and Myles Brown. Cistromics of hormone-dependent cancer. *Endocr Relat Cancer*, 16(2):381–389, 2009.

[53] Elaine R Mardis. Chip-seq: welcome to the new frontier. *Nat Meth*, 4(8):613–614, 2007.

[54] Troels T. Marstrand, Jes Frellsen, Ida Moltke, Martin Thiim, Eivind Valen, Dorota Retelska, and Anders Krogh. Asap: A framework for over-representation statistics for transcription factor binding sites. *PLoS ONE*, 3(2):e1623, Feb 2008.

[55] G.A. Maston, S.K. Evans, and M.R. Green. Transcriptional regulatory elements in the human genome. *Annu. Rev. Genomic Hum. Genet.*, 7:29, 2006.

[56] Michael Mayhew. Coding regions under non-coding selection: Implications for transcriptional and post-transcriptional gene regulation. Master's thesis, McGill University, Montreal, Quebec, Canada, july 2008.

[57] Steen Mollerup, Kjersti Jrgensen, Gisle Berge, and Aage Haugen. Expression of estrogen receptors [alpha] and [beta] in human lung tissue and cell lines. *Lung Cancer*, 37(2):153 – 159, 2002.

[58] B Morgenstern, K Frech, A Dress, and T Werner. DIALIGN: finding local similarities by multiple sequence alignment. *Bioinformatics*, 14(3):290–294, 1998.

[59] Cedric Notredame, Desmond G. Higgins, and Jaap Heringa. T-coffee: a novel method for fast and accurate multiple sequence alignment. *Journal of Molecular Biology*, 302(1):205 – 217, 2000.

[60] Su-Ja Oh, Markku M. Jeltsch, Ralf Birkenhger, John E. G. McCarthy, Herbert A. Weich, Bodo Christ, Kari Alitalo, and Jrg Wilting. Vegf and vegf-c: Specific induction of angiogenesis and lymphangiogenesis in the differentiated avian chorioallantoic membrane. *Developmental Biology*, 188(1):96 – 109, 1997.

[61] Chester Olson. *Statistics: Making Sense of Data*. Allyn and Bacoon, Inc., 1987.

[62] Nilesh M. Pandya, Naranjan S. Dhalla, and Dev D. Santani. Angiogenesis–a new target for future therapy. *Vascular Pharmacology*, 44(5):265 – 274, 2006.

[63] Heather B. Patisaul, Patricia L. Whitten, and Larry J. Young. Regulation of estrogen receptor beta mrna in the brain: opposite effects of 17[beta]-estradiol and the phytoestrogen, coumestrol. *Molecular Brain Research*, 67(1):165 – 171, 1999.

[64] Giulio Pavesi, Giancarlo Mauri, and Graziano Pesole. Methods for pattern discovery in unaligned biological sequences. *Briefings in Bioinformatics*, 2(4):417, 2001.

[65] Giulio Pavesi, Giancarlo Mauri, and Graziano Pesole. In silico representation and discovery of transcription factor binding sites. *Brief Bioinformatics*, 5(3):217–36, Sep 2004.

[66] Giulio Pavesi and Federico Zambelli. Prediction of over represented transcription factor binding sites in co-regulated genes using whole genome matching statistics. *Applications of Fuzzy Sets Theory*, 4578:651–658, 2007.

[67] Amol Prakash and Martin Tompa. Discovery of regulatory elements in vertebrates through comparative genomics. *Nat Biotech*, 23(10):1249–1256, 2005.

[68] Pascal Pujol, Jean-Marc Rey, Philippe Nirde, Pascal Roger, Marguerite Gastaldi, Francois Laffargue, Henri Rochefort, and Thierry Maudelonde. Differential Expression of Estrogen Receptor-alpha and -beta Messenger RNAs as a Potential Marker of Ovarian Carcinogenesis. *Cancer Res*, 58(23):5367–5373, 1998.

[69] R Development Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2005. ISBN 3-900051-07-0.

[70] Gordon Robertson, Martin Hirst, Matthew Bainbridge, Misha Bilenky, Yongjun Zhao, Thomas Zeng, Ghia Euskirchen, Bridget Bernier, Richard Varhol, Allen Delaney, Nina Thiessen, Obi L Griffith, Ann He, Marco Marra, Michael Snyder, and Steven Jones. Genome-wide profiles of stat1 dna association using chromatin

immunoprecipitation and massively parallel sequencing. *Nat Meth*, 4(8):651–657, 2007.

[71] Helge G. Roider, Thomas Manke, Sean O'Keeffe, Martin Vingron, and Stefan A. Haas. PASTAA: identifying transcription factors associated with sets of co-regulated genes. *Bioinformatics*, 25(4):435–442, 2009.

[72] Benita S Katzenellenbogen and John A Katzenellenbogen. Estrogen receptor transcription and transactivation: Estrogen receptor alpha and estrogen receptor beta - regulation by selective estrogen receptor modulators and importance in breast cancer. *Breast Cancer Res*, 2(5):335–344, 2000.

[73] Keisuke Sako, Shigetomo Fukuhara, Takashi Minami, Takao Hamakubo, Haihua Song, Tatsuhiko Kodama, Akiyoshi Fukamizu, J. Silvio Gutkind, Gou Young Koh, and Naoki Mochizuki. Angiopoietin-1 Induces Kruppel-like Factor 2 Expression through a Phosphoinositide 3-Kinase/AKT-dependent Activation of Myocyte Enhancer Factor 2. *J. Biol. Chem.*, 284(9):5592–5601, 2009.

[74] A. Sandelin, W. Alkema, P. Engstrom, W.W. Wasserman, and B. Lenhard. Jaspar: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res.*, 32:91, 2004.

