Transcription Factors Locations Analysis in Topological Associated Domains

Mohammad Sefat

School of Computer Science

McGill University, Montreal

August 2021

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Master of Science in Computer Science

Abstract

3D genome structure is a very complex organization inside the nucleus. Genomic interactions that play an important role in cell development are organized through 3D genome structure and chromosomal folding. Transcription factors have an important impact on such interactions. To understand and predict how they locate in the genome structure, many methods have been already offered. However, none of them are comprehensive and the exact interpretation of transcription factors in 3D DNA structure is not fully understood. The understanding of the probability models should be useful to model the 3D genome organization and genomic interactions. This thesis covers some aspects of the computational inference of transcription factors binding sites from experimental data. This thesis aims to study deterministic and probabilistic methods used for transcription factors binding site prediction and suggest a probability model to analyze and predict the transcription factor binding sites and their interactions.

Abrégé

La structure du génome 3D est une organisation très complexe à l'intérieur du noyau. Les interactions génomiques qui jouent un rôle important dans le développement cellulaire sont organisées à travers la structure du génome 3D et le repliement chromosomique. Les facteurs de transcription ont un impact important sur ces interactions. Pour comprendre et prédire comment ils se localisent dans la structure du génome, de nombreuses méthodes ont déjà été proposées. Cependant, aucun d'entre eux n'est complet et l'interprétation exacte des facteurs de transcription dans la structure de l'ADN 3D n'est pas entièrement comprise. La compréhension des modèles de probabilité devrait être utile pour modéliser l'organisation du génome en 3D et les interactions génomiques. Cette thèse couvre certains aspects de l'inférence informatique des sites de liaison des facteurs de transcription à partir de données expérimentales. Cette thèse vise à étudier les déterministes et des méthodes probabilistes utilisées pour la prédiction des sites de liaison des facteurs de transcription et suggèrent un modèle de probabilité pour analyser et prédire les sites de liaison des facteurs de transcription et leurs interactions.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Prof.Mathieu Blanchette for his supervision and guidance in every stage of this research and for providing me with full financial support while I was pursing my Master's degree at McGill University.

Table of Contents

List of Figures

List of Tables

Chapter 1

Literature Review

1.1 Introduction

The emergence of studying a three-dimensional genome organization (3D) has occurred in the last two decades after its importance in genome activity was revealed. For instance, some researches have shown that transcription, replication, and chromosome translocation are impossible without chromosomal folding [8], [31]. Although much work has already been done on the 3D genome organization, its structure details are not fully understood yet [51], [13], [53].

A multi-layer structure model for the genome is offered by several recent research by imaging and computational methods in biochemistry [8], [31], [51]. The genome organization changes dynamically, like other cellular events. In recent years, research shows that in a small region of the genome, chromatin pattern often changes while the construction of chromatin in the global view remains unchanged [8], [31], [51], [13], [53]. This behavior gives the possibility of dynamic variations and interactions between different small sections of chromatin as long as its global structure has a stable formation [8].

Several kinds of genomic biochemical activities, such as transcription factor (TF) binding, chromatin accessibility, and transcription and histone modification, can be measured through sequencing-based genomic assays. Genomic datasets' availability from cellular

conditions such as varying tissues, individuals, disease states, and drug perturbations is essential to develop the integrative analysis algorithms and utilize them [35], [42].

Regulatory factors such as Transcription Factors (TFs) find their genome locations based in part on the other TFs' sites, called co-localization. TFs interactions regulate the most genomic activities, like gene expression, the genome's physical structure, etc. It is important to recognize genomic co-localization of the TFs interactions for understanding genome regulation and the function of regulatory factors [8], [48].

In [60], a robust approach is suggested for the exact global localization of TF binding sites (TFBS) by combining chromatin immunoprecipitation (ChIP) with the paired-end ditag (PET) sequencing to map p53 targets in the human genome.

Spatial and temporal forms of gene expression are regulated by thousands of regulatory DNA sequence elements encoded from the human genome [42]. The procedure that regulatory elements like enhancers act on the target genes is maintained through chromosomal looping. This looping brings the distal enhancer closer to the target genes. This long-range regulatory interaction has a substantial impact on gene expression and determines regulatory variation. There have been many methods suggested to detect these interactions. However, it is unclear by which principles these regulatory factors act on their target genes. Therefore integrative approaches are required to utilize multiple regulatory genomic data sets [48], [49].

Most real world systems in nature are considered a network of dynamic vertices interacting with each other in [39] and [52]. In other words, wherever some information is exchanged there is a network. The Internet, World Wide Web (WWW), social interactions between people, biological systems and neural networks are only a few examples of such networks [7], [41]. In fact, we live in a world of networks. Research, which has been done in the recent years on different fields, shows that in most of these networks there are several common characteristics. These networks are not in the general random networks or deterministic networks. Because of the complicated structure of such networks, they are

called complex networks.

1.2 Biological aspects and importance of 3D genome organization

Several factors may impact chromatin structure within a cell, such as the radial position of chromatin within the nucleus, the cell cycle stage, transcriptional activity, long-range chromatin interaction, etc., or a combination of all of these factors [31].
It gets more obvious by recent results that the global chromatin structure is more complicated than what was expected before [8], [31], [51], [13]. Consequently, in this structure, only through high-depth sequencing or new techniques, important factors like enhancer-promoter interactions, subdomain organizations, etc., can be extracted reliably. DNA structure with high resolution is still unclear, despite the fact that the nucleosome structure model dates back to four decades ago [31]. Thus research on chromatin organization has been done very well by applying different methods that are not comprehensive by themselves.
One of the most potent methods which can give beneficial information about relative positioning of the genomic regions at the single-cell level is the microscopy-based approach. Recently, light microscopy and electron microscopy methods increased our understanding of chromatin's three-dimensional organization within the nuclear space and its relationship with transcriptional regulation [31] but their results are only available in a few areas of interest (i.e. it is restricted to a small numbers of genetic loci and does not permit a comprehensive analysis of the whole genome). This is why the structure and dynamics in the interphase nucleus in vivo and how this structure is related to transcriptional regulation is not clear. Compared with that, chromosome conformation capture (3C)-based methods are about genome-wide, but their output most likely does not show the individual genome's stable structure (i.e. may show a superimposition structure).The 3C exper-

iment identifies interactions between a pair of genomic loci. For instance, it can be used to validate a candidate for promoter-enhancer interaction [24]. Therefore, this technique should be used when there is a prior information about the interacting regions. Since 3C methods cannot give information about dynamics, it is not fully understood how the interaction within a single-cell changes during time. Also, 3C-methods cannot explain the interaction frequency and how they are related to cell cycles and differentiation. All kind of 3C methods start with the same experimental steps which is explained in next sections. It is still unclear whether transcription regulation is a cause or consequence of these factors. Sequencing-based methods cannot directly localize genomic regions but provide information to obtain a spatial relationship between genomic regions and spatial proximities, which is very useful for improving the in silico modeling of the 3D genome. By generalizing these methods for different applications, it is clear that combination of them can give more robust approaches to access the 3D chromatin structure.

The hierarchical folding of DNA at different layers is a more accepted model for 3D than other models. The layers are nucleosomes, chromatin fibers, Topological Associated Domains (TADs), whole chromosomes, and whole-genome (see figure 1.1). Other patterns, such as loops and meta-TADs, also exist along with the genome folding. It has been proved that while TADs are consistent between cell types and species, chromatin loops and compartmentalization are cell type-specific [8]. Proteins such as mediators, cohesins, and CTCF also have high impacts on genome organizations' architecture, specifically on loops formation, but how they work together and the details of their effects are still unclear.

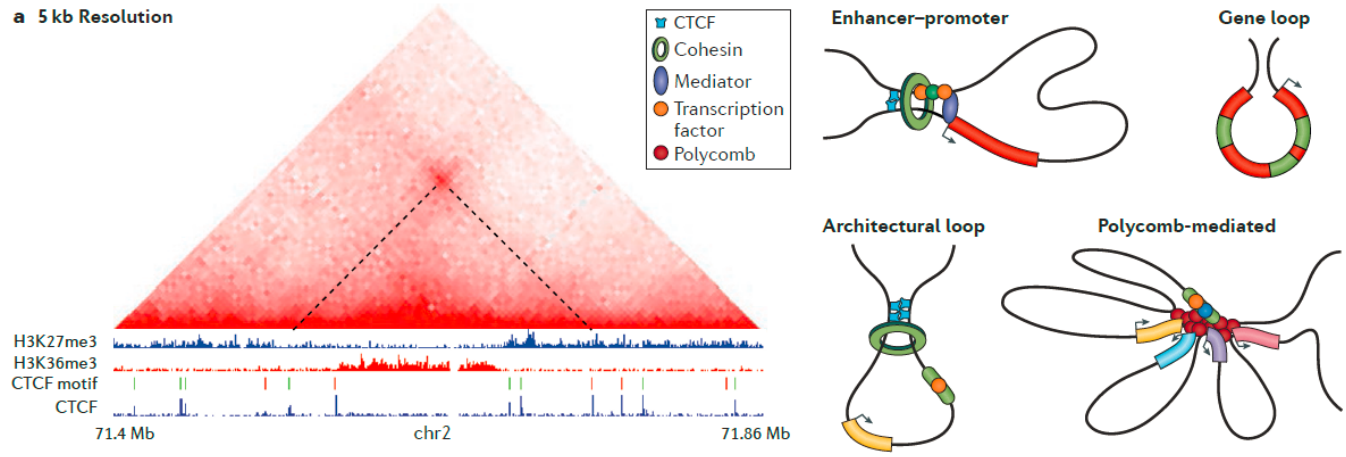


Figure 1.1: Different types of chromatin loop that can locate within a domain (enhancer–promoter loop, Polycomb-mediated loop, gene loop or architectural loop). On the left is an example of an architectural loop as seen in high-resolution Hi-C data (regions participating in loop formation are separated with dotted lines) [8].

Nucleosome is the smallest scale of chromatin structure. Although it was thought to arrange arrays with solenoid or zigzag shapes since long time ago, recently it was revealed that it is more flexible and are shaped in heterogeneous groups called clutches [8]. In order to eliminate the effect of long distance between some regulatory elements through linear genome, chromatin loops are formed in 3D structure. The interactions through the loops are not restricted to enhancer -promoter ones. In a type of chromatin loops which is called gene loops, there are interactions between transcription site of a gene and its own promoter. Recently it is revealed through high resolution Hi-C that there is a correlation between chromatin loops and transcription [8].

Chromosomes are mostly divided spatially into smaller domains such as Topologically associating domains (TADs). TAD is known as a contiguous square domain (sub-megabase pair scale) along the Hi-C maps (introduced in section 1.4) . Regions inside a same TAD interact with each other more often than than the regions in different TADs.

TFs have an impact on cell function by interpreting DNA in the genome. TFs identify DNA in a specific manner. According to the three-dimensional protein-DNA structure

organization, the operation underlying this specificity is recognized for many TFs. There are many concepts have already been understood how TFs identify their cognate DNA binding sites in the genome to initiate gene regulatory function [8], [31], [51]. Since the potential target regions are located in several parts of the genome, it is not fully understood how TFs can recognize their binding locations in genomic regions. Although it is already known that related proteins that attach binding sites to operate in vivo function, the operation of how each TFs select their binding region is unclear. By several processes, at different levels, TFs choose their target regions. Although some of these processes are well understood, a valid model that can combine all of the factors impacting TF-DNA specificity is yet unknown. The high complexity of interaction between various elements of in vivo binding and variability and dependency on many unknowns results in the difficulty of designing such models [51].

Studies in two parallel fields that recently their research are combined, genomics and structural biology, on DNA binding sites specificities results in current knowledge. Searching for protein-DNA recognition code is first done by structural biology. After co-crystal structures of protein-DNA complexes were solved in the 1980s, more than 1600 entries of protein-DNA forms have been inserted in the Protein Data Bank. These structures have revealed why many TFs prefer to attach a specific DNA sequence [51].

Physical interaction between amino acid side chains of protein and the accessible edge of the base pairs that are attached results in the preference for a given protein at a specific position. This form of protein-DNA contact is called base readout (Figure 1.2A). The structural features of DNA binding sites can be identified by TFs as well. Shape readout is called the phenomena of identifying sequence-dependent DNA structure. The static and dynamic properties of DNA structure shape is included in the DNA shape concept (Figure 1.2B). For a given protein, historically, these two identification mechanisms of protein-DNA, which is also called direct and indirect readout, are recognized as mutually exclusive forces for DNA recognition. However, recent studies show that in realistic conditions, most of the proteins use both base and shape readout to identify their cog-

nate binding sites. The combination and contribution of these two forces changes across protein families. (Figure 1.2C)

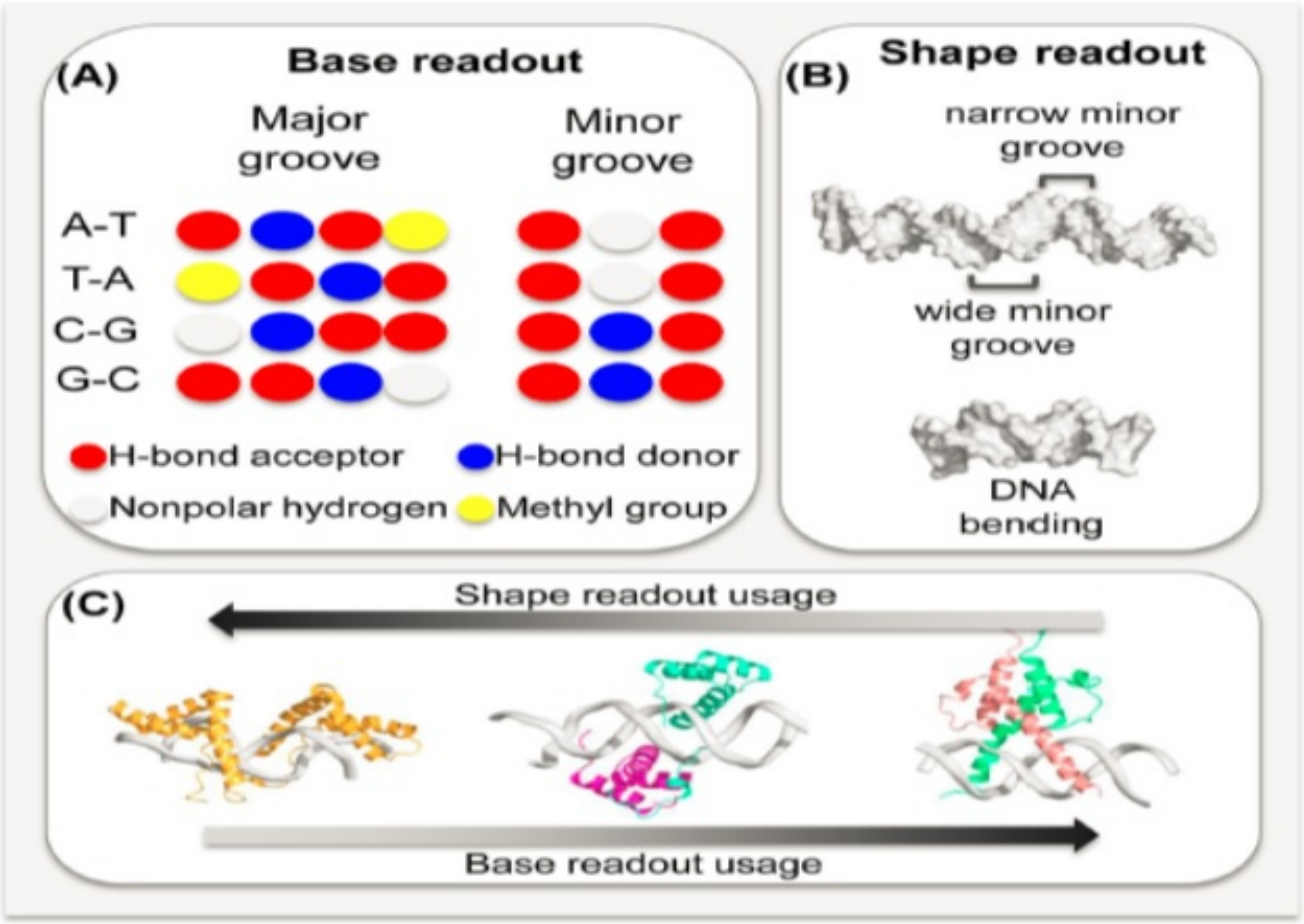


Figure 1.2: (A) Direct interactions between amino acids and functional groups of the bases is described by base readout. Here the hydrogen bond acceptors (red) and donors (blue), heterocyclic hydrogen atoms (white) and the hydrophobic methyl group (yellow) is base pair specific is major groove.(B) Any kind of structural readout based on global and local DNA shape features, is a shape readout. The IFN-β enhanceosome (top) is various in minor groove shape. The human papilomavirus E2 protien (bottom) binds to a binding site with intrinsic curvature(C) Most TFs change between base and shape-readout frequently but the value of each mode is different for each TF. Shape readout contributes mostly for the minor groove -binding HMG box protein (left). Base readout dominates in DNA recognition by the bHLH protein Pho4 (right). Both readout have a about same contribution in the DNA binding of a Hox-Exd heterodimer (center). [51]

1.3 Computational TF binding site prediction

One of the modern genomic's main problem is recognizing all functional elements in genomes and identifying their regulatory impacts. Understanding regulatory sequences is still challenging, although the annotation of human protein-coding sequences is very comprehensive. It is necessary to have a correct map of the regulatory sites to understand gene regulation. Besides, genetic variation in regulatory elements is one of the main reason for diseases [42]. Predicting the operational effect of non-coding variants is challenging since there is little knowledge of which regions in the genome contribute to gene regulation [35], [42]. At characteristic sequence motif, most TFs bind DNA preferentially and provide the sequence specificity, which is essential to direct complex gene regulation operation, and it is one of the primary functions of gene regulation. For individual TF, one of the standard methods to discover its binding locations is Chromatin immunoprecipitation with parallel sequencing (ChIP-seq). The main problem of ChIP is that it considers only one TF in each experiment and underestimate the simultaneous effects of other TFs across different conditions in one single experiment. Therefore it needs thousands number of experiments to consider the impacts of all TFs. Although several computational methods alternatively have provided TF binding site prediction, TFs only are attached to a small fraction of those matching motifs [42]. Even though there is an improvement in the result if other information like experimental data is incorporated, the error rate still is not low. Studies show recently that TF binding sites are correlated with genome-wide assays such as chromatin accessibility measured by DNase I sensitivity, therefore it can be used to analyze and predict TFBS. [35].

