Master Thesis

Sequence-Based Predictions of Chromatin Compartments

Julie Prost

School of Computer Science
McGill University
Montreal, Quebec, Canada

Supervisor
Dr. Mathieu Blanchette

May 2019

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

Abstract

Spatial genomic organization is known to be critical for proper gene regulation. It is based on a hierarchical model where chromosomes are divided into megabase-sized cell-type specific A and B compartments, associated with open and closed chromatin. In this thesis, we present a computational pipeline for sequence-based annotations of chromatin compartments. The first step of the pipeline consists in selecting relevant sequence features using a Random Forest algorithm. Then, compartments annotations are produced by a stacked artificial neural networks model. Our approach is validated on different cell types and species, with a cross-validation AUC score ranging between 82% and 90%. We observe conserved compartment establishment rules between mouse and human and study compartment evolution across mouse neural differentiation. Finally, in an effort to gain insights into the underlying biological processes leading to compartments establishment, we interpret our model features and identify key sequence determinants related to the determination of chromatin A and B compartments.

Abrégé

L'organisation spatiale du génome est connue pour être un élément critique de la régulation des gènes. Elle repose sur un modèle hierarchique où, à l'échelle de la mégabase, les chromosomes sont divisés en deux types de compartiments, A et B. Les compartiments sont spécifiques à un type cellulaire et associés à l'euchromatine et l'hétérochromatine respectivement. Dans cette thèse, nous présentons un pipeline de calcul pour l'annotation des compartiments A et B. La première étape consiste en la sélection de variables pertinentes pour la constitution des compartiments par un algorithme Random Forest. Ensuite, une combinaison de réseaux de neurones est utilisée pour annoter les compartiments. Notre approche est validée sur différents types cellulaires ainsi que pour différentes espèces. Elle atteint des scores AUC compris entre 82% et 90% sur l'ensemble des jeux de données testés. Nous observons une conservation des règles de formation des compartiments entre la souris et l'homme et nous étudions également la différentiation neuronale chez la souris. Finalement, dans le but de mieux comprendre les processus biologiques menant à la formation des compartiments, nous interprétons les variables de notre modèle et identifions des déterminants séquentiels clefs pour la formation des compartiments A et B.

Acknowledgements

I would like to express my deepest appreciation to my supervisor, Dr. Mathieu Blanchette, for his support and guidance through my entire project. I would also like to thank Christopher Cameron for his advice as well as for the data processing he helped me with.

My thanks also go to the examiner of my Master thesis.

Finally, I would like to thank all the individuals who helped me during insightful conversations: Faizy Ahsan, Alexander Butyaev, Samy Coulombe, Chris Drogaris, Elliot Layne, Vincent Mallet, Nikou Sefat, Juliette Valenchon, ZiChao Yan and Yanlin Zhang.

Contents

List of Figures

List of Tables

Acronyms

3C	**C**hromosome **C**onformation **C**apture
3D	**3 D**imensions
ANN	**A**rtificial **N**eural **N**etwork
AUC	**A**rea **U**nder the **C**urve
BART	**B**ayesian **A**dditive **R**egression **T**ree
bp	**b**ase **p**air
CN	**C**ortical **N**euron
CNN	**C**onvolutional **N**eural **N**etwork
DT	**D**ecision **T**ree
EI	**E**xpected **I**mprovment
ESC	**E**mbryonic **S**tem **C**ell
GLM	**G**eneralized **L**inear **M**odel
IF	**I**nteraction **F**requency
LSTM	**L**ong **S**hort **T**erm **M**emory
MDI	**M**ean **D**ecrease in **I**mpurity
NPC	**N**euron **P**rogenitor **C**ell
PCA	**P**rincipal **C**omponent **A**nalysis
RF	**R**andom **F**orest
RNN	**R**ecurrent **N**eural **Network**
ROC	**R**eceiving **O**perating **C**urve
SVM	**S**upport **V**ector **M**achine
TAD	**T**opologically **A**ssociated **D**omain
TE	**T**ransposable **E**lement
TF	**T**ranscription **F**actor
TFBS	**T**ranscription **F**actor **B**inding **S**ite
UCB	**U**pper **C**onfidence **B**ound

1

Introduction

Spatial genomic organization is known to be critical for proper gene regulation. It is based on a hierarchical model where chromosomes are divided into megabase-sized cell-type specific A and B compartments, associated with open and closed chromatin. In this thesis, we aim at predicting chromatin compartments from DNA sequence-features alone. In this introductory chapter, we will first discuss the stakes related to the three-dimensional (3D) organization of the genome, explain what chromatin A and B compartments are and why they are an important topic of study. Then, we will provide a necessary machine learning background, including detailed descriptions of the different algorithms used in this work. Finally, we will see how machine learning has been previously used to infer elements of DNA's spatial structure as well as how this type of methods can take advantage of DNA sequence information to make useful predictions, paving the way for a machine learning approach to predict chromatin A and B compartments from sequence-level features.

1.1 3D Genomics

1.1.1 Spatial Genome Organization

DNA folding and compaction in the nucleus of cells is critical at least for physical reasons. Indeed, the entire human DNA would be more than 2 meters long if entirely spread out and has to fit into a micron-sized cell nucleus, which is equivalent to fitting 20km of fine thread

into a tennis ball. In addition to this physical reason, it has been found that DNA folding is not random and has an impact on many cellular processes such as gene expression and regulation [Gorkin et al., 2014, Gibcus and Dekker, 2013, Bickmore, 2013], replication timing [Ryba et al., 2010] and nuclear organization [Yaffe and Tanay, 2011]. For instance, by putting in close spatial proximity DNA loci that are far apart in the linear sequence, regulatory elements can interact and impact gene expression [Christopher J.F. Cameron and Dostie, 2016].

The development of chromosome conformation capture (3C) technologies such as Hi-C (see Section 1.1.2) has then allowed increasing insights into genome folding [Lieberman-Aiden et al., 2009]. Mammalian genomes were found to be hierarchically organized in 3D (see Figure 1.1). At the highest level, chromosomes have specific locations in the nucleus and form structures called chromosome territories. As an example, human chromosomes 18 and 19 have been found to occupy different territories in the nucleus, a trend which is linked to their respective gene-density [A. Croft et al., 1999]. Each chromosome is then grouped into multi-megabase-sized A and B compartments. Domains belonging to one type of compartment preferentially interact with domains of the same compartment type [Lieberman-Aiden et al., 2009] and are cell type specific [Dixon et al., 2015, Bonev et al., 2017]. Their known characteristics will be described further in this thesis (see Section 1.1.3). Compartments are themselves partitioned into self-interacting genomic regions known as Topologically Associated Domains (TADs) of an average size varying between 0.5 and 1 megabase [Jesse R. Dixon and Ren, 2012]. Contrary to compartments, TADs are relatively stable across cell types. Moreover, they were found to be highly conserved across species, making them an inherent feature of mammalian genomes [Jesse R. Dixon and Ren, 2012]. This same study also highlights the enrichment of TAD boundaries for the binding sites of the CTCF DNA binding protein, as well as in short interspersed (SINE) transposable elements [Jesse R. Dixon and Ren, 2012]. As such, TADs are more and more thought as having a critical role in genome regulation [Gibcus and Dekker, 2013]. At a higher resolution, TADs contain chromatin loops, which are formed by the interaction of distal chromatin-binding proteins such as CTCF. These interactions can either restrict or encourage the spatial closeness of regulatory DNA elements [Sanyal et al., 2012].

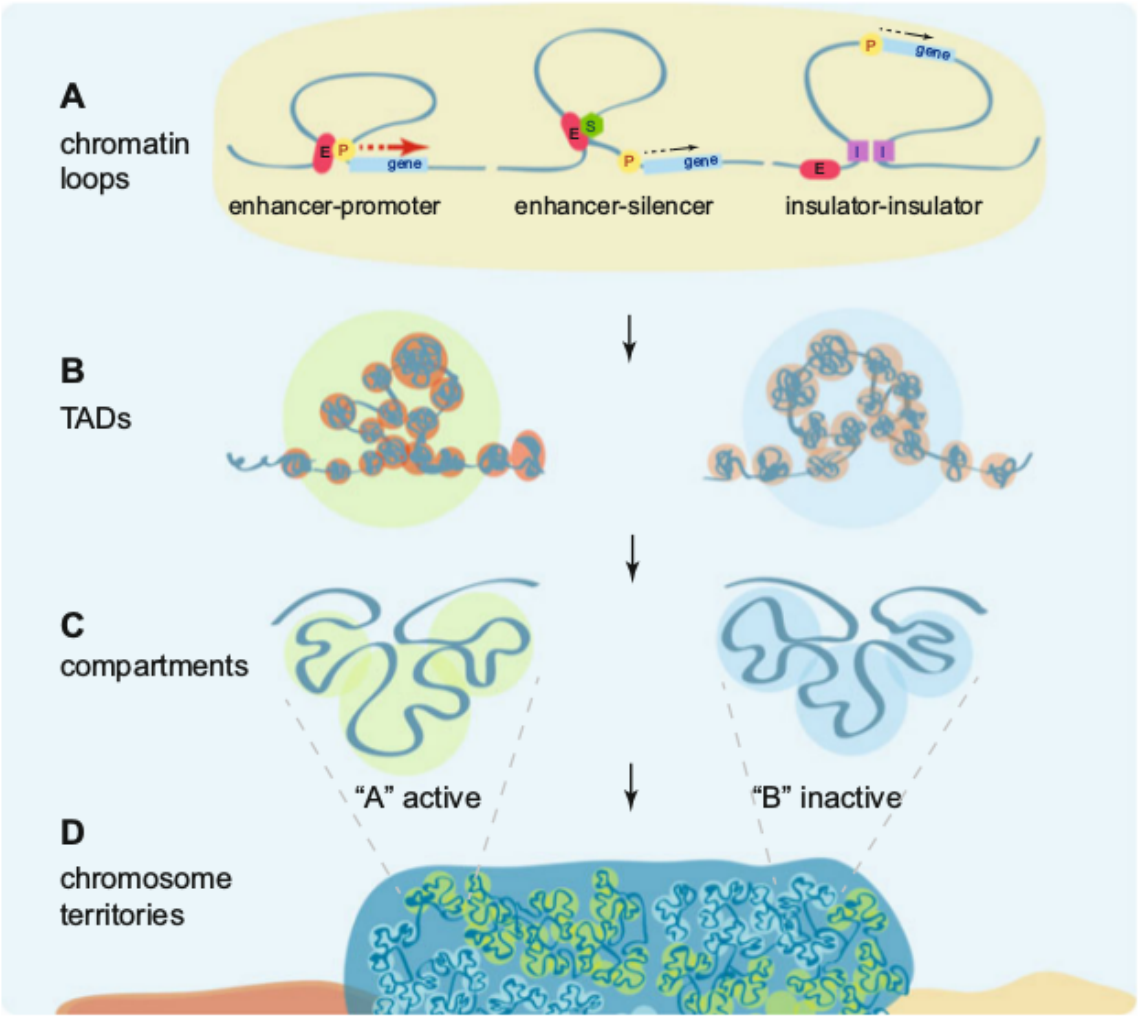

Figure 1.1: Genome 3D Organization reproduced from [Christopher J.F. Cameron and Dostie, 2016]. A: at a high resolution, the genome is organized in a succession of chromatin loops formed by the interaction of chromatin-binding proteins. B: chromatin loops are grouped within larger (0.5 to 1Mb on average) Topologically associated Domains (TADs). C: Compartments divide chromosomes into opposite regions of the genome in terms of transcriptional activity. D: Finally, chromosome territories define the localization of chromosomes within the cell nucleus.