[75] Albin Sandelin, Wyeth W. Wasserman, and Boris Lenhard. ConSite: web-based prediction of regulatory elements using cross-species comparison. *Nucl. Acids Res.*, 32(suppl2), 2004.

[76] Thomas D. Schneider and R.Michael Stephens. Sequence logos: a new way to display consensus sequences. *Nucl. Acids Res.*, 18(20):6097–6100, 1990.

[77] Scott Schwartz, W. James Kent, Arian Smit, Zheng Zhang, Robert Baertsch, Ross C. Hardison, David Haussler, and Webb Miller. Human-Mouse Alignments with BLASTZ. *Genome Research*, 13(1):103–107, 2003.

[78] Scott Schwartz, Zheng Zhang, Kelly A. Frazer, Arian Smit, Cathy Riemer, John Bouck, Richard Gibbs, Ross Hardison, and Webb Miller. PipMaker A Web Server for Aligning Two Genomic DNA Sequences. *Genome Research*, 10(4):577–586, 2000.

[79] Nicolas Sierro, Yuko Makita, Michiel de Hoon, and Kenta Nakai. DBTBS: a database of transcriptional regulation in Bacillus subtilis containing upstream intergenic conservation information. *Nucl. Acids Res.*, 36(suppl1):D93–96, 2008.

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

[81] Gary D. Stormo, Thomas D. Schneider, and Larry Gold. Quantitative analysis of the relationship between nucleotide sequence and functional activity. *Nucl. Acids Res.*, 14(16):6661–6679, 1986.

[82] Gert Thijs, Magali Lescot, Kathleen Marchal, Stephane Rombauts, Bart De Moor, Pierre Rouze, and Yves Moreau. A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics*, 17(12):1113–1122, 2001.

[83] Julie D. Thompson, Desmond G. Higgins, and Toby J. Gibson. CLUSTAL W: 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.

[84] Martin Tompa, Nan Li, Timothy L Bailey, George M Church, Bart De Moor, Eleazar Eskin, Alexander V Favorov, Martin C Frith, Yutao Fu, W James Kent, Vsevolod J Makeev, Andrei A Mironov, William Stafford Noble, Giulio Pavesi, Graziano Pesole, Mireille Regnier, Nicolas Simonis, Saurabh Sinha, Gert Thijs, Jacques van Helden, Mathias Vandenbogaert, Zhiping Weng, Christopher Workman, Chun Ye, and Zhou Zhu. Assessing computational tools for the discovery of transcription factor binding sites. *Nat Biotech*, 23(1):137–144, 2005.

[85] Wyeth Wasserman and Albin Sandelin. Applied bioinformatics for the identification of regulatory elements. *Nat Rev Genet*, 5(4):276–287, 2004. 10.1038/nrg1315.

[86] Elizabeth D. Wederell, Mikhail Bilenky, Rebecca Cullum, Nina Thiessen, Melis Dagpinar, Allen Delaney, Richard Varhol, YongJun Zhao, Thomas Zeng, Bridget Bernier, Matthew Ingham, Martin Hirst, Gordon Robertson, Marco A. Marra, Steven Jones, and Pamela A. Hoodless. Global analysis of in vivo Foxa2-binding sites in mouse adult liver using massively parallel sequencing. *Nucl. Acids Res.*, 36(14):4549–4564, 2008.

[87] E. Wingender, P. Dietze, H. Karas, and R. Knuppel. Transfac: a database on transcription factors and their dna binding sites. *Nucleic Acids Res.*, 24:238, 1996.

[88] Tyra G. Wolfsberg, Johanna McEntyre, and Gregory D. Schuler. Guide to the draft human genome. *Nature*, 409(6822):824–826, 2001.

[89] C. Workman and G. Stormo. ANN-spec: a method for discovering transcription factor binding sites with improved specificity. *Pac. Symp. Biocomput.*, 5:464–475, 2000.

[90] Xiaohui Xie, Jun Lu, E. J. Kulbokas, Todd R. Golub, Vamsi Mootha, Kerstin Lindblad-Toh, Eric S. Lander, and Manolis Kellis. Systematic discovery of regulatory motifs in human promoters and 3[prime] utrs by comparison of several mammals. *Nature*, 434(7031):338–345, 2005.

[91] Federico Zambelli, Graziano Pesole, and Giulio Pavesi. Pscan: finding over-represented transcription factor binding site motifs in sequences from co-regulated or co-expressed genes. *Nucl. Acids Res.*, 37(suppl2):W247–252, 2009.

[92] J Zhu and MQ Zhang. SCPD: a promoter database of the yeast Saccharomyces cerevisiae. *Bioinformatics*, 15(7):607–611, 1999.

Glossary

A adenine

bp basepair

C cytosine

ChIP chromating inmunoprecipiation

CPFM corrected position frequency matrix

CRM Cis-regulatory module

CRUNCS coding regions under non-coding selection

DNA Deoxyribonucleic acid

EM expectation maximization

ER estrogen receptor

FDR False discovery rate

G guanine

GATOR Genome-wide Analysis of TFBS Over-Representation

HMM hidden Markov model

IC information content

IUPAC International Union of Pure and Applied Chemistry

kb kilobases

mb megabyte

MCB McGill Centre for Bioinformatics

mRNA messenger ribonucleic acid

nt nucleotides

PFM position frequency matrix

PHP PHP Hypertext Preprocessor

PSSM position-specific scoring matrix

PWM position weight matrix

RNA ribonucleic acid

RNAP RNA polymerase

SELEX systematic evolution of ligands by exponential enrichment

T thymine

TF transcription factor

TFBS transcription factor binding site

TSS transcription start site