Unsupervised annotation of regulatory domains by integrating functional genomic assays and Hi-C data

by

Neda Shokraneh

B.Sc., Sharif University of Technology, 2018

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
School of Computing Science
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY
Fall 2020

Declaration of Committee

Name:	**Neda Shokraneh**
Degree:	**Master of Science**
Thesis title:	**Unsupervised annotation of regulatory domains by integrating functional genomic assays and Hi-C data**
Committee:	**Chair:** Ke Li Assistant Professor, Computing Science
	Maxwell Libbrecht Supervisor Assistant Professor, Computing Science
	Manolis Savva Committee Member Assistant Professor, Computing Science
	Leonid Chindelevitch Examiner Lecturer Faculty of Medicine, School of Public Health Imperial College London

Abstract

In each cell type, chromosomes are organized into a specific 3D structure that controls the function of a cell through different mechanisms including domain-scale regulation. Because of the correlation between genome structure and its function, different methods have been proposed to integrate 1D functional genomic and 2D Hi-C data to identify domain types. Existing methods rely on an assumption that directly connected genomic regions are more probable to have the same domain type, however, spatial clustering of genomic regions is based on both their first-order and second-order proximities.

Here, we present an integrative approach that uses 1D functional genomic features and 3D interactions from Hi-C data to assign labels to genomic regions that can discriminate both spatial and functional genomic patterns. We use graph embedding to learn latent variables for nodes (genomic regions) that preserve the Hi-C graph second-order proximity. Such latent variables summarize spatial information in Hi-C data, and we feed them in addition to existing 1D functional features to the Segway, a genome annotation method, to infer domain states. We show that our labels distinguish a combination of the spatial and functional states of the genomic regions, for example, loci locating in the nucleus interior can be furthermore clustered into significantly and moderately expressed domains. We also found the importance of each of the spatial and functional features to explain different cell activities including replication timing and gene expression profile, and how coupling two feature types improve the prediction of such activities. Finally, we showed that incorporating spatial features allow finding domain types, which are co-regulated even in large genomic distance from each other. Our framework can be generalized to aggregate different 1D genomic assays and 3D interactions from Hi-C to find the mechanisms behind the association of genome 3D structure and epigenetic profile.

Keywords: Epigenomics, Genome 3D structure, Graph embedding, Genome domain annotation

Dedication

To my great family and friends.

Acknowledgements

I would like to thank my wonderful supervisor, Prof. Maxwell Libbrecht for constantly guiding me through this project patiently, and providing financial support for my study. I also would like to thank Prof. Manolis Savva for his insightful comments on my project and for serving as my committee member. Further, I would like to acknowledge my other committee members, Prof. Leonid Chindelevitch and Prof. Ke Li, for their time and serving as my committee members. Lastly, I want to express my gratitude to my labmates in Comp Bio lab for their friendly supports and helpful feedback on my project.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Chromatin in the nucleus is segregated into nuclear compartments [44], each of these compartments having their specific spatial properties such as radial position and relative distance to other nuclear compartments that are correlated with 1D functional genomic properties like epigenetic marks. These spatial and functional properties drive many cellular processes, including gene regulation and DNA replication. A popular way to understand genome regulation and compartmentalization is to annotate the genome into several categories of *domains*, such that each domain type has particular properties and activity. For example, the chromatin structure has been probed using Hi-C assay that outputs interaction frequencies between every pair of genomic regions, and Rao et al [44] categorize the genome into subcompartments based on Hi-C data, such that domains having the same pattern of interaction with the rest of the genome are annotated with the same subcompartment.

The structure of a genome is correlated with epigenetic properties of it, for example, genomic regions located in the nuclear interior are more transcriptionally active and enriched for epigenetic marks associated with active transcription, while the genomic regions near the nuclear periphery are transcriptionally repressed and depleted of active epigenetic marks. Therefore, both structural assays such as Chromatin Conformation Capture (3C) or Hi-C [35, 44, 56] and functional genomic assays [27] such as ChIP-seq and ATAC-seq provide complementary information to identify such domains in a genome. Since Hi-C data is represented as a 2D matrix and functional genomic data are in a 1D signal format, integrative analysis methods are needed that integrate both types of information to produce a comprehensive understanding of chromatin regulation.

Segmentation and genome annotation (SAGA) methods are widely-used for integrative analysis of multiple data types in 1D signal format. SAGA methods take multiple signal tracks as an input. They segment the genome and output a label for each segment of a genome such that segments with the same label have a similar pattern of genomics signal tracks [33, 25]. While most SAGA methods can only handle 1D data sets such as ChIP-seq, two SAGA methods, Segway-GBR and SPIN [34, 54] can incorporate both genomics assays and Hi-C data to infer more accurate domain annotations. Segway-GBR does that

by encouraging bins having high interaction in Hi-C data, to get the same label using graph-based regularization. SPIN [54] defines a Markov random field that includes edges defined on Hi-C contacts. Both of these methods have an assumption that bins that have more contact with each other, should get the same functional label. However, analysis of Hi-C interaction data has shown that genomic regions that are not directly connected but having the same neighborhood vector (interaction pattern with rest of the regions in the genome) share the same spatial features such as radial positioning.

In this paper, we aim to use recent advances in Graph Neural Networks to incorporate Hi-C data without the specific assumptions made by previous methods. Graph Neural Networks (GNNs) can embed graph structures into a low-dimensional feature space. We propose an integrative approach for identifying chromatin domains that leverages GNN methods. We learn latent features for each genome locus through their interaction graph. We concatenate these structural features with data from 1D ChIP-seq and DNase-seq data sets as input to a SAGA algorithm. The resulting domain annotations combine both biochemical activities driven from functional genomics assays and the structure of the genome defined by Hi-C data. These domains form a comprehensive picture of domains in the genome that can be used to discover new categories of domain activity and elucidate their influence on cellular activity such as gene regulation. In particular, we show that our inferred domain states show a more accurate explanation for different functional and structural properties of a genome, including replication timing and distance to nuclear compartments.

Chapter 2

Literature Overview

2.1 Overview

Proteins are complex molecules in cells, and each of them has a specific function in a cell. For example, they can act as an enzyme to catalyze metabolic reactions in a cell, or provide structural support for a cell. Each cell controls its functions by regulating the amount and type of proteins it manufactures. DNA and the proteins bound to DNA carry the information essential for this regulation.

DNA, or deoxyribonucleic acid, is the hereditary material in humans and almost all other organisms, containing the genetic information of life and acts as a set of instructions for how to make the proteins that living things need to grow. The information in DNA is stored as a code made up of four chemical bases: adenine (A), guanine (G), cytosine (C), and thymine (T). A DNA sequence can be represented as a string of the letters ACTG.

The DNA molecule is packaged into thread-like structures called **chromosomes**. The DNA material in chromosomes is composed of coding and non-coding regions. The coding regions are known as **genes** and contain the information necessary for a cell to make proteins. **Gene expression** is the process by which the instruction in a gene is converted into a protein. It has two major steps: transcription and translation. In transcription, the sequence of nucleotides in a gene is copied as **RNA**. In eukaryotes, the RNA must go through additional processing steps to become a messenger RNA (mRNA). In translation, the sequence of nucleotides in the mRNA is translated into a protein chain.

Each human cell contains approximately 2 m of DNA in its nucleus, which is only about 6 µm in diameter. In fact, the **histone** proteins bind to the DNA and organize it into a compact structure called **chromatin** (histone proteins act as spools for the DNA to wrap around). The chromatin structure is not consistent across different cell types, and **histone modifications** play a key role in shaping this structure. These modifications can be seen as biochemical tags that mark the genome sequence, and **Epigenetics** is the study of such biochemical tags and their role in the cellular differentiation (prefix epi in epigenetics implies features that are "on top of" the traditional genetic information (DNA sequence)).

Transcription factors (TFs) are proteins involved in the transcription process that coverts the DNA into the RNA. TFs can initialize the transcription of a specific gene through binding to its **transcription start site** (TSS) or its corresponding **enhancer** (genomic region that can enhance the transcription of a gene over large genomic distance but small spatial distance). The TFs can bind to the accessible regions of DNA. Such DNA accessibility is determined through epigenetic modifications and consequently the way the DNA is packaged in the 3D space. Therefore, the epigenetic profile of a cell defines its purpose and gene expression profile (the measurement of the activity of all the genes in a cell as a global picture of cellular function).

Conclusively, a wide variety of behaviors and illnesses have been linked with epigenetic modifications. The epigenetic related studies and comparative analysis (for example when we want to infer epigenetic modifications corresponding to a particular behavior like cancer) rely on two things: the epigenetic processes to study as underlying factors and the mechanisms that link the epigenetic processes to that behavior. Many types of epigenetic processes have been identified and measured experimentally so far, including DNA methylation, DNA acetylation, histone modification, etc. Different computational and biophysical models have been proposed to model these processes, their relationships, and the way they affect cell behavior. For example, different chromatin domain types have been inferred based on different epigenetic processes, that can explain **domain-scale regulation** of the genes (explained in detail in the future). In this chapter, first, we overview some of the epigenetic processes measured by the genomic assays and then introduce some biological phenomena related to the genome structure and function, derived from modeling the epigenetic processes, and the computational methods used for such models.

2.2 Epigenomic datasets

2.2.1 Overview

A histone is a protein that binds to the long chromosomal DNA and compacts it into the microscopic space that fits in the cell nucleus. This complex of DNA and protein is called chromatin. A **nucleosome** is the basic unit of chromatin and is composed of eight histone proteins (histone octamer) and 147 base pairs (bp) of DNA around it. The histone octamer consists of two copies of each of the four core histone proteins including H2A, H2B, H3, and H4. The nucleosomal units are connected by sequences of **linker DNA**, which are between $20 - 90$ bp long and are associated with the linker histone H1.

The proteins are modified in different ways after the translation from mRNA through a process called Post-translational modification (PTM). PTMs diversify and extend protein function, resulting in the increased complexity of the proteome (entire set of proteins expressed by a genome) relative to the genome (1 million proteins compared to $20{,}000-25{,}000$ genes). The histone proteins also undergo the PTM, known as histone modification, that

affects their interaction with DNA. The histone modification is characterized by the name of the modified histone, type, and position of affected residue (amino acid residue in histone chain), the type of modification (at least nine different types of modifications have been discovered [3]) and the number of modifications. For example, H3K4me1 denotes the monomethylation (me1) of the 4th lysine residue (K4) from the start of the H3 protein. The combination of these parameters results in different types of histone modifications.

Histone modifications act in diverse biological processes in a cell such as a gene regulation and DNA replication through altering chromatin condensation and DNA accessibility. For example, H3K27me3 is a histone modification associated with the downregulation of nearby genes as a result of forming densely packed chromatin regions, while H3K27ac is associated with the higher genes transcription and open chromatin regions. Such epigenetic processes can be probed through ChIP-seq assay that can identify genome-wide binding sites for a particular protein such as a modified histone. There have been also techniques such as DNase-seq and ATAC-seq to assess genome-wide chromatin accessibility. We briefly explain these techniques and the information they provide in the first two subsections of this section.

Previous studies have shown the correlation between epigenetic properties of the genomic regions and their positioning relative to different nuclear components [21, 54]. The TSA-seq method estimates chromosomal distances from a particular nuclear component genome-wide. The 3D structure of a genome also has been probed through the Hi-C assay. We also explain these two techniques in this section.

Finally, the epigenetic state of a cell plays a key role in determining different behaviors of a cell such as a gene expression and replication timing profile. We explain RNA-seq, a sequencing-based method that quantify the gene expression, and Repli-seq, a protocol to analyze the replication timing profile of a cell, in more detail at the end of this section.

2.2.2 Histone modification ChIP-Seq assays

To identify the epigenetic state of a cell, we would like to identify the locations in a genome bound by different histone modifications. ChIP-seq is able to capture this information through the following steps:

- Use formaldehyde (or everything like glue) to glue all (even the ones we are not interested in) of the proteins bound to the DNA together with the DNA.
- Cut the DNA up into small (about 300 bp) fragments.
- An antibody is a large and Y-shaped protein component of the immune system that binds like a lock-and-key to its particular target like a virus (to fight infections). Their arms contain sites that can recognize and bind to specific proteins. ChIP-seq isolates the protein we are interested in using its corresponding antibody and washes everything else away.