1.1.2 Hi-C Technology

Hi-C, a technology probing the spatial organization of entire genomes, was introduced by Lieberman et al. in 2009 [Lieberman-Aiden et al., 2009]. Unlike previous chromosome conformation capture (3C) technologies, Hi-C is not restricted to the study of specific predetermined loci but allows the study of the spatial organization of whole genomes. An overview of the Hi-C process is shown on Figure 1.2.

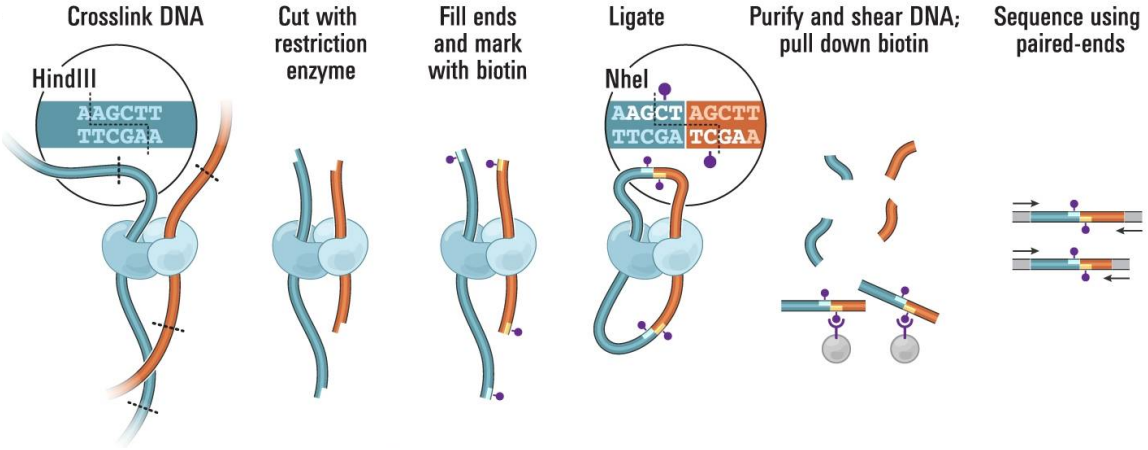

Figure 1.2: Overview of the process of a Hi-C experiment reproduced from [Lieberman-Aiden et al., 2009]. Cells are first crosslinked; then DNA is digested with a restriction enzyme like HindIII, NcoI or DpnII; free ends are filled with a biotinylated residue which are then ligated. A Hi-C library of DNA reads is then created with these ligation products and analyzed with massive parallel DNA sequencing.

Hi-C [Lieberman-Aiden et al., 2009], the process described in Figure 1.2, is the high-throughput application of 3C technology. Briefly, in the Hi-C protocol, DNA is first cross-linked using formaldehyde (to strengthen covalent bonds) and then digested/cut at specific sites by a restriction enzyme, such as HindIII or MboI. The resulting fragmented DNA is biotinylated (to identify true cut sites) and then complementary, overhanging ends are ligated. These ligated products are finally purified and the biotin ends are selected to create a library of spatially close DNA fragments. This Hi-C library is then analyzed using massive parallel sequencing, hence creating a catalog of interacting fragments that can be used to

quantify the frequency of interaction of two genomic loci. To get an idea of the amount of data generated by a Hi-C experiment, for mouse embryonic stem cells, Dixon et al. (2012) [Dixon et al., 2012] produced 806 million total read pairs and in 2017, Bonev et al. [Bonev et al., 2017] produced 7,260 billion read pairs.

In more details, once the raw Hi-C library is obtained, it can be mapped to a reference genome (mouse, human...) using tools like the HiCUP pipeline [Wingett et al., 2015]. First, the reads are truncated at restriction sites to remove bases not originally present in the DNA sequence. Then, the purified reads are aligned to the reference genome and a first raw contact matrix can be built. Indeed, the typical output of a Hi-C experiment is presented as a contact matrix where the genome is divided into bins forming the rows and the columns of the matrix and where each entry represents the frequency of interaction of the two corresponding genomic bins, sometimes also called genomic loci. In order to have exploitable data, different biases of Hi-C experiments, such as the distance between restriction sites or GC content, need to be corrected. This can also be done using HiCUP [Wingett et al., 2015] or other tools such as the probabilistic method implemented in Hicpipe by Yaffe and Tanay (2011) [Yaffe and Tanay, 2011]. By using an iterative methods, other papers avoid pre-defining biases when analyzing raw Hi-C data. This is the case for HiC-Lib, introduced by Imakaev et al. (2012) [Imakaev et al., 2012], where Hi-C contact maps of relative contact probabilities are produced. Knight and Ruiz (2007) [Knight and Ruiz, 2007] propose an iterative method relying on matrix balancing to solve this problem, and the HOMER software [Heinz S., 2010] can also use an iterative method to correct Hi-C matrices. The final version of the matrix is called the interaction frequency matrix (IF matrix) and is the one used for further analysis of the genome's spatial organization. Figure 1.3 shows an example of Hi-C matrix normalization.

The size of each genomic loci used to form the Hi-C contact map is called the resolution of the Hi-C experiment. As of now, Hi-C experiments have reached a resolution of up to 1kb [Rao et al., 2014], in comparison to the 1Mb resolution reached with the initial introduction of Hi-C. Given the large size of DNA sequences (more than 3.2 mega-basepairs for the whole human genome) studied in each experiment, high-resolution contact matrices are

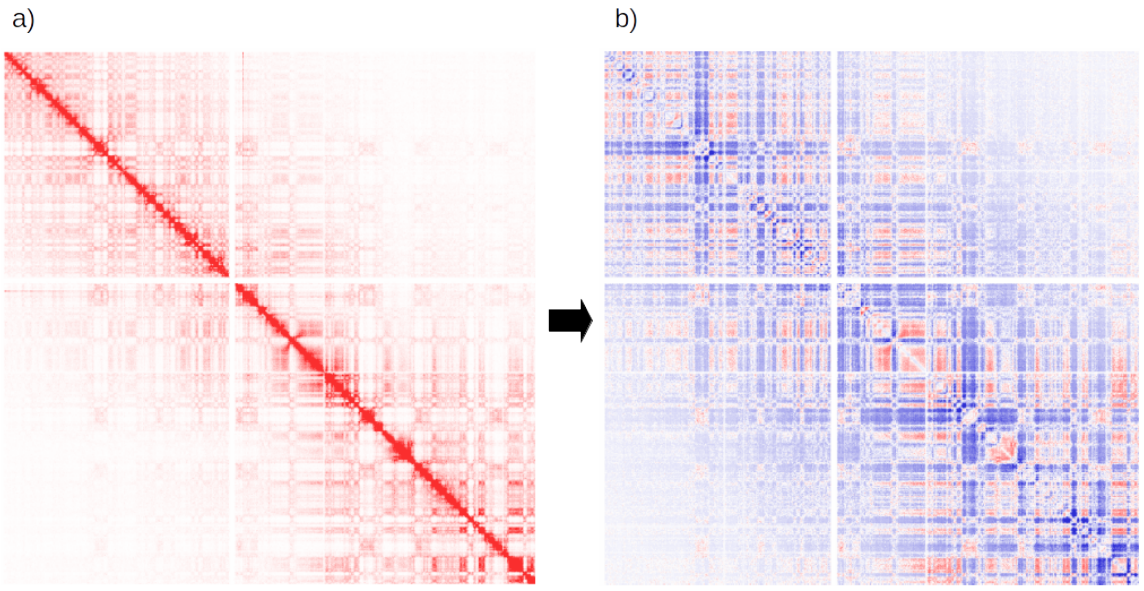

Figure 1.3: Example of the normalization of a Hi-C contact map, reproduced from [Heinz S., 2010]. a) Read count (raw) Hi-C contact map, red represents high contact counts and white low contact counts. b) Normalized version of the a) matrix with the HOMER software, $\log(\frac{observed}{expected})$. Here the matrix was corrected for genomic distance and sequence coverage. Red represents entries where the oberved read counts were higher than expected and blue represents entries where the observed read counts were lower than expected given the genomic distance between the two interacting bins and sequencing depth.

filled with a large number of 0s, e.g. are sparse matrices, that is to say that many possible interactions have never been observed. It should be noted that the definition of resolution in Hi-C is problematic. Indeed, resolution first depends on the precision of the restriction enzyme used in the experiment, more precisely on the size of the resulting restriction fragments (from 400bp for MboI to 4kb for HindIII for instance). Then, the main limiting factor in Hi-C is the obtained sequencing coverage. The final resolution of the contact map is a trade-off between high-resolution and low sequence coverage e.g. between obtaining a high-resolution contact map and having a reasonable number of samples in each entry of the matrix. Studies to tackle this trade-off and obtain up to restriction-fragment resolution contact maps have been made [Cameron et al., 2018, Zhang et al., 2018c].