Most of the computational methods need a DNA motif sequence at the binding site because the generic assays information can be connected with specific TFs [42]. Most of the approaches start by scanning the genome for all positions with a known sequence motif. Such an approach assumes that those regions occupied by TFs are different in several aspects than the other regions. For instance, there is a considerably higher probability

that those regions bound with TFs are more related to open chromatin or associated with active histone marks and represent evolutionary sequence conservation. Then all of the areas are divided into two categories: occupied or non-occupied. Finally, by calculating the posterior probability for each region, it is assigned to one category. The likelihood of experimental data at a single motif is

$$P(D|G) = P(D|Bound)P(Bound|G) + P(D|NotBound)P(NotBound|G)$$

where G refers to genomic information, and D is experimental cell-specific experimental data [42].

Semi-automated genome annotation (SAGA) methods are a category of algorithms that can get genome-wide datasets such as ChIP-seq, DNase-seq, and Repliseq data from a specific cell-type and output an annotation of genomic activity such as active promoters and repressed regions. By changing the non-automated part of the algorithm (which is traditionally done by human) to the automated one (which is done by algorithm), [35] suggested a new, fully automated strategy to measure each genomic region's importance. To solve the non-automated part's problem, the interpretation step done by a human in SAGA is performed by a machine learning classifier. This new measurement is called conversation-associated activity score to perform integrative modeling of various genomic datasets.

In [34] different machine learning methods to determine TF binding site, promoters and enhancers that used extracted data from next-generation sequencing (NGS) data are studied. They reviewed the different sources of NGS data for machine learning methods. Then they studied unsupervised learning techniques such as Bayesian mixture models, Hidden Markov Models (HMM) and Dynamic Bayesian Networks (DBN). After that they review supervised learning approaches such as regularized linear models, Random Forests, methods based on RNA transcripts and DNA sequence properties and multiple kernel learning. At then end they studied briefly deep learning , recurrent neural networks and

suggested some new approaches to be explored.

In [12] a theoretical model for 3d chromosome structure is suggested to predict active TFBSs. This model provides a probabilistic method to integrate different types of epigenetic data. These data include some histone modifications, DNase I sensitivity applied in a standard motif based model. More precisely they explained how they change epigenetic data to prior probabilities in the first step and then they describe the procedure of motif sequence combination with these priors to calculate posterior.

In [3] DeepBind technique as the first deep learning method is introduced to discover specific sequences where regulatory factors such as TFs attach to. It applies a set of sequences and a binding score is corresponded to that for training. It calculates four step function $f(s)$ for each sequence s as follows:

$$f(s) = net_W(pool(rect_b(conv_M(S))))$$

Through sequences, motif detectors with parameters M are detected at the convolution stage. M_k is a $4 \times m$ matrix, similar to Position Weight Matrix (PWM) but it does not need coefficients to be probabilities or log odds ratio. At the rectification step, by changing b_k units the response of M_k and forcing all negative values to zero, position with a good pattern are separated. Then the maximum and average of detector's response calculated at pooling stage for learning effect of longer motifs and cumulative shorter motifs respectively. After that, output is plugged in nonlinear neural networks with weight W to provide a score by combining the responses.

In [27] the chromatin modification states and some active promoters and enhancers are revealed and predicted by a computation algorithm applying distinct chromatin signatures which can predict regulatory elements in human genome. It used ChIP-chip analysis to discover 30 Mb chromatin structure at 38-bp resolution along 44 human loci chosen by ENCODE contribution. The patterns of core histone H3 and five histone modifications are studied. To recognize active promoters RNA polymerase II (RNAPII) and TBP-associated

factor 1 (TAF1) which are two components of the basal transcriptional machinery, are tested. Conventional ChIP on RNAPII are performed and examined for 121 sites in the ENCODE regions by applying quantitative real time PCR. Similarly transcriptional coactivator p300 are tested to identify active enhancers.
In [57] different tools for regulatory elements prediction are reviewed and also a benchmark of data sets are introduced as future tools inputs. It made a data-sets including known biding sites, and it is ran by thirteen different tools. Then the tools predictions were compared with known biding sites by applying different statistics to evaluate correctness of predictions.

1.4 Computational 3D genomics for analysis of Hi-C data

Several sequencing-based methods have been suggested to model the 3D genome structure. They are mostly high throughput because they are based on parallel sequencing technology. The chromosome conformation capture (3C) method recognizes physical contact between different loci, gives information about relative distance in the nucleus has shed light on the systematic analysis of the 3D organization of DNA [36], [48].

Joint with high throughput sequencing, genome-wide conformation capture assays referred to as Hi-C [10]. Hi-C is one of the popular methods to study the 3D structure of the whole genome. In a Hi-C experiment, those pairs of fragments that are close by distance are recognized through sequencing. A matrix called contact map is then created by the number of ligated fragments spanning two genomic regions and represent loci proximity. To do this procedure, first, close chromatin segments are cross-linked with formaldehyde, and then chromatin is digested with a restriction enzyme. Then because of digestion, overhang segments are filled by biotinylated nucleotides. After that, chimeric DNA fragments are ligated under specific conditions that cause the combining of connected fragments. By cutting DNA and choosing fragments labeled with biotin, a Hi-C library is created. Finally, paired-end sequencing is conducted with the Hi-C library. These

genomic positions are recognized by aligning the two ends of chimeric fragments to the reference genome. The number of contacts is proportional to sequencing coverage, and a high enough coverage cannot be achieved because of budget limitation. Some regions are difficult to recognize through sequencing (like paralogs), and the Hi-C experiment does not detect far separated regions. Therefore Hi-C matrix is very sparse, especially at the highest resolution. To make the matrix less sparse, neighboring regions are grouped to a fixed bin like 1 Mb (mega-base-pair) resolution. Hi-C has provided new insights to reveal the concept of many biological processes such as gene regulation, DNA replication, somatic copy number alteration, and epigenetic change [10], [36], [48]. A basic question is then how to rebuild the 3D structure of the genome from this map. Two general methods have been suggested so far. (i) The consensus method provides a unique mean structure of data at inferring (ii) The ensemble method, which provides a population of structures.
Consensus methods: When a Hi-C contact map is given, inferring a set of 3D coordinates in such a way that the associated chromatin structure fits the contact map [58].
Multidimensional scaling (MDS) is the most popular approach for this method. In this approach for loci pair (i, j) a Hi-C contact $f_{i,j}$ is converted to distance $d_{i,j}$ by power-law conversion where $d_{i,j} = \frac{1}{f_{i,j}^{\alpha}}$. MDS is a classical method aiming at finding N-dimension embeddings for a set of objects such that the given pairwise distances are preserved as well as possible. In the chromatin reconstruction problem, N is set as 3. Since power-law conversion is not defined at zero contact, researchers usually assign a predefined large distance to genomic pairs with zero connection to make the MDS work. However, the arbitrary assignment of these distances often makes the reconstruction weird.
Ensemble methods: For a given Hi-C, several probabilistic methods have been suggested that are divided into two categories. First, those approaches provide a group of structures that fit the contacts map almost equally well. Second, those approaches offer combinations of forms that fit the contact map by considering the additive effect. The only difference in the first category with the consensus method is that instead of inferring a unique solution, it provides multiple optimal solutions. It usually considers the objective func-

tion in consensus structure from a probabilistic perspective and concludes the various solutions through sampling [58]. In [58], a consensus 3D model of a genome from Hi-C data is proposed. This approach applies a statistical model of contact counts, assuming the counts between two loci follow a Poisson distribution. The arbitrary loss function is minimized by a multidimensional scaling-based method by a likelihood function obtained from a statistical model.

High throughput sequencing helps to identify particular regions such as binding sites through experiments. These neighborhoods map to individual genes; we can apply gene enrichment analysis approaches. A new technique called BEHST is presented in [36] to leverage long-range chromatin interaction information from Hi-C datasets for genomic enrichment analysis. These datasets contain chromatin loops that bring distal regions up to a hundred kilobases away within spatial proximity. It uses chromatin loops to associate genes to genomic regions accurately and provide a functional annotation list that corresponds to those genes.

1.5 Joint analyses of TFBS and Hi-C

Understanding the genomic co-localization in TFs interaction networks is essential to analyze and predict genome regulations and functions of regulatory factors in this network. In [36] , a statistical approach is suggested to design a system based on the regulatory factors' interactions by considering their conditional dependence relationship. In the conditional dependence network, there is an edge between two variables conditionally dependent. Because DNA sequences contain transcription factor binding sites and biological experiments are costly to identify these regions, different computational methods are proposed to handle this problem. KEGRU is a recurrent neural network model offered in [48] to predict binding sites based on DNA sequence. It uses the word2vec mapping function to map a part of the DNA sequence, computes features, and predicts binding

sites through Bidirectional Gated Recurrent Unit (BiGRU).

In [44] TAD boundaries and protein attached to them were recognized and predicted by DNA motifs in Drosophila melanogaster and compared with human genome case.

A main mechanism that enhancers as regulatory elements interact with a target gene is through chromosomal looping. Chromosomal looping brings a distal enhancer close to a target gene in three-dimensional space and makes a long-range interaction, which is very important and necessary for tissue-specific expression and regulatory variation interpretation [50]. Several methods such as Hi-C are suggested for detecting these interactions and these methods are mainly different in their resolutions and genomic coverage of the region [46]. Several components of the transcription machinery, such as TFs, make these interactions easier. However, the principles of these interactions which cause tissue-specific expression are not fully understood. It is also unclear the relationship between long-range interaction and other one-dimensional regulatory signals, including transcription factor occupancies or chromatin modification. Therefore, it is necessary to utilize multiple regulatory genomic data sets to characterize enhancer-promoter interactions. Methods that combine different regulatory genomic data sets are beneficial to recognize enhancers and rebuilding transcriptional regulatory networks, but several problems need to be solved to predict enhancer-promoter interactions in a cell line-specific manner. The first problem is to find the most informative measurements to predict interactions in another cell type. It is shown by experiments recently that interactions are mostly cell-specific. Therefore, using one cell-type trained data for another cell-type data causes a bias problem. To overcome this issue, one of the offered approaches is to use a classifier trained on one cell type to predict in another cell type, but it is unclear which cell line's classifier needs to be measured in the new cell lines to predict interactions. The second problem is that additional regulatory genomic data sets are useful rather than those (like CTCF, DNase I, etc.) usually applied for enhancer-promoter interactions. The third problem is that most methods find a unique classifier for all cell lines, while building a specific classifier for each cell line is very important because a classifier can discriminate

between different cell lines. For instance, interaction in one cell line can be inactive due to the inactivity of the promoter or enhancer in that cell line, or they do not interact because of the chromosomal domain [49], [46].

Recently identifying long-range interactions between transcription factors is made possible by improvements in high-throughput chromosome conformation capture such as 3C, 4C, and Hi-C and the availability of Genome-wide 3C datasets. In [49], a multi-task clustering algorithm is presented to use Hi-C data to identify common aspects of genome architecture. Given two or more Hi-C data from different tissues or cell types, it is not clear how to simultaneously recognize clusters in all of them and compare those clusters to identify common and context-specific patterns. By applying inferred clusters, a systematic comparative study of the extent conservation and divergence between matched cell lines of different species and the same species' cell lines is given. Long-range regulatory interaction plays an important role in gene expression. While detecting such interactions helps predict gene expression behavior, a limited number of genome-wide data sets makes this job challenging. A predictive modeling approach is suggested in [46] to leverage limited datasets in cell line-specific to identify long-range regulatory interaction. RIPPLE's supervised machine learning-based method uses 5C experiments in a specific cell type as a training data set. RIPPLE has three main steps. First, it learns a single cell-specific classifier. Then it determines the minimal number of datasets to predict interaction in different cell lines, and finally, it ensembles learning to predict interaction in new cell lines. In [50], a deep neural network model called SPEID is presented to predict enhancer-promoter interactions only based on sequence-based features when the locations of these regulatory factors are given. However, this method does not consider three-dimensional chromatin structure and higher-order chromatin interactions. SPEID, similar to other deep learning models a sequence of feature representations. It consists of three layers. The first convolution layer learns a massive array of kernels. The input sequence is convoluted with short weighted patterns called kernels to compute that pattern's match at each position of the input. Then to learn predictive kernel features, the

second layer called the recurrent layer re-weights each kernel match. At the final layer named dense layer, linear classifier learned on top of the combinations of sequence features output of the recurrent layer.
In [29] a model is purposed to reveal interactions between different part of chromatin and predicts TADs boundaries based on cell type datasets which includes Hi-C interactions and histone mark Chip-seq data.

1.6 Network Estimation

Network modeling assist in creating new methods in studying of interactions between biological entities. Among these methods, a biological network, which is a graph of a connected biological entities plays an important role. Nowadays, by extracting more accurate datasets, these networks assist in understanding and predicting biological phenomena. On the other hand, the urgent need to apply such models in bioinformatics cannot be ignored, because making inference on them can give us deep insight on molecular interactions involved in complex phenotypes such as cancer. The importance of this subject has attracted more attention to biological networks and its behavioral modeling. So much research has been done to improve and utilize such network effects. In this research, we introduce a probability model in biological networks to predict TFSBs.
The results of studies show there are common structural characteristics in many complex networks. For instance, we can point out the small world characteristic [59], scale free [5] and high clustering index [40]. These characteristics have a large effect on network dynamics [41].
Generally, researching for characteristics of a complex network is based on a collected data set from that network. This data set includes information related to network vertices (i.e. person in social networks) and interaction between them (i.e. friendship in social networks). On the other hand, studying this structure and behavior of an enormous network (including around ten thousand vertices or more) and achieving a knowledge about the

whole network is costly (time and storage space) [4], while most of the real world networks include large numbers of vertices and edges and have complicated structures. For example, Facebook currently has more than 1.44 billion monthly active users.

A solution to this problem reduces network complexity by sampling and estimation of characteristics of structure and behavior of that network from its incomplete data sets obtained by sampling. Sampling from a network is extracting a subset of vertices and edges of a large graph. Therefore, accuracy in results of the research in the field of complex networks, is strongly dependent on a given estimation of these networks' characteristics. Generally, traditional statistical methods that independently sample only from vertices or edges have inaccurate results since they ignore the fact that vertices and edges are correlated factors [32]. In this section, we concentrate on sampling methods from a network and estimating characteristics of network elements (vertices and edges) from an incomplete data set.

There are two different cases for networks with respect to availability of network graphs:

- The network graph is observable (i.e. known network). In this case, it is assumed that the whole graph is observable (known) and we have a general knowledge about that such as vertices degree distribution, etc. The goal of sampling here is compressing and scale reduction with extracting a subgraph of the network graph [33], [43]. The main challenge in this approach is the method of selection of vertices and edges in such a way that the final subgraph has the same general characteristics as the main network. This approach that is called coarse-graining in some literature is popular in statistical physics.

- The network graph is hidden. In this case, it is assumed that the graph is unknown to us and our knowledge about the network is local. Therefore, initially we do not know about the situation of vertices and edges. Thus, sampling is being done based on graph surveying.

1.6.1 Network characteristics measurement process

Since vertices and edges are the main elements of a network, studies on the network characteristics can be categorized in three groups:

- Research that is concentrated on network vertices (i.e. the people's age in a social network or the loads on a server in the network of servers)
- Research that is focused on network edges (i.e. friendship status in a social network or information transmitted between two servers)
- Research that is concentrated on characteristics of both network elements (vertices and edges).

The measurement process is shown in the figure 1.3.

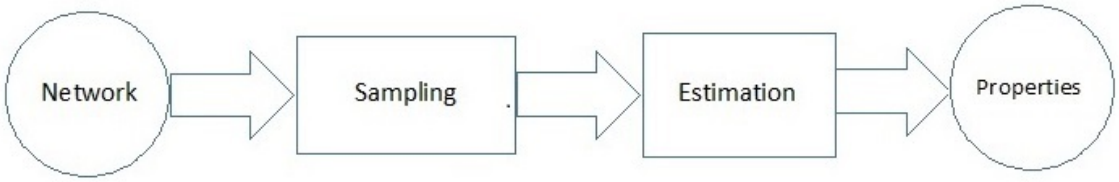


Figure 1.3: The measurement process of network characteristics

This process contains two steps:

- **Sampling:** In this step, the dataset is collected by a sampling method. The sampling method defines how to select network elements (vertices and edges) and the related selection probabilities. A data set for example can include Logs of stored response from HTTP request to an online social network website like Facebook. The raw collected dataset in this step should be processed with an encapsulation function and its output is an extracted sample.

  Since our focus in this research is on hidden networks, the practical solution for sampling design from such networks is applying adaptive sampling. Specifically by starting from an initial vertex (or vertices), its neighbors will be recognized. After

that, one or all of its neighbors will be assigned to a sample set of vertices. This process continues until observing all vertices or achieving a preassigned fixed value for the number of sample set elements. An assigned probability, which the neighbors of a vertex are selected based on, determines the sampling probability of vertices and edges in the network.

According to the neighbor selection step, these methods can be categorized into two groups: random walk methods and Snowball methods. In random walk methods, only one of the neighbors is visited. visiting neighbors can be with or without replacement. In contrast, in Snowball methods all of the neighbors are visited. Breadth First Search (BFS) is one of the Snowball methods.