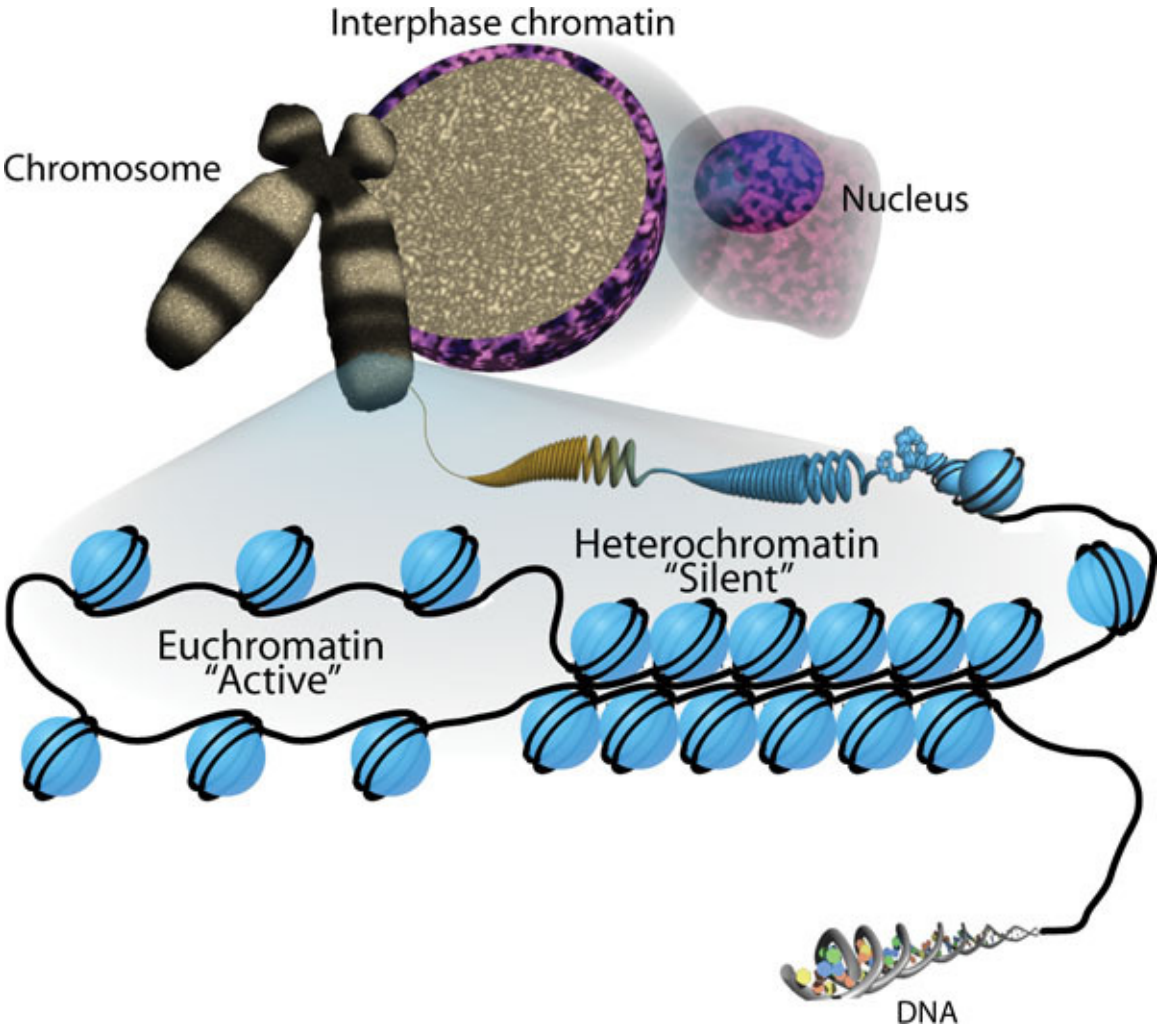


Figure 2.1: Reprinted from [47]. Every 147 bp of DNA is wrapped around a histone octamer (blue spheres), making a basic unit of the chromatin known as a nucleosome. Different epigenetic processes, including DNA and histone modifications, influence nucleosome positioning, and chromatin compaction. The level of the compaction consequently influences DNA-dependent processes like the transcription level. For example, the nucleosomes in an active region are lightly packed and more open to the transcription factors essential for the gene transcription, while the silent regions are densely packed and inaccessible to transcription factors and therefore transcriptionally silent.

- Reverse the formaldehyde glue by warming up everything, and isolate the DNA fragments by washing away the proteins.
- All the above processes are usually applied to a pool of 6 million cells, resulting in lots of DNA fragments from a lot of cells.
- A sequencing library is prepared from the DNA fragments, and after processing, high-quality reads are aligned to a genome.
- Now, we have a long list of genomic coordinates for all of the reads (usually between 50 and 100 million reads), and we can use these reads to create a genome-wide track (signal) showing the number of reads associated with each genomic region.

  Note: Repetitive regions of a genome (patterns of DNA that occur in multiple copies throughout the genome) are more probable to be assigned to the read, resulting in a high concentration of reads, although they are not significantly enriched with a target protein. Therefore, the control track is made by skipping the antibody step in the ChIP-seq experiment (keeping all DNA fragments to sequence) to verify that a high concentration of reads in a ChIP-seq track is due to a protein-binding there and not because of a lot of reads mapped to a repetitive region.

2.2.3 Chromatin accessibility assays

We introduce two biochemical assays that measure the chromatin accessibility, DNase-seq and ATAC-seq in this section:

- DNase-seq

  DNase I is an enzyme that is able to cleave the bonds between nucleotides of the DNA. DNase I hypersensitive (HS) sites are regions of chromatin that are sensitive to cleavage by the DNase I enzyme [6]. These regions include nucleosome-free regions in which chromatin has lost its condensed structure, and is open to regulatory elements like transcription factors. Therefore, identifying DNase I HS sites gives information about chromatin accessability and the location of regulatory elements in the genome. DNase-seq is a high-throughput method that identifies DNase I HS sites across the whole genome through the following steps:

  - Stabilize chromatin structure by formaldehyde
  - Digest chromatin with a small amount of DNase I that preferentially cuts at a DNase I HS sites so results in the DNA fragments corresponding to the open chromatin regions.
  - A sequencing library is prepared from the DNA fragments, and after processing, high-quality reads are aligned to a genome.

- Now, we have a long list of genomic coordinates for all of the reads, and we can use these reads to create a genome-wide track (signal) showing the number of reads associated with each genomic region that is informative of open chromatin and TF binding sites.

- ATAC-seq

  A transposable element (TE, transposon, or jumping gene) is a DNA sequence that can change its position within a genome [9]. Transposase is an enzyme that binds to the end of a transposon and catalyzes its movement to another part of the genome by a cut and paste mechanism [8]. In standard next-generation sequencing library preparation pipelines, first, genomic DNA from a sample is randomly fragmented, and sequencing adaptors are then attached to these fragments. Tagmentation is a recently developed transposition-based method of sequencing library preparation, in which transposase are loaded by adapter on their ends, and insert **randomly** into DNA in a cut and paste reaction. Because the DNA ends (attached adapters) are free, this effectively fragments the DNA while adding on the adapter sequences required for PCR amplification and sequencing [7]. So, two steps of fragmentation and adding an adapter to the fragments can be done simultaneously. While naturally occurring transposases have a low level of activity and insert an adapter sequence into all regions of the genome uniformly, ATAC-seq employs the mutated hyperactive transposase (Tn5) that inserts sequencing adapters into open regions of the genome. Therefore, sequencing the fragments can be used to infer regions of increased accessibility [4].

ATAC-Seq can be performed with significantly fewer cells (50,000 cells for ATAC-Seq compared to millions of cells for the other methods [5], and also is faster and more sensitive compared to DNase-seq. [4].

2.2.4 TSA-seq

The assays we introduced so far, including ChIP-seq, DNase-seq, and ATAC-seq, measure chromatin structural and functional properties. Previous studies have demonstrated the association between genome loci functional properties like transcriptional activity and their positioning relative to other nuclear components (see Table 2.1). For example, there is a functional link between the activity of a gene and its relative distance from nuclear lamina (nuclear periphery) and speckle (nuclear interior), as genes close to nuclear lamina are mostly silent, whereas active genes are localized toward the nuclear interior [49]. Therefore, the genome-wide mapping of distances of genome loci relative to a particular nuclear component gives a new insight into the genome 3D structure and function. TSA-seq is a genomic method capable of estimating such distances. We skip the explanation of a method since it requires lots of biological background, however, it outputs a genome-wide signal showing the estimated distances to a target nuclear component.

The nucleus contains more components than just chromosomes like a nuclear envelope, nuclear lamina, and nucleoplasm [11]. The nuclear envelope surrounds the nucleus to separates it from the cell cytoplasm. The **nuclear lamina** is a fibrillar network lying on the inner nuclear membrane that supports the nuclear envelope to maintain the overall shape and structure of the nucleus. The nuclear lamina is made of proteins including **A- and B-type lamins** and lamin-associated proteins, which can participate in compaction of peripheral chromatin (chromatin near the membrane) [10]. Nucleoplasm fills the empty space between chromosomes and other compartments in the nucleus and contains various proteins and enzymes dissolved within it. Nuclear bodies are membrane-less structures found in the nucleoplasm like **nuclear speckle**. Nuclear speckles are sites for splicing factor (a protein involved in the mRNA processing, so essential for transcription) storage and modification. All of these components work together to conduct cell-type-specific functions like the gene expression profile.

Table 2.1: Components in the nucleus

2.2.5 Chromosome conformation capture techniques

The 3D structure of a genome can be estimated indirectly through identifying pairwise interactions between genomic loci, as close loci are expected to have more interactions with each other and vice versa. Chromosome Conformation Capture (3C) assays are techniques used to study chromatin structure in this way. Hi-C is a 3C derivate technology that is genome-wide and outputs the frequency of the interactions between all pairs of genomic loci through the following steps:

- Fixing chromatin using formaldehyde to preserve its organization (interacting loci to be bound to one another). This process is called crosslinking.
- DNA fragmentation with the restriction enzyme (a protein that recognizes a specific, short nucleotide sequence and cuts the DNA only at that specific site). Note that interacting loci remain linked after fragmentation.
- Now, there are X-shaped linked fragments. Their two ends are ligated to each other to concatenate two DNA fragments, and then their link is canceled through reverse formaldehyde crosslinking. This results in a genome-wide library of ligation products, corresponding to pairs of fragments that were originally in close proximity to each other in the nucleus [51].
- The read pairs are aligned to the reference genome resulting in long list of pairs of genomic coordinates that were interacted to each other. We can construct a contact ma-

trix with specific resolution from this list. For example, contact matrix for chromosome 1 (length = 249,250,621) with 100 kb resolution is a $2493 \times 2493(\frac{249,250,621}{100,000} = 2493)$ matrix M such that M_{ij} is the number of pairs of genomic coordinates in a list with one coordinate in ith bin and the other one in jth bin (ith bin means region $[(i-1) * 100000, i * 100000]$ of a chromosome).

2.2.6 Repli-Seq

We introduced genes transcription as a DNA-dependent process so far, however, there are also other DNA-dependent processes that are affected by chromatin structure, including DNA repair, replication, and recombination. We used replication timing profile data to evaluate our experiments, and we explain it here.

The cells go through a process called **cell cycle** to replicate and make two daughter cells. The cell cycle consists of four discrete phases in which the cell increases in size (G_1), copies its DNA (S), prepares to divide (G_2), and divides (M). The process of a copy of a DNA during the S phase is called **DNA replication**. DNA replication is initiated at discrete sites, termed replication origins. There are about 50,000 DNA replication origins along the chromosomes [37] and replication does not start at all of these origins at once. Rather, there is a defined temporal order in which these origins fire [12]. The Repli-seq technology aims to quantify the DNA replication time as a function of a genome position [31]. They sort the cells into six fractions according to the DNA synthesis phase of a cell division (G_1, S_1, S_2, S_3, S_4, G_2), and then the cells from different stages are sequenced independently. After an alignment of the reads, the number of the sequences for each cell-cycle fraction is calculated over non-overlapping windows with a size of interest (for example 5 kb), and then normalized with genome-wide number of sequences, and finally normalized at each genomic position to calculate the percentage of the replication occurred in each stage for all of the positions. For example, "*chr1* 10000 15000 36 26 14 14 10 0" is a row in an output file showing that region $[10000, 15000]$ on chromosome 1 is replicated with ratio $36, 26, 14, 14, 10, 0$ during stages G_1, S_1, S_2, S_3, S_4, G_2 respectively.

2.3 Genome higher order organization

In eukaryotes, the genome does not exist as a linear molecule but is folded in an organized way inside the nucleus [57]. Analysis of Hi-C data has revealed three main principles governing genome 3D structure: 1) Chromosome territories, 2) Chromosome compartments, and 3) Topologically associating domains (TADs). We briefly describe each of these concepts below.

2.3.1 Chromosome territories

The well-known X-shaped structure of chromosomes, which is very condensed and distinguishable from other components in a cell nucleus, just exists in a metaphase stage of a cell, in which each pair of chromosomes is ready to be separated into two identical chromosomes. Contrastingly, during interphase, a portion of the cell cycle during which cells conduct their normal cellular functions, i.e. grow, read DNA and produce proteins, replicating their DNA, chromosomes are less uniform and more difficult to distinguish [1] (figure 2.2a). Chromosomes organization in interphase control the cell functions, such as the amount of each protein to be produced, in each cell type.

During interphase, each chromosome occupies a limited space in the nucleus which is known as a chromosome territory (CT). They are organized radially around the nucleus, some chromosomes closer to the nucleus periphery, while others occupy interior positions. CTs are associated with chromosomes activities, for example, gene-rich chromosomes are mostly around interior space, while gene-poor chromosomes locate near the membrane. Although chromosomes have their own territories, they still share some spaces, and build up and interconnected chromosomes network. Different models have been proposed to explain chromosome territories and their connections. We describe two of these models briefly here:

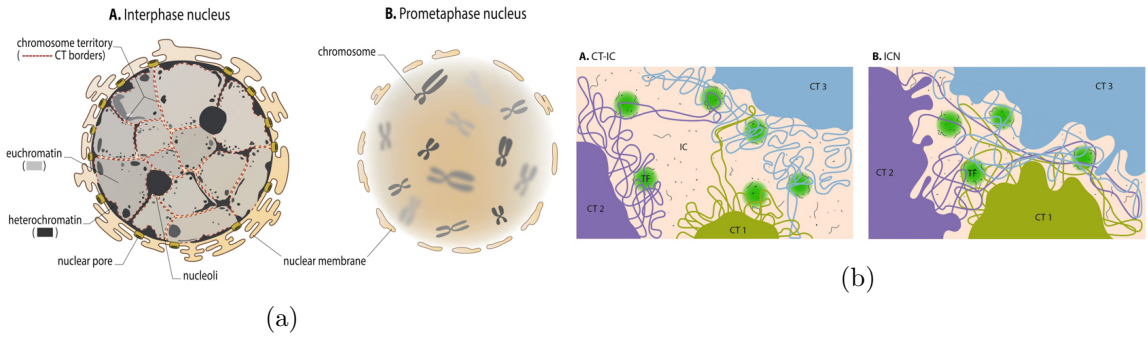
- Chromosome territory-interchromatin compartment model:
  This model [15] introduces two principle nuclear compartments: chromosome territories and an interchromatin compartment. In this model, the space between CTs is called the interchromatin compartment (figure 2.2b, left). Within each CT, the chromosome is divided into decondensed and more condensed regions. Decondensed regions of different chromosomes are adjacent to the shared interchromatin compartment and are in close proximity to each other, so they often make contact with each other, and share the same transcription factories.