1.1.3 Chromatin A/B Compartments

With the introduction of Hi-C, mammalian genomes were found to be partitioned into two types of megabase-sized compartments [Lieberman-Aiden et al., 2009]. Compartments are responsible for the plaid pattern that can be observed on Hi-C contact maps, as shown in Figure 1.4. They are derived by performing principal component analysis (PCA) on the contact map, the sign of the first principal component dividing the genome into A and B compartments [Lieberman-Aiden et al., 2009]. PCA [Karl Pearson, 1901] is a mathematical linear transformation which projects a data set on a new system of coordinates, called principal components, such that the first coordinate, i.e the first principal component, is the one capturing the largest part of the data set variance. A data set is understood here as a set of n experiments each characterized by a set of m features and can be represented by a matrix $\mathrm{X}_{n\times m}$. In the context of Hi-C and compartments determination, the experiments are the rows of the Hi-C contact map and the features are the columns. Since the Hi-C contact map is symmetric, we could equivalently view the rows as the features and the columns as the experiments. Intuitively, rows of the Hi-C contact map that behave similarly will have similar principal component values while rows that behave oppositely will have opposite principal component values, which is how A and B compartments are set apart. Mathematically, if $\mathrm{X}_{n,n}$ represents the contact map, performing PCA on X will produce a set of n-dimensional coefficient vectors $w_k = (w_1, ..., w_n)_k$ mapping each row vector $x_i \in \mathrm{X}$ to its projected value i.e to a vector of principal component scores s_i such that $s_{k,i} = x_i \cdot w_k$. To maximize the variance, the first weight vector needs to satisfy:

$$w_1 = \text{argmax}_{||w||=1} \sum_i (x_i \cdot w)^2$$

where each coefficient vector is constrained to be a unit vector. In matrix notations, this is equivalent to:

$$\begin{aligned} w_1 &= \text{argmax}_{||w||=1} \; || \, Xw \, ||^2 \\ &= \text{argmax}_{||w||=1} \; w^T X^T X w \\ &= \text{argmax}_w \; \frac{w^T X^T X}{w^T w} \end{aligned}$$

If the matrix $X^T X$ is positive semidefinite, i.e if $\forall z \in \mathcal{R}_n^*,\ z^T X^T X z > 0$, then the maximum value is reached when w_1 is equal to the first eigenvector of the matrix, i.e to the eigenvector corresponding to the largest eigenvalue of the matrix. An eigenvector of a matrix A is a vector v such that $Av = \lambda v,\ \lambda \in \mathcal{R}$ where λ is the corresponding eigenvalue. In practice, the HOMER software can be used to derive A/B compartments. Other papers like [Dixon et al., 2015, Fraser et al., 2015] also use variations of the original PCA method to annotate compartments. Nagano et al. (2017) then proposed to use k-means clustering with K=2 to identify A/B compartments [Nagano et al., 2017]. To the best of our knowledge, no comprehensive review of compartment calling methods was made and for this work, unless otherwise stated, we used the PCA method implemented in HOMER.

In addition to their characteristics on Hi-C contact maps, compartments also have distinct biological properties. A(ctive) compartments have been linked to euchromatin and are gene rich, transcriptionally active regions while B (inactive) compartments are associated with inactive genomic regions and heterochromatin. A compartments were also found to have a high GC content and to be enriched in activating H3K36me3 histone marks [Lieberman-Aiden et al., 2009]. A further study by Dixon et al. (2015) [Dixon et al., 2015] highlighted that compartments are cell type specific and variable across differentiation, with 10% of compartments being subject to alterations during the differentiation of human embryonic stem cells into neuron progenitors. They furthermore show that up to 36% of compartment switch type at least once during the differentiation of human embryonic stem cells into four distinct cell types, namely neuron progenitors, mesendoderm, mesenchymal and trophoblast-like cells. As such, compartments are thought to play a part in cell-type specific gene expression profiles [Dixon et al., 2015]. However, the specific determinants of compartments establishment and their impact on gene regulation remain unclear [Adriaens, 2018].

In human, Rao et al. (2014) [Rao et al., 2014] showed that A/B compartments were further partitioned into six types of sub-compartments, each one with its own genomic and epigenomic characteristics. For instance, they detected that even though two A type sub-compartments, A1 and A2, had different replication times, GC content and different

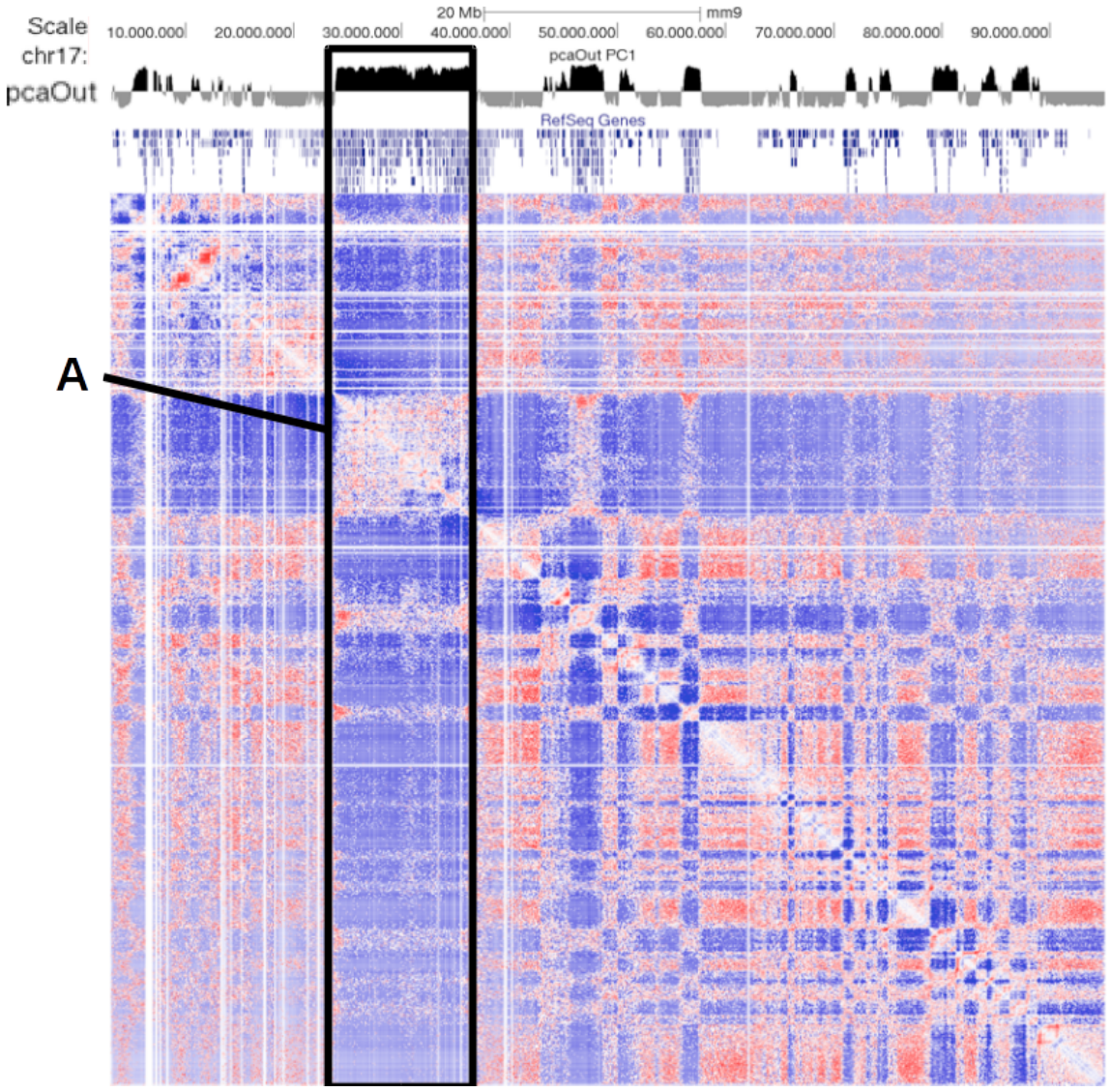

Figure 1.4: Example of the output of a Hi-C experiment after normalization, reproduced from [Heinz S., 2010]. Regions in red indicate high contact frequency and regions in blue low contact frequency. The PCA track above the contact map represent the value of the first principal component for the corresponding genomic region. The A box identify one portion of the genome belonging to the A compartment.

associations with the H3K9me3 chromatin mark, they were both enriched in histone marks such as H3K79me2, H3K27ac and H3K4me1. Another paper by Ma et al. (2018) [Ma et al., 2018] studies the spatial co-localization of transcription factor binding sites and their

occupancy. Their work suggests that cell type specific Transcription Factor (TF) spatial networks can account partially for the previously mentioned sub-compartments. Furthermore, they show that chromatin spatial organization can help understanding the functioning of genome-wide gene regulatory networks.

As compartments are thought to play a part in cell-type specific gene expression profiles [Dixon et al., 2015] and that there is a lack of knowledge about their determinants and formation mechanism, work in this direction seems necessary.

1.2 Machine Learning

Technologies such as Hi-C produce large amounts of data and demand important computational power to be analyzed. Machine Learning is booming and has fast-growing applications in numerous fields, including bioinformatics, as it is well suited to detect patterns invisible to the human eye in large datasets. In this work, we focus on applying and combining different machine learning algorithms for the detection of A/B compartments, that is to say for a supervised binary classification problem. In particular, we experimented with decision trees and the Random Forest (RF) algorithm as well as with Artificial Neural Networks (ANNs). Both are powerful families of algorithms capable of learning complex non-linear decision functions. Bayesian optimization, as a tool to determine optimal hyperparameters combinations for our models, is also presented in this section.

1.2.1 Random Forest

Random Forest is an ensemble method introduced by Breiman in 2001 [Breiman, 2001] and composed of a combination of decision tree (DT) classifiers [Breiman, 1984]. DTs are non-parametric classifiers which can learn complex functions with a set of if-and-then rules on a data set's features. They can be learned recursively by choosing the best test in terms of information gain in a classification setting or of mean squared error in a regression setting. If one of the test outcomes contains a single class, a leaf is created and the algorithm ends.