When an identifier can be assigned to each element of a network, Simple Random Sampling methods can be applied. This method in statistical inference literature is so called ego-centric. For instance, in Twitter's network for each user a unique 32-bit identifier is specified. Therefore, by providing the 32-bit identifier randomly and extracting the user's information (corresponding to the identifier), we can apply sampling of network vertices but since identifiers space is sparse, many requests will remain without response. Therefore, such scenarios are very expensive [45].

- **Estimation:** In this step, by applying an estimator, the characteristics of the main network from the data sample is estimated. An Estimator is a function that uses sampled data as input and gives out the estimation of the desired characteristic of network as an output. We can consider two general approaches to estimate from sampled data: 1) Design-based 2) Model-based.

  In the Design-based approach, it is assumed that the characteristics of vertices and edges are fixed and the sampling method is done only because of the bias existence in observations. Therefore, the network characteristics extraction is only based on the observed sample set. Some prominent works in the design-based approach include the network sampling methods [6], adaptive cluster sampling [55], [54] and some methods in the Snowball sampling field [18], [19], [22]. The design-based ap-

proach can be applied for estimating of network characteristics when the selection probability of each network element can be calculated according to the sampling method. In practice, this has so many challenges. This fact makes researchers study more model-based techniques which include maximum likelihood [56], Bayes [11]. In the model-based approach, a model is considered for the network (i.e. Exponential Random Graph Model) where the collected data set plays as the observed values for that model. Thus our concentration on model parameter estimation designs a mechanism to produce that data set. In practice the real world network modeling is very difficult and complicated therefore it is the weakness of these methods compared with design-based models. In the third chapter of [2] a complete review on the different network models has been done. We should notice that there is generalization for the design-based approach in which a model is applied to direct estimation in the design-based approach [16], [47], [23]. This approach in fact is a combination of two approaches: Design-based and model-based, which is explained in [56] in detail.

It should be mentioned that statistical data, which is missed in the first step of the network characteristics measurement process, can not be recovered in the next steps. Specifically if the obtained sample from the sampling step does not contain sufficient statistical information to have exact estimation, there is no estimator that can compensate and give an exact estimation. Therefore, in addition to the importance of the estimator, we should find a better encapsulation function or more efficient sampling design. According to that, we can see in the table below different aspects of network characteristics estimation using sampled data.

According to the measurement process which include sampling and estimation steps, we can realize the common mistake in some previous research. For example the characteristics of the suggested network in [33], [37] and [32] which are used as references so many times in the complex networks field, all are characteristics of the extracted sample network while they are not estimated characteristics of the main

Table 1

Subject	Variates
Network characteristic	Vertex, Edge, Vertex and Edge
Sampling design	Simple Random Sampling, Random Walk, Snowball
Vertex selection	With replacement, Without replacement
Estimation approach	Design-based, Model-based, Combination

Table 1.1: Different aspects of network characteristics estimation using sampled data

network graph. It means these papers authors assumed the collected sample is representative of the whole network. Also in some studies, they tried to give an exact estimate without mentioning that the extracted sample does not include sufficient statistical information for an estimator. For instance, we can mention [15] which tries to give an estimation of the flow size distribution in Internet.

Generally, based on the estimation approach (design-based or model-based) and network characteristic (vertex, edge, both) research, which has already been done, can be categorized into:

- **Design-based/Vertex:** The research focus here is on extraction related to the network vertices with the design based approach for estimation. Ove Frank's work exists in this category [17]. In [9], [21] full reviews have been done on the research in this category.
- **Design-based/Edge:**

  In this category, the estimation is based on the design for network edge characteristics from sampled data. Most research in this category also originated from Ove Frank's research [20]. [20] is one of the first that extracted the structural characteristics of the network edges based on an incomplete observation.
- **Design-based/Vertex and Edge:** There is little research about simultaneous estimation of vertices and edges characteristics. Recently, in [23] a method is suggested for this purpose based on Random Walk. Specifically, a design-based

strategy supported by a network model is offered to estimate the probability of network elements (vertex and edge).

- **Model-based/Vertex:** There is no serious research on model-based approach to estimate vertex characteristics of a network. It can be justified in this way that if network characteristics are independent of connective structures (edges), there is no need to use the network methods (i.e. adaptive sampling methods). Otherwise, simultaneous vertex and edge modeling (the last category) is required. In practice, it is not clear if we can define a scientific application without considering the connective structure of the network.
- **Model-based/Edge:** Already there is little work that has been done for model-based to estimate characteristics of the network. In [56], the first systematic method is suggested. This method is generalized in [25] in the full class of Exponential Random Graph Models(ERGM).
- **Model-based/Vertex and Edge:** This category is concentrated on vertex and edge characteristics simultaneously. [1] and [38] are two outstanding works on this subject. The suggested idea in both papers is based on the network connective structures (edges) modeling with considering the contagion process in network. In [28], a definite application of this kind of model (i.e. simultaneous network and contagion parameters modeling) is discussed.

Our concentration in this research is on the second category. Some of such methods can be generalized for the estimation of network edge characteristics. For future works, we can pay more attention to the model-based approach to estimate the network vertex and edge characteristics from the Snowball methods.

1.6.2 Subject importance and its application

Analysis of real world complex networks helps us to study the dynamics of different processes (i.e. contagion a new idea or an infection) in a network and its different outputs

like purchasing a production or infection diffusion. However, studying these networks with more than a thousand vertices is very expensive. The costs here means time $T(n)$ and necessary memory space $S(n)$ to run those algorithms, which are applied to analyze big data correspondent to such networks (n is the number of the vertices in networks). In most of the real world complex networks, the order of edges are close to the order of vertices (O(n) or O(nlog(n))). Such networks are considered sparse. On the whole, most of the real world networks contain so many vertices and have a complicated but sparse structure.

The space complexity is the problem of storing data in proper space. The space complexity to store sparse networks can be decreased by proper data structure definition (i.e. linked list). Theoretically, those problems which can be solved by an algorithm with polynomial complexity are considered easy problems. In practice when n is a large number, algorithms with a time complexity of $O(n^2)$ or more, can be very slow. Therefore, when the time complexity is high, accessing even fast super computers does not help that much.

However, the time complexity of most algorithms which are applied to analyze corresponding big datasets are $O(n^c)$ that $c \geq 2$ [30]. Therefore, in practice these algorithms are unusable for enormous networks. To solve this problem, a strategy is to design algorithms where c is as small as possible in algorithms. Most of the research, which follows this approach, work on heuristic algorithms, which only offer estimation of desired parameters. The second strategy for decreasing complexity is to decrease n (i.e. decreasing number of vertices) by sampling from a main network. A combination strategy that simultaneously intends to decrease n and c can also be considered. In this research, we concentrate on the second strategy (i.e. decreasing n). On the other hand, the complete dataset of a network mostly is not available for the following reasons:

- These datasets belong to the most valuable companies' wealth and they are protected. Therefore, borrowing those datasets directly from the owners of those companies is very difficult.

- Collecting an enormous data that contains millions of vertices and edges for instance friends list, profiles, images and videos in a social network, has so many challenges. For example, overloads of data collection for testing social networks (i.e. Facebook) connectivity is one of them. In Facebook, each user is coded with a 4-byte code identifier. On average each Facebook user has 130 friends, thus for friends list extraction we need to download an HTML page with 220 KB. Therefore, to access only the Facebook friends connections, we need about 260 GB (500 millions users* 130*4 bytes). More importantly, there are the overloads of data collection: in order to collect 260 GB we need to download 110 TB of information (500 millions users*220 KB) for HTML data.

- In most cases, we cannot access all vertices (or edges) of the networks. For instance, in a social network like Facebook a user by upgrading his/her own privacy profile would not let the other users access his/her information (i.e. his/her own information and friends list).

In respect to application, network analysis is important in different fields like sociology, biology and telecommunication. Besides social sciences fields, our studies have applications in sampling design of hidden networks. For instance, the hidden network in these fields can be a network of drug addicted people or HIV infected persons. A question, which research tries to answer in this field, can be: What percentage of society (i.e. a city) is infected by HIV? The persons (network vertices) are mostly unknown in such networks and they tend not to disclose their information or their friends' information (i.e. neighbor vertices). By different sampling methods, we can estimate the hidden population of this network. Responses to such questions will be very effective to improve plans in the future.

Chapter 2

Transcription Factor Occupancy Analysis in Topological Associated Domains

Most of research that has already been done about TFBS was related to the analysis and prediction of their location based on other epigenetic information such as histone modification data, DNase I information, etc.[12]. In this chapter we study a practical problem in TFBS analysis and predict binding sites along chromosomes based only on HiC data and TFBS information for other TFs. Due to our knowledge it is the first time that HiC data is borrowed to predict TFBS.

Some important gene characteristics such as gene expression is related to complex diseases like cancer. These characteristics are regulated by the regulatory factors such as TFs. Therefore, how TFs interacts and initiate these function are crucial subjects. The interaction of TFs are very dependant on their binding sites on the genome. That is exactly why TFBS analysis is very essential to understand the initiation of the regulatory functions in the genome. The TFs repeatedly attach and detach on binding sites along chromosomes. Therefore, their positions are very dynamic. We can only get their binding sites information for one instance, Thus, this information cannot be used directly to analyze and predict the regulatory functions initiated by TFs. The TFs preferences to be located in specific sites are based on the location of other TFs nearby. Therefore the proba-

bility of an existing TF in a particular area is dependant on other TFs in the chromosome. It may be the first time that this probability is calculated based on the other TFBS. We can formulate this concept here.

There are two sets of data available. One is training data which includes the binding sites of a set of TFs. A HiC matrix related to that DNA sequence is also available which shows the interactions between these regions. Another one is test data, which includes the position of the same TFs except TF_n along different chromosomes. A HiC matrix related to interactions between these regions is also available. All of these data is obtained through ChIP-sequencing. ChIP-seq experiments was done by ENCODE project (https://www.encodeproject.org/). We used the human genome hg19 data and it is explained how they generate the data in https://genome.ucsc.edu/cgi-bin/hgGateway. The file includes the location of TFs along chromosomes. They include 161 different TFs from 91 different cell types and 947 experiments. The Hi-C data which contains TADs information is achieved by [14] through TopDom program for human embryonic stem cell (hESC) lines. The goal is to calculate the probability that TF_n attaches a set of binding sites in test data.

$$Goal : estimate \quad Pr[TF_n(l_s) = 1 | HiC\ matrix\ \ \&\ \ Training\ data\ \ \&$$

$$Test\ data\ for\ TF_1, ..., TF_{n-1}] \quad 0 < s < m^{'} + 1$$

For instance the following training data is given with the existing interactions between locations. Binding sites of TFs in training data (i.e. bins in training data) are shown by l_1, l_2,...,l_m. Binding sites of TFs (i.e. bins in test data) in test data are shown by $l_1^{'}$, $l_2^{'}$,...,$l_m^{'}{}^{'}$. The test data for another cell is a similar table with existing interactions between locations but unknown location of TF_n.

The goal is to predict the locations of TF_n in the second table.

As we explained in the previous chapter, we can estimate network parameters by network sampling and estimation methods. In this problem, The TFs bound along chromosomes

TFs /Locations	l_1	l_2	.	.	.	.	l_m
TF_1	1	0	.	.	.	.	0
TF_2	0	1	.	.	.	.	0
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
TF_n	0	0	.	.	.	.	1

Table 2.1: The training data for TFs locations

TFs /Locations	l_1'	l_2'	.	.	.	.	$l_{m'}'$
TF_1	1	0	.	.	.	.	0
TF_2	0	1	.	.	.	.	0
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
TF_{n-1}	0	1	.	.	.	.	1
TF_n	?	?	?	?	?	?	?

Table 2.2: The test data for TFs locations

have interactions with each other. We can create a network model based on the training data and HiC matrix including 161 vertices. Each vertex is the representative of a specific TF. The weight of edges between the vertices is representative of interaction tendency between TFs. Since we do not have access to all interactions between TFs along the genome, and from the available data we only sample part of that, to estimate the edge weight $w_{i,j}$, between the vertices i and j, we have to sample from a hidden network. Besides, since we sample from edges and estimate their weights, the method is design based approach to estimate edge characteristic. After we estimated the characteristic of the network, we can use this model to infer the test data. Let b_{TF_i} be a representation of TF_i locations. It is the set of all l_s where TF_i bounds i.e

$$b_{TF_i} = \{l_s : TF_i \; bound \; to \; \; l_s\}$$

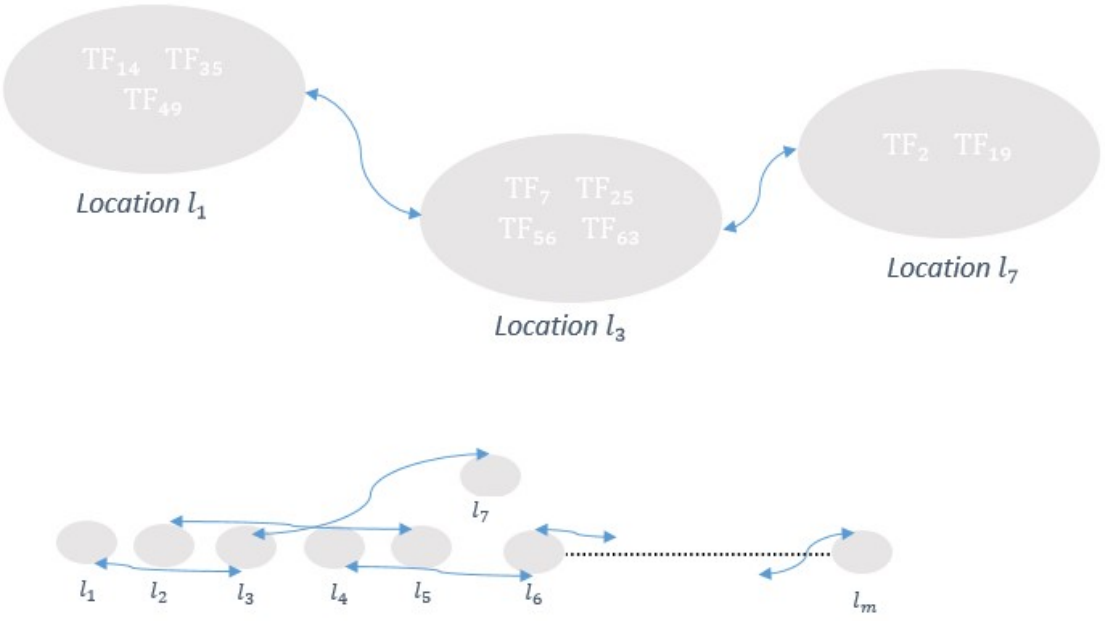


Figure 2.1: An example of HiC interactions between three different bins l_1, l_3 and l_7 when different TFs occupy them(up). The interactions between bins can be occurred randomly. An example of whole interactions map between different bins that obtained by HiC data.(bottom)

Also let $I_{l_s,l_t}(TF_i, TF_j) = 1$ if TF_i and TF_j at bins l_s and l_t respectively (i.e. $l_s \subseteq b_{TF_i} \& l_t \subseteq b_{TF_j}$) and l_s and l_t have an interaction in 3D structure (based on HiC data) otherwise $I_{l_s,l_t}(TF_i, TF_j)$ is set to zero. Let $I(l_s, l_t)$ be one when there is an interaction between l_s and l_t and also there is at least one TF in each l_s and l_t otherwise it is zero. Finally $N(l_s)$ is defined as the number of TFs which exists in bin l_s i.e $N(l_s) \geq 0$.

From Table 2.1 we can calculate the probability of TF_n existing at bin $l_s, 0 < s \leq m'$, in condition that there is an interaction between l_s and l_t and TF_i bounds in bin l_t. The probability $p(l_s \subseteq l_{TF_n} | l_t \subseteq l_{TF_n}, I(l_s, l_t) = 1)$ can be calculated in the following way. We calculate the interaction coefficient $W_{i,j}$ between two different TFs. We can define $W_{i,j}$ in two different version. The first version is more exact and needs more computation but the second version is an approximation of the first version and is less accurate while requiring less computation. Although we use the first version to solve the problem without any

computational issue, in cases calculating the first version cause computational problems, the second version can be considered as an alternative.