- Interchromatin network model:

  This model says that although chromosomes have their own territories, they intermingle significantly with each other (figure 2.2b, right), that cannot be explained by the separation of chromosome territories and interchromatin compartment [19]. The intermingling regions are decondensed regions of chromosomes having functional significance in the nucleus, and associated with euchromatin, which we explain in detail in the genome compartmentalization section.

2.3.2 Genome compartmentalization

Earlier microscopy experiments introduced two different types of chromatin which are spatially separated within the nucleus, heterochromatin, and euchromatin [32, 44, 35]. Genome-



(a) (b)

Figure 2.2: (a) Chromosomes in Interphase nucleus vs Metaphase nucleus, Reprinted from [2], (b) Chromosome territory-interchromatin compartment model vs Interchromatin network model to explain chromosome territories and their connections, Reprinted from [1]

wide analysis of epigenetic assays has shown that euchromatin is associated with more decondensed structure, gene-rich and active histone marks, while heterochromatin is less accessible due to the condensed structure, and is transcriptionally inactive, and bound with inactive histone marks. Filion et al [26] also identified 5 principal chromatin types based on genome-wide location maps of 53 chromatin proteins and 4 histone modifications in Drosophila cells that are associated with a unique combination of proteins. Moreover, analysis of Hi-C data has also revealed that euchromatin and heterochromatin are spatially segregated, as euchromatin is mainly positioned in the nucleus interior space, while heterochromatin is located close to the nuclear periphery, and they also can be further separated into different subcompartments [44]. Different methods have been proposed to identify different types of chromatin using Hi-C data. These methods are described later in domain annotation using the Hi-C data section.

2.3.3 Topologically Associating Domains

At a finer resolution (scale of tens to hundreds kb), chromosomes fold into domains that have more intra-domain interactions than interactions with the neighboring domains. These contact domains are called topologically associating domains (TADs) and they appear along the diagonal of a Hi-C observed contact count matrix as blocks with increased self-interaction frequencies (figure 2.3a) [48].

2.4 Domain annotation

Different genome-wide chromatin properties have been measured through biochemical techniques such as ChIP-Seq, DNase-Seq, Repli-Seq, and ATAC-Seq in the last decades. Such 1D signals, together with Chromosome Conformation Capture (3C) assays gave a detailed insight into the genome structure and function. However, it is hard to get a comprehensive picture of the genome from these diverse properties. For example, it is not obvious how to

compare the available data for different cell types to understand the mechanisms behind the difference in their gene expression profiles. So, computational methods are needed to model these properties and transform the inputs into an interpretable output like an activity or structural pattern of each genomic region. Previous studies have shown that the genome functional behaviors are in part the consequence of domain-scale regulation, in which regions of hundreds or thousands of kilobases known as domains are regulated as a unit, or have the same biochemical properties [34, 20, 18]. Therefore, different methods have been proposed to identify domains based on different input data types. These methods could be classified according to the data types they can use as an input, including 1D genomic signals, Hi-C matrix, and their combination. In this section, we explain some of the methods in each of these classes.

2.4.1 Domain annotation using 1D genomic signals

Semi-automated genome annotation (SAGA) methods jointly analyze different 1D properties to discover known and novel patterns associating with different biological phenomena [25, 33]. More formally, SAGA methods take multiple genome-wide signals as an observation and learn a model to assign a label to each genomic region such that regions with the same label have similar patterns in the input data. The domain annotation is the genome annotation in the domain scale (length of genomic regions in the range from 10 Kbp to 1 Mbp), which results in identifying domain types that summarize the input datasets. We briefly explain the input, model, and output of these methods in this section.

Input data

Assuming that we have a signal representative of the input data, we need to perform some pre-processing:

- Binning the signal into bins
  We need input signals in different resolutions according to the type of annotation we need. For example, if we want to do the domain annotation task, we need 10-kbp to 100-kbp resolution input signals. We can use different statistics to downsample 1-bp resolution signal into bins, for example, weighted average or maximum of signal values in each bin. We use the weighted average in our experiments.

- Stabilizing the signal variance
  Since the input signals are from different experiments, they have different mean values, and it has been shown that the variance of the genomic signals is dependent on their mean value [17]. So, it is good to transform the signals using $asinh(x)$ or log_2^x. The transformation also reduces the distorting effects of high data values in input signals. We used $asinh(x)$ transformation in our experiments.

Model

Most of the SAGA methods are based on a Hidden Markov Model (HMM) that model the genomic signals as sequences of observations. Assuming that such observations depend on an unobserved Markov process (sequence of hidden states), HMM aims to learn the most probable Markov process by observing the input signals. The learned hidden states provide the clustering for the genomic bins. We explain a Segway method that we used in our study in methods section in more detail.

Output labels

Since SAGA methods are unsupervised, the output labels are just integers without any meaning. Ideally, each label in our resolution setting (domain-scale) corresponds to a domain type having its own biological properties, for example, expressed or inactive domain. We need to assess the enrichment of different biological phenomena for each label and assign an interpretation to them.

2.4.2 Domain annotation using Hi-C data

As explained in the Genome higher-order organization section, analysis of Hi-C data has revealed different types of chromatin which are spatially segregated, and they have their own spatial properties like a relative distance to other nuclear components. In this section, we describe different methods used to compartmentalize a genome using Hi-C data. Before that, we need to define a few terms related to the Hi-C matrix.

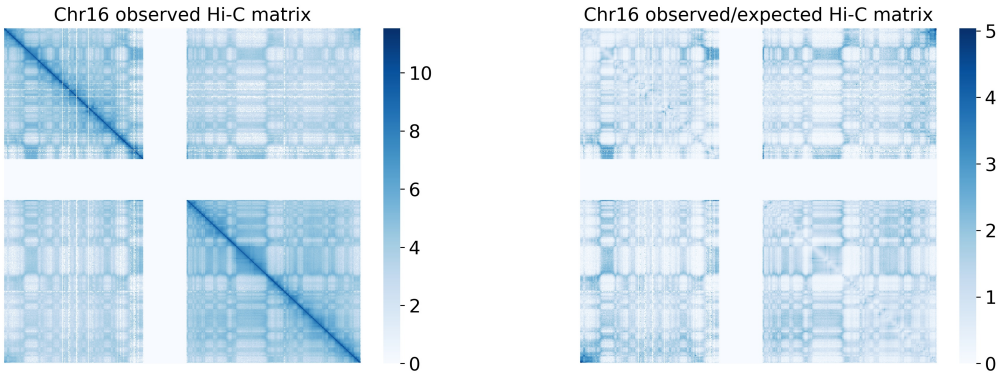
- Intra-chromosomal vs inter-chromosomal Hi-C matrix

  The intra-chromosomal Hi-C matrices are defined for each chromosome separately and include interactions with the source and target in that particular chromosome.

  The inter-chromosomal interactions are the interactions that their source and target are located in different chromosomes. The inter-chromosomal Hi-C matrix could be defined for every two different chromosomes separately, however, it is not possible to define a genome-wide matrix that includes all inter-chromosomal interactions (rows and columns should not have an overlap in terms of the chromosome to exclude intra-chromosomal interactions), so Rao et al [44] defined an odd-even inter-chromosomal matrix. In this matrix, rows and columns are respectively associated with the genomic bins in odd and even chromosomes.

- Observed vs observed/expected Hi-C matrix

  The observed Hi-C matrix includes the raw interactions counts between the regions (figure 2.3a).



(a) (b)

Figure 2.3: (a) Chromosome 16 observed Hi-C matrix (arcsinh-transformed) has larger values around the diagonal, (b) Chromosome 16 observed/expected Hi-C matrix (arcsinh-transformed). The effect of genomic linear distance on interactions counts is removed after normalization.

The observed/expected (O/E) Hi-C matrix includes the ratio of observed to expected interactions by assuming that regions are expected to interact depending on their linear distance along the chromosome (figure 2.3b). The expected number of intra-chromosomal interaction for chromosome i is calculated for each genomic distance d, as a mean of interactions counts between two loci in chromosome i with linear distance $= d$, and then all observed interactions counts are normalized with the expected counts corresponding with their distance.

The expected number of inter-chromosomal interactions between chromosome i and j is calculated as a mean of all interactions counts with one end located in chromosome i and the other end in chromosome j.

Eigenvector analysis of intra-chromosomal Hi-C matrix [35]

The plaid pattern of the O/E intra-chromosomal Hi-C matrix (figure 2.3b) suggests that each chromosome can be segregated into two compartments such that contacts within each compartment are enriched and contacts between compartments are depleted [35]. Also, the analysis of Hi-C data in finer resolution has shown that each compartment can be decomposed into different subcompartments [44]. To accurately find the transitions between compartments, Lieberman et al [35] used principal component analysis to find the principal components (PC) of the O/E intra-chromosomal Hi-C matrix for all chromosomes. They showed that the first PC reflects the plaid pattern (positive values defining one compartment and negative values the other). Moreover, they observed that the clusters within each chromosome were consistent among different chromosomes, meaning that cluster labels in

different chromosomes can be mapped to each other so that regions with the same label have similar contact profiles, and regions with different labels have different contact profiles. So, they partitioned the entire genome into two compartments (A and B), which are spatially separated. They showed that A compartment is associated with euchromatin that includes transcriptionally active regions and is located in the nuclear interior, while B compartment is associated with heterochromatin, including transcriptionally inactive regions, and is located near the nuclear periphery.

Clustering the rows of the inter-chromosomal Hi-C matrix [44, 56]

Yaffe et al [56] clustered genomic loci into 3 clusters: active, inactive centromer-proximal and inactive centromer-distal. First, they define a genome-wide contact map D such that each entry D_{ij} represents a normalized contact between loci i and j:

$$D_{ij} = \frac{O_{ij}}{E_{ij} \times N_i \times N_j}$$

E_{ij} is the number of the expected contacts between the regions i and j, which they calculated using their own model counting for few biases. $N_i = \frac{O_i}{E_i},\ (O_i = \sum_j O_{ij},\ E_i = \sum_j E_{ij})$ is a normalization factor that takes into account the variable total coverage for different loci. Then, they adopt the standard k-means clustering algorithm to perform contact profile clustering on the normalized matrix D. Each of the rows in a matrix represents proximity between a corresponding region to that particular row and all other genomic regions. To ignore the intra-chromosomal contacts (they cause a bias toward the coclustering of the loci in the same chromosome), their values are removed from the contact matrix D. Given these definitions, k-means clustering assigns labels to each genomic loci according to the similarity of their inter-chromosomal contact profile.

Later, Rao et al [44] generated very dense Hi-C data containing 4.9 billion contacts. They defined the odd-even Hi-C matrix instead of using a genome-wide matrix to ignore the intra-chromosomal contacts, such that rows and columns are respectively associated with the genomic bins in odd and even chromosomes. Then, they calculated the log of the entries and applied the z score on each of the rows for normalization (to remove the effect of variable total coverage for different loci). Assuming the matrix rows as samples and the columns as features, they trained HMM on the normalized matrix to infer the label for each genomic locus in the odd chromosomes. They did the same process (matrix normalization and clustering) for the transpose of the odd-even Hi-C matrix to find the labels for the even chromosomes loci. They paired the learned labels for odd and even chromosomes such that paired labels show a similar contact pattern. They introduced 6 subcompartments (A1, A2, B1, B2, B3, B4) having distinct spatial and functional patterns. We describe each of these subcompartments' functional and spatial properties in more detail in the results chapter.

Clustering the latent variables from embedding the inter-chromosomal Hi-C graph [16]

The sparsity of the Hi-C matrix and the fact that A/B compartments pattern in the Hi-C matrix is captured with the first principal component of a matrix suggests that contacts information in the $N \times N$ Hi-C matrix could be represented in a denser $d \times N$ matrix such that $d << N$.

The node embedding methods aim to encode nodes in a graph so that similarity in the embedding space approximates similarity in the original graph. The Hi-C matrix can be represented as a graph such that its nodes represent the genomic loci, and edges show the interactions counts between pairs of loci. Applying the node embedding methods on a Hi-C graph projects the contacts information in a graph into the embedding for all loci in a vector space. Ashoor et al [16] used a node embedding method, LINE, to learn an embedding for genomic loci, and then apply k-means on the learned embeddings to cluster the genomic loci according to their interaction pattern. They also introduced five subcompartments (two active and three inactive) each of them having their own spatial and functional properties.

2.4.3 Joint analysis of 1D genomic signals and Hi-C data

In this section, we explain two unified frameworks, Segway-GBR and SPIN, that integrate 1D genomic signals and 2D Hi-C matrix to assign a label to genomic regions.

Segway-GBR

Previous studies have shown that colocalized chromatin domains tend to have a similar activity [34, 35, 44], so Graph-Based Regularization Segway (Segway-GBR) [34] aims to extend the Segway to integrate chromatin contacts information from Hi-C data using the pairwise prior that positions close in 3D should be more likely to be identified as the same domain type.

The Segway models the observed variables (X_O) depending on the hidden states (X_H) using a probability distribution parameterized by θ, $p_\theta(X_H, X_O)$. To train a model given an instantiation of observed variables (x_O, input signals), the following objective function should be optimized:

$$maximize_\theta \ \mathcal{J}(\theta) = \mathcal{L}(\theta) + \mathcal{R}(\theta)$$

$$\mathcal{L}(\theta) = \log p_\theta(x_O) = \log \sum_{x_H} p_\theta(x_H, x_O)$$

where $\mathcal{L}(\theta)$ is the likelihood of the parameters given the observed variables, and $\mathcal{R}_\theta$ is a regularizer that expresses prior knowledge about the parameters.