Otherwise, another test is chosen. The algorithm C4.5 [Salzberg, 1994] implements this method with binary tests in the case of classification. In the case of regression, the CART learning algorithm [Breiman, 1984] can be used. One of the main advantages of DTs is that they are easy to interpret, which is valuable in the context of biological applications. Moreover, little data processing is needed to train them, an advantage in the case of large-scale datasets, and their learning algorithms are fast. However, DTs are prone to overfitting and thus do not always generalize well to unseen data. Indeed, when a tree is fully grown such that each leaf node only contains one class, many tests are often irrelevant and induce errors during generalization.

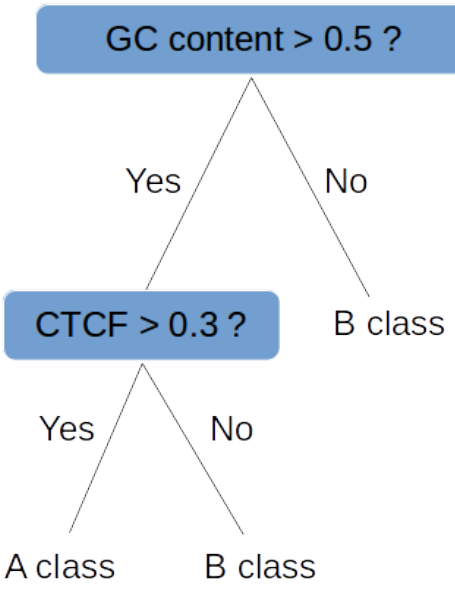

Figure 1.5: Example of a decision tree with depth = 2. A set of if-and-then rules divides the data set into two distinct classes: at the first node, if the GC content of the example is smaller than 0.5 then the example is classified as a belonging to a B compartment, else a second test is run; if the example's CTCF is greater than 0.3 than it is classified as belonging to the A compartment, otherwise to the B compartment.

Although techniques such as tree pruning, where deeper nodes with low information gain are removed, can help in mitigating this issue of DTs, a Random Forest (RF) classifier can also be used instead of a single decision tree. Indeed, ensemble methods such as

RF regroup multiple simple estimators to avoid overfitting. In the case of Random Forest, several DTs are regrouped and their predictions averaged to make a classification. More precisely, RF is known as a bagging method, that is to say that each DT it is composed of is trained independently with a bootstrap sample from the training set as well as with a random subset of features, to yield slightly different estimators. Then, their probability predictions can be averaged to make the final Random Forest prediction for a sample. The use of bootstrap samples (i.e samples drawn with replacement from the training set) introduces randomization in the training procedure, a source of reduced model complexity (i.e a source of bias) but also a cause of decreased variance with the predictions averaging. In summary, this method reduces the expressivity of the RF classifier by averaging, which has the beneficial consequence of reducing its variance. However, this reduction in expressivity is also made at the cost of introducing bias, i.e as the model looses complexity, it is less able to capture complex patterns in the data set. In our case as well as in a majority of applications, the trade-off is mainly beneficial.

Because it consists of an ensemble of multiple decision trees, Random Forest looses a bit of interpretability compared to a single decision tree. However it is still possible to estimate relative feature importance and retrieve a ranking of the most important features for the decision process. In a single tree, an estimate of the relative importance of a feature can be obtained by considering the expected fraction of samples this feature contributes to classifying. This number is combined in scikit-learn [Pedregosa et al., 2011] with the decrease in impurity reached by adding the feature of interest to the tree. This method is also known as the mean decrease in impurity (MDI) [Louppe, 2014]. In a forest of trees, this estimate can be averaged over all tress, hence reducing its variance and providing a more reliable estimate of the feature's predictive power. In this work, we exploited this characteristic of Random Forest to use it as a feature selection algorithm, allowing us to reduce the total number of input features to our model while keeping informative ones.

1.2.2 Artificial Neural Networks

Another powerful family of non-linear classifiers is the one of Artificial Neural Networks (ANNs). ANNs were originally developed to reproduce the information processing system of brains [Rosenblatt, 1962]. They are composed of nodes, also called neurons, linked to each other with weighted connections, in reference to the synapses that can be found in the brain, and organized in successive layers. When the number of such stacked layers increases, the term 'deep learning' can be used to describe these networks and their functioning. Here, we briefly describe two subsets of ANNs that are used in this project. First, we will look into feedforward neural networks or Multi-Layer Perceptrons (MLPs), which are the original type of ANNs. Then, the family of Recurrent Neural Networks (RNNs), particularly well-suited to make predictions on sequence-structured data, is studied. Indeed, RNNs and their declinations are widely used in natural language processing and more broadly in all kinds of sequence-labelling problems, which makes them strong candidates to tackle the compartment prediction problem.

Multi-Layer Perceptron (MLP) MLPs are supervised classifiers that can learn complex non-linear functions by stacking successive layers of neurons. Figure 1.6 shows an example of a one-hidden layer MLP. The input layer copies the input vector values X. The hidden layer then linearly combines the inputs, $W \cdot X$ where W is a matrix of learnable weight parameters, and applies a non-linear function known as an activation function to the results. In this project, we chose the commonly used logistic sigmoid function $f : x \rightarrow \frac{1}{1+exp(-x)}$ as activation function. The output layer finally receives these non-linear transformations of the inputs and transforms them into output values. In the case of classification, the softmax function $f : \mathbf{R}^O \rightarrow \mathbf{R}^O$, where O is the number of classes, defined as:

$$f(x_i) = \frac{e^{x_i}}{\sum_{k=1}^{O} e^{x_k}}$$

is often used as output function.

Once the architecture (i.e. the number of hidden layers and the number of nodes per

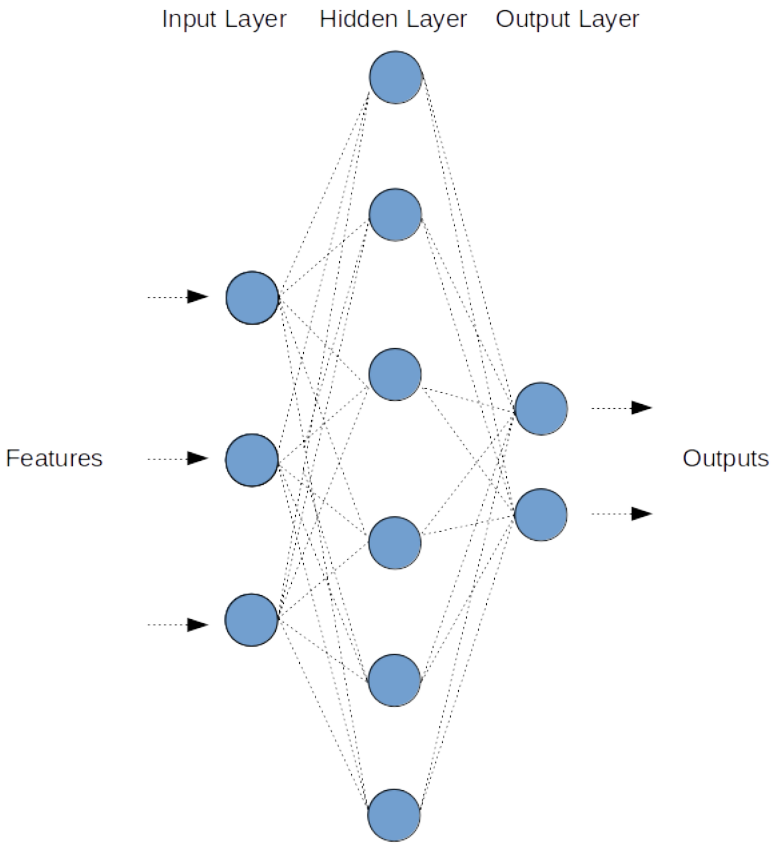

Figure 1.6: One-hidden layer MLP with three input features and two output neurons. The input layer copies the input vector. The hidden layer then linearly combines the inputs and applies a non-linear function to the results. The output layer finally receives these non-linear transformations of the inputs and transforms them into output values.

layers) and the characteristics of the network (the activation function for instance) are chosen, the network needs to be trained in order to make accurate predictions. Training a MLP is done by determining weights which will minimize a loss function, i.e an objective function measuring mathematically how far the MLP's predictions are from the ground truth. In this project, we are in the setting of binary classification and hence choose cross-entropy as a loss function. For a data set of size N examples, the cross-entropy loss function can be written as :

$$J(w) = -\frac{1}{N}\sum_{n=1}^{N}\left[y_n \log \hat{y}_n + (1-y_n)\log(1-\hat{y}_n)\right]$$

where $\hat{y}_n$ is the prediction of sample n by the network and y_n is its true value or label.

In order to obtain the best predictor possible with this model, the weights of the network need to be chosen such that the errors represented by the loss function are as few and of the smallest amplitude as possible. The network is hence trained to minimize this loss function. This is done by applying gradient descent to it, an optimization method designed to find the minimum of a function [Cauchy, 1847, Robins and Monro, 1951]. Intuitively, gradient descent consists in taking 'small' steps inversely proportional to the gradient of the loss function with respect to the weights of the network toward a minimum of the function. The proportionality coefficient defining how small the steps are is called the learning rate. The step during which the weights of the networks are updated is known as the back-propagation of the errors, first described by Bryson and Ho in 1969 [Bryson and Ho, 1969].

Like decision trees, MLPs are prone to overfitting. Different techniques such as dropout [Srivastava et al., 2014] or L2 regularization can help tackle this issue. Here, we opt for L2 regularization. This method consists in adding a penalization term on the weights w of the network in the loss function, hence effectively reducing the model parameters and preventing overfitting. The new loss function would then be written:

$$J(w) = -\frac{1}{N}\sum_{n=1}^{N}\left[y_n \log \hat{y}_n + (1-y_n)\log(1-\hat{y}_n)\right] + \lambda\left[\frac{1}{2}\sum_i w_i^2 + \frac{1}{2}\sum_{i,j} W_{i,j}^2\right]$$

with λ being the L2 regularization rate, W the weights between the input layer and the hidden layer and w the weights between the hidden layer and the output layer.