- Version 1 :

$$W_{i,j} = \sum_{s=1}^{m} \sum_{t=1, I_{l_s,l_t}(.)=1}^{m} \frac{I_{l_s,l_t}(TF_i, TF_j)}{\sum_{k=1}^{n} I_{l_s,l_t}(TF_k, TF_j)}$$

The value of $W_{i,j}$ in each chromosome is very dependent on the number of regions bound by TF_i and TF_j in that chromosome. On the other hand the number of each TF_i in each chromosome is very different. If the number of TF_i and TF_j in a chromosome is high, there is a greater chance to have a larger value of $W_{i,j}$ in that chromosome. To overcome on this issue and define a less dependent parameter on the number of TF_i in each chromosome, we can calculate the parameter $w_{i,j}$ which is a normalized version of $W_{i,j}$. For each TF_i in each chromosome, we add all tendency factor $W_{i,j}$, then divide each $W_{i,j}$ over this addition. Therefore, the value of $w_{i,j}$ is always between zero and one i.e. $0 \leq w_{i,j} \leq 1$. Thus comparing $w_{i,j}$s in different chromosomes gets more reasonable.

$$w_{i,j} = \frac{W_{i,j}}{\sum_{h=1}^{n} W_{h,j}}$$

We can assume $w_{i,j}$ as the interaction tendency between two different TFs, does not change from one chromosome to another chromosome. Based on this assumption, we can calculate $p(l_s \subseteq b_{TF_n} | l_t \subseteq b_{TF_j}, I(l_s, l_t) = 1)$ based on $w_{n,j}$.

$$p(l_s \subseteq b_{TF_n} | l_t \subseteq b_{TF_j}, I(l_s, l_t) = 1, N(l_s) = 1) = w_{n,j}$$

We can generalize this idea for more than one transcription factor to calculate $p(l_s \subseteq b_{TF_n} | l_t \subseteq b_{TF_i}, l_t \subseteq b_{TF_j}, I(l_s, l_t) = 1)$. Let $I_{l_s,l_t}(TF_i, TF_{(j_1,j_2)}) = 1$ if TF_i, TF_{j_1} and TF_{j_2} exist at bins l_s, l_t and l_t respectively (i.e. $l_s \subseteq b_{TF_i}$ and $l_t \subseteq b_{TF_{j_1}}, l_t \subseteq b_{TF_{j_2}}$) and l_s have an interaction with l_t in 3D structure (based on HiC data) otherwise

$I_{l_s,l_t}(TF_i, TF_{(j_1,j_2)})$ is set to zero. Therefore,

$$p(l_s \subseteq b_{TF_n} | l_t \subseteq b_{TF_{j_1}}, l_t \subseteq b_{TF_{j_2}}, I(l_s, l_t) = 1, N(l_s) = 1) = \frac{W_{n,(j_1,j_2)}}{\sum_{k=1}^{n} W_{k,(j_1,j_2)}} = w_{n,(j_1,j_2)}$$

where

$$W_{i,(j_1,j_2)} = \sum_{s=1}^{m} \sum_{t=1, I(l_s,l_t)=1}^{m} \frac{I_{l_s,l_t}(TF_i, TF_{(j_1,j_2)})}{\sum_{k=1}^{n} I_{l_s,l_t}(TF_k, TF_{(j_1,j_2)})}$$

If we can apply Bayes formula we have:

$$p(l_s \subseteq b_{TF_n} | l_t \subseteq b_{TF_{j_1}}, l_t \subseteq b_{TF_{j_2}}, I_{l_s,l_t} = 1, N(l_s) = 1) = \frac{A * B}{C}$$

where $A = p(l_s \subseteq b_{TF_n} | I(l_s, l_t) = 1, N(l_s) = 1)$, $B = p(l_t \subseteq b_{TF_{j_1}}, l_t \subseteq b_{TF_{j_2}} | l_s \subseteq b_{TF_n}, I(l_s, l_t) = 1, N(l_s) = 1)$, and $C = p(l_t \subseteq b_{TF_{j_1}}, l_t \subseteq b_{TF_{j_2}} | I(l_s, l_t) = 1, N(l_s) = 1)$.

If we calculate all of the terms A, B, and C based on the training set, the final result will be very biased toward training set. The reason is that all of the data are plugged in the formula are from same dataset, therefore the final answer is also very matched with that set. To avoid being biased with the training set we can calculate $p(l_t \subseteq b_{TF_{j_1}}, l_t \subseteq b_{TF_{j_2}} | I(l_s, l_t) = 1, N(l_s) = 1)$ based on the test set.

- Version 2 :

$$w_{i,j} = \frac{\sum_{s=1}^{m} \sum_{t=1, I_{l_s,l_t}(.)=1}^{m} I_{l_s,l_t}(TF_i, TF_j)}{\sum_{s=1}^{m} \sum_{t=1}^{m} I(l_s, l_t)}$$

The second version uses less computation to calculate $w_{i,j}$. Because it assumes if there is an interaction between two end of a TAD where each end include one of TF_i or TF_j, the HiC interaction originates from these two transcription factors without considering the impact of other TFs on this HiC interaction. This assumption simplifies the computation of $w_{i,j}$, but simultaneously is less accurate.

In our experimental results, we only used the first version. We consider two different cases. The first case is for $l_s = l_t$. In this case, for calculating the interaction tendency between two TFs, we consider them only when they locate on the same site. The second case is for $l_s \neq l_t$. In this case, for calculating the interaction tendency between two TFs, we consider them only when they locate on different sites.

2.1 Data Filtering

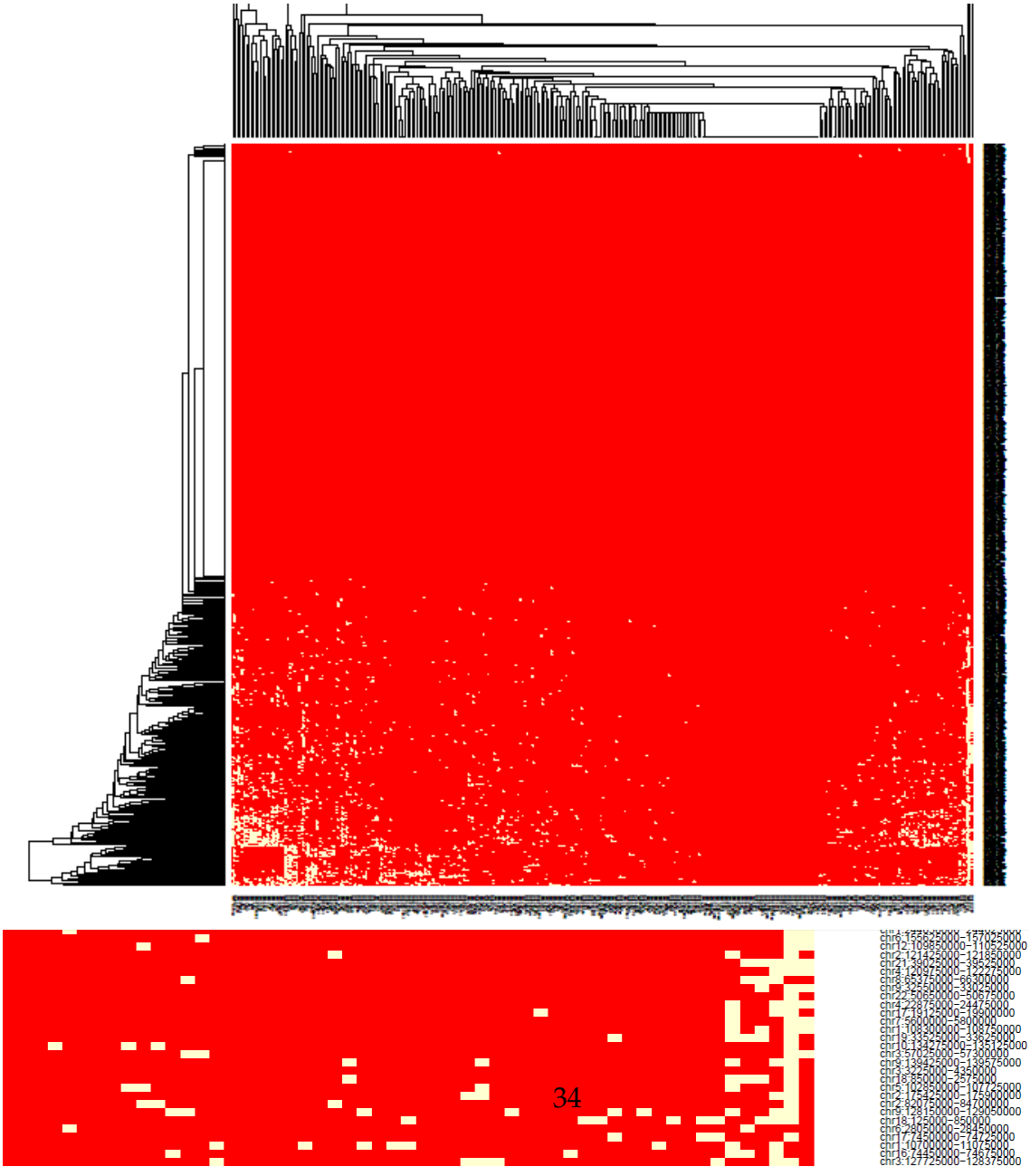
We consider those TFs located in $\pm 10000bp$ of the beginning or end of TAD's boundaries. Then we can create a frequency table for a pair of TFs situated at the two different boundaries of a TAD (i.e., one at the beginning and one at the end). Based on the number of TADs in the chromosome and hypergeometric distribution, we calculate the probability of randomly locating those TFs in the same TAD as explained in the following part.
A Hypergeometric distribution is a discrete probability distribution that models the probability of k successes in n randomly draws for objects with definite feature without replacement from a population N with K objects having that feature [26]. The hypergeometric test applies this distribution to measure the statistical significance of extracting a random sample with a constant number of k successes in n total draws from a population of size N containing K successes. In a test for over-representation of successes in the sample, the hypergeometric p-value is calculated as the probability of randomly drawing k or more successes from the population in n total draws. In this experiment we count the number of each possible pair of TFs located at the boundaries of TADs (each pair of TFs located at the boundaries of a specific TAD) and call it k. For instance, suppose there are k pairs of (TF_i, TF_j), with TF_i located at the start boundary of the TADs and TF_j located at the end boundary of the TADs. Let the total number of the TADs be N. If the total number of TF_i located at the start of a TAD is n_1 and the total number number of TF_j located at the end part of a TAD is n_2, let $K = min(n_1, n_2)$ that is the maximum possible pairs of TF_i, TF_j in this structure (i.e. maximum successes) and $n = n_1 + n_2 - k$ that is number of

TADs occupied by TF_i at the start boundary of TAD or TF_j at the end boundary of a TAD . Then we can calculate what is the probability that k pairs are selected randomly given K, n and N. Then based on the hypergeometric over-representation test, we calculate the probability that k as a number of these TFs pairs occurred randomly. Based on this test, if the p-value is more than 0.05, we ignore their interaction for the next part (i.e. we assume these number of pairs happened by accident) ; otherwise, we consider them. We also prepared a heatmap (clustering of TFs) based on these interactions. In these calculations, we distinguished between the start and the end of the TAD. The figure 2.2 show a part of this probability table. Based on the table some of values which are more than $.05$ (lighter color) are eliminated.

The heatmap figure (2.3) are shown on the next page. A heatmap (or heat map) is a method to observe hierarchical clustering. Heatmaps are a way to analyze both clusters of samples and features together. Hierarchical clustering is done for both the rows and the columns of the data matrix. Then columns and rows of the data matrix are rearranged based on the hierarchical clustering result, where similar observations get close to each other. A color scheme is used to show the data matrix based on the high and low values of data [61]. Therefore, in the figure (2.3) columns which are different parts of chromosomes and rows which are TFs are rearranged in such a way similar TFs in terms of their binding sites get close to each other. As we see in figure (2.3) $CTCF_{first}$ and $RAD21_{first}$ are beside each other in the corner. In next sections we see RAD21 between all TFs has the most interactions with CTCF. We also see TFs with more frequencies have more brighter colors in the map which shows the number of their binding sites in the different part of chromosomes.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Protein Table	ZBTB33	CEBPB	CTCF	TAF1	GABPA	USF1	SP1	EGR1	FOXA1	RUNX3	MAZ	RAD21	SMC3	MAFF	MAFK	BHLHE40	FOSL2	JUND
2	ZBTB33	0.601641887	0.120392133	0.349822	0.271785	0.011945	0.026703	0.070505	0.004371	0.803053	0.508572	0.00618	0.238486	0.482801	0.434477	0.552177	0.021772	0.738417	0.764796
3	CEBPB	0.096226987	0.001397207	8.08E-05	0.052071	0.072549	0.005373	0.004784	0.002105	0.001817	0.514062	0.058598	0.000448	0.01829	0.264931	0.332654	0.001351	0.259747	0.089839
4	CTCF	0.040022701	0.001305771	4.76E-05	6.41E-06	0.009335	3.90E-07	0.000216	9.00E-05	0.004856	0.543583	0.004207	0.00081	0.117764	0.80218	0.002636	0.036192	0.004939	2.00E-05
5	TAF1	0.088985973	9.75E-06	0.001555	0.072499	0.198355	0.001918	0.009725	0.000928	0.019664	0.118725	0.000197	0.026075	0.006363	0.810395	0.534098	0.308463	0.077964	0.004857
6	GABPA	0.326500148	0.001584452	0.053925	0.001729	0.079184	0.012708	0.015102	0.000807	0.35507	0.052466	4.52E-05	0.338224	0.476606	0.257229	0.524392	0.13417	0.008045	0.185924
7	USF1	0.006271416	0.00058348	0.000135	0.017225	0.000357	2.04E-05	0.027303	0.000318	0.621474	0.709571	0.028162	0.000263	0.021972	0.584912	0.661804	0.051862	0.04793	0.006391
8	SP1	0.089127937	0.000118678	2.42E-06	1.76E-06	0.040842	0.000668	4.25E-06	0.000541	0.027052	0.029203	0.002714	0.000243	0.020607	0.438249	0.237472	0.055093	0.009595	0.163585
9	EGR1	0.015497549	0.057622715	0.000127	2.82E-05	0.279044	0.002092	0.000584	5.26E-06	0.075113	0.001486	8.66E-06	0.001468	0.186849	0.900501	0.259704	0.03903	0.616783	0.016576
10	FOXA1	0.114239397	0.619782895	0.11093	0.202879	0.028717	0.071088	0.080959	0.459717	0.200449	0.58832	0.021762	0.086589	0.142395	0.107578	0.934717	0.219944	0.694328	0.01452
11	RUNX3	0.001189309	0.011195787	0.219986	0.264616	0.45627	0.027927	0.243391	0.059001	0.223048	0.312934	0.533433	0.163087	0.253321	0.675245	0.956186	0.649101	0.156972	0.088981
12	MAZ	0.177148882	0.017320371	5.51E-05	0.045816	0.006796	0.161107	0.001958	0.005626	0.186471	0.171391	1.68E-07	0.0082	0.097332	0.947453	0.748533	0.171613	0.028896	0.171595
13	RAD21	0.01577725	0.044071338	0.001474	0.006823	0.051193	0.006358	0.004763	0.000366	0.007948	0.829987	0.110696	0.029598	0.087085	0.290617	0.012592	0.023164	0.223076	0.017941
14	SMC3	0.699637127	0.025761582	0.209251	0.050452	0.459959	0.12393	0.079955	0.228088	0.016155	0.983153	0.396512	0.027151	0.558184	0.228571	0.101632	0.123716	0.100127	0.013472
15	MAFF	0.072744765	0.044294808	0.331391	0.088229	0.617516	0.259466	0.481743	0.721045	0.137015	0.345536	0.389534	0.894527	0.711232	0.043793	0.082553	0.144226	0.857687	0.459141
16	MAFK	0.627862478	0.057925801	0.804736	0.796847	0.398859	0.677736	0.661556	0.032084	0.000476	0.953801	0.421469	0.061322	0.633492	0.772569	0.824064	0.439636	0.67409	0.231853
17	BHLHE40	0.602964364	0.010976322	0.017513	0.040782	0.081128	0.069198	1.87E-05	0.000165	0.424194	0.009015	0.00436	0.398041	0.105314	0.361765	0.555311	0.022796	0.029027	0.175292
18	FOSL2	0.16091643	0.415755079	0.006624	0.10747	0.684539	0.369725	0.145135	0.02042	0.986015	0.711669	0.440408	0.221732	0.106694	0.589156	0.510302	0.857252	0.451719	0.090105
19	JUND	0.457968474	0.07400143	0.000518	0.00573	0.004743	0.021888	0.050362	0.026313	0.406638	0.167384	0.219808	0.002614	0.002852	0.727152	0.938382	0.022248	0.25041	0.000985
20	E2F6	0.004397071	0.000178955	0.001728	0.000173	0.000428	0.000169	2.75E-06	1.57E-06	0.260625	0.053438	0.001714	0.008229	0.012679	0.458721	0.90218	0.0295	0.023098	0.003872
21	MAX	0.018575309	1.06E-06	0.005234	0.004156	0.00844	1.85E-06	4.81E-06	0.000243	0.000465	0.065432	0.000718	0.018443	0.019608	0.432427	0.348554	0.445138	9.89E-06	0.00043
22	POLR2A	0.043479078	1.34E-06	7.28E-08	0.003484	0.001161	9.24E-06	0.000323	3.21E-06	0.002294	0.102706	4.39E-07	1.09E-05	0.000521	0.357242	0.398356	0.07792	0.037462	0.000584
23	PAX5	0.773608958	0.015071377	0.019048	0.131131	0.366627	0.028085	8.96E-06	0.01246	0.287834	0.316733	0.001005	0.044331	0.005926	0.39815	0.454861	0.022336	0.015511	0.121332

Figure 2.2: Part of p-value table for a pair of TFs in TADs structure. The colorful value is less than threshold value 0.05



2.2 Experimental Results

The objective of the practical experiments is to apply and evaluate the theoretical formula of the previous sections on the real dataset. We divide the whole data set into two parts: The training set and the test set. The chromosome data alternatively correspond to the training and test set. The data of Chromosome one, three, five,... up to chromosome X are used as the training set and the data of chromosome two, four, six,.. ,twenty two , and Y are used for the test set. Since the size of the chromosome is decreasing from chromosome one to chromosome Y, this data partition is done to make close the number of total TFs in the training and test set. Some TFs still do not appear in one of these sets. This fact affects the prediction accuracy for such TFs and decreases the average prediction accuracy. We assume there are only interactions between the start and end of each TADs along chromosomes. We show the frequency of TFs in the training and test set in the figure 2.4.