The probability distribution used in Segway is chain-structured, meaning that variables at position i are independent of all but variables at positions $i-1$ and $i+1$. So, Segway-GBR aims to add pairwise dependencies prior expressed in the input graph (Hi-C data in

this case) to the Segway model. They employ a strategy based on posterior regularization to integrate this prior. This is done by introducing an auxiliary joint distribution $q(X_H)$, and encouraging q to be similar to p_θ, while expressing a pairwise prior knowledge in the graph. So, the regularizer objective for the parameters is:

$$\mathcal{R}_{PR}(\theta) = \max_q \mathcal{R}_{PR}(\theta, q)$$

$$\mathcal{R}_{PR}(\theta, q) = -D(q(X_H) \parallel p_\theta(X_H|x_O)) + \mathcal{PR}(q)$$

where, D is the KL divergence to encourage the distributions q and p_θ to be close to each other, and $\mathcal{PR}(q)$ is a posterior regularizer that expresses a pairwise prior from a graph. Given a weighted, undirected regularization graph over the hidden variables $G_R = (H, E_R)$, where $E_R \subseteq H \times H$ is a set of edges with non-negative weights ($w : E_R \to \mathbb{R}_+$), the marginal distribution of two hidden variables X_u and X_v should be similar if $w(u, v)$ is large. The marginal distribution of a hidden variable h, $q_h^M(X_h)$ is defined as $q_h^M(x_h) = \sum_{x_{H/h}} q(x_H)$. So, considering λ_G as a hyperparameter controlling the strength of regularization, the posterior regularization is:

$$\mathcal{PR}(q) = -\lambda_G \sum_{(u,v)\in E_R} w(u, v) D(q_u^M(X_u) \parallel q_v^M(X_v))$$

And the full objective is:

$$maximize_{\theta,q}\, \mathcal{J}_{GBR}(\theta) = \mathcal{L}(\theta) - D(q(X_H) \parallel p_\theta(X_H|x_O)) - \lambda_G \sum_{(u,v)\in E_R} w(u, v) D(q_u^M(X_u) \parallel q_v^M(X_v))$$

The dynamic programming algorithms used for inference in the chain-structured model cannot be used in the GBR model. So, they developed a new algorithm alternating between two steps, inference and message passing, iteratively. In the inference step, the model gets evidence from the past message passing step as a form of distribution $r_h^M(X_h)$ for each hidden variable h. These distributions are used in the original chain model (maximizing the first term of $\mathcal{J}_{GBR}$, $\mathcal{L}(\theta)$) to compute a posterior distribution over the labels. Then, in the next message passing step, the algorithm updates r^M to minimize the KL penalties (second and third term of $\mathcal{J}_{GBR}$). The inference and message passing steps are iterated until convergence.

So, Segway-GBR incorporates the 3D structure information of a genome by adding a posterior graph-based regularization term to the objective function. They introduced five types of domains: (1) broad expression, (2) specific expression, (3) facultative heterochromatin, (4) constitutive heterochromatin, (5) quiescent. These labels are more described in the result section.

SPIN

Several genome-wide mapping assays measured the association between chromosome regions and specific nuclear compartments. For example, TSA-Seq [21] is the first method capable of estimating cytological distances of chromosome loci genome-wide relative to a particular nuclear compartment like nuclear speckle or lamina. Guelen et al [29] also used DamID to measure contact frequencies between chromatin with the nuclear lamina. The Hi-C data also provides information about the genome loci localization indirectly, as the regions with more interactions are closer to each other. Therefore, SPIN (Spatial Position Inference of the Nuclear genome) [54] integrates TSA-Seq, DamID, and Hi-C data in a unified framework based on hidden Markov random field to infer genomic regions spatial state. In this section, we explain their model to integrate the 1D signals and 2D Hi-C data.

They use four different 1D genomic signals, TSA-Seq for nuclear speckle, TSA-Seq for nuclear lamina, DamID for nuclear lamina, and DamID for nucleoli (the input signals are binned at 25 kbp resolution). They use a type of probabilistic graphical model called hidden Markov random field (HMRF) to capture the dependencies (edges) between the 25 kbp genomic regions (nodes). More formally, HMRF can be represented as an undirected graph $G = (V, E)$, where each node represents a non-overlapping 25 kbp genomic region, and E represents the set of edges coming from significant Hi-C interactions, adjacent nodes on chromosome and adjacencies introduced by large structural variations specific to their experiment cell type.

Each node i is associated with a hidden state H_i, representing the spatial localization of genomic region i (note that state types are dependent on the input types) and $O_i \in \mathbb{R}^d$, representing the observed inputs for genomic region i (d is the number of the input signals). H_i is dependent on O_i and $\{H_j | j \in V, (i,j) \in E\}$. They aim to maximize the following joint probability:

$$P(\overrightarrow{H}, \overrightarrow{O}) = \propto \frac{1}{Z} \prod_{i \in V} P_V(O_i|H_i) \prod_{(i,j) \in E} P_E(H_i, H_j)$$

where $P_V(O_i|H_i)$ corresponds to the node potential of observation O_i given the hidden state H_i, and $P_E(H_i, H_j)$ corresponds to the edge potential of two nodes i and j with hidden states H_i and H_j. They model the node potential with multivariate Gaussian distribution:

$$P_V(O_i|H_i = h_a) = \frac{1}{\sqrt{(2\pi)^d |\Sigma^{h_a}|}} exp\{-\frac{1}{2}(O_i - \mu^{h_a})^T [\Sigma^{h_a}]^{-1} (O_i - \mu^{h_a})\}$$

where O_i follows multivariate Gaussian distribution $N(\mu^{h_a}, \Sigma^{h_a})$ given state $H_i = h_a$. The edge potential is modeled by the transition probability matrix between pairs of states h_a and h_b:

$$P_E(H_i = h_a, H_j = h_b) \propto t(h_a, h_b)$$

To maximize a joint probability $P(\overrightarrow{H}, \overrightarrow{O})$, first, they estimate the parameters in the model using the Expectation-Maximization (EM) algorithm [23], and then infer the hidden states given the observations and parameters using the loopy belief propagation algorithm [39].

They identified 10 SPIN states that represent nuclear compartmentalization patterns in the K562 cell line. As expected, the states also have a distinct distribution of functional genomic signals, although the input for SPIN does not use any functional genomic data.

Chapter 3

Thesis Problem

Our goal is to identify domain states that explain the variance of both genome-wide regulation activities (such as gene expression and histone modifications) and spatial compartmentalization by integrating functional assays data including histone modification ChIP-seq and DNase-seq together with Hi-C data.

Although there have been studies showing that the genome spatial compartmentalization and epigenetic marks are highly correlated and can be predicted from each other [14, 58, 42], domain states inferred from each of these data types are good for identifying pattern of specific genomic activities. For example, subcompartments from Rao et al [44] capture genome spatial positioning very well, as they explain the variance of replication timing profile and TSA-seq data which are dependent on spatial positioning. However, they do not stratify transcriptional activities as well as domain states inferred using epigenetic assays. For example, A and B compartments show a clear distinction for transcriptional activities, as A regions are densely populated by active genes while B regions are mostly depleted of them. However, subcompartments within each compartment do not have different gene expression patterns (A1: 21.1, A2: 21.06, B1: 4.39, B2: 3.34, B3: 4.6 (numbers are average gene expression in RPKM unit)). On the other hand, domain types that are based on the genome epigenetic profile better stratify the transcription patterns comparing to subcompartments, while do not explain the variance of positional properties like TSA-seq data measuring chromosome distances to Lamin and SON (figure 5.3c).

Therefore, we propose a framework to incorporate both 1D functional assays signals and 2D Hi-C matrix to identify domain states (fig 3.1). In which, we first extract spatial features from the Hi-C matrix using a graph embedding method, and then feed the inferred spatial features in addition to the functional features coming from epigenetic assays to the Segway, a genomic segmentation method, that infer domain states using a dynamic bayesian network.

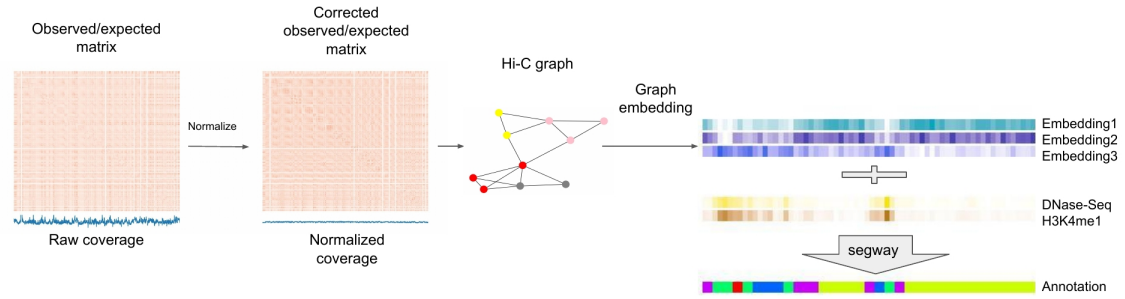


Figure 3.1: Our framework: We construct a Hi-C graph after preprocessing a genome-wide Hi-C matrix, and then learn embeddings that represent spatial properties of genomic regions (nodes in a Hi-C graph). We pass the learned embeddings together with existing 1D genomic signals to the Segway to infer the domain states.

Chapter 4

Methods

4.1 Data processing

4.1.1 Hi-C data

We downloaded GM12878 cell line processed Hi-C data with GEO entry 63525 [1]. Then, observed over expected (O/E) contact frequency files for every pair of chromosomes at 100K resolution were extracted using a Juicer tool [24]. SCI [16] uses interchromosomal interactions to construct a Hi-C graph, but we found out that ignoring intrachromosomal edges results in a larger similarity of embeddings within a chromosome, which makes it unfeasible to cluster same domain types from different chromosomes together. So, we use both intra and inter chromosomal O/E matrices to construct a genome-wide Hi-C matrix. We observed that different bins have different visibility in genome-wide O/E matrix, meaning that the sum of the matrix rows is spread out over a large range of values, which also results in dependency of embeddings on bins visibility. Therefore, we do the normalization on this genome-wide matrix based on mean of its rows, so each entry O_{ij} in matrix is being normalized by $O_i * O_j$ ($O_i = mean_j(O_{ij})$). Finally, we do hyperbolic arcsine transformation on matrix O. We observe that this preprocessing allows us to use both intra and inter chromosomal contact frequency information simultaneously to learn embedding for all genomic bins in the same space.

4.1.2 Genomic functional assays

We downloaded GM12878 cell line (E116) ChIP-seq data sets targeting DNase, H2A.Z, and 10 histone modifications (H3K4me1, H3K4me2, H3K4me3, H3k9ac, H3K9me3, H3K27ac, H3K27me3, H3K36me3, H3K79me2, H4K20me1) from the Roadmap Epigenomics data por-

[1] `https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE63525`

tal[2]. To calculate a signal value for each 100k length bin, we get a weighted average of signal value over that bin.

4.2 Models

In general, our framework has two steps (Figure 3.1). In the first step, we infer spatial features for genome loci through embedding a Hi-C graph into the feature space. In the second step, we input such spatial features in addition to the functional features into the genome annotation methods. In this section, we first introduce a graph embedding problem and specifically explain the LINE and Autoencoder methods that we use in this work, and then we describe genome annotation methods.

4.2.1 Graph embedding

The graph data structure has been employed extensively in different fields to model relational knowledge about interacting entities. One main problem to answer within a graph is to classify nodes based on their structural properties in a graph. Traditional machine learning approaches first extract some graph statistics (e.g., degree or centrality), and then use such features as an input to the model, however, they are very limited as they rely on inflexible hand-engineered features [30]. Therefore, node embedding methods have emerged to automate the first step and learn representations that preserve graph structural information.

Formally, the node embedding problem is defined as follows: Given a graph $G = (V, A)$, with V as the node set and A as the adjacency matrix, the goal is to learn a function $V \rightarrow \mathbb{R}^d$ that maps each vertex to a d-dimensional vector that capture its structural properties [43].

Hi-C matrix represents observed/expected (O/E) contacts between all pairs of non-overlapping 100KB length bins, so it can be represented as a graph such that its nodes are the genomic bins, and the edges represent O/E contact frequency between node pairs. We use graph embedding methods to learn latent features for every node of a Hi-C graph to map structural information in the Hi-C graph to structural features for each genomic bin. Ashoor et al [16] predict subcompartments by applying LINE [50] method on interchromosomal Hi-C graph, followed by k-means clustering on learned embeddings. We also use the LINE method on the whole Hi-C graph to learn structural features in this study. As an alternative model, we used an autoencoder architecture proposed in SNIPER paper [55] to learn the embeddings for the genomic bins. We are going to explain these two graph embedding methods, autoencoder, and LINE, in the following section.

[2] `https://egg2.wustl.edu/roadmap/data/byFileType/signal/consolidated/macs2signal/foldChange/`

Autoencoder

An autoencoder is an unsupervised neural network that learns to encode input data to the lower-dimensional space so that it can reconstruct the input data from the encoded representation as close as possible to the original input data. Autoencoders consists of 3 main parts: encoder, decoder, and reconstruction loss. The encoder maps the N-dimensional input data to the d-dimensional encoded representation that $d << N$, and the decoder maps the encoded representation to the output data which is close to the input data according to the reconstruction loss function.

The neighborhood matrix of a weighted graph $G = (V, E, w)$ (V: nodes, E: edges, w: weight function that map edges to real values) is a $|V| \times |V|$ matrix M that entry $M_{ij} = w(E_{ij})$. The neighborhood matrix can be viewed as a feature matrix, in which rows are the samples (nodes) and the columns are features (neighborhood vector of a node). The neighborhood matrix can be fed into an autoencoder to extract latent variables corresponding to each node (genomic bin) that represents its contact pattern.