One difficulty that needs to be taken into account with the use of neural networks is that the loss function being minimized during training is non-convex. This implies that this function has multiple local minimum and that the global optimum might not be reached during training. Here, we decide to use the Adam optimizer during training of the network. Adam [Kingma and Ba, 2014] is a gradient-based optimization algorithm which is widely used in the deep learning community and yields high performances for our experiments.

Finally, many parameters like the number of hidden layers, the number of neurons per hidden layer, the initial learning rate of the Adam optimizer as well as the L2 regularization rate need to be tuned in order to optimize the performance of the MLP classifier. These parameters are known as hyperparameters and tuning them efficiently is a major challenge when opting for any type of neural network. We will see in Section 1.2.3 how we address this problem.

Recurrent Neural Networks RNNs [E. Rumelhart et al., 1986] are different from MLPs in the way their neurons are connected. In a MLP, each layer is connected to the next without connections between neurons belonging to the same layer. On the other hand, RNNs allow such connections and take advantage of this structure to account for sequence data, where an input at time t can be dependent on that at time $t-1$. The underlying assumption in many machine learning algorithms is that the input data is identically and independently (iid) sampled from an unknown distribution. In the case of sequence data, this assumption is false as each input can have dependencies on past or future inputs. This type of structure is omnipresent in natural language or time series but can also be seen in DNA sequences for instance.

RNNs also exploits parameter sharing: the same weight matrix is used for recurrent connections in a layer of the network:

$$h_t = tanh(W_h * h_{t-1} + W_i * x_t + b)$$

Here, h_t is the hidden state of the network at timestep t, x_t is the input data at timestep t, W_h and W_i are the shared parameter matrices between hidden states and inputs respectively and b the bias vector.

One of the drawbacks of RNNs is that they have difficulties learning long-term dependencies [Yoshua Bengio and Frasconi, 1994]. Indeed, when applying backpropagation to a

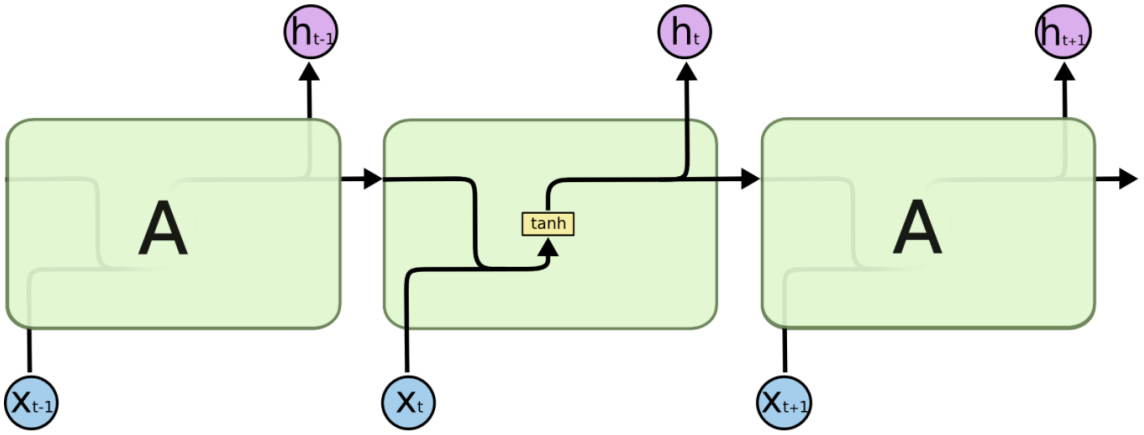

Figure 1.7: Overview of a RNN unit reproduced from [Colah, 2015]. x_t: input data at timestep t, h_t: hidden state at timestep t, $tanh$: non-linear activation function.

RNN, the shared parameter matrix is multiplied several times, causing exploding or vanishing gradients that prevent the algorithm from learning. This problem can be solved with Long-Short Term Memory networks (LSTMs) [Hochreiter and Schidhuber, 1997], a variant of RNNs where a different repeated unit is used, as can be seen by comparing Figures 1.7 and 1.8. The equations for this new type of unit are the following:

$$
\begin{aligned}
f_t &= \sigma(W_f \cdot [h_{t_1}, x_t] + b_f) \\
i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\
\tilde{C}_t &= tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\
C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t
\end{aligned}
$$

In these equations, all Ws represent parameter weight matrices and all b's are bias vectors. f is known as the forget gate, its role is to decide how much of the previous cell information the current cell should keep. As the sigmoid function σ ranges between 0 and 1, a 0 in this gate is equivalent to not keeping any information about the previous state while a 1 means to keep everything. Then, i and $\tilde{C}$ form the input gate and indicate to the network what information it should keep about the current state. Finally, the cell state C is updated with a linear combination of the previous elements. It is this linear operation which avoids the multiple matrix multiplications present in RNNs while capturing the necessary information

about the previous and the current state of the network. To get the final output for the current state, the next two operations are performed:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t * tanh(C_t)$$

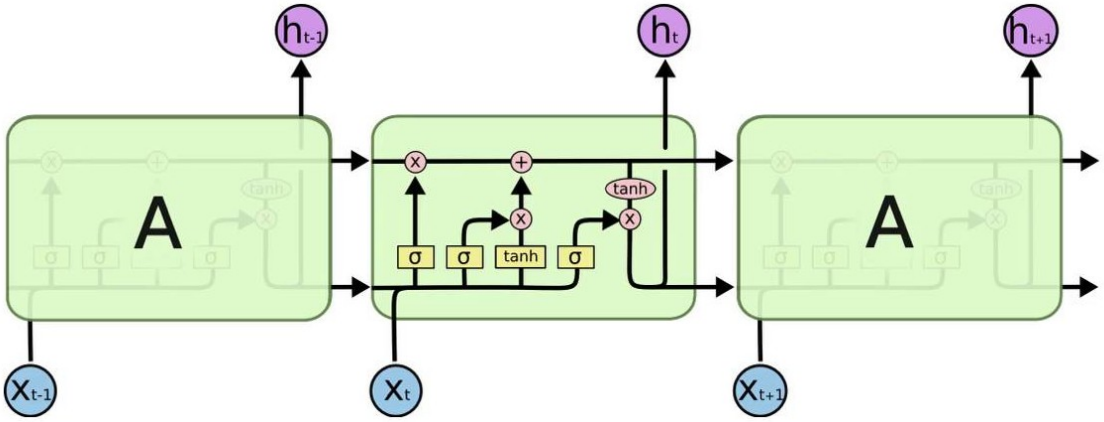

Figure 1.8: Overview of a LSTM unit reproduced from [Colah, 2015]. x_t: input data at timestep t, h_t: hidden state at timestep t.

In this project, we experimented with RNNs and LSTMs but the results were not very different from what was obtained with a combination of MLPs and the use of an input window around each input vector. However, the RNNs models are longer to train so we decided to keep working with the stacked MLPs model.

1.2.3 Bayesian Hyper-parameter Optimization

As was seen in the previous section, many hyper-parameters need to be tuned for neural networks to function optimally. Hyper-parameter tuning is an active field of research. Currently, the most widely used method is a brute-force approach which consists in performing grid search, that is to say in exhaustively testing combinations of pre-defined hyper-parameters values. However this method is computationally costly in the number of hyper-parameter combinations tested. Another drawback is that if the impactful parameters are

not known beforehand, which is the case in many settings, the algorithm looses efficiency. Indeed, many equivalent combinations will be tested hence wasting computing resources and time. To balance these disadvantages, randomized grid search [Bergstra and Bengio, 2012] is also an option, where hyper-parameters combinations are chosen at random in the grid. However this technique is not optimal in the way the combinations are sampled.

Another way to tackle this problem is to view hyper-parameter tuning as the optimization of a black-box function. Bayesian optimization can then be a good choice compared to other global optimization algorithms [Jones, 2001, Snoek et al., 2012]. Like in randomized grid search, only a subset of hyper-parameters combinations are tested. The difference with randomized grid search resides in the way this subset of combinations is chosen. In Bayesian optimization, the performance of a machine learning algorithm is viewed as a sample from a gaussian process and the posterior distribution for this function allows insight into the confidence of the function's value in the hyper-parameter space. There are then two options to choose the next combination to be tested: either sample where the performance function is at its highest to gain in performance or explore another subset of the hyper-parameter space where not a lot of information is available. This is known as the exploitation and exploration trade-off and has been thoroughly studied. Methods like the Gaussian Process Upper Confidence Bound (GP-UCB) [Srinivas et al., 2010] and Expected Improvement (EI) [Mockus et al., 2014] can be used in practice to address it.

Here, we choose to use the Bayesian optimization software Spearmint [Snoek et al., 2012], which implements a Bayesian treatment of EI. The user only needs to specify the hyper-parameters and the ranges that the software needs to explore. A user-chosen number of random hyper-parameter combinations are then tested before the start of the optimization part. The user also specifies the total number of iterations performed by the optimizer.

1.3 Applications of Machine Learning to 3D Genomics

As mentioned in the previous section, machine learning is now widely used in multiple fields including bioinformatics. For instance, RNNs were first used in bioinformatics for proteins secondary structure predictions in 1999 [Baldi et al., 1999]. More recently, many applications of deep learning to biological problems emerged, like for the prediction transcription factors binding sites [Alipanahi et al., 2015] or of DNA accessibility [Kelley et al., 2016]. Here, we will focus on machine learning applications in the field of 3D genomics and more specifically for Hi-C derived predictions and sequence-based predictions of DNA's 3D structure elements.

1.3.1 Machine Learning in Hi-C data analysis

Prediction of an entire Hi-C contact map Attempts to predict entire Hi-C contact maps from different sources of data have been made, notably by Farré et al. (2018) [Farré et al., 2018]. In this paper, the authors model the DNA sequence as a distribution of DNA-bound chromatin-associated factors extracted from ChIP-seq data. They then predict the Hi-C interaction frequency matrix by combining a convolutional filter and a feedforward neural network. This architecture yields a Pearson correlation coefficient of 0.68 between the original and the predicted contact maps for *Drosophila Melanogaster* embryos. Although these results are encouraging, the authors present their work as a proof-of-concept and not a fully developed predictor of Hi-C contact maps.