We have already collected TFs information at the start and end of each TAD. We only

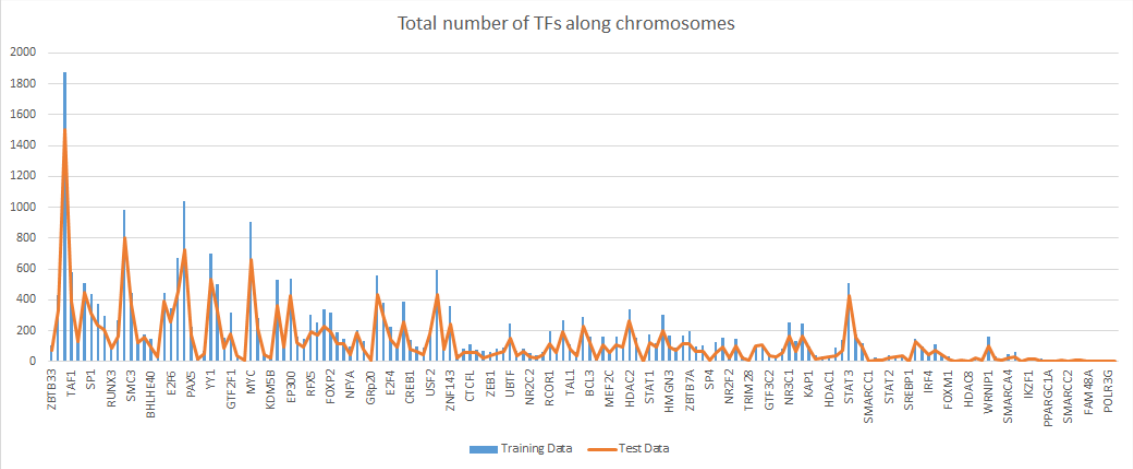


Figure 2.4: The comparison between the number of regions bound by each TFs in the training and test set

consider such TFs in our calculation. We calculate $W_{i,j}$ and $w_{i,j}$ for any two TF_i and TF_j for both training and test sets. We also calculate the number of each TF in each chromosome part. The interesting observation for this parameter is that although $W_{i,j}$ is very different between the training and test set, the normalized version $w_{i,j}$ are more similar in

such a way that 90 percent of them are close with difference less than .01. The dissimilarity between $W_{i,j}$s was predictable since the number of each TF in the training and test set are very different, and normalizing helps eliminate this factor and create new parameter that can have approximately same value in the training and test set.

Protein Ta	ZBTB33	CEBPB	CTCF	TAF1	GABPA	USF1	SP1	EGR1	FOXA1	RUNX3	MAZ	RAD21	SMC3	MAFF	MAFK	BHLHE40	FOSL2	JUND	E2F6	MAX	POLR2A	PAX5
ZBTB33	0.00179	0.01599	0.10186	0.01088	0.00154	0.01693	0.02213	0.01121	0.01495	0.00617	0.00261	0.05299	0.02817	0.01173	0	0.00143	0.00211	0.01988	0.01435	0.01939	0.03672	0.003
CEBPB	0.00479	0.02705	0.08678	0.02027	0.0068	0.02652	0.01587	0.00745	0.01062	0.00686	0.00499	0.04147	0.0214	0.00925	0.01102	0.00501	0.00112	0.01899	0.01091	0.02044	0.0379	0.009
CTCF	0.00655	0.02959	0.09861	0.01786	0.0053	0.02283	0.01508	0.00995	0.01132	0.00333	0.00822	0.04675	0.02016	0.00992	0.00772	0.0043	0.00193	0.02389	0.01038	0.01695	0.0342	0.005
TAF1	0.00409	0.0297	0.12931	0.01185	0.0038	0.02084	0.01997	0.01481	0.01525	0.0079	0.00349	0.06617	0.02353	0.01986	0.01328	0.00292	0.0016	0.02389	0.00944	0.01262	0.02784	0.004
GABPA	0.01727	0.01495	0.09556	0.00986	0.00379	0.01917	0.01428	0.00352	0.01377	0.00367	0.01335	0.05008	0.02658	0.01272	0.0156	0.00853	0.00118	0.04809	0.01256	0.02463	0.03448	0.004
USF1	0.00457	0.01546	0.0907	0.01538	0.00614	0.02075	0.0166	0.01107	0.0066	0.00475	0.00582	0.04081	0.01449	0.00955	0.01007	0.00328	0.0003	0.01836	0.01074	0.02096	0.02973	0.005
SP1	0.0055	0.01936	0.09055	0.01759	0.00589	0.01471	0.01755	0.00847	0.01361	0.00298	0.00753	0.0411	0.01858	0.00973	0.01018	0.00489	0.00254	0.0225	0.01423	0.02272	0.03118	0.007
EGR1	0.00576	0.02924	0.10807	0.01277	0.00795	0.02096	0.01236	0.01085	0.01033	0.0032	0.00419	0.04577	0.01586	0.00764	0.00818	0.00632	0.00227	0.01989	0.01572	0.01941	0.03583	0.004
FOXA1	0.00912	0.02839	0.12475	0.01043	0.00354	0.0186	0.0221	0.01085	0.00774	0.00572	0.00617	0.05174	0.01901	0.01045	0.0196	0.00311	0.0014	0.02444	0.00642	0.02401	0.0455	0.006
RUNX3	0.00154	0.03812	0.11603	0.01568	0.0037	0.00742	0.01929	0.01489	0.0096	0.00476	0.00749	0.0393	0.00645	0.00133	0	0.00271	0	0.02477	0.01071	0.02424	0.02927	0.004
MAZ	0.00209	0.01974	0.11639	0.01521	0.00286	0.01605	0.01479	0.00874	0.01037	0.0031	0.01156	0.04497	0.01656	0.00978	0.00252	0.00239	0.0005	0.02517	0.01323	0.01782	0.03475	0.007
RAD21	0.00577	0.02869	0.10431	0.01801	0.00441	0.02183	0.01687	0.00902	0.01061	0.00282	0.0061	0.05302	0.02215	0.0106	0.0112	0.00352	0.00148	0.02624	0.0104	0.01744	0.03711	0.005
SMC3	0.00283	0.02515	0.10789	0.01878	0.0067	0.01984	0.01459	0.00723	0.01322	0.00483	0.00483	0.04219	0.01366	0.01115	0.00475	0.00332	0.00303	0.02662	0.01092	0.01354	0.03623	0.007
MAFF	0.01202	0.01652	0.06978	0.02085	0.00468	0.02234	0.01472	0.01181	0.00913	0.00346	0.00285	0.03466	0.02123	0.0263	0.01683	0.0056	0.00198	0.0246	0.01462	0.01907	0.04809	0.003
MAFK	0.00258	0.03425	0.13225	0.01516	0.00395	0.02279	0.0075	0.00841	0.0022	0.00108	0.00106	0.08004	0.03679	0.02805	0.01179	0.0023	0.00178	0.00896	0.00962	0.02136	0.04606	0.00
BHLHE40	0.01243	0.0401	0.07757	0.00635	0.00297	0.02344	0.01183	0.01017	0.00884	0.00534	0.00407	0.04725	0.01726	0.01006	0.00473	0.00403	0.00093	0.04784	0.01473	0.01463	0.02525	0.009
FOSL2	0	0.02268	0.13281	0.02833	0.00321	0.02085	0.02833	0	0.00244	0.01473	0.00565	0.06671	0.0241	0	0	0.00244	0.00673	0.01015	0.02377	0.01704	0.01704	0.009
JUND	0.00357	0.01972	0.09637	0.01652	0.0046	0.02178	0.01064	0.00962	0.01281	0.00733	0.00811	0.04535	0.02192	0.00912	0.01259	0.00275	0.00113	0.02416	0.01215	0.01909	0.03448	0.005
E2F6	0.00095	0.02186	0.09636	0.02163	0.00606	0.0314	0.01641	0.01251	0.00987	0.00203	0.00563	0.04539	0.01825	0.00482	0.00811	0.00337	0.00195	0.02293	0.02217	0.01777	0.03009	0.005
MAX	0.00537	0.02253	0.11048	0.01463	0.00495	0.02358	0.0173	0.00956	0.01285	0.00549	0.00435	0.04907	0.01951	0.00922	0.00925	0.00558	0.00193	0.02955	0.01546	0.01706	0.02602	0.006
POLR2A	0.00552	0.02265	0.10116	0.01804	0.00901	0.02166	0.01494	0.01044	0.01055	0.00313	0.00506	0.04555	0.01782	0.01225	0.01011	0.00159	0.00156	0.02133	0.01112	0.01673	0.03153	0.00
PAX5	0.00388	0.03075	0.09176	0.01425	0.00311	0.0234	0.0099	0.01032	0.01906	0.00459	0.00424	0.04006	0.02388	0.02069	0.0155	0.00911	0.00272	0.02836	0.0126	0.01357	0.03852	0.005
PHF8	0	0.02873	0.08605	0.0352	0	0.03957	0.01705	0.01326	0.01566	0.03957	0	0.01566	0	0.00919	0.01432	0	0.00407	0.02734	0.00407	0.01705	0.03784	0.004
PML	0.0045	0.05031	0.12391	0.01614	0.00764	0.02827	0.03307	0.00499	0.00418	0.0035	0.00768	0.018	0.02898	0.00269	0	0.00148	0	0.04496	0.02151	0.00418	0.01452	0.001
YY1	0.00676	0.02233	0.10954	0.01569	0.00539	0.02069	0.01271	0.00864	0.00862	0.00517	0.00362	0.05154	0.02441	0.01374	0.00976	0.00355	0.00109	0.02526	0.01117	0.01619	0.02899	0.005

Protein Ta	ZBTB33	CEBPB	CTCF	TAF1	GABPA	USF1	SP1	EGR1	FOXA1	RUNX3	MAZ	RAD21	SMC3	MAFF	MAFK	BHLHE40	FOSL2	JUND	E2F6	MAX	POLR2A	PAX5
ZBTB33	0.00836	0.02044	0.10867	0.01531	0.00624	0.0236	0.01692	0.02069	0.01025	0.00865	0.00698	0.04693	0.00567	0.02193	0.01262	0.00276	0.00108	0.00714	0.01134	0.0169	0.02531	0.002
CEBPB	0.00599	0.02083	0.08152	0.01859	0.00717	0.02134	0.01961	0.01025	0.00671	0.00472	0.00685	0.03554	0.01226	0.01109	0.00955	0.00451	0.00089	0.0127	0.00992	0.02209	0.03785	0.006
CTCF	0.00565	0.023	0.09053	0.01758	0.00675	0.01685	0.01415	0.01112	0.01327	0.00311	0.00874	0.04056	0.01544	0.007	0.00565	0.00455	0.00189	0.01817	0.01029	0.02033	0.04008	0.009
TAF1	0.0024	0.01962	0.09888	0.01473	0.00859	0.01609	0.01845	0.01297	0.0079	0.00297	0.00917	0.03983	0.01619	0.00896	0.00685	0.00422	0.00138	0.01635	0.01355	0.02055	0.03468	0.006
GABPA	0.00142	0.01727	0.08554	0.01193	0.01392	0.01889	0.01594	0.00929	0.01563	0.00172	0.00611	0.02917	0.01133	0.00219	0.00979	0.00236	0	0.01169	0.01547	0.01961	0.03999	0.004
USF1	0.0062	0.02671	0.08449	0.01895	0.00527	0.02407	0.01552	0.01223	0.0099	0.00393	0.00751	0.03732	0.01677	0.00593	0.0046	0.0034	0.0016	0.01315	0.01276	0.02431	0.04159	0.005
SP1	0.00347	0.0118	0.08842	0.01445	0.00505	0.0149	0.01686	0.01368	0.00923	0.00527	0.01102	0.03895	0.01423	0.0059	0.00531	0.00858	0.00086	0.01925	0.01288	0.02028	0.03381	0.011
EGR1	0.00371	0.01656	0.08613	0.01828	0.00683	0.02345	0.0163	0.01512	0.0048	0.00302	0.01119	0.04334	0.01409	0.00451	0.01042	0.00682	0.0021	0.0157	0.01241	0.02117	0.0398	0.006
FOXA1	0.00318	0.01925	0.08808	0.0212	0.006	0.01354	0.01246	0.01284	0.00824	0.00422	0.00769	0.04551	0.02033	0.00771	0.0145	0.00227	0	0.0134	0.01324	0.02484	0.03919	0.004
RUNX3	0.00722	0.00565	0.07953	0.01741	0.01027	0.01048	0.01168	0.0216	0.00545	0.0005	0.00653	0.02896	0.00759	0.02097	0.00143	0.01016	0.00091	0.01288	0.0083	0.01822	0.04076	0.003
MAZ	0.00402	0.01615	0.08083	0.01772	0.01112	0.02137	0.01429	0.0166	0.0106	0.00152	0.01139	0.02814	0.01206	0.00566	0.01008	0.005	0.00126	0.01478	0.01126	0.01966	0.0425	0.005
RAD21	0.00494	0.02235	0.08384	0.01764	0.00476	0.02083	0.01363	0.01175	0.01374	0.00467	0.00953	0.03702	0.01747	0.00472	0.0097	0.00362	0.00182	0.01626	0.01074	0.01886	0.03833	0.007
SMC3	0.00643	0.02244	0.08031	0.01504	0.00557	0.02201	0.0147	0.01445	0.01352	0.00358	0.0096	0.04035	0.01458	0.00549	0.01056	0.00377	0.00095	0.01986	0.0117	0.02188	0.043	0.007
MAFF	0.01464	0.03645	0.09109	0.01354	0.0046	0.01025	0.0117	0.00544	0.0183	0.00267	0.00198	0.04659	0.01952	0.0181	0.01076	0.00075	0.00127	0.01296	0.00885	0.0216	0.0362	0.009
MAFK	0.00609	0.02943	0.09447	0.0128	0.00618	0.01106	0.0151	0.01166	0.00962	0.00346	0.00861	0.04662	0.01792	0.00571	0.00206	0.00525	0.00029	0.01193	0.01181	0.02205	0.03092	0.007
BHLHE40	0.01088	0.02493	0.0842	0.01417	0.01496	0.02948	0.01494	0.01366	0.01619	0.00054	0.00863	0.04057	0.01761	0.00257	0.008	0.00565	0	0.01191	0.01258	0.01445	0.03378	0.005
FOSL2	0.0016	0.00572	0.05631	0.01559	0.01521	0.01687	0.02374	0.00854	0.00805	0.001	0.00858	0.02554	0.01907	0.01009	0.00549	0.0092	0	0.01127	0.00787	0.03978	0.03329	0.009
JUND	0.00344	0.01749	0.1124	0.01831	0.00408	0.01716	0.01266	0.0111	0.01142	0.00145	0.00376	0.04177	0.01364	0.0055	0.00997	0.00537	0.00401	0.01556	0.01217	0.02276	0.03082	0.005
E2F6	0.00346	0.00799	0.0878	0.01423	0.00655	0.02384	0.02071	0.01255	0.01087	0.00374	0.01051	0.04284	0.01258	0.00754	0.00606	0.00862	0.00134	0.01291	0.01041	0.01961	0.03838	0.006
MAX	0.00274	0.01616	0.08796	0.01879	0.00928	0.02202	0.01703	0.01301	0.01015	0.00318	0.00864	0.0411	0.01494	0.00644	0.00684	0.00426	0.00154	0.01507	0.01253	0.02344	0.03536	0.007
POLR2A	0.00679	0.01941	0.08716	0.01868	0.0075	0.02085	0.01675	0.01227	0.00757	0.00332	0.0078	0.03467	0.0176	0.00488	0.00755	0.00447	0.00184	0.01552	0.01409	0.02512	0.04036	0.005
PAX5	0.00793	0.01139	0.12169	0.01587	0.00847	0.02613	0.01239	0.01232	0.00273	0.00299	0.00532	0.04854	0.0117	0.00939	0.01088	0.0046	0.00049	0.01095	0.01612	0.02061	0.0416	0.012
PHF8	0.0132	0.01221	0.15078	0	0	0.01592	0.03228	0.01592	0.00937	0	0	0.10228	0.00937	0.01908	0	0	0	0	0.01592	0.02158	0.01592	0.009
PML	0.0022	0.00229	0.07087	0.01833	0.00378	0.03857	0.0188	0.0146	0.00601	0.00357	0.01376	0.02063	0.01859	0.00316	0	0.00962	0.00151	0.0176	0.01577	0.01856	0.03664	0.0
YY1	0.0046	0.02098	0.08777	0.01663	0.00706	0.01668	0.01522	0.01484	0.00826	0.00336	0.01079	0.03968	0.01582	0.00618	0.01071	0.00476	0.00049	0.01328	0.01209	0.024	0.03699	0.006

Figure 2.5: The values of $w_{i,j}$ in training(bottom) and test(up) set are very close.

The next step to figure out if there is any reasonable relationship between $w_{i,j}$ and the probability of existing TF_i and TF_j in the same TAD. Based on the definition $p(TF_n \in l_s/TF_j \in l_t, I_{l_s,l_t} = 1 \& N(l_s) = 1)$, to satisfy the conditions in this probability, we need to consider only those TADs with at most one TF at the one end. The number of such TADs is very low. In all of them, the value of $w_{i,j}$ is low. Based on the definition of probability, we need a high number of experiments to approximate the probability value (i.e. greater than 100). Lack of this situation in this simulation and simultaneously existing impacts on the experiments make it difficult to accurately obtain the probability. That's why we change our direction a little bit. Instead of finding the probability for each bin i.e.

$\{l_t|1 \leq t \leq m^{'}\}$, we can assign a score. Then we can rank the bins based on these scores which show the tendency of TF_n to attach the bins.

The tendency of TF_n to attach a bin is dependent on presence of other TFs nearby to that bin. In this research we assume other TFs can interact with TF_n only when they are present in the same TAD. We can assume TF_n select its binding sites to have interactions with other TFs. In this work, we only consider TFs relative positions to each other as a factor to predict binding sites. Therefore, the bin's score can be approximated by $\sum_{i_j=1}^{k} w_{i_j,n}$ where k is the number of TFs in that bin . We assume here $w_{i_j,n}$ for each i_j does not change in the presence of other TFs. Consequently, we make a ranking between bins preferred to be occupied by TF_n in each chromosome. It is worth mentioning here that the whole system is dynamic, and the bins of each TFs change continuously during the time, and for one single TF, selecting a bin is done locally (selecting a bin between its close neighborhood) rather than globally. Therefore for a fixed number of TFs along a chromosome, there may be potential bins more than three times that fixed number.

Creating ranking can be applied to multiple factors one by one, and we investigate if the overlap between the potential bins (bins with higher scores that is predicted to be occupied by specific TF) and occupied ones is increasing. If it is not increasing, we can go back and reformulate the factor. There are two main issues which we face in this problem.