SNIPER [55] is a new computational method that predicts subcompartments annotation for cell types with low-coverage Hi-C data (Hi-C data from low coverage sequencing data) based on an autoencoder and multi-layer perceptron (MLP) classifier. Although Hi-C data from GM12878 cell type has almost 5 billion mapped read pairs, most available Hi-C datasets have relatively low coverage, 400 million to 1 billion mapped read pairs, and their contact matrices are too sparse to predict the subcompartments. Therefore, computational methods have been developed to enhance the resolution of the low-coverage Hi-C data. They rely on learning a neural network that maps the sparse Hi-C matrix to the dense Hi-C matrix through the training data from available high coverage Hi-C datasets. SNIPER utilizes an autoencoder that takes the low-coverage odd-even inter-chromosomal Hi-C matrix as an input to recover the high-coverage odd-even inter-chromosomal Hi-C matrix and uses an MLP classifier to predict the subcompartments from the learned representations in the middle layer of an autoencoder (Note that they use the contact probability instead of the contact count in the Hi-C matrix. Contact probability between loci i and j, $P_{ij} = exp(-\frac{1}{C_{ij}})$ where C_{ij} is the contact frequency between loci i and j). We use their autoencoder architecture to compress the inter-chromosomal Hi-C contact map. It contains 9 sequential linear layers with N_{loci}, 1024, 512, 256, 128, 256, 512, 1024, N_{loci} neurons, respectively (N_{loci} is the number of rows in the input contact matrix). Activation of neurons is computed by:

$$z_i(x_i) = g_i(W_i x_i + b_i)$$

where z_i is the activated output vector of layer i, x_i is the n-dimensional input to layer i, W_i is the $m \times n$-dimensional weight matrix of layer i (m is the output dimensionality of the layer i), b_i is the m-dimensional bias vector of layer i, and g_i is the activation function used in the layer i. They use the rectified linear unit (ReLU) activation function for the hidden

layers:

$$ReLU(z) = \begin{cases} z & \text{if } z > 0 \\ 0 & \text{if } z \leq 0 \end{cases}$$

, the linear activation function for a latent layer, and the sigmoid activation function for an output layer:

$$\sigma(z) = \frac{1}{1 + exp(-z)}$$

to limits the values in an activated output to be within $[0, 1]$ to match the range of values in the input probability matrix.

They use binary cross-entropy (BCE) loss function to measures the difference between an input contact probability vectors and the reconstructed ones.

$$\hat{\theta} = arg\min_{\theta}\{-\sum_{i=1}^{N}(y_i^T \log(\hat{y}_i) + (1 - y_i)^T \log(1 - \hat{y}_i))\}$$

where the autoencoder parameters θ are optimized to minimize the cross entropies between predicted outputs $\hat{y}_i$ and target outputs (same as inputs here) y_i for all N training inputs.

We use the available high-coverage Hi-C data for GM12878 as both input and output of an autoencoder, and use the learned latent variables in a middle layer as genomic bin embeddings in our downstream analysis.

LINE

Large-scale Information Network Embedding (LINE) [50] has been used to map very large networks into low-dimensional vector space. They learn embeddings for nodes in a graph such that pairwise distances in embedding space would be representative of the proximity of pair of nodes in a graph. We chose LINE because it can address the embedding problem for large and weighted graphs. LINE can be optimized based on the first-order or second-order proximity loss function.

First-order proximity is the local pairwise proximity between two nodes. The greater weight between nodes indicates more proximity between nodes. By optimizing the first-order proximity objective, we encourage nodes with a large weight between them to get similar embeddings. To model this problem, they define a joint probability distribution $(p_1(.,.))$ over node pairs $(V \times V)$ space:

$$p_1(v_i, v_j) = \frac{1}{1 + exp(-\overrightarrow{u_i}^T . \overrightarrow{u_j})}$$

where $\overrightarrow{u_i} \in \mathbb{R}^d$ is the representation of node i in the vector space. This joint probability distribution is supposed to be close to empirical distribution $\hat{p_1}(v_i, v_j) = \frac{w_{ij}}{W}$, that $W = \sum_{(i,j)\in E} w_{ij}$. To model the closeness between two distributions, they use KL-divergence, so

after eliminating constants, the first-order proximity objective would be:

$$O_1 = -\sum_{(i,j)\in E} w_{ij} \log p_1(v_i, v_j)$$

On the other hand, second-order proximity is based on the similarity between how two nodes are connected to all other nodes. For example, if we show the first-order proximity of node u with all other nodes with $p_u = (w_{u,1}, ..., w_{u,|V|})$, second-order proximity of nodes u and v is based on the similarity between p_u and p_v. To model the second-order proximity, they define two vectors corresponding with each node i, one is its embedding (u_i), and the other one is its context (u'_i) that comes from the embedding of its neighbors. Then, they define $p_2(.|v_i)$, distribution over all nodes, for each node i, that determine the probability that the context of each node has been generated using embedding of node i.

$$p_2(v_j|v_i) = \frac{exp(\overrightarrow{u}'^T_j \overrightarrow{u}_i)}{\sum_{k=1}^{|V|} exp(\overrightarrow{u}'^T_k \overrightarrow{u}_i)}$$

To preserve the second-order proximity, $p_2(.|v_i)$ should be close to the empirical distribution coming from graph, which they define as $\hat{p}_2(v_j|v_i) = \frac{w_{ij}}{d_i}, d_i = \sum_j w_{ij}$. They minimize weighted sum of distance between $p_2(.|v_i)$ and $\hat{p}_2(.|v_i)$ over all nodes. Weights show the importance of the nodes in a graph, which are equal to degree of the nodes here. They use KL divergence for measuring closeness of distributions again, and second-order proximity objective function after eliminating the constants would be:

$$O_2 = -\sum_{(i,j)\in E} w_{ij} \log p_2(v_j|v_i)$$

Previous studies on compartmentalization used bins interaction patterns with the rest of the genome to infer their labels. This is equivalent to the second-order proximity in the LINE method. We tried both 1st and 2nd orders embeddings, and 2nd order embeddings result in meaningful compartmentalization as expected. So we just use embeddings from the second-order proximity in this study.

4.2.2 SAGA

Semi-automated genome annotation (SAGA) methods jointly analyze different 1D properties to discover known and novel patterns associating with different biological phenomena [25, 33]. More formally, SAGA methods take multiple genome-wide signals as an observation and learn a model to assign a label to each genomic region such that regions with the same label have similar patterns in the input data. We pass learned spatial features and

functional features from epigenomic assays together into a Segway, a SAGA method that we explain in this section.

Segway

Segway is a SAGA method relying on a Dynamic Bayesian Network (DBN) model, which generalizes the basic HMM by allowing for more hidden state variables. We explain it in detail here:

- Mathematical formulation: The Segway models n input signals as continuous observation tracks $X^{(i)} \in \{X^{(1)}, ..., X^{(n)}\}$. $X^{(i)}$ is a sequence of length T, and $X_t^{(i)}$ is the value of signal i at position t. $X^{(i)}$ might have a missing value at some positions, so $\mathring{X}_t^{(i)}$ is an indicator variable to mark whether $X_t^{(i)}$ is defined ($\mathring{X}_t^{(i)} = 1$) or undefined ($\mathring{X}_t^{(i)} = 0$). $X_t^{(i)}$s are only dependent on the hidden segment label variable at that position Q_t when an indicator variable $\mathring{X}_t^{(i)} = 1$, and $X_t^{(i)}$ does not have a dependence on Q_t if $\mathring{X}_t^{(i)} = 0$.

- Weighting: The contribution of each of the tracks in the overall probability is affected by the number of available data points in that track. Therefore, Segway weights the contribution of each observation track i by the number of available data points for that track $N^{(i)} = \sum_t \mathring{X}_t^{(i)}$ divided by the maximum number of available data points for any particular track $N^* = max_j N^{(j)}$:

$$P'(X_t^{(i)}|Q_t) = P(X_t^{(i)}|Q_t)^{\frac{N^{(i)}}{N^*}}$$

  So, changes in the probabilities corresponding to the tracks with more missing values would be amplified.

- The duration model: The Segway model uses additional random variables to specify the minimum and maximum length of segments. Such limitations are helpful in 1 bp resolution setting, as they allow modeling broader behavior when the observed variables show a transition. Since we binned the signals into 100 Kbp genomic regions, observed variables are robust and we do not need to have those constraints.

- Parameters: The Segway models $P(X_t^{(i)}|Q_t)$ with an $m \times n$ matrix (μ) of scalar Gaussians, with one Gaussian for each combination of n observation tracks and m hidden states. The values in a matrix are the Gaussian mean parameters (μ_{ij} is the Gaussian mean for the track j if it is emitted from the state i), and all the Gaussian distributions share the variance parameter σ^2. The $m \times m$ transition probability matrix (T) includes the conditional probabilities for each pair of the hidden states ($T_{ij} = P(Q_t = i|Q_{t-1} = j)$).

- Training: The Segway uses the expectation-maximization algorithm [23] to train the model. They set initial parameters, and iteratively define an expected value of the model likelihood with respect to the current estimates of the parameters, and update the parameters to maximize the defined expected value. They continue this process until the convergence of the model likelihood. They perform ten separate instances of training and use parameters of an instance with the highest final likelihood to infer the hidden states.

- Decoding: Given a sequence of observed variables and the learned parameters, we aim to know the most probable sequence of hidden states Q^*. The Segway uses the Viterbi algorithm [52] to find Q^*. The Viterbi is a dynamic programming algorithm for finding the most likely sequence of hidden states, the Viterbi path, that results in a sequence of observed events.

4.3 Evaluation methods

4.3.1 Signal variance explained:

We use variance explained (VE) to measure the fraction of signal variance that is explainable using an annotation prediction. Given a genome annotation $a_{1:n} \in \{1...K\}^n$ (K is the number of labels) and a signal vector $s_{1:n} \in \mathbb{R}^n$, first we compute mean of the signal over positions with label k for all labels.

$$\mu_k = \frac{\sum_{i=1}^{n} \mathbb{1}(a_i = k) s_i}{\sum_{i=1}^{n} \mathbb{1}(a_i = k)}$$

Predicted signal vector is $s_i^p = \mu_{a_i}$. VE is computed as the difference of total variance $\text{var}(s_i)$ and the variance of residuals of the prediction $\text{var}(s_i - s_i^p)$.

$$VE = \frac{var(s_i) - var(s_i - s_i^p)}{var(s_i)} = 1 - \frac{var(s_i - s_i^p)}{var(s_i)} = 1 - \frac{\sum_{i=1}^{n} (s_i - s_i^p)^2}{\sum_{i=1}^{n} (s_i - \overline{s})^2}$$

VE is bounded by the range [0,1], and higher values show more agreement between annotation and signal values.

4.3.2 Random Forest classification

To evaluate the quality of the embeddings, we use a random forest classifier to predict the known subcompartments from Rao et al [44] based on embeddings and use the model test accuracy to assess how good predictors the embeddings are. A decision tree learning

is a non-parametric supervised learning method used for regression and classification. It uses a tree structure to model the decision problem in such a way that each non-leaf node corresponds to a feature and its branches correspond to the decision rule based on that feature, and leaves represent classification labels (or regression outputs). A random forest consists of a large number of decision trees that are fitted independently based on a random subset of features and training data. Each tree in the classification random forest returns a predicted class, and the most seen class will be the final prediction for a classification problem. We used the RandomForestClassifier function from the sklearn package in Python for the implementation.

Chapter 5

Results

5.1 Overview of the experiments

Our framework incorporates both 1D functional assays signals and 2D Hi-C matrix to identify domain states (fig 3.1). It consists of two steps: first, we should extract spatial features from the Hi-C matrix using a graph embedding method, and then feed the inferred spatial features in addition to the functional features coming from epigenetic assays to the Segway.

More formally, we construct a normalized observed/expected whole genome Hi-C matrix M at 100 kb resolution (described in Method section), and define its corresponding Hi-C graph $G = (V, E, w)$ such that $V = \{g_1, g_2, ..., g_{|G|}\}$ (g_i is the ith genomic bin, and $|G|$ is the number of 100 kb bins in the whole genome), $E = \{(g_i, g_j),\ \forall g_i, g_j\ s.t\ M[g_i, g_j] > 0\}$ and $w = \{(g_i, g_j) \mapsto M[g_i, g_j],\ \forall (g_i, g_j) \in E\}$.

We use a node embedding method, LINE, to embed the nodes of a graph G into a vector space. LINE outputs $\{l^1_{1:d}, l^2_{1:d}, ..., l^{|G|}_{1:d}\}$, which are d-dimensional latent representations for each node. These latent variables preserve the structural information in a Hi-C graph, and we utilize them as d spatial features defined over the whole genome $\{S_i = [l^1_i, l^2_i, ..., l^{|G|}_i],\ \forall i \in \{1...d\}\}$. Finally, we pass the spatial features S together with the functional features coming from binned epigenetic signals to the Segway, a genome annotation method, to assign a label to each genomic bin.

We use an autoencoder (SNIPER architecture) to get the embeddings as an alternative method. We also tried using TSA-seq signals instead of Hi-C embeddings as a baseline, since they also represent spatial properties of a genome. The input types in different experiments are summarized in table 5.1.

In the following sections, first, we show that our inferred spatial features are a good representation for genome positional information in the whole Hi-C matrix, and then we introduce the resulting domain states from our method and their biological interpretation. Finally, we demonstrate their improved ability to identify different regulation activities as

Experiment name	Input types
segway_fa	1D functional genomic assays
segway_LINE	Hi-C embeddings from LINE method
segway_LINE_fa	1D functional genomic assays + Hi-C embeddings from LINE method
segway_SNIPER_fa	1D functional genomic assays + Hi-C embeddings from SNIPER
segway_TSA_fa	1D functional genomic assays + Lamin and SON TSA-seq data

Table 5.1: Experiments input types

well as respecting the 3D organization of a genome comparing to methods that rely on using one data type.