Another promising study was made simultaneously by Zhang et al. (2018) [Zhang et al., 2018b]. The authors use one dimensional signals including chromatin marks, chromatin accessibility and protein binding to predict Hi-C contact counts at 5kb resolution with a Random Forest regression-based approach called HiC-Reg. One of their interesting contributions is to use data from multiple cell types to enhance the model's predictive power. HiC-Reg obtains an average AUC score for the distance-stratified Pearson correlation of about 0.60 for human GM12878 cells. At a higher resolution of 200bp, Zhu et al. (2016) [Zhu et al., 2016] also use one-dimensional vectors, including histone marks, chromatin ac-

cessibility and RNA-sequencing, to identify spatial interactions within TADs. Their predictor consists in an unsupervised method named EpiTensor, the use of unsupervised learning preventing them from relying on possibly uncertain labels produced by biological experiments such as Hi-C. On specific tasks like active promoter-enhancer interaction predictions, EpiTensor reaches a high AUC score of 0.87 with Hi-C detected interactions as ground truth in human IMR90 cells.

Instead of relying on machine learning to infer spatial genome organization, other studies use polymer physics to model genome folding and predict Hi-C contact maps, either using a bead-and-spring polymer to model chromatin fiber [Brackley et al., 2016] or going further, a block copolymer model for entire epigenomic regions [Jost et al., 2014]. In this last model, successive monomer blocks representing fix-sized DNA sections form block copolymers accounting for the local epigenomic state.

In general, these studies are difficult to compare as they are often made on different cell types and species, with different performance measures and even though they all aim at describing genome folding, they do not always produce entire contact maps but sometimes only subsets of such maps. However they all have in common the use of external biological data as input features as opposed to the primary DNA sequence.

TADs prediction Other studies have focused on specific elements of Hi-C contact maps, such as the localization of TAD boundaries in the genome. For instance, Rennie et al. (2018) [Rennie et al., 2018] use a Generalized Linear Model (GLM) to predict TAD boundaries from gene expression data. Huang et al. (2015) [Huang et al., 2015] decided to take advantage of ChIP-seq tracks of histone modifications to predict TAD boundaries as detected in Hi-C contact maps. The prediction task is made by a Bayesian Additive Regression Tree (BART), a model averaging the predictions from an ensemble of regression trees. Finally, Oluwadare and Cheng (2017) [Oluwadare and Cheng, 2017] have a different approach as they annotate TAD boundaries from Hi-C data. The authors produced an unsupervised clustering tool, ClusterTAD, in order to detect TADs more accurately in Hi-C contact matrices.

BART and the GLM methods use different input data and machine learning algorithms but have similar performances on the tested cell types. Indeed with BART, Huang et al. (2015) [Huang et al., 2015] reach an AUC score of 0.774 on human IMR90 cells while Rennie et al. (2018) [Rennie et al., 2018] get an AUC score of 0.73 in human GM12878 lymphoblastoid cells. Since the prediction problem is different in [Oluwadare and Cheng, 2017], the results are not comparable. The author tested ClusterTAD on simulated Hi-C data and compared their annotations with two other TAD annotation methods on two mouse data sets.

1.3.2 Prediction of A/B Compartments

Aside from TAD boundaries and whole Hi-C contact maps, work has also been done to infer A/B compartments from various biological sources of data. As an example, Di Pierro et al. (2017) [Di Pierro et al., 2017] combine a neural network with an energy landscape model for chromatin organization to predict chromatin architecture de novo. They use as features publicly available ChIP-seq data, including 84 protein-binding experiments and 11 histone modification tracks for human GM12878 lymphoblastoid cells. Another example is the one of Fortin and Hansen (2015) [Fortin and Hansen, 2015], who use eigenvector analysis of epigenetic data correlation matrices to reconstruct compartments. Their method is similar to the original method used to detect compartments, that is to say PCA performed on a Hi-C contact map.

These two studies achieve accurate results on the tested data sets, with [Di Pierro et al., 2017] reaching 89% accuracy on human lymphoblastoid cells GM12878 and [Fortin and Hansen, 2015] reaching up to 86% agreement for a human EBV data set and 80% on IMR90 fibroblast cells using the eigenvector of DNAse hypersensitivity correlation matrix as a predictor of compartments. [Di Pierro et al., 2017] also predicts sub-compartments as defined in [Rao et al., 2014] but with less success. None of these methods were tested on the data sets that were used for our study and the corresponding biological data are not all available

for the cell types we experimented on so it was not possible to perform a direct comparison between our method and theirs. However, as will be seen in the next section, our results seem similar to these performances while obtained directly from the DNA sequence as a source of features.

The different studies mentioned in the two previous paragraphs show that a broad range of machine learning algorithms can be successfully used to infer elements of DNA's spatial structure, from TADs to compartments and even for the inference of whole Hi-C contact maps.

1.3.3 Sequence-based Predictions in 3D Genomics

So far, all the methods that were cited rely on some form of external biological data to make predictions and are hence dependent on the availability of that data for a given genome and cell type. Leveraging only DNA sequence features would reduce these constraints and make predictions possible for any sequenced genome. Moreover, previous work succeeded in predicting elements of DNA's spatial structure from sequence-level features. For instance, Nikumbh and Pfeifer (2017) [Nikumbh and Pfeifer, 2017] successfully predict long-range chromatin interactions using a genetic sequence-based Support Vector Machine (SVM) predictor. Whalen et al. (2016) [Whalen et al., 2016] use candidate enhancers and promoters genetic sequences to predict enhancer-promoters interactions with Ensemble Boosted Trees. And more recently, Zhang et al. (2018) [Zhang et al., 2018a] predict CTCF-mediated chromatin loops from sequence-level features, using a combination of a trained word2vec model and relevant biological features to summarize DNA sequences that they then feed to a boosted tree classifier.

1.4 Thesis outline

The studies detailed in Section 1.3.3 prove that different elements of DNA's spatial organization can be inferred from sequence-level features. In this project, we work under the hypothesis that compartments are at least partially determined by the underlying DNA sequence and that a machine learning algorithm would have the ability to learn these dependencies. A second goal will be to interpret the algorithm in order to get insights about the underlying compartments' establishment mechanisms. We propose a machine learning pipeline, named Sequence-based Annotator of Chromatin Compartments by Stacked Artificial Neural Networks (SACCSANN), to infer compartments based solely on features derived from the DNA sequence. Such a model enables compartment annotations for genomes where only the DNA sequence is available, in what we differ from [Di Pierro et al., 2017] and [Fortin and Hansen, 2015]. This in turn can lead to applications to reconstructed ancestral genomes, for which Hi-C or ChIP-seq experiments cannot be performed. Our model achieves high performances on both human and mouse embryonic stem cells (ESC) as well as on different mouse cell types, including neuron progenitor (NPC) and cortical neurons (CN) from different published datasets. We demonstrate that our model is transferable between species: SACCSANN learns a set of rules to describe compartments that is applicable to both mouse and human. In addition, we provide a thorough evaluation of features used by our model to gain insights into the underlying sequence-level biological processes that contribute to the formation of A/B compartments.

Chapter 2 of this thesis will be submitted shortly for publication in Genome Biology.

2

Sequence Determinants of Chromatin Compartments

Abstract

Background: Spatial genomic organization is known to be critical for proper gene regulation. It is based on a hierarchical model where chromosomes are divided into megabase-sized cell-type specific A and B compartments, associated with open and closed chromatin. We propose SACCSANN, a machine learning pipeline composed of stacked neural networks to predict compartments using features derived from genomic sequence alone.

Results: Our results are shown to be highly accurate and reproducible across different cell types and genomes. We also identify key sequence determinants related to the determination of chromatin A/B compartments.

Keywords: A/B compartments, Chromatin, Hi-C, 3D Genomics, Neural Networks

2.1 Background

With the introduction of Hi-C experiments by Lieberman et al. in 2009 [Lieberman-Aiden et al., 2009], mammalian genomes were found to be segmented into two types of megabase-sized compartments: 1) A(ctive) compartments, which have been linked to euchromatin and are gene rich, transcriptionally active regions; 2) B (inactive) compartments, associated with heterochromatin and inactive regions. The typical output of a Hi-C experiment

is a contact map where each entry represents the frequency of interaction of the two corresponding genomic loci. Compartments are then derived by performing PCA on the Hi-C contact map, the sign of the projection onto the first principal component dividing the genome into A and B compartments [Lieberman-Aiden et al., 2009].

A compartments were found to have a high GC content and to be enriched in activating H3K36me3 chromatin marks [Lieberman-Aiden et al., 2009]. A further study by Dixon et al. (2015) highlighted that compartments are cell type specific and variable across differentiation, with 10% of compartments being subject to alterations during differentiation of human embryonic stem cells into neuron progenitors and up to 36% of overall alterations during the course of differentiation into four distinct cell types. As such, compartments are thought to play a part in cell-type specific gene expression profiles [Dixon et al., 2015].

Compartments are themselves composed of one or more Topologically Associated Domains (TADs), regions enriched in local chromatin contacts. In human, Rao et al. (2014) [Rao et al., 2014] then showed that A/B compartments were further partitioned into six types of sub-compartments, each with its own genomic and epigenomic characteristics. For instance, they detected an enrichment of A type subcompartments in histone marks such as H3K79me2, H3K27ac and H3K4me1. Furthermore, a study by Ma et al. (2018) [Ma et al., 2018] suggests that cell type specific Transcription Factor (TF) spatial networks can account partially for these sub-compartments. They show that the genome's spatial organization can help understand genome-wide gene regulatory networks. However, the specific determinants of A/B compartment formation remain unclear. In this paper, we study the problem of compartment prediction from sequence information alone in the hope of gaining insight into the underlying biological processes.