1- The designed parameter for each factor in the training and test set are not close.

2- Designing a parameter that adequately describes the factor so that interfering with this parameter in the bins ranking increases the overlap between the potential bins and occupied ones.

To handle these two issues independently, we can try the experiment on both the test and training set to distinguish the cases where the first issue occurs. Reasonably it seems the second issue is more challenging than the first one, and if it is solved, the first issue can be handled to some extent. Here we study experimental results in two different cases and compare them with each other.

2.2.1 1st case

As we explained before, in the 1st case we only consider the interactions between those TFs that are located in the same bins. We calculate $W_{i,j}$ only based on these interactions. In the first case, it seems the interaction tendency between TFs is more robust than in other cases. This fact originates from short distances between TFs in this case that makes their interactions stronger. We obtain $w_{j,i}$ in both training and test sets. The values of $w_{j,i}$ in both training and test set as we compared them in the figure 2.6.

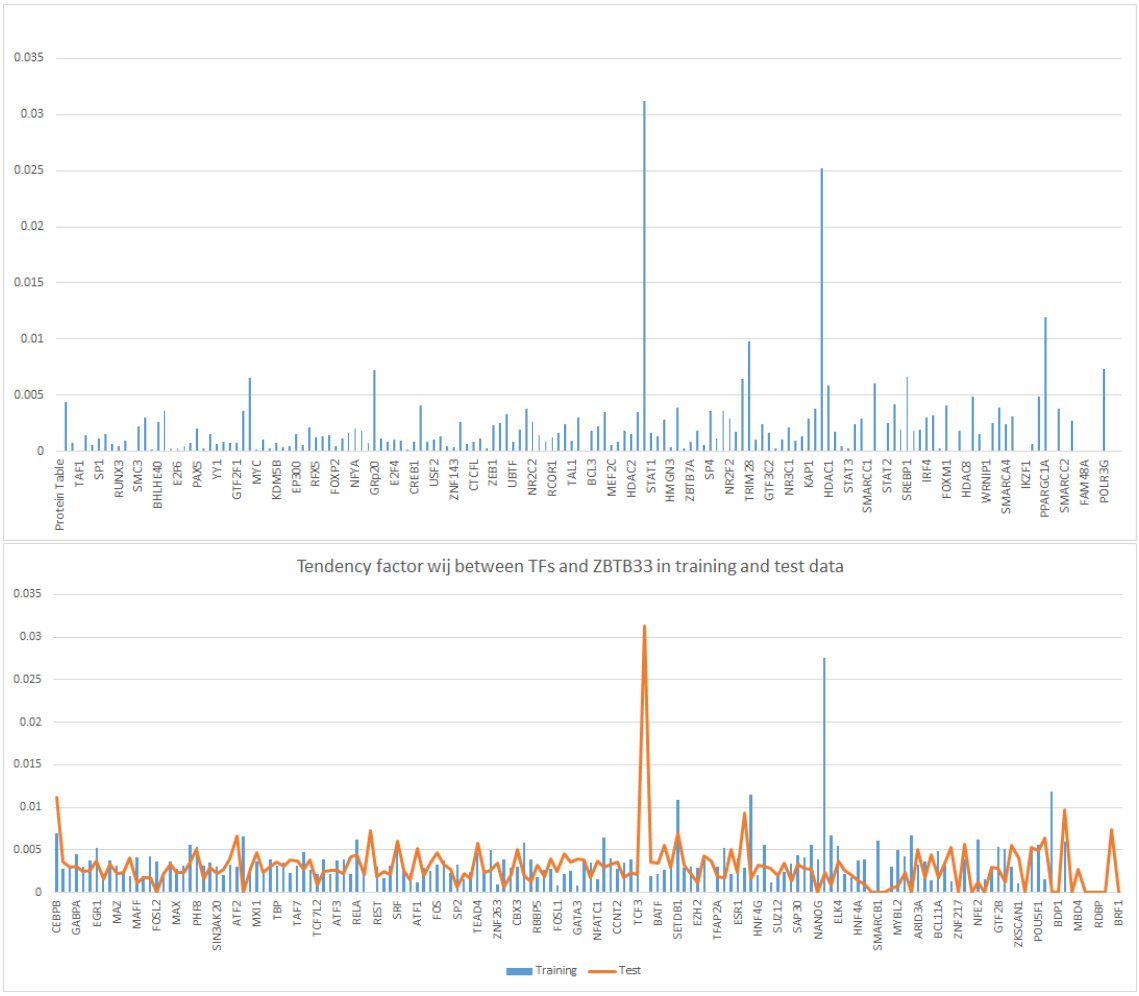


Figure 2.6: The absolute difference of interaction tendencies $|(w_{j,1} \ |_{Training\ set}) - (w_{j,1} \ |_{Test\ set})|$ between TF_1 which is ZBTB33 and other TFs in training and test set(up). Tendency factor $w_{i,1}$ between ZBTB33 and other TFs in training and test data set (bottom).

We started our experiments to calculate $w_{i_j,i}$s as the first factor, and it can handle the first issue. Ranking bins for each TF_i is based on the sum of $w_{i_j,i}$ of any TF_{i_j} that potentially can have an interaction with TF_i, $\sum_{j=1}^{k} w_{i_j,i}$ where k is the number of TFs can have an interaction with TF_i (i.e. based on dataset they locate in other side of TAD that have interaction with that bin). You can calculate the approximate likelihood by normalizing this sum at each chromosome and multiplying with the number of TF_i at that chromosome over the total number of possible bins in that chromosome. The number of each TF_i and possible bins is the same in a specific chromosome. Thus, we only need to calculate the sum of $w_{i_j,i}$s to rank the bins in the specific chromosome. To rank bins for each TF_i in the specific chromosome, we suppose that the TF_i locations are unknown. Then we rank the bins based on the position of the other TFs in that chromosome. To do that, for each bin, we add $w_{i_j,i}$s calculated in training data if TF_{i_j} locates in that bin. We call this summation as the score of that site for TF_i. Based on this score, we can rank all the bins inside the chromosome. After ranking, we calculate a range of possible areas for TF_i. This range is a number between the total number of TF_i in the chromosome and the total number of possible bins in the chromosome because the locations of TF_i is dynamic. We supposed the range R is (2*number of TF_i in the chromosome + the total number of bins in the chromosome)/3. This range is selected in such a way that in both cases that $\frac{\text{the number of } TF_i \text{ in the chromosome}}{\text{the total number of bins in the chromosome}}$ is small or large gives a reasonable value of range between the number of TF_i in the chromosome and the total number of bins in the chromosome. Then we calculate what percentage of TF_is exists in the range, and this number shows the true prediction ratio of our model. This number is called the true prediction ratio of the model. For the first case, the average of this number is about 70 percent, while if TF_i is distributed independently from the tendency weight, the average true prediction ratio should be 34 percent that is far from 70 percent. We can see the comparison between the true prediction ratio and the random prediction in the figure 2.7.

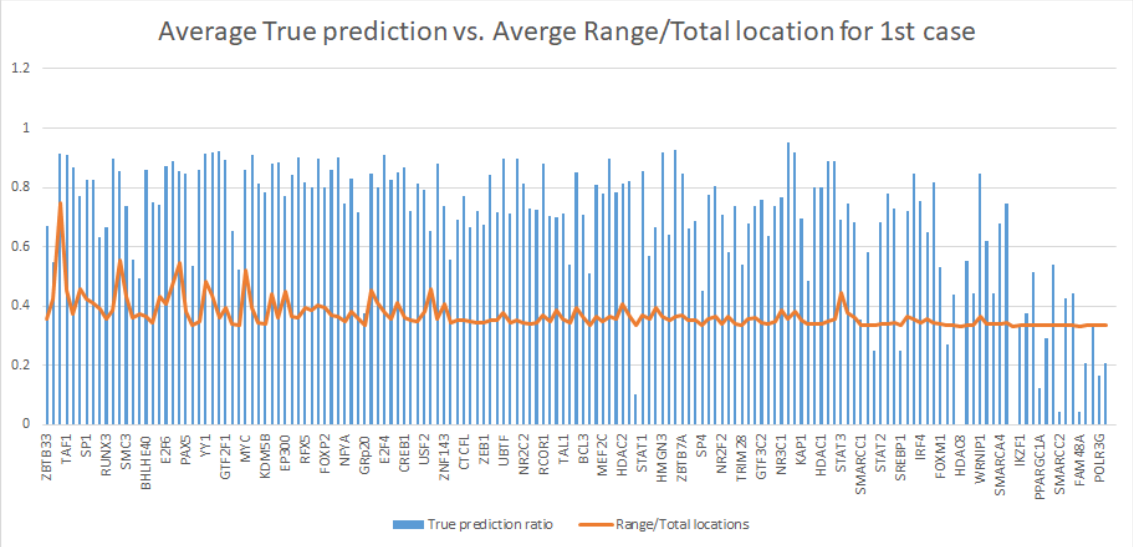


Figure 2.7: The average true prediction ratio vs. range over the total bins $\frac{R}{T}$ which shows the prediction of random ranks for bins. T is the total number of bins in the chromosome.

The factors that heavily impacts our prediction is the number of TF_i in each chromosome. As we showed in the theory section, the chance of TF_i to attach each bin is proportional to each TF_i in the chromosome. We can show this fact in the figure 2.8. The true prediction ratio average is the highest in the first chromosomes, where more number of TFs exists on average.

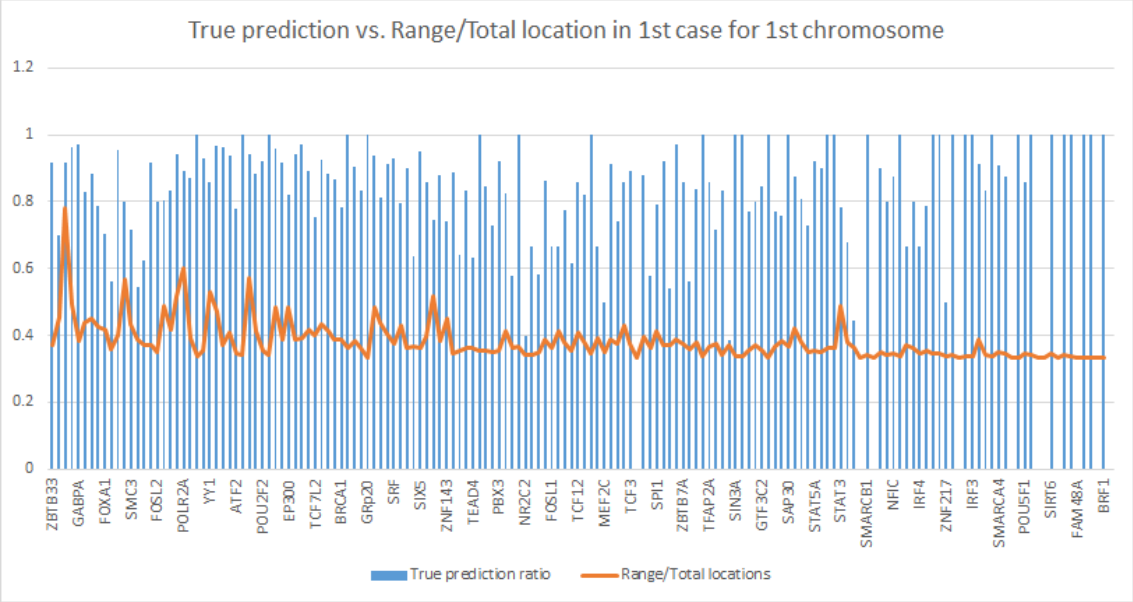


Figure 2.8: The average true prediction ratio vs. range over the total bin shows the prediction of random ranks for bins.

Besides, in each chromosome, the true prediction ratio is much better for those TFs with higher frequency in that chromosome. It is explained in the previous section to have a better output from the probability model; we need more frequencies of TFs in the training and test set. We can see this fact for the first and the second chromosome in the figure 2.9.

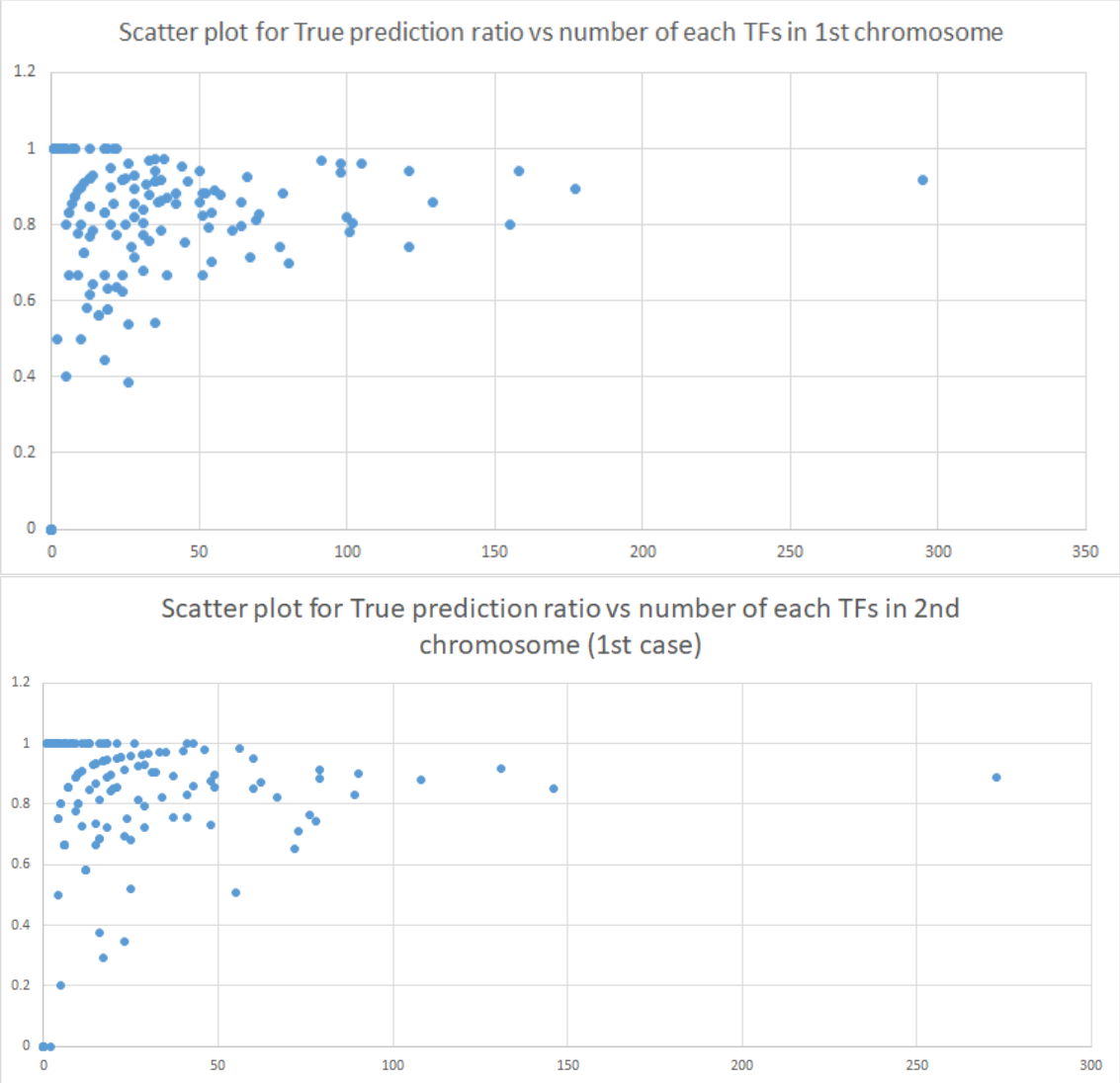


Figure 2.9: The scatter plot of true prediction ratio (y-axis) in terms of number of TFs in the first and second chromosome (x-axis) shows increasing TFs results in a better prediction.

Therefore, if we have more frequencies for each TF_i, we have a more accurate prediction for its interaction tendency with other TFs. Thus, to have a better comparison we can multiple this factor to the interaction tendency $w_{j,i}$. For CTCF, we can see this matter in the next figure. Based on the figure 2.10, there is a TF which has the most interaction tendency in both training and test set. This TF is RAD21. We can conclude RAD21 have the most tendency to have interactions with CTCF. This conclusion is validated in ChIA-PET

experiments as well [53]. Through the same approach, we can find more pair of TFs with the most interaction tendency to initialize regulatory functions.

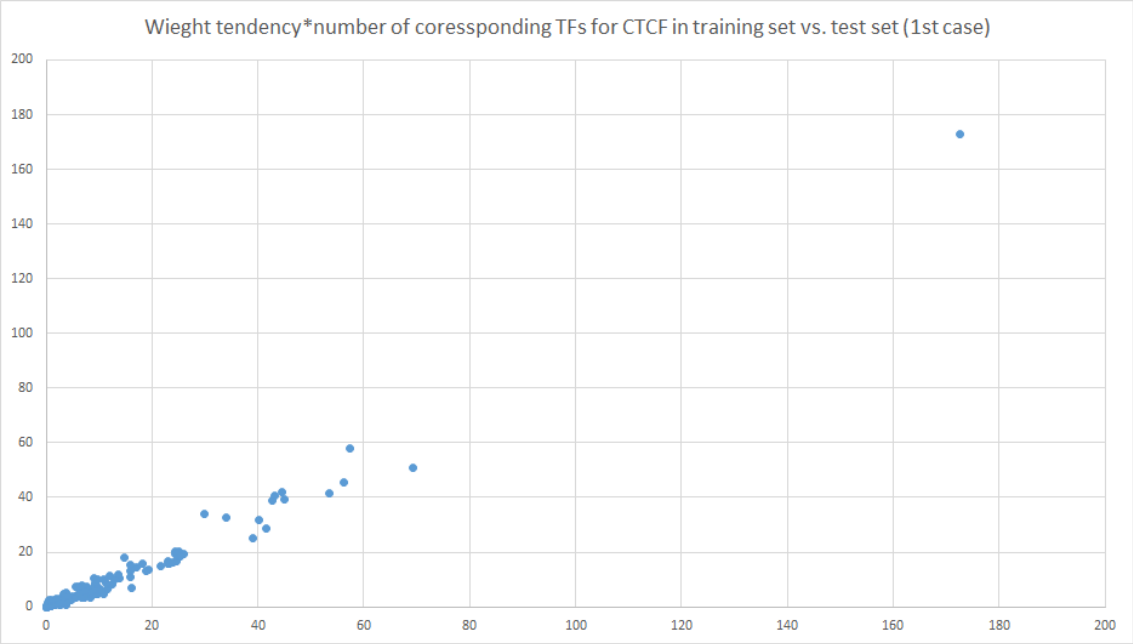


Figure 2.10: interaction tendency multiplied by the number of corresponding TFs for CTCF in training set vs. test set (1st case) for different TFs. It is maximized for RAD21 in both training and test sets.