5.2 Extracted spatial features are good representations for the whole Hi-C matrix

Chromosomes are nonrandomly positioned within the nuclear space [38], and their spatial organization has been probed using different technologies including Hi-C, FISH and TSA-seq [35, 21, 53]. Hi-C captures interaction frequency between pairs of loci, represented in the 2D matrix. Based on the fact that close loci should have the same interaction profile, different computational methods have been proposed to identify spatial compartmentalization states genome-wide [16, 56, 44, 54]. Ashoor et al [16] first embeds Hi-C graph into the vector space, and then infer compartments by applying k-means on inferred loci features. We use their idea of feature extraction from the Hi-C graph using graph embedding methods and show that our extracted features having information about the spatial properties of loci are good representations for the whole Hi-C matrix to be used in downstream domain annotation task. We use the graph embedding method, LINE, which can learn embeddings that capture the similarity between nodes' second-order proximity (described in detail in the Method section). We expect that loci having similar interaction profiles to have similar embeddings as well. To evaluate the goodness of embeddings, we reconstruct the correlation matrix from learned embeddings and compare it to the correlation matrix of the original Hi-C matrix. The correlation matrix of original Hi-C data is calculated as a pairwise correlation between rows of observed/expected Hi-C matrix. Comparison of reconstructed and original correlation matrices has been shown in figures 5.1a, 5.1b for different values of LINE two hyperparameters, embedding size (d) and sampling size (s). As we see, an embedding size of 10 is enough to capture most of the information in the whole Hi-C matrix, and embeddings converge after 100 millions of sampling iterations. We use these extracted embeddings of size 10 as spatial features of loci in our downstream domain annotation task to infer domain labels that capture positional properties as well as functional properties.

We also evaluated the quality of the embeddings based on their performance in predicting Hi-C subcompartments from Rao et al. We used random forest classifier in scikit-learn

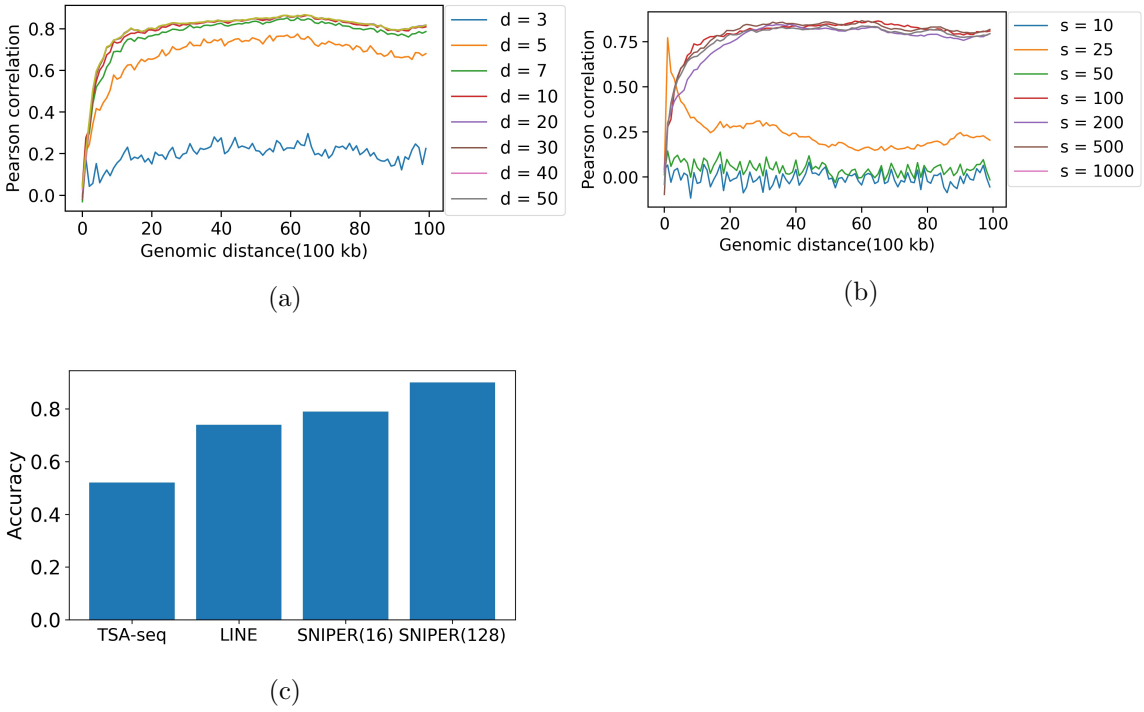


Figure 5.1: (a,b) Comparison of chr16 intrachromosomal correlation matrices obtained using the observed/expected Hi-C matrix and learned embeddings (100-kb resolution) for different LINE hyperparameters (d: embedding size, s: sample size/1M). To compare two matrices, the correlation of off-diagonal regions of two matrices is computed and plotted as a function of distance from the main diagonal. (c) Hi-C subcompartments prediction accuracy based on different features. Accuracy is a percentage of correct predictions.

[40] to predict the subcompartment labels from different inputs including SNIPER embeddings of size 16 and 128, LINE embeddings of size 10, and TSA-seq signals of size 2 (associated with Lamina and SON). We used 75% of data for the training set and the remaining as a test set. The accuracy of a model was measured by the percentage of correct predictions(5.1c). We can see that available 1D genomic signals (TSA-seq) associated with genome spatial positioning are not as informative as Hi-C embeddings, so it is reasonable to use Hi-C embeddings instead of just available 1D spatial genomic signals in our downstream analysis. SNIPER embeddings of size 128 show a good performance, however concatenating 128 spatial features and 12 functional features in our downstream analysis results in domain annotations that are highly correlated with spatial properties, therefore we need embeddings of smaller size. Line embeddings of size 10 and SNIPER embeddings of size 16 show a comparable result. We use LINE embeddings in our downstream analysis, since the embeddings of all chromosomes are in the same space, while SNIPER embeddings of odd and even chromosomes are in different spaces.

5.3 Domain types based on the aggregation of functional and structural genomic features stratify both patterns of regulatory activity and genome compartmentalization

Previous studies have shown the correlation between genome regulatory activities and compartmentalization [44, 34, 55]. Since we used both functional assays and Hi-C data, we also expect our domain states to stratify patterns of regulatory activities including histone modifications enrichment and gene expression profile as well as positional properties including replication timing profile and relative distance to other nuclear compartments such as Lamin and SON measured by TSA-seq assay. We tried a different number of states from 5 to 10 and observed that 7 domain states can represent the transcriptional activity and compartmentalization patterns. We assigned a name to the states based on their overlap with Rao et al subcompartments and transcription level. Different genomic properties of domain states have been summarized in table **??**.

Figure 5.2a shows enrichment of functional genomic signals, 6 phase replication timing profile, and Hi-C subcompartments from Rao et al for different domain states. Our domain states have a strong correlation with both transcriptional activities and previously defined subcompartments. Although there is a correlation between genomic bins positioning and transcriptional activity states [54], they do not have a monotonic relationship, as there are transcribed regions in the B compartment (inactive regions) and differently transcribed regions in the A compartment (active regions). We observed that using both spatial and functional features helps to find domain states that represent a combination of such properties. For example, A1a and A1b are both corresponded with the A1 subcompartment, which is close to the nuclear speckle, however showing different transcriptional activities, as A1a is more enriched with histone marks that are associated with transcriptional activation, and includes genes that are significantly more expressed comparing to A1b ($p - value < 4.5E - 124$). A2a, A2b and B1 also show roughly similar localization (figure 5.2d), however having different transcription profile. A2a is enriched with H3K27ac and highly transcribed genes and is associated with the A2 subcompartment. A2b is also mostly overlapped with the A2 subcompartment, however, showing significantly less transcriptional activities. B1 is depleted of all active histone marks and enriched with H3K27me3, a histone modification associated with facultative heterochromatin, which represses cell-type-specific regions [34]. B2' and B3' are associated with B2 and B3 subcompartments, which are permanently repressed regions near Lamin, but they show different replication timing profiles and transcription levels, and B2' is more enriched with repressive H3K27me3 mark, while B3' is more enriched with H3K9me3.

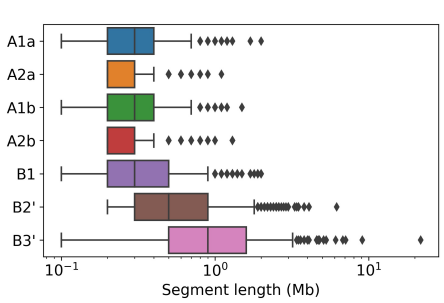
We showed that the integration of spatial features extracted from Hi-C data and functional features result in domain states that represent a combination of different positional and transcription states. To further evaluate the importance of each type of feature for

learned domain types, we compared the proportion of variance explained for different genomic properties using only spatial features, only functional features, and their combination. The result is shown in 5.3c. 1D functional genomic signals are highly correlated with each other, and passing them as an input to the Segway results in domain states that explain a large proportion of variance in most of the functional genomic signals, except H3K9me3 which is moderately enriched in all of the domain states. However, functional domain states are not good predictors for signals such as TSA-seq and replication timing, which are dependent on genomic positioning. On the other hand, feeding extracted spatial features into the Segway result in domain states that are a good representation for replication timing and TSA-seq signal, however, they explain a small proportion of variance of functional genomic signals and gene expression data. The domain states coming from our aggregated approach can capture variance of positional-dependent genomic properties, while also preserve variance of functional features.

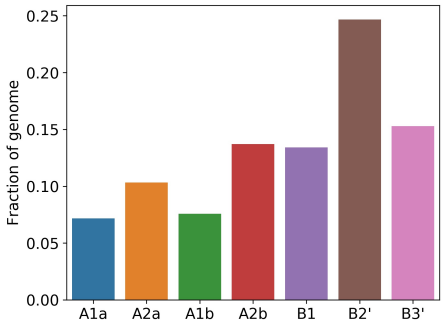
5.4 Domain types based on the aggregation of spatial and functional features improve gene expression and replication timing prediction

DNA replication is essential for the transmission of genetic information and is intimately tied to chromosome structure and function [45], such that active domains in the nuclear interior replicate early, whereas more condensed and inactive chromatin domains near the nuclear periphery replicate later [36]. We used 6 phase Repli-seq data for GM12878 to evaluate the performance of our domain states to explain it. Processed Repli-Seq data represents the percentage of DNA replicated during each of the six cell cycle phases (G1, S1, S2, S3, S4, G2) for each genomic bin. Our domain states show distinct replication timing (RT) profile as shown in figure 5.2a. We found that RT profile neither changes exactly according to genomic positioning nor epigenomic profile. For example, A1b is replicated earlier than A2a because of its positioning state that is closer to the nuclear interior, although it is less enriched with active epigenomic marks. On the other hand, B3' is replicated later than B2', since it has a bit less transcriptional activity comparing to B2', while they have similar Lamin and SON TSA-Seq enrichment. To further evaluate the contribution of each type of feature, we ran Segway on just functional genomics assays (Segway_fa), just on extracted structural features from LINE method (Segway_LINE) and their combination (Segway_LINE_fa), and calculated variance explained for each of 6 phases signals according to domain states inferred from each of input data variations (5.3a). We found that for G1 and G2 phases signals, in which both functional and positional features are good predictors, their combination improves the proportion of variance explained. However, in S phases, functional features are not as good as positional features in explaining signals variance, and domain states from their combination are affected by functional features and show

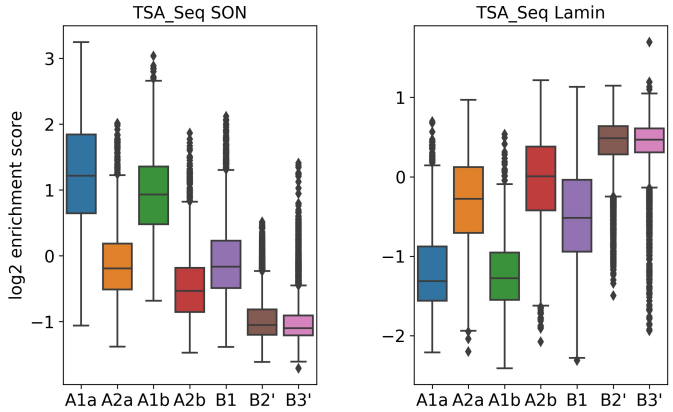
(a)



(b)



(c)



(d)

Figure 5.2: (a) Enrichment of different 1D functional signals, replication timing phases, and Rao et al subcompartments for each domain state. (b) Segment length distribution for each domain type. (c) Fraction of genome covered by each domain type. (d) Boxplot of SON and Lamin TSA-seq log2 enrichment for each domain type

Data type	Figure	A1a	A2a	A1b	A2b	B1	B2'	B3'
Median segment length	5.2b	0.3	0.2	0.3	0.2	0.3	0.5	0.9
Average Lamin TSA-seq enrichment	5.2d	−1.2	−0.3	−1.2	−0.1	−0.5	0.4	0.4
Genes / MB		24.6	11.2	14.3	6.8	5.7	2.3	1.2
Average gene expression (RPKM)		27.6	25.7	10.4	12.1	2.1	2.7	0.9
Average gene expression z-score		0.16	0.15	−0.04	−0.04	−0.21	−0.21	−0.26
Hi-C subcompartments	5.2a	A1	A2	A1	A2-B	B1	B2-B3	B2-B3

Table 5.2: Summary of learned domain types. A1a and A1b are both almost overlapped with the A1 subcompartment and located in the nuclear interior, but they show distinct patterns of epigenetic marks, as A1a is associated with a higher level of active histone marks and more transcription level, whereas A1b is less transcriptionally active. A2a and A2b are almost overlapped with the A2 subcompartment, but A2a shows a high level of transcription, while A2b is less transcriptionally active. The B1 state is very similar to the B1 subcompartment in terms of both positioning and epigenetic pattern, so we did not change the name. B2' and B3' states are both overlapped with constitutive heterochromatin (B2 and B3 subcompartments), however, they show different epigenetic patterns, so we named them in this way.

a little bit worse performance comparing to just using positional features as an input to annotation task. We got an average of variance explained over different replication timing phases (RT_mean signal in figure 5.3c), and our domain states (from segway_LINE_fa) shows an improvement compared with other variations of input.