Work to infer compartments from epigenetic data has been done previously. Di Pierro et al. (2017) [Di Pierro et al., 2017] use a neural network to predict chromatin architecture de novo from ChIP-seq data, reaching high accuracy for the prediction of compartments. Similarly to the original PCA on a Hi-C contact map, Fortin et al. (2015) [Fortin and Hansen, 2015] use eigenvector analysis of epigenetic data correlation matrices to reconstruct com-

partments. However these methods are dependent on the availability of epigenetic data for the given species and cell type. On the other hand, Nikumbh and Pfeifer (2017) [Nikumbh and Pfeifer, 2017] successfully predict long-range chromatin interactions using a genetic sequence-based SVM predictor and Whalen et al. (2016) [Whalen et al., 2016] also use candidate enhancers and promoters genetic sequences to predict enhancer-promoters interactions with Ensemble Boosted Trees.

Here, we describe a stacked neural network model for the prediction of chromosomal compartments called 'Sequence-based Annotator of Chromosomal Compartments by Stacked Artificial Neural Networks' or SACCSANN. SACCSANN takes as input features derived solely from the reference DNA sequence of a species, enabling compartment annotations for genomes where only the genome sequence is available. Our model achieves high performances on both human (hESC) and mouse embryonic stem (mESC) cells [Dixon et al., 2012] as well as on mouse cell types resulting from a neural differentiation [Bonev et al., 2017]. We also demonstrate that our models are transferable between species: SACCSANN learns a set of rules to describe compartments that is applicable to both mouse and human embryonic stem cells. In addition, we provide a thorough analysis of features used by our model to gain insights into the underlying sequence-level biological processes contributing to the formation of A/B compartments.

2.2 Results

Hi-C data from a mouse neural differentiation data set containing three distinct cell-types (Embryonic Stem Cells (ESC), Neuron Progenitors Cells (NPC) and Cortical Neurons (CN) from [Bonev et al., 2017] and ESC, NPC and Neurons from [Fraser et al., 2015]) were retrieved for compartment annotation. Hi-C data from human and mouse ESC from [Dixon et al., 2012] was also used. When available, we used both the author's compartment annotations and those produced by the HOMER software [Heinz S., 2010] to train our model. See the Methods section for GEO references and pre-processing steps of the different data sets.

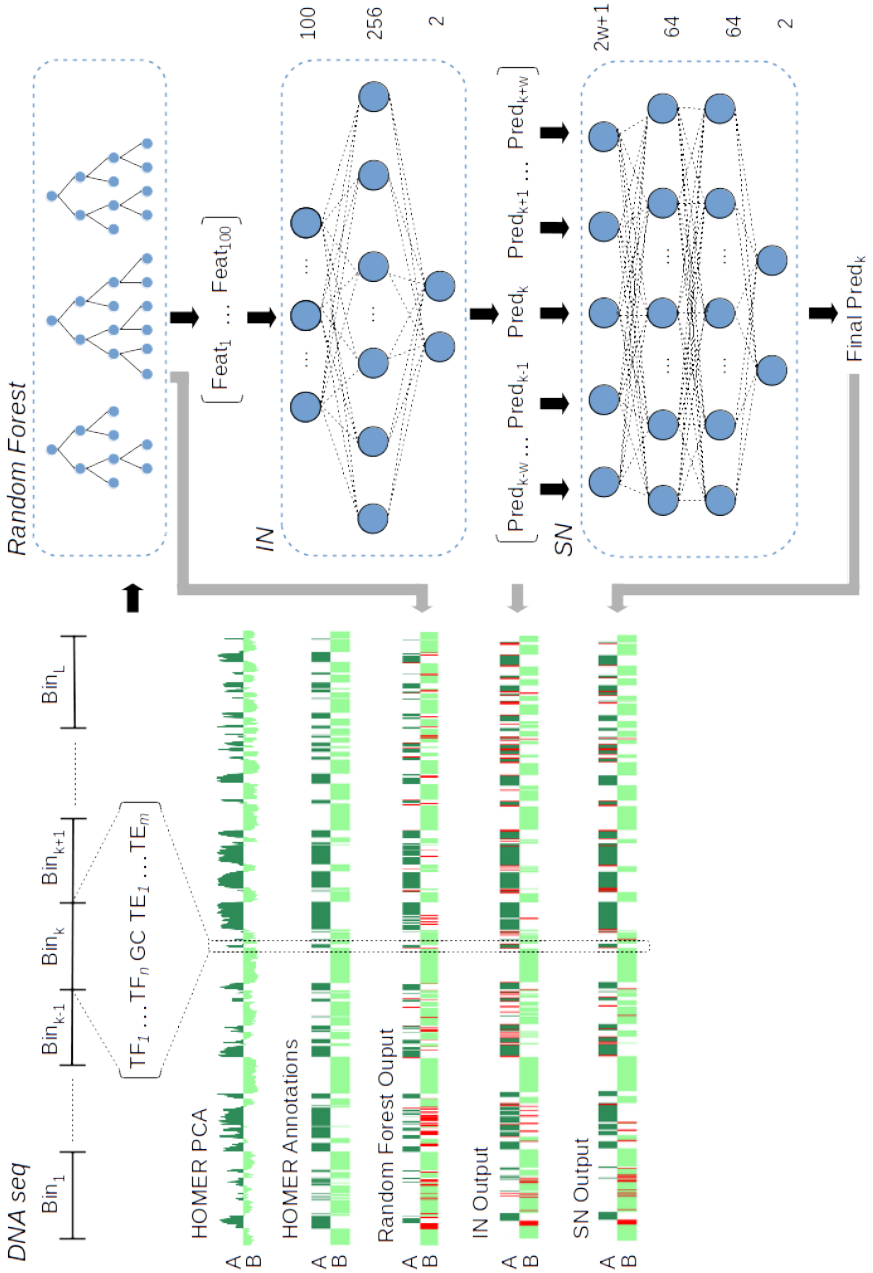

Figure 2.1: The DNA sequence is first divided into fixed-sized bins from which the features are extracted. A Random Forest (RF) algorithm then selects the top 100 features that will be the input to SACCSANN. SACCSANN is itself a stack of two fully connected Artificial Neural Networks (ANN), named Intermediate Network (IN) and Smoothing Network (SN) respectively, which predicts the compartment type of each input genomic bin. On the left, an illustration of compartment annotations and predictions for mouse Embryonic Stem Cells (ESC) chromosome 3. The errors in the predictions are represented in red.

Figure 2.1 gives an overview of the machine learning approach developed for this study. The sequence-level features needed to train our model are extracted by dividing the genome into 100kb bins and counting the occurrences of specific motifs in each bin. More precisely, we count predicted binding sites for 334 transcription factors in human and mouse, as well as 35 transposable elements (TEs) in mouse and 41 in human in addition to the GC content of each bin. The extracted features first go through a species and cell-type specific Random Forest algorithm which is used to select the top 100 features that will be the input to our model. This model, SACCSANN, is composed of two stacked artificial neural networks that classify each input vector as belonging into an A or a B compartment. Once the top 100 features have been selected, SACCSANN is trained for each species and cell type using chromosome-wise leave-one-out cross-validation.

2.2.1 Chromosomal compartments can be predicted from sequence-level features

SACCSANN is accurate across all 8 tested data sets (average AUC $> 80\%$, see Figure 2.2), which indicates that chromosome compartmentalization is at least partially determined by the underlying DNA sequence. We observe that SACCSANN is more performant at predicting HOMER compartment calls (average AUC score $> 88\%$) than authors' compartment calls on mouse ESC and thus decided to focus on these annotations for further analysis (see Table 2.2 for a complete list of the data sets used).

Biological Supporting Evidence of Compartment Annotations A-compartments were previously reported to correlate positively with different histone marks such as H3K36me3, H3K4me1 and H3K27ac [Lieberman-Aiden et al., 2009, Rao et al., 2014]. They are also associated with open chromatin, which can be measured by DNAse hypersensitivity [Song and E Crawford, 2010, Tsompana and Buck, 2014], and highly expressed regions as can be measured by RNA sequencing [Gerstein Z. Wang, 2009]. We study publicly available data sets for these markers in mouse embryonic stem cells, binned at 100 kb resolution (see Methods section for a complete list of references). Each bin is associated with an epigenetic

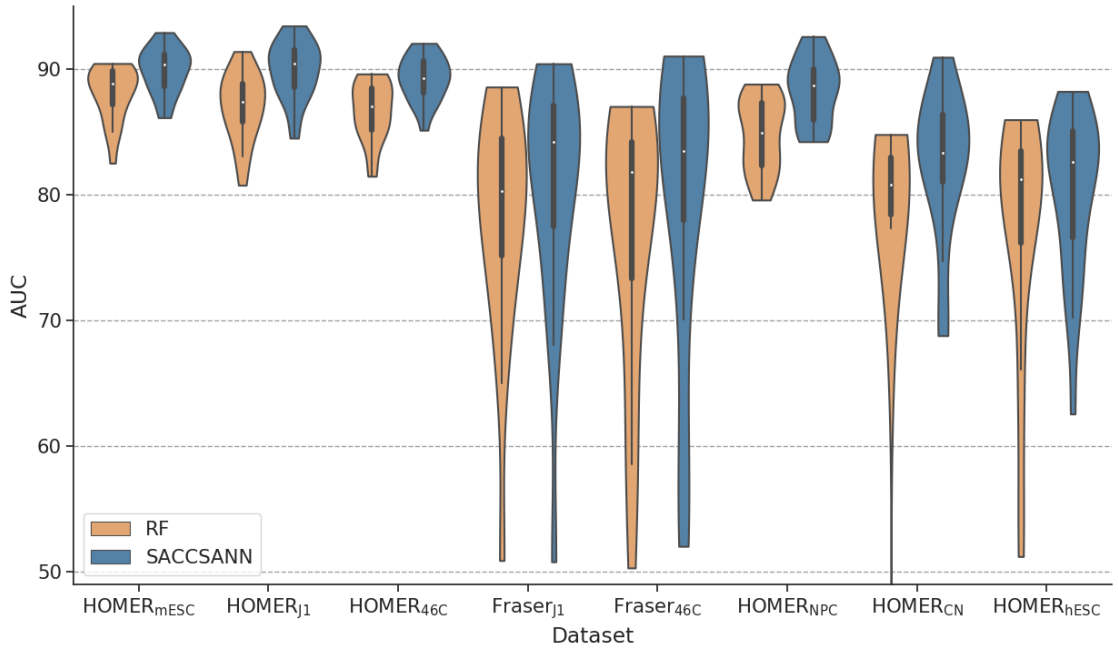

Figure 2.2: Evaluation of SACCSANN on different datasets, cell types and species against a Random Forest (RF) baseline algorithm at 100kb resolution. Each violin plot represents the AUC score distribution obtained by performing chromosome-wise leave-one-out cross validation on different datasets i.e each violin plot contains 19 or 22 AUC scores depending on the species. x label: $Annotator_{celltype}$. $Annotator$: compartment annotator used to get the training compartment annotations, either HOMER software [Heinz S., 2010] or author's name when compartment annotations are provided.