Local Ranking

As we mentioned before, TFs most likely select their location locally rather than globally. Therefore it may impacts the ranking procedure, which was done globally before. Local ranking means to rank bins based on their scores which compete in the local range around each bin (i.e. amount of the range is specified exactly in each ranking procedure). That was why we also tried local ranking to compare the results with global ones. The local ranking procedures can be performed in different ways. We start with a tournament style. Two neighbor bins (i.e., the start and end of a TAD) compete with each other, and the bin with the higher score goes to the next step. In each step, the winners of previous steps compete locally with each other, and half of them go to the next step until the final winner

gets definite. Local competing means each bin competes with one of its two neighbors in that stage located just before or after that bin. To start this procedure perfectly, we need 2^n bins to begin. Since the number of bins in each chromosome is not exactly the power of two, we need to add some artificial bins with an average score to balance the total number of bins. After finishing the competition, we start ranking the bin from the last stage (tournament final) to the first stage. In this procedure, we do not give any rank to those artificial bins. We can calculate the local prediction ratio similar to the prediction ratio in the global ranking. We can see the result of the tournament ranking in prediction in the figure 2.12.

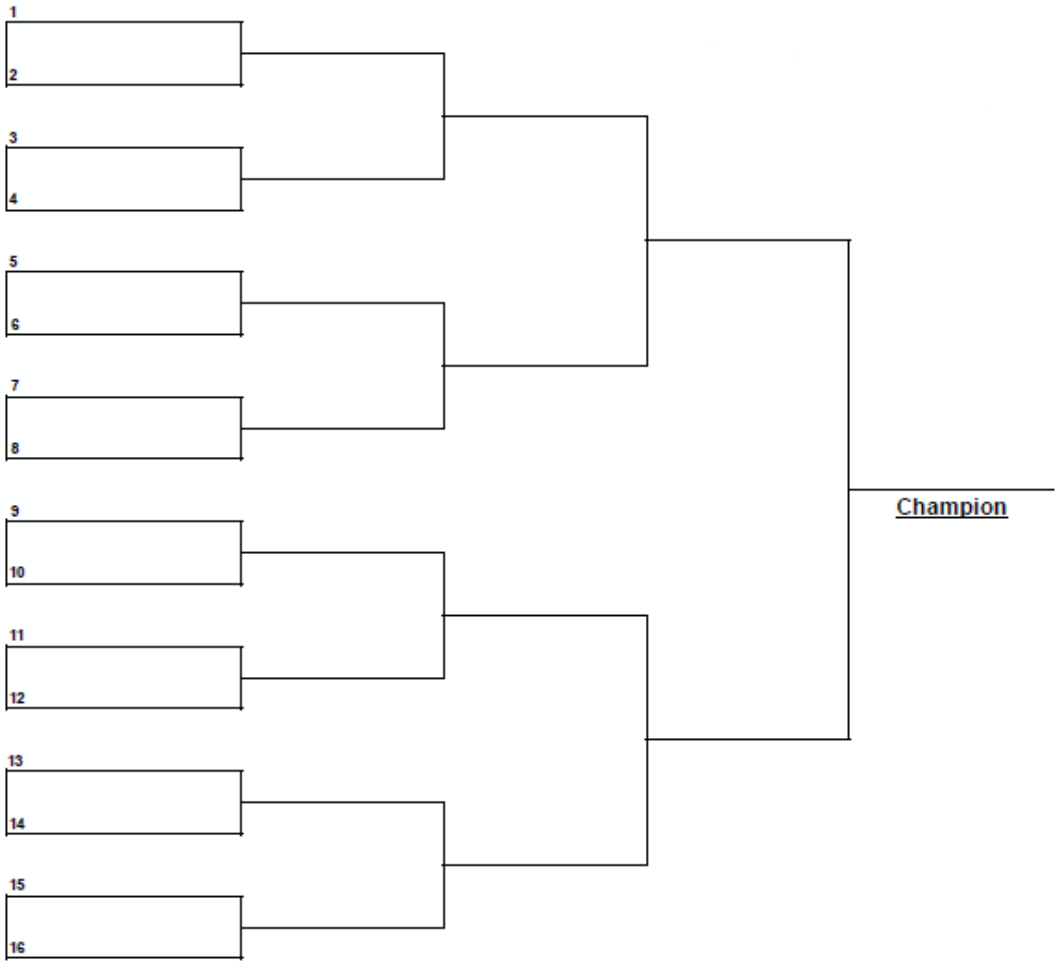


Figure 2.11: The general representation of tournament competition to rank competitors when there are 16 competitors.

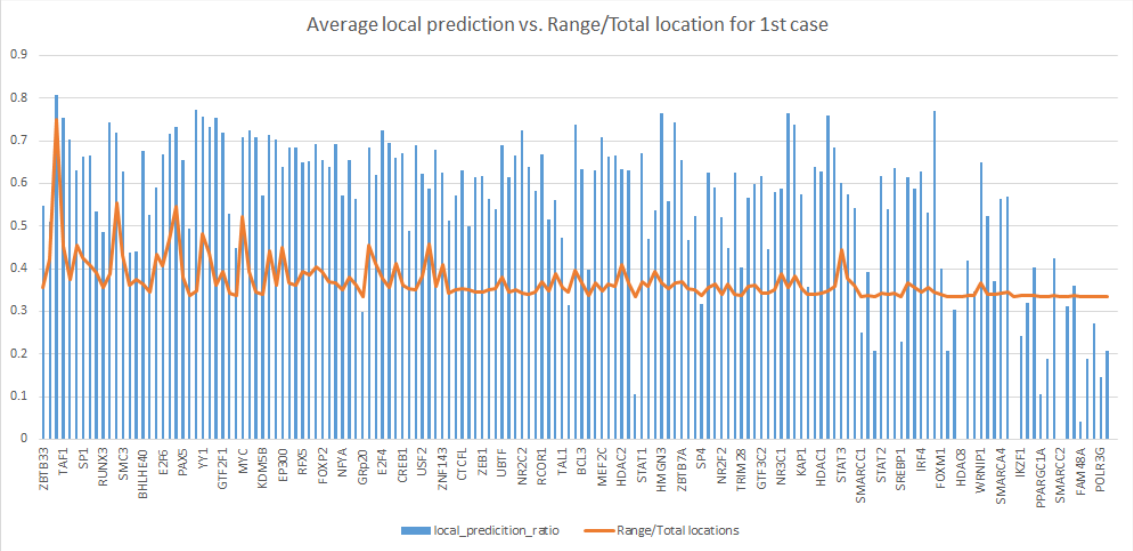


Figure 2.12: The scatter plot of true prediction ratio in terms of number of TFs in the first and second chromosome shows increasing TFs results in a better prediction.

We can change our algorithm by modifying the way we compare the score of bins. For instance, the number of bins in each local competition can be changed from two to a number greater than two. We did this modification to see the impact of that in results. The results got worse a little bit. We can see them in the following figures. Another modification in this algorithm can be done by group ranking instead of Tournament. In group ranking, the fixed number of neighbor bins are located in one group. The score of each group is equal to the maximum of the score of each group member. Then groups can be ranked in a local ranking procedure like a tournament or a global ranking. In the end, We can assign a rank to group members, first by comparing their group rank and if it is equal (i.e. both are in the same group), the score of each group member. This procedure originated from this assumption that TFs are more attracted not only to those bins with high scores but also the close neighbors of high score bins have a very high chance to attract them and it can happen when there is not enough space around the high score bin.

2.2.2 2nd case

In the second case, we only consider the interactions between TFs located at the two different ends of a TAD. All the procedures and algorithms that have been done in the first case are repeated in the second case as well. As we explained before, these interactions are weaker than the first case interactions. Therefore, the prediction results based on that are worse than the first case ones. In the second case, it seems the interaction tendency values between TFs in the training and test set are not very close like the first cases. This fact originates from long distances between TFs in this case that makes their interactions weaker. It is shown in the figure 2.13. Like the first case, we started our simulation to cal-

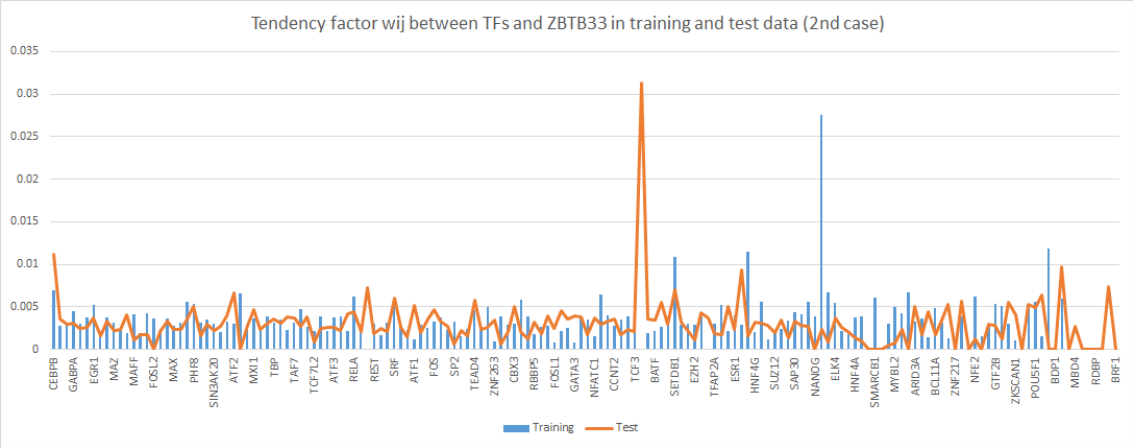


Figure 2.13: Tendency factor $w_{i,1}$ between TF_1 which is ZBTB33 and other TFs in the training and test data set for the second case.

culate $w_{i,j}$s as the first factor, and it can handle the first issue. Ranking bins for each TF_i is based on the sum of $w_{j,i}$ of any TF_j that potentially can have an interaction with TF_i, $\sum_{j=1,j\neq i}^{k} w_{i_j,i}$ (i.e. based on dataset they locate in other side of TAD that have interaction with that bin). The rest part will exactly the same as the first case.

For the second case, the average of true prediction ratio is about 50 percent, while if TF_i is distributed independently from the tendency weight, the average true prediction ratio should be 34 percent. We can see the comparison between the true prediction ratio and the random prediction in the figure 2.14 . The factors that heavily impacts our predic-

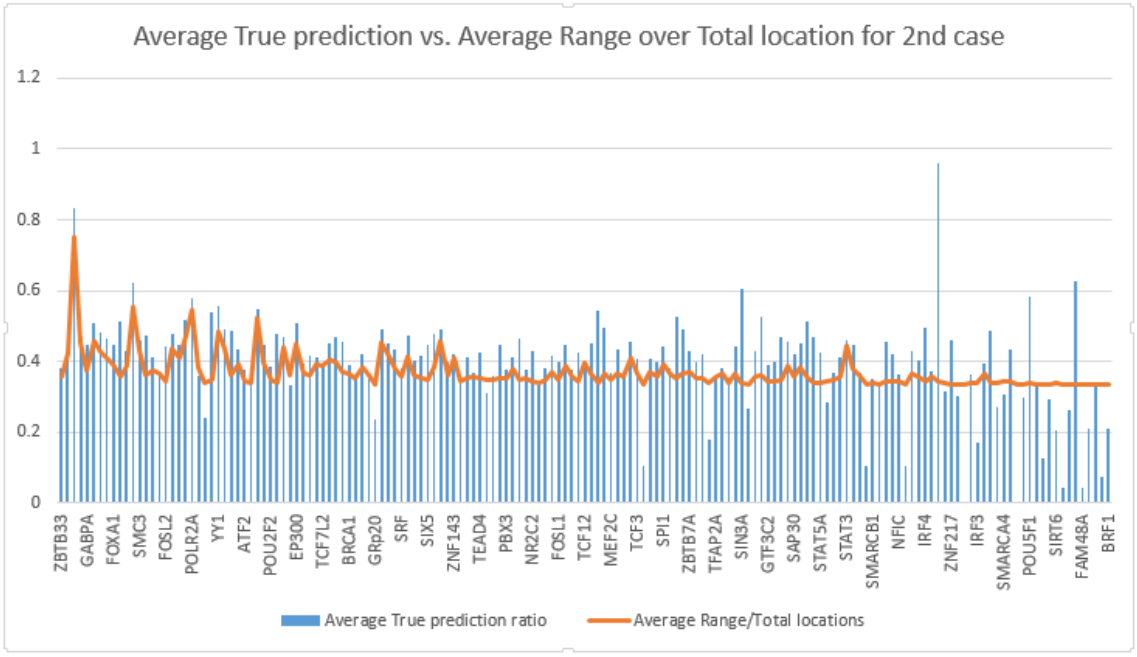


Figure 2.14: The average true prediction ratio vs. range over the total number of bins shows the prediction of random ranks for bins in the 2nd case.

tion is the number of TF_i in each chromosome. As we showed in the theory section, the chance of existing TF_i in each bin is proportional to each TF_i in the chromosome. We can show this fact in the figure 2.15. The true prediction ratio average is more in the first chromosomes, where more TFs on average. Besides, in each chromosome, the true prediction ratio is much better for those TFs with higher frequency in that chromosome. We can see this fact for the first and the second chromosome in the figure 2.16. We can compare the interaction tendency between the training and test set. For CTCF, we can see this matter in the figure 2.17. Based on the figure 2.17, there is a TF with have the most interaction tendency in both training and test set. This TF is KDM5A . We can conclude KDM5A have the most tendency to have the 2nd-case interactions with CTCF. This conclusion be more investigated through HiC data and intra-TAD interactions. Through same approach, we can find the TF pairs with the most 2nd case interaction tendencies.

For local ranking, we can have same approach as in the 1st case. We can see the result of the tournament ranking prediction in the figure 2.18. We can see the local ranking

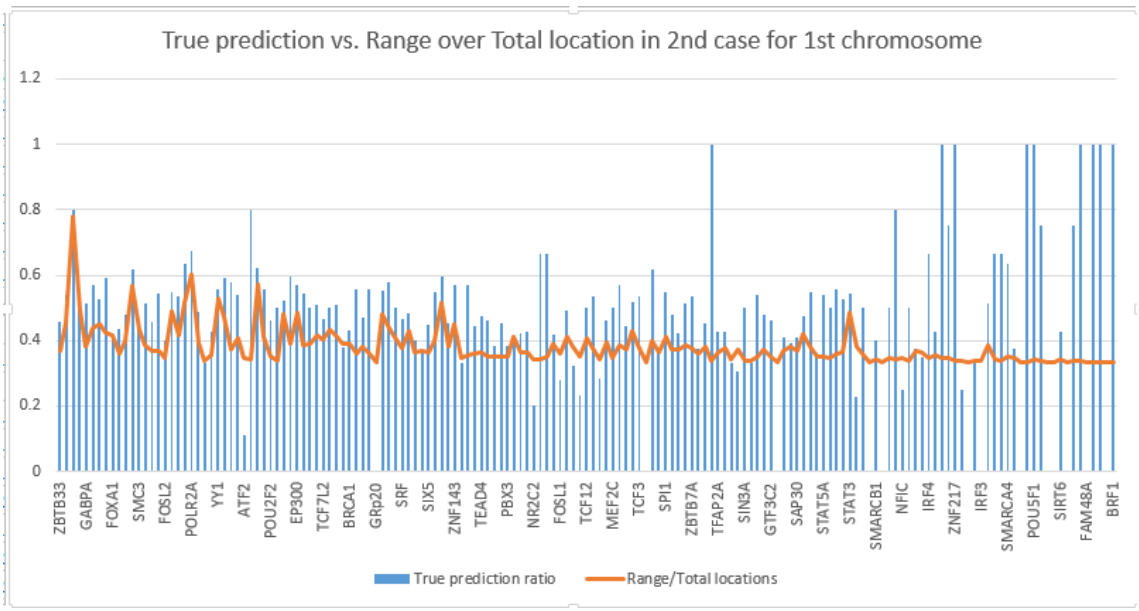


Figure 2.15: The average true prediction ratio vs. range over the total bin shows the prediction of random ranks for bins.

prediction is a little bit worse than global ranking like the first case.

Figure 2.16: The scatter plot of true prediction ratio in terms of number of TFs in the first and second chromosome shows increasing TFs results in a better prediction.