Our domain states also stratify transcription activity patterns as well as domain states that just rely on functional genomic signals. For example, we showed variance explained for the signals DNase, H3K27ac, H3K9me3, H3K27me3, H3K4me3, and H3K79me2 in figure 5.3c, and we can see that inputting both spatial and functional features improves the variance explained of the functional signals.

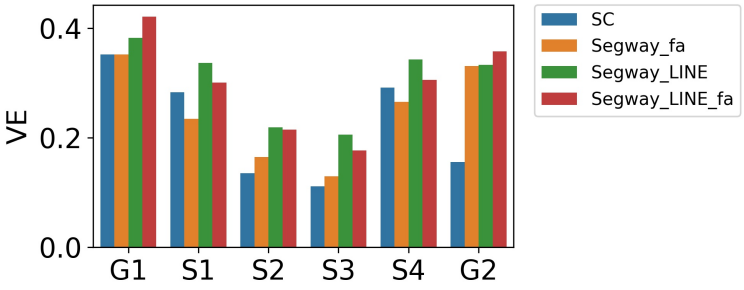
Furthermore, we compared our domain states with GRO-seq data that measures the genes' transcription level in the cell. The information is summarized in a table **??**. We found that two active subcompartments, A1 and A2, from Rao et al [44] can be further separated as moderately (A1b, A2b) and increasingly (A1a, A2a) transcribed domains. We also calculated variance explained of the genes transcription level according to the domain types they belong to (figure 5.3c), and we observed an improvement in the proportion of variance explained based on domain types from aggregating spatial and functional features.

The gene expression z-score represents the distance between the transcription level of a gene in GM12878 cell type and the population (57 cell types gene expression data [13]) transcription level mean in units of the standard deviation. It shows that our more refined compartmentalization can distinguish cell-type specific and broadly expressed genes, as A1a and A2a include genes that are significantly expressed in GM12878, while A1b and A2b are enriched with broadly expressed genes (genes that are transcribed in all cell types). We also observed that gene density is more correlated with genomic positioning than transcription activity, as A1a and A1b have higher gene density compared to A2a and A2b, and all inactive subcompartments are more depleted of genes.

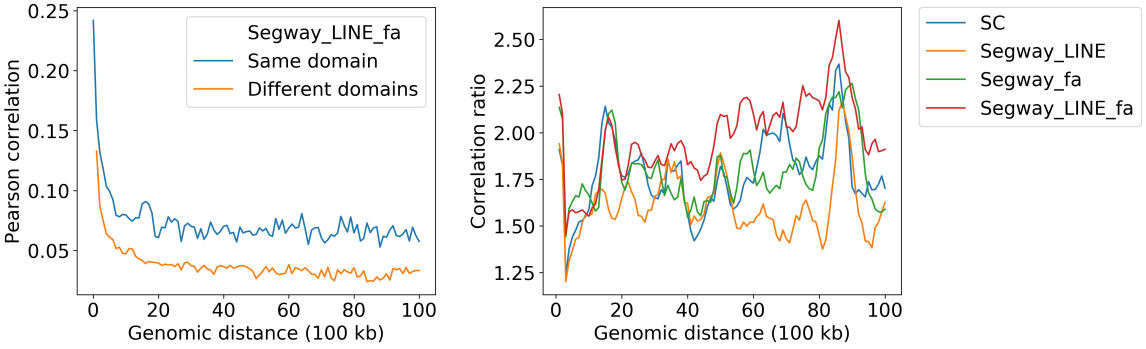
5.5 Genes in the same domain types are coexpressed

The 3D organization of the genome plays a critical role in gene regulation. Enhancers are key regulatory elements that control cell-type-specific gene expression profile by contacting with their target genes. Enhancers can target genes at great genomic distances from them as a result of spatial co-localization. Thus, we should consider both linear genome and spatial localization to predict effective enhancer-promoter pairs [46].

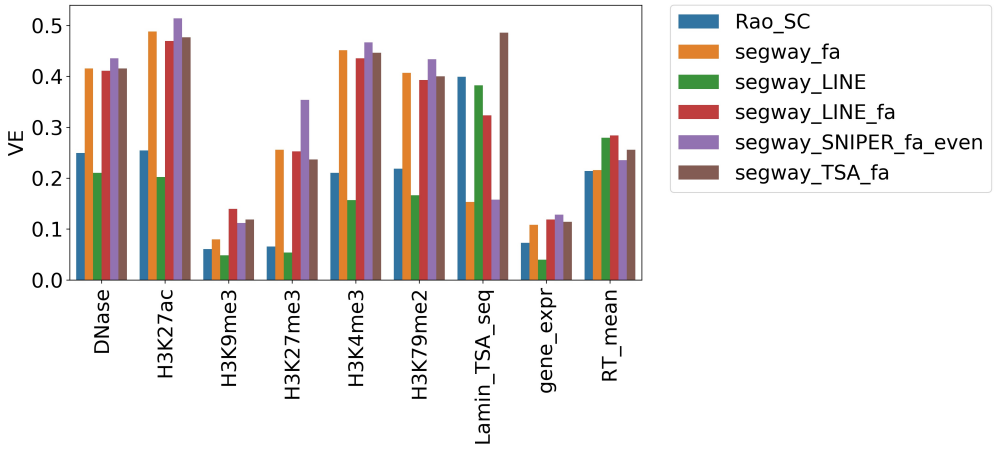
Enhancer activity as both cis-regulatory and trans-regulatory element suggests that domain inference based on the combination of functional and spatial features result in domain types that capture co-regulation of genes at great genomic distances. To test this hypothesis, we calculate pairwise Pearson correlation of gene pairs, and compare the average correlation of gene pairs within the same domain and across different domains for each genomic distance (0-10MB, step = 100KB) (figure 5.3b). As expected, gene pairs within the same



(a)



(b)



(c)

Figure 5.3: (a) Variance explained for different replication timing phases according to domain states from Rao et al subcompartments (SC), running Segway on 1D functional genomic features (Segway_fa), learned structural features from LINE (Segway_LINE) and their combination (Segway_LINE_fa). (b) Left. Average pairwise gene-expression correlations vs their genomic distance for gene pairs having the same or different domain types. (b) Right. The ratio of gene-expression correlations of gene pairs with same domain type to gene pairs with different domain types plotted for domain states from SC and running Segway on different feature types. (c) Variance explained for different genomic signals and gene expression according to domain states from running Segway on different feature types.

domain show more co-regulation comparing to gene pairs from different domains. To further evaluate the impact of each of the functional and spatial features, we compared the ratio of the average correlation of gene pairs from the same domain to different domains based on domain types coming from different experiments (described in experiment section). We observed that the domains based on just epigenomic profile can capture local co-regulated genomic regions, as Segway_fa domains show a high correlation ratio for genomic distances less than 1 Mb. However, as genomic distance increases, incorporating spatial features allow finding domain types, which are co-regulated even in large genomic distance from each other. This finding suggests that our domains can be used as candidate regions to find long-range enhancer-promoter contacts.

Chapter 6

Discussion and future work

Packaging of DNA in a cell nucleus is not randomly but in a cell-type-specific manner and controls different cell activities such as transcriptional activity. Analysis of genome-wide chromosome conformation capture assay (Hi-C) data has revealed that chromosomes loci can be clustered into different subcompartments, and loci belonging to the same subcompartment are colocalized [56, 44]. Such clusters based on chromatin contacts also have different epigenomic profiles. On the other hand, different methods have been developed to cluster a genome into different domain types according to their epigenetic marks [33, 25]. Domain types inferred from different data types overlap each other, for example, separation of a genome into active and inactive regions has been seen through the analysis of both Hi-C and epigenetic signals data, as subcompartments located in the nucleus interior are more enriched with active epigenetic marks like H3K27ac and subcompartments close to nucleus periphery are depleted from active marks and enriched with inactive epigenetic marks like H3K9me3. These observations suggest that chromosomes are positioned in the nucleus in a way that controls the clustering of epigenetic marks and coregulated genes. Therefore, we propose a framework that uses genome spatial organization together with epigenetic modifications to identify more accurate and detailed chromatin domains and understand the relationship between domain types inferred from Hi-C and 1D epigenetic signals.

We identified 7 domain types that represent both spatial positioning of a genome and its functional features. 4 domain types are associated with active chromatin, and 3 of them include inactive regions. We observed that 2 active domain types are almost overlapped with known $A1$ subcompartment from Hi-C data, while the other 2 active domain types are associated with $A2$ subcompartment. Previous works [44, 54] studied differences between these two active subcompartments. Rao et al [44] showed that although both $A1$ and $A2$ have highly transcribed genes and are enriched with active epigenetic marks, they have different replication timing patterns, and $A2$ is more enriched with H3K9me3 than $A1$ and has lower GC content. SPIN [54] also studied their difference in terms of relative positioning to other nuclear components, and showed that $A1$ is closer to nuclear speckles compared to $A2$ subcompartment. We found that each of these subcompartments can be further subdivided

into two different domain types with distinct epigenetic modifications and replication timing patterns. For example, $A1$ can be separated into $A1a$ and $A1b$ domain types and $A1a$ is significantly more enriched with active histone marks including H3K79me2, H3K27ac, and H3K4me2 compared to $A1b$, and it also is replicated earlier than $A1b$ (the same separation happens for $A2$ for $A2a$ and $A2b$). We also identified known facultative heterochromatin [34] which is almost overlapped with $B1$ subcompartment and is enriched with H3K27me3. We also found two domain types corresponding with constitutive heterochromatin that have distinct replication timing patterns but have similar positioning in the nucleus since they have the same average relative distance to both nuclear lamina and speckles.

Domain organization of a genome controls gene transcription and other cellular programs like replication timing. We showed that replication timing is more associated with genomic positioning in the nucleus, as its variance is better explained by spatial properties, and variance of gene transcription is better explained by functional properties as expected since they directly capture enrichment of transcriptional factors essential for gene expression. Therefore, coupling spatial and functional features helps to infer domain types that can capture both cellular activities better than just using one feature type.

Spatial organization of a genome results in colocalization of distant regions of the genome that could be functionally significant or not, meaning that the colocalization has an effect on coregulation of the genes in those regions or is random. We showed that our domain types can capture the higher co-expression of gene pairs falling in the same domain type compared to the genes in different domain types since both genome positioning and epigenetic modifications have an impact on genes coregulation. Therefore, our domain types can be used as candidate regions to find long-range contacts in the chromatin.

Previous integrative methods [54, 34] relied on the assumption that spatially close genomic regions are more probable to be assigned the same label. These methods are sensitive to the procedure we use to determine spatially close genomic regions, as we are going to push non colocalized regions to get the same label if we incorporate random contacts. On the other hand, methods that infer genome subcompartments are based on clustering genomic regions' interaction patterns, which is equivalent to the second-order proximity of the nodes (genomic regions) in a Hi-C graph. The novelty of our framework lies in leveraging spatial properties of genomic regions according to their interaction pattern with the rest of the genome instead of their direct interactions that results in identifying domain types capable of capturing both genomic positioning and functionality.

To incorporate chromatin conformation information, we used node embedding methods to map nodes of a Hi-C graph into vector space. As expected, we observed that our learned embeddings are highly correlated with available 1D spatial genomic signals like Lamina and SON TSA-seq signals, however, they encode more information than such 1D signals. As future work, we aim to find the important learned features responsible for the accuracy of downstream tasks like predicting known Hi-C subcompartments and evaluate their ef-

fectiveness in encoding node properties such as degree, closeness centrality, and clustering coefficient to find what kind of information the embeddings encode[22].

Overall, our findings suggest that the spatial organization of a genome and its epigenetic profile provide complementary information to understand a domain scale regulation of cellular behaviors. Although we just used 12 1D epigenomic signals together with Hi-C data to annotate domain types, different variations of 1D signals can be used as an input to our framework according to the type of domains we are looking for. Also, our framework can be applied to different cell types, or a cell type in different time points to understand mechanisms behind cell differentiation.

Chapter 7

Extended results

7.1 Comparison of different graph embedding methods

In general, graph embedding methods consist of an encoder to map each node of a graph to a low-dimensional vector, a decoder to reconstruct a user-specified graph statistic from node embeddings, and a loss function to quantify the difference between the decoder output and target graph statistic. A graph statistic is determined based on the information that the node embeddings have to represent. For example, to capture local pairwise proximity, edge weights between node pairs are used to reconstruct from embeddings, while to capture more global structures in a graph, nodes' higher-order proximity (for example neighborhood vector as second-order proximity) is used to reconstruct.

We tried 4 different node embedding methods including LINE (optimized through first-order proximity), LINE (optimized through second-order proximity) [50, 16], autoencoder [55] and node2vec [28]. We observed that optimizing LINE based on first-order proximity results in embeddings that are close to zero vector and cannot capture the genome positional-related properties such as TSA-seq signals like embeddings based on second-order proximity. We guess that the problem arises due to including all Hi-C contacts in a graph that might be random. Since the Hi-C graph is dense, encouraging directly connected nodes to get similar embeddings pushes the embedding of the nodes to converge to each other. Therefore, filtering Hi-C edges and just including significant interactions in the Hi-C graph might help to infer meaningful node embeddings based on first-order proximity. However, in this study, we just focused on optimization based on second-order proximity.