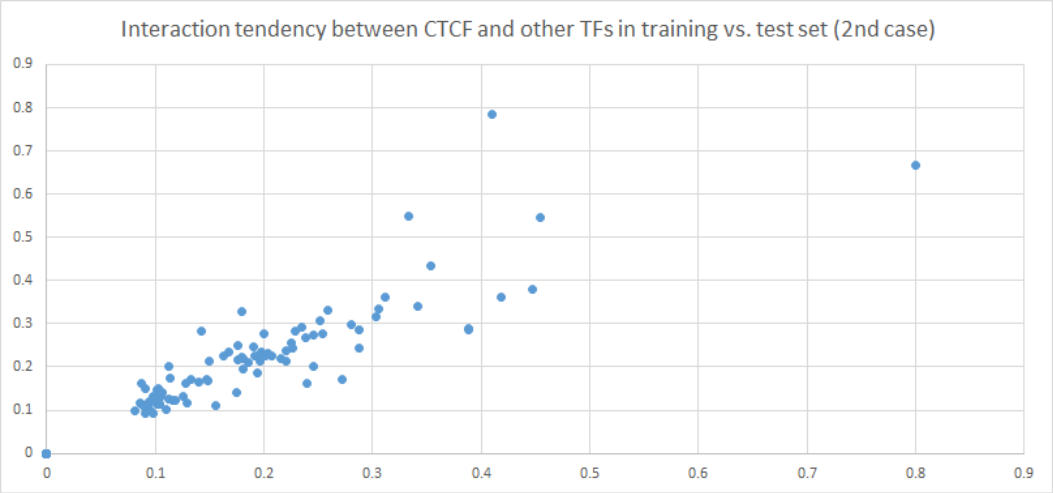


Figure 2.17: Interaction tendency between CTCF and other TFs in training set vs. test set (2nd case).

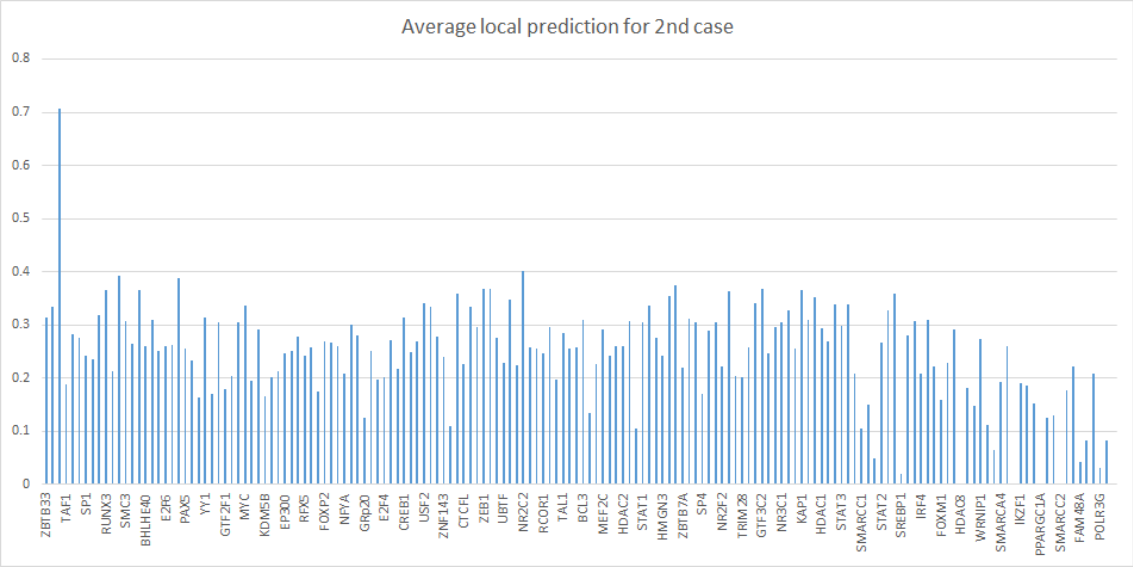


Figure 2.18: The average local ranking prediction ratio for different TFs in 2nd case.

Chapter 3

Conclusion

In the last decade, it gets more obvious that studying 3D genome structure plays an essential role in analyzing regulatory functions in the genome after advancements in methods for studying chromatin contacts at the genome level. As a part of this improvement, the discovery of different domains such as TADs and accessing to Hi-C data increase our knowledge about the impact of those genome domains and interactions inside them between regulatory factors such as TFs. The interactions between TFs are very dependant on their co-localization (i.e., their situations relative to each other). TFs choose their target sites by several different mechanisms. However, a model that can integrate all of these mechanisms is not available because these mechanisms are complicated and depend on many unknown factors. Many data integration strategies have been suggested to tackle this problem. Different intelligent methods such as supervised and unsupervised machine learning, deep learning, etc., leverage next generation sequencing data to extract novel patterns for predicting TF binding sites.

Recently genome sequences of an increasing number of organisms have been extracted, but because our knowledge about transcription factor binding motif is limited, and the ability of computational tools to provide high-resolution data is restricted, obtaining practical information from these data sequences still is a severe challenge.

This thesis presents a mechanism to predict the transcriptional regulatory element loca-

tions independent of motif sequence features. Here we take advantage of probability models for predicting TF binding sites based on the TADs and TFs locations along different chromosomes. In our model, the tendency of the TFs to be close to each other in each TADs domain is the only feature used for their binding site prediction. In this analysis, two different cases for the interaction tendency are studied and compared with each other. As shown in the results, the first case has the best prediction accuracy between all of these cases. Because of the shortest distance between TFs in the first case compare with the other case, the impacts of TFs on each other's location increase.

Although there is no single agreed feature for predicting TFBS, we found in this framework, we can test more features than the relative distance between TFs. Although our model is straightforward and considers only one feature for TFBS prediction, it still sheds light on the TFs co-localization fact and can be used as a guideline for interpreting other vital features to have a more accurate prediction. As a future direction for this research, other potential features can be tested through a similar strategy. To do so, we need to consider two main steps. The first step is designing a proper parameter for each potential feature. The second step is to modify that parameter so that their difference in the test and training set is negligible. There are three main points that we can get from this strategy.

The first point is that our co-localization-based model provides a means to measure the interaction tendency of TFs in respect of each other with a high degree of sensitivity. In other words, although the prediction model was trained only from the human genome data, the sensitivity of the model is strong enough to handle other independent data sets as well. Even though generalizing our model to add cell types and other components data is potentially possible, the results of two cases confirm the flexibility of our model to predict the location of TFs in different chromosomes and cell types.

The second point is that important collaboration between pairs of TFs to initialize regulatory functions can be revealed through our predictive model. As we explained in the section 1.6, we can find the important pair of TFs that collaborate with each other by weight tendencies scatter plot. This finding is an extra advantage of this model which

primarily is designed for TFBS prediction. We figured out the results of these plots is very accurate and is not dependent of the true prediction rate for TFBS. The accuracy of some of these results have been validated through ChIA-PET experiments. Other part of results can be investigated by more experiments.

The third point is that our approach will also be valuable to explain the co-localization process's dynamics. Despite of considerable effort to date, this dynamic process remains a beautiful and complicated challenge for researchers. In this study, we present a tendency interaction parameter between TFs as a piece of primary evidence that leads us to design a long term model for dynamic TFs co-localization. To our knowledge, our model was the first one to borrow Hi-C interactions data between TFs in TADs to predict TFBS. In the second case we consider HiC interactions between TFs located at two different sides of TADs. We observed these interaction are weaker than the the interactions between TFs inside a same bin. Therefore the interactions between TFs inside a same bin are more crucial to take in to account for TFBS prediction. Besides, we consider the strong interactions between TFs located in close sites at TAD boundaries. Given these two types of interactions, it may be useful in explaining the dynamic 3D structure of the TADs, consequently, chromatin reorganization since the dynamic co-localization process may be associated with the dynamic structure of the 3D genome structure. Revealing this fact would advance our understanding of the mechanism of TADs formation.

Bibliography

[1] Ryan Admiraal. *Dynamic network models based on revealed preference for observed relations and egocentric data*. PhD thesis, Ph. D. Thesis, University of Washington, Seattle, WA, 2009.

[2] Charu C Aggarwal, Haixun Wang, et al. *Managing and mining graph data*, volume 40. Springer, 2010.

[3] Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, 33(8):831–838, 2015.

[4] Phipps Arabie and Yoram Wind. Marketing and social networks. *SAGE FOCUS EDITIONS*, 171:254–254, 1994.

[5] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[6] Zygmunt William Birnbaum, Monroe G Sirken, et al. Design of sample surveys to estimate the prevalence of rare diseases: Three unbiased estimates. 1965.

[7] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D-U Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4):175–308, 2006.

[8] Boyan Bonev and Giacomo Cavalli. Organization and function of the 3d genome. *Nature Reviews Genetics*, 17(11):661, 2016.

[9] Peter J Carrington, John Scott, and Stanley Wasserman. *Models and methods in social network analysis*, volume 28. Cambridge university press, 2005.

[10] Davide Chicco, Haixin Sarah Bi, Jüri Reimand, and Michael M Hoffman. Behst: genomic set enrichment analysis enhanced through integration of chromatin long-range interactions. *bioRxiv*, page 168427, 2019.

[11] Mosuk Chow and Steven K Thompson. Estimation with link-tracing sampling designs–a bayesian approach. *Quality control and applied statistics*, 49(6):613–616, 2004.

[12] Gabriel Cuellar-Partida, Fabian A Buske, Robert C McLeay, Tom Whitington, William Stafford Noble, and Timothy L Bailey. Epigenetic priors for identifying active transcription factor binding sites. *Bioinformatics*, 28(1):56–62, 2012.

[13] Jesse R Dixon, Inkyung Jung, Siddarth Selvaraj, Yin Shen, Jessica E Antosiewicz-Bourget, Ah Young Lee, Zhen Ye, Audrey Kim, Nisha Rajagopal, Wei Xie, et al. Chromatin architecture reorganization during stem cell differentiation. *Nature*, 518(7539):331–336, 2015.

[14] JR Dixon, I Jung, S Selvaraj, and B Ren. Global reorganization of chromatin architecture during embronic stem cell differentiation. *Gene Expression Omnibus*, 2013.

[15] Nick Duffield, Carsten Lund, and Mikkel Thorup. Estimating flow distributions from sampled flow statistics. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 325–336. ACM, 2003.

[16] Martín H Félix-Medina and Steven K Thompson. Combining link-tracing sampling and cluster sampling to estimate the size of hidden populations. *Journal of Official Statistics*, 20(1):19, 2004.

[17] Ove Frank. *Statistical inference in graphs*. Foa Repro, 1971.

[18] Ove Frank. Survey sampling in graphs. *Journal of Statistical Planning and Inference*, 1(3):235–264, 1977.

[19] Ove Frank. Sampling and estimation in large social networks. *Social networks*, 1(1):91–101, 1979.

[20] Ove Frank. A survey of statistical methods for graph analysis. *Sociological methodology*, 12:110–155, 1981.

[21] Ove Frank. Social network analysis, estimation and sampling in. In *Computational Complexity*, pages 2845–2863. Springer, 2012.

[22] Ove Frank and Tom Snijders. Estimating the size of hidden populations using snowball sampling. *JOURNAL OF OFFICIAL STATISTICS-STOCKHOLM-*, 10:53–53, 1994.

[23] Krista J Gile. Improved inference for respondent-driven sampling data with application to hiv prevalence estimation. *Journal of the American Statistical Association*, 106(493), 2011.

[24] Ofir Hakim and Tom Misteli. Snapshot: chromosome conformation capture. *Cell*, 148(5):1068–e1, 2012.

[25] Mark S Handcock and Krista J Gile. Modeling social networks from sampled data. *The Annals of Applied Statistics*, pages 5–25, 2010.

[26] William L Harkness. Properties of the extended hypergeometric distribution. *The Annals of Mathematical Statistics*, 36(3):938–945, 1965.

[27] Nathaniel D Heintzman, Rhona K Stuart, Gary Hon, Yutao Fu, Christina W Ching, R David Hawkins, Leah O Barrera, Sara Van Calcar, Chunxu Qu, Keith A Ching, et al. Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome. *Nature genetics*, 39(3):311–318, 2007.

[28] Timothy Classen Hesterberg. *Advances in importance sampling*. PhD thesis, Stanford University, 2003.

[29] Jialiang Huang, Eugenio Marco, Luca Pinello, and Guo-Cheng Yuan. Predicting chromatin organization using histone marks. *Genome biology*, 16(1):1–11, 2015.

[30] Christian Hubler, Hans-Peter Kriegel, Karsten Borgwardt, and Zoubin Ghahramani. Metropolis algorithms for representative subgraph sampling. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 283–292. IEEE, 2008.

[31] Michael R Hübner, Mélanie A Eckersley-Maslin, and David L Spector. Chromatin organization and transcriptional regulation. *Current opinion in genetics & development*, 23(2):89–95, 2013.

[32] Sang Hoon Lee, Pan-Jun Kim, and Hawoong Jeong. Statistical properties of sampled networks. *Physical Review E*, 73(1):016102, 2006.

[33] Jure Leskovec and Christos Faloutsos. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 631–636. ACM, 2006.

[34] Yifeng Li, Chih-yu Chen, Alice M Kaye, and Wyeth W Wasserman. The identification of cis-regulatory elements: A review from a machine learning perspective. *Biosystems*, 138:6–17, 2015.

[35] Maxwell W Libbrecht, Oscar L Rodriguez, Zhiping Weng, Jeffrey A Bilmes, Michael M Hoffman, and William Stafford Noble. A unified encyclopedia of human functional dna elements through fully automated annotation of 164 human cell types. *Genome biology*, 20(1):1–14, 2019.

[36] Scott M Lundberg, William B Tu, Brian Raught, Linda Z Penn, Michael M Hoffman, and Su-In Lee. Chromnet: Learning the human chromatin network from all encode chip-seq data. *Genome biology*, 17(1):1–19, 2016.

[37] Alan Mislove, Massimiliano Marcon, Krishna P Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 29–42. ACM, 2007.

[38] Martina Morris et al. *Local Acts, Global Consequences: Networks and the Spread of HIV*. National Institutes of Health, 2007.

[39] Mark Newman, Albert-Laszlo Barabasi, and Duncan J Watts. *The structure and dynamics of networks*. Princeton University Press, 2006.

[40] Mark EJ Newman. Scientific collaboration networks. i. network construction and fundamental results. *Physical review E*, 64(1):016131, 2001.

[41] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[42] Roger Pique-Regi, Jacob F Degner, Athma A Pai, Daniel J Gaffney, Yoav Gilad, and Jonathan K Pritchard. Accurate inference of transcription factor binding from dna sequence and chromatin accessibility data. *Genome research*, 21(3):447–455, 2011.

[43] Davood Rafiei. Effectively visualizing large networks through sampling. In *Visualization, 2005. VIS 05. IEEE*, pages 375–382. IEEE, 2005.

[44] Fidel Ramírez, Vivek Bhardwaj, Laura Arrigoni, Kin Chung Lam, Björn A Grüning, José Villaveces, Bianca Habermann, Asifa Akhtar, and Thomas Manke. High-resolution tads reveal dna sequences underlying genome organization in flies. *Nature communications*, 9(1):1–15, 2018.

[45] Bruno Ribeiro, Pinghui Wang, and Don Towsley. On estimating degree distributions of directed graphs through sampling. *University of Massachusetts CMPSCI Technical Report UM-CS-2010-046*, 2010.

[46] Sushmita Roy, Alireza Fotuhi Siahpirani, Deborah Chasman, Sara Knaack, Ferhat Ay, Ron Stewart, Michael Wilson, and Rupa Sridharan. A predictive modeling approach for cell line-specific long-range regulatory interactions. *Nucleic acids research*, 43(18):8694–8712, 2015.

[47] Carl-Erik Särndal, Bengt Swensson, and Jan Wretman. *Model assisted survey sampling*. Springer Science & Business Media, 2003.

[48] Zhen Shen, Wenzheng Bao, and De-Shuang Huang. Recurrent neural network for predicting transcription factor binding sites. *Scientific reports*, 8(1):1–10, 2018.

[49] Alireza Fotuhi Siahpirani, Ferhat Ay, and Sushmita Roy. A multi-task graph-clustering approach for chromosome conformation capture data sets identifies conserved modules of chromosomal interactions. *Genome biology*, 17(1):1–18, 2016.

[50] S Singh, Y Yang, B Poczos, and J Ma. Predicting enhancer-promoter interaction from genomic sequence with deep neural networks. biorxiv, 085241. 2016.

[51] Matthew Slattery, Tianyin Zhou, Lin Yang, Ana Carolina Dantas Machado, Raluca Gordân, and Remo Rohs. Absence of a simple code: how transcription factors read the genome. *Trends in biochemical sciences*, 39(9):381–399, 2014.

[52] Steven H Strogatz. Exploring complex networks. *Nature*, 410(6825):268–276, 2001.

[53] Zhonghui Tang, Oscar Junhong Luo, Xingwang Li, Meizhen Zheng, Jacqueline Jufen Zhu, Przemyslaw Szalaj, Pawel Trzaskoma, Adriana Magalska, Jakub Wlodarczyk, Blazej Ruszczycki, et al. Ctcf-mediated human 3d genome architecture reveals chromatin topology for transcription. *Cell*, 163(7):1611–1627, 2015.

[54] SK Thompson and GA Seber. F (1996). adaptive sampling.

[55] Steven K Thompson and Linda M Collins. Adaptive sampling in research on risk-related behaviors. *Drug and Alcohol Dependence*, 68:57–67, 2002.

[56] Steven K Thompson and Ove Frank. Model-based estimation with link-tracing sampling designs. *Survey Methodology*, 26(1):87–98, 2000.

[57] 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, et al. Assessing computational tools for the discovery of transcription factor binding sites. *Nature biotechnology*, 23(1):137–144, 2005.

[58] Nelle Varoquaux, Ferhat Ay, William Stafford Noble, and Jean-Philippe Vert. A statistical approach for inferring the 3d structure of the genome. *Bioinformatics*, 30(12):i26–i33, 2014.

[59] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[60] Chia-Lin Wei, Qiang Wu, Vinsensius B Vega, Kuo Ping Chiu, Patrick Ng, Tao Zhang, Atif Shahab, How Choong Yong, YuTao Fu, Zhiping Weng, et al. A global map of p53 transcription-factor binding sites in the human genome. *Cell*, 124(1):207–219, 2006.

[61] Leland Wilkinson and Michael Friendly. The history of the cluster heat map. *The American Statistician*, 63(2):179–184, 2009.