Other node embedding methods can capture the global network structure by defining stochastic graph statistics to reconstruct [28, 41]. In general, they perform multiple random walks starting from each node, and then the node pairs falling in the same random walks are more close to each other in the embedding space. According to the length of random walks and the way we choose the next node in a random walk, we are going to preserve local or more global structures of a graph. Node2vec [28] introduces a biased random walk procedure to add flexibility in exploring neighborhoods and learn embeddings that preserve more

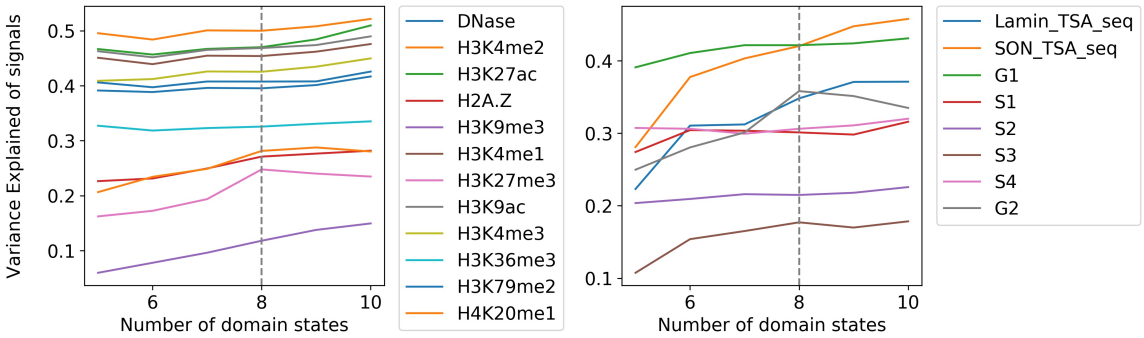


Figure 7.1: Proportion of variance explained for 12 epigenomic signals (left figure) and 2 TSA-seq signals and 6 phase Repli-seq signals (right figure) for different number of labels (domain states)

global structures in a graph. In general, random walks choose the next node in the walk according to the weights of edges between the current node and its neighbors. Node2vec controls exploring nodes further away from the source node or closely around the source node by remembering the previous node in a walk as well as the current node. For example, to explore the nodes further away from the source node, node2vec chooses the shared neighbors between current and previous nodes in a walk with less probability compared to other neighbors of a current node. Since the next step is dependent on both current and previous nodes, the transition matrix space complexity has a growth rate of $O(a^2|V|)$ where a is the average degree of the graph. Due to its large space complexity, we found it impossible to run the node2vec on a genome-wide matrix. Also, there is another random-walk based node embedding method with naive random walk procedure, DeepWalk [41], that was compared to the LINE embedding in Ashoor et al. paper [16]. Since they showed that DeepWalk embeddings preserve the node properties in the Hi-C graph like different centralities less than LINE embeddings, we did not use any of random-walk based embeddings in our framework.

7.2 Choosing the number of domain types

To estimate the number of domain types, we calculated the variance explained for different signals including 12 functional genomic, 2 TSA-seq, and 6 phases Repli-seq signals as a function of the number of domain types. We observed that the proportion of variance explained for most of the signals is not going to improve after about 8 clusters (figure 7.1). We also found that additional states (number of states = 9 or 10) result in states with similar patterns, therefore we choose 8 as the number of domain types, however, one of the states covered just a few separated regions, and we did not find any interpretation for that state, so we did not show it in our results.

Bibliography

[1] *What are chromosomes and chromosome territories?*, 2018 (accessed October 25, 2020). https://www.mechanobio.info/genome-regulation/what-are-chromosomes-and-chromosome-territories/.

[2] *What is chromatin, heterochromatin and euchromatin?*, 2018 (accessed October 25, 2020). https://www.mechanobio.info/genome-regulation/what-is-chromatin-heterochromatin-and-euchromatin/.

[3] *histone modification*, accessed November 12, 2020. https://www.abcam.com/epigenetics/histone-modifications.

[4] *ATAC-seq*, accessed November 13, 2020. https://en.wikipedia.org/wiki/ATAC-seq.

[5] *ATAC-seq*, accessed November 13, 2020. https://www.activemotif.com/blog-atac-seq.

[6] *DNase hypersensative site*, accessed November 13, 2020. https://en.wikipedia.org/wiki/DNase_I_hypersensitive_site.

[7] *tagmentation*, accessed November 13, 2020. https://bitesizebio.com/13567/too-good-to-be-true-what-can-nextera-do-for-you/.

[8] *transposase*, accessed November 13, 2020. https://en.wikipedia.org/wiki/Transposase.

[9] *transposon*, accessed November 13, 2020. https://en.wikipedia.org/wiki/Transposable_element.

[10] *Lamina*, accessed November 14, 2020. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6406483/.

[11] *Nucleus compartments*, accessed November 14, 2020. https://www.kenhub.com/en/library/anatomy/cell-nucleus.

[12] *Replication timing*, accessed November 14, 2020. https://en.wikipedia.org/wiki/Replication_timing.

[13] *57 cell types gene expression data*, (accessed November 22, 2020). https://egg2.wustl.edu/roadmap/data/byDataType/rna/expression/.

[14] Ziad Al Bkhetan and Dariusz Plewczynski. Three-dimensional epigenome statistical model: genome-wide chromatin looping prediction. *Scientific reports*, 8(1):1–11, 2018.

[15] Heiner Albiez, Marion Cremer, Cinzia Tiberi, Lorella Vecchio, Lothar Schermelleh, Sandra Dittrich, Katrin Küpper, Boris Joffe, Tobias Thormeyer, Johann von Hase, et al. Chromatin domains and the interchromatin compartment form structurally defined and functionally interacting nuclear networks. *Chromosome research*, 14(7):707–733, 2006.

[16] Haitham Ashoor, Xiaowen Chen, Wojciech Rosikiewicz, Jiahui Wang, Albert Cheng, Ping Wang, Yijun Ruan, and Sheng Li. Graph embedding and unsupervised learning predict genomic sub-compartments from hic chromatin interaction data. *Nature communications*, 11(1):1–11, 2020.

[17] Faezeh Bayat and Maxwell Libbrecht. Variance-stabilized units for sequencing-based genomic signals. *bioRxiv*, 2020.

[18] Wendy A Bickmore and Bas van Steensel. Genome architecture: domain organization of interphase chromosomes. *Cell*, 152(6):1270–1284, 2013.

[19] Miguel R Branco and Ana Pombo. Intermingling of chromosome territories in interphase suggests role in translocations and transcription-dependent associations. *PLoS Biol*, 4(5):e138, 2006.

[20] Lyubomira Chakalova, Emmanuel Debrand, Jennifer A Mitchell, Cameron S Osborne, and Peter Fraser. Replication and transcription: shaping the landscape of the genome. *Nature Reviews Genetics*, 6(9):669–677, 2005.

[21] Yu Chen, Yang Zhang, Yuchuan Wang, Liguo Zhang, Eva K Brinkman, Stephen A Adam, Robert Goldman, Bas Van Steensel, Jian Ma, and Andrew S Belmont. Mapping 3d genome organization relative to nuclear compartments using tsa-seq as a cytological ruler. *Journal of Cell Biology*, 217(11):4025–4048, 2018.

[22] Ayushi Dalmia and Manish Gupta. Towards interpretation of node embeddings. In *Companion Proceedings of the The Web Conference 2018*, pages 945–952, 2018.

[23] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[24] Neva C Durand, Muhammad S Shamim, Ido Machol, Suhas SP Rao, Miriam H Huntley, Eric S Lander, and Erez Lieberman Aiden. Juicer provides a one-click system for analyzing loop-resolution hi-c experiments. *Cell systems*, 3(1):95–98, 2016.

[25] Jason Ernst and Manolis Kellis. Chromhmm: automating chromatin-state discovery and characterization. *Nature methods*, 9(3):215–216, 2012.

[26] Guillaume J Filion, Joke G van Bemmel, Ulrich Braunschweig, Wendy Talhout, Jop Kind, Lucas D Ward, Wim Brugman, Inês J de Castro, Ron M Kerkhoven, Harmen J Bussemaker, et al. Systematic protein location mapping reveals five principal chromatin types in drosophila cells. *Cell*, 143(2):212–224, 2010.

[27] Jean-Philippe Fortin and Kasper D Hansen. Reconstructing a/b compartments as revealed by hi-c using long-range correlations in epigenetic data. *Genome biology*, 16(1):180, 2015.

[28] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

[29] Lars Guelen, Ludo Pagie, Emilie Brasset, Wouter Meuleman, Marius B Faza, Wendy Talhout, Bert H Eussen, Annelies de Klein, Lodewyk Wessels, Wouter de Laat, et al. Domain organization of human chromosomes revealed by mapping of nuclear lamina interactions. *Nature*, 453(7197):948–951, 2008.

[30] William L Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*, 2017.

[31] R Scott Hansen, Sean Thomas, Richard Sandstrom, Theresa K Canfield, Robert E Thurman, Molly Weaver, Michael O Dorschner, Stanley M Gartler, and John A Stamatoyannopoulos. Sequencing newly replicated dna reveals widespread plasticity in human replication timing. *Proceedings of the National Academy of Sciences*, 107(1):139–144, 2010.

[32] Erica M Hildebrand and Job Dekker. Mechanisms and functions of chromosome compartmentalization. *Trends in Biochemical Sciences*, 2020.

[33] Michael M Hoffman, Orion J Buske, Jie Wang, Zhiping Weng, Jeff A Bilmes, and William Stafford Noble. Unsupervised pattern discovery in human chromatin structure through genomic segmentation. *Nature methods*, 9(5):473, 2012.

[34] Maxwell W Libbrecht, Ferhat Ay, Michael M Hoffman, David M Gilbert, Jeffrey A Bilmes, and William Stafford Noble. Joint annotation of chromatin state and chromatin conformation reveals relationships among domain types and identifies domains of cell-type-specific expression. *Genome research*, 25(4):544–557, 2015.

[35] Erez Lieberman-Aiden, Nynke L Van Berkum, Louise Williams, Maxim Imakaev, Tobias Ragoczy, Agnes Telling, Ido Amit, Bryan R Lajoie, Peter J Sabo, Michael O Dorschner, et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *science*, 326(5950):289–293, 2009.

[36] Claire Marchal, Jiao Sima, and David M Gilbert. Control of dna replication timing in the 3d genome. *Nature Reviews Molecular Cell Biology*, pages 1–17, 2019.

[37] Marcel Méchali. Eukaryotic dna replication origins: many choices for appropriate answers. *Nature reviews Molecular cell biology*, 11(10):728–738, 2010.

[38] Tom Misteli. Spatial positioning: A new dimension in genome function. *Cell*, 119(2):153–156, 2004.

[39] Kevin Murphy, Yair Weiss, and Michael I Jordan. Loopy belief propagation for approximate inference: An empirical study. *arXiv preprint arXiv:1301.6725*, 2013.

[40] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[41] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.

[42] Yifeng Qi and Bin Zhang. Predicting three-dimensional genome organization with chromatin states. *PLoS computational biology*, 15(6):e1007024, 2019.

[43] Jiezhong Qiu, Yuxiao Dong, Hao Ma, Jian Li, Kuansan Wang, and Jie Tang. Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 459–467, 2018.

[44] Suhas SP Rao, Miriam H Huntley, Neva C Durand, Elena K Stamenova, Ivan D Bochkov, James T Robinson, Adrian L Sanborn, Ido Machol, Arina D Omer, Eric S Lander, et al. A 3d map of the human genome at kilobase resolution reveals principles of chromatin looping. *Cell*, 159(7):1665–1680, 2014.

[45] Juan Carlos Rivera-Mulia and David M Gilbert. Replication timing and transcriptional control: beyond cause and effect—part iii. *Current opinion in cell biology*, 40:168–178, 2016.

[46] Stefan Schoenfelder and Peter Fraser. Long-range enhancer–promoter contacts in gene expression control. *Nature Reviews Genetics*, page 1, 2019.

[47] KBLA Sha and Laurie A Boyer. The chromatin signature of pluripotent cells. *Stem Book*, 2009.

[48] Quentin Szabo, Frédéric Bantignies, and Giacomo Cavalli. Principles of genome folding into topologically associating domains. *Science advances*, 5(4):eaaw1668, 2019.

[49] Takumi Takizawa, Karen J Meaburn, and Tom Misteli. The meaning of gene positioning. *Cell*, 135(1):9–13, 2008.

[50] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077, 2015.

[51] Nynke L Van Berkum, Erez Lieberman-Aiden, Louise Williams, Maxim Imakaev, Andreas Gnirke, Leonid A Mirny, Job Dekker, and Eric S Lander. Hi-c: a method to study the three-dimensional architecture of genomes. *JoVE (Journal of Visualized Experiments)*, (39):e1869, 2010.

[52] Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269, 1967.

[53] Emanuela V Volpi and Joanna M Bridger. Fish glossary: an overview of the fluorescence in situ hybridization technique. *Biotechniques*, 45(4):385–409, 2008.

[54] Yuchuan Wang, Yang Zhang, Ruochi Zhang, Tom van Schaik, Liguo Zhang, Takayo Sasaki, Daniel Peric-Hupkes, Yu Chen, David M Gilbert, Bas van Steensel, et al. Spin reveals genome-wide landscape of nuclear compartmentalization. *bioRxiv*, 2020.

[55] Kyle Xiong and Jian Ma. Revealing hi-c subcompartments by imputing inter-chromosomal chromatin interactions. *Nature communications*, 10, 2019.

[56] Eitan Yaffe and Amos Tanay. Probabilistic modeling of hi-c contact maps eliminates systematic biases to characterize global chromosomal architecture. *Nature genetics*, 43(11):1059, 2011.

[57] Hui Zheng and Wei Xie. The role of 3d genome organization in development and cell differentiation. *Nature Reviews Molecular Cell Biology*, page 1, 2019.

[58] Yun Zhu, Zhao Chen, Kai Zhang, Mengchi Wang, David Medovoy, John W Whitaker, Bo Ding, Nan Li, Lina Zheng, and Wei Wang. Constructing 3d interaction maps from 1d epigenomes. *Nature communications*, 7(1):1–11, 2016.