state vector and these are then hierarchically clustered for comparison (Figure 2.3). On the right of these clusters, the HOMER PCA value for each bin is plotted as well as the compartment category each bin belongs to. Here, the following four compartment categories are considered: bins that were annotated as A by both HOMER and SACCSANN, B by both methods, A by HOMER and B by SACCSANN and finally B by HOMER and A by SACCSANN. Three main clusters seem to emerge, which broadly correspond to the two types of compartments. Indeed, the top and the bottom clusters (A type) show high enrichment in H3K36me3, H3K9ac, H3K27ac, H3K4me3, H3Kme3 and CTCF with high values of DNAse hypersensitivity and RNA-seq while the middle cluster (B type) is depleted for these markers, which is consistent with the trends observed in previous studies[Rao et al.,

2014, Lieberman-Aiden et al., 2009]. And accordingly, the top cluster (type A) is almost exclusively composed of bins annotated as A by both approaches (contains 97% of $A \rightarrow A$ which make up for 23% of the total number of bins in the $A \rightarrow A$ category). The bottom one also contains a majority of $A \rightarrow A$ bins (57% of them) while the middle cluster mainly composed of $B \rightarrow B$ bins (92% of them). Interestingly, the two remaining compartment categories, in which the compartment annotators disagree, can be found in majority in the B-type cluster (56% of the $A \rightarrow B$ bins and 66% of the $B \rightarrow A$ bins respectively). More precisely, they are mostly found in a sub-cluster of the B-type cluster, mixed with $A \rightarrow A$ and $B \rightarrow B$ bins as well. This sub-cluster shows a particular epigenetic pattern with enrichment of CTCF and relatively high levels of DNAse but depletion for the other markers. Another characteristic of these disagreeing bins is that their corresponding PCA values are close to 0, which might indicate that they are harder to predict in general. See Figure 2.10 for the distribution of PCA values in the four compartment categories.

After the discovery of A/B compartments, Rao et al. (2014) [Rao et al., 2014] found a further partitioning of compartments into six sub-compartment types in human lymphoblastoid cells, each with its own genetic and epigenetic characteristics. To see if these results could be replicated with this biological clustering, we studied the sub-clusters of Figure 2.3. The two A-type clusters could correspond to the two A-type sub-compartments found in [Rao et al., 2014]. Indeed, as was reported in their study, the top A-type cluster shows higher enrichment of histone marks H3K9ac, H3K27ac and H3K4me3 than the bottom one and slightly higher enrichment of H3K4me1, which points toward the top A-type cluster being the A1 sub-compartment and the bottom one the A2 sub-compartment. However, we were not able to retrieve a similar separation in the B-type cluster. This might be in part due to the resolution at which we performed this study (100kb against 1kb in [Rao et al., 2014]) as well as to the available epigenetic data for this clustering ([Rao et al., 2014] use 20 markers while only 8 were used in this clustering).

To further compare the annotations made by SACCSANN and those by HOMER, we considered the previous epigenetic state vector clustering (Figure 2.3) as being a third compartment annotation method and plotted Venn diagrams comparing the three methods (see

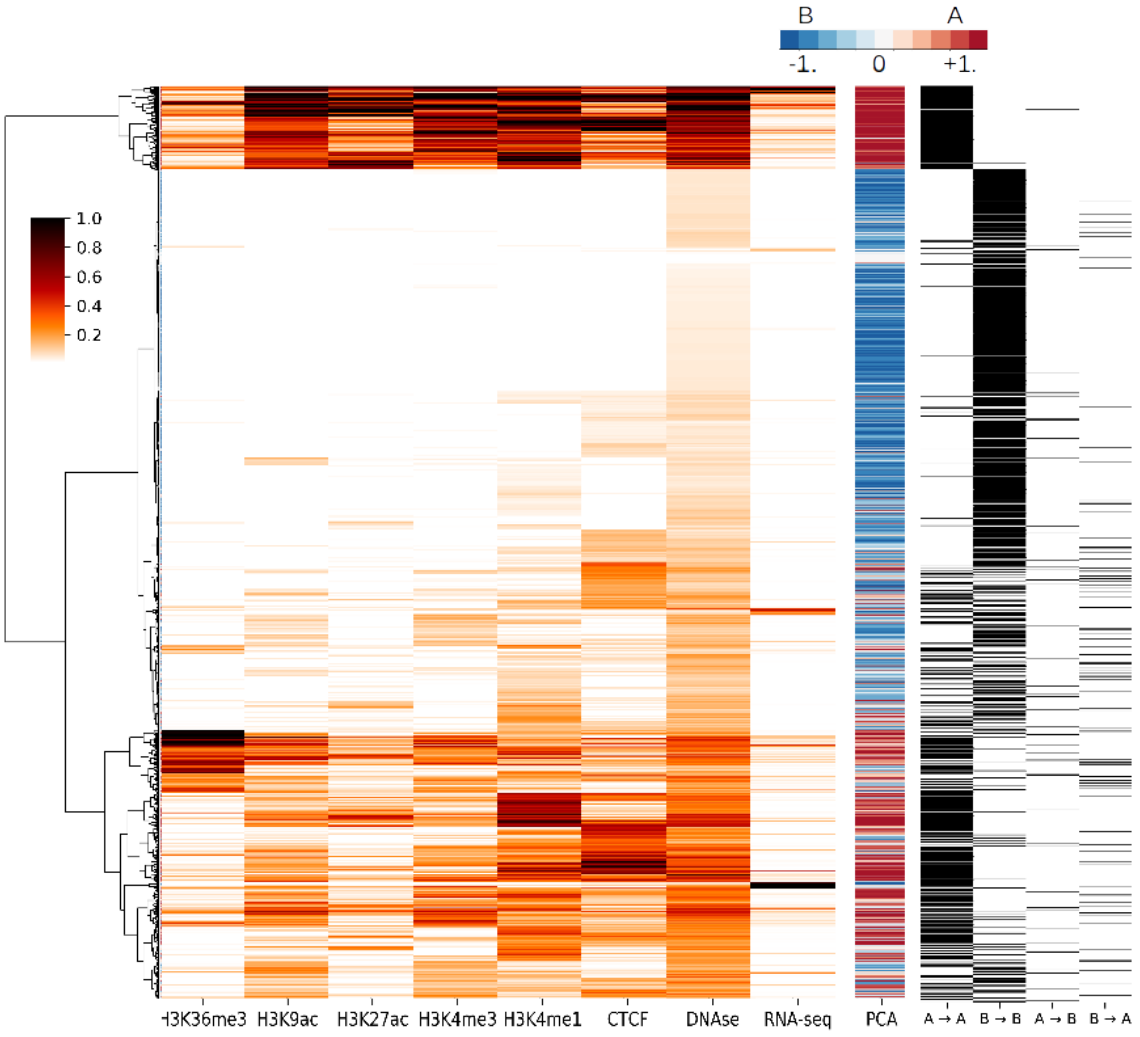

Figure 2.3: Hierarchical clustering of the epigenetic state vectors in mESC (23964 bins) with the euclidean metric and 'ward' method was performed using the Scipy package cluster.hierarchy [Jones et al., 2001, Müllner, 2011]. Each biological track was scaled between 0 and 1 with clipping of the bottom and top 1% of the values for visibility purposes. On the right of the clusters: average PCA value for the corresponding bin as outputted by the HOMER software. Further right, each bin is classified as belonging to one out four distinct categories: bins annotated as A by both HOMER and SACCSANN ($A \rightarrow A$), B by both HOMER and SACCSANN ($B \rightarrow B$), A by HOMER and B by SACCSANN ($A \rightarrow B$) or B by HOMER and A by SACCSANN ($B \rightarrow A$).

Figure 2.4). 78% of the bins are found to be annotated consistently by the three methods. On the other hand, the remaining 22% of the bins are almost evenly distributed in the other domains of the Venn diagrams, which points out that no particular method stands out compared to the others. A comparative measure to assess their respective performance one these bins is lacking.

Feature importance for compartment prediction

In an effort to understand how SACCSANN uses its input sequence features to produce compartment predictions, we correlated each of its 100 individual input features against its final prediction score. Importantly, since GC-content is a major determinant of compartments and a co-variate of many TFBS counts, we performed this analysis controlling for GC content, using partial correlations (see Methods section). It should be noted that this analysis does not directly interpret how the features are used in the model but only how they correlate to its results. Figure 2.5 (see also additional file 2) contrasts this measure of feature importance for predictors trained in human and mouse ESCs. Overall, there is a strong correlation between the way the human and mouse predictors use features (Spearman correlation coefficient=0.83, p-value $= 1.15 * 10^{-16}$), suggesting a conservation of the mechanisms of compartment establishment across species in embryonic stem cells.

This analysis reveals that SACCSANN predictors rely in part on the presence of various transposable elements to make their predictions. Notably, we observe that the Alu transposable element has the highest partial correlation in mouse (0.39) and the second in human (0.19). In mouse, Alus refer to the B1 family of TEs, which was found to be similar to primate Alus [Quentin, 1994]. In human, Alus are known to be found in gene rich regions [Natale et al., 2018], which is consistent with a positive correlation with A compartments. Moreover, Alu elements were found to play a part in regulating the expression of their neighbouring genes in human [Cordaux and Batzer, 2009] and have an impact on primate transcriptome through cis-regulation of RNA editing [Daniel et al., 2014], which supports the hypothesis of a link between Alus and regulation. Surprisingly, L1 is positively correlated with A compartment predictions in human (0.13), but negatively in mouse (-0.19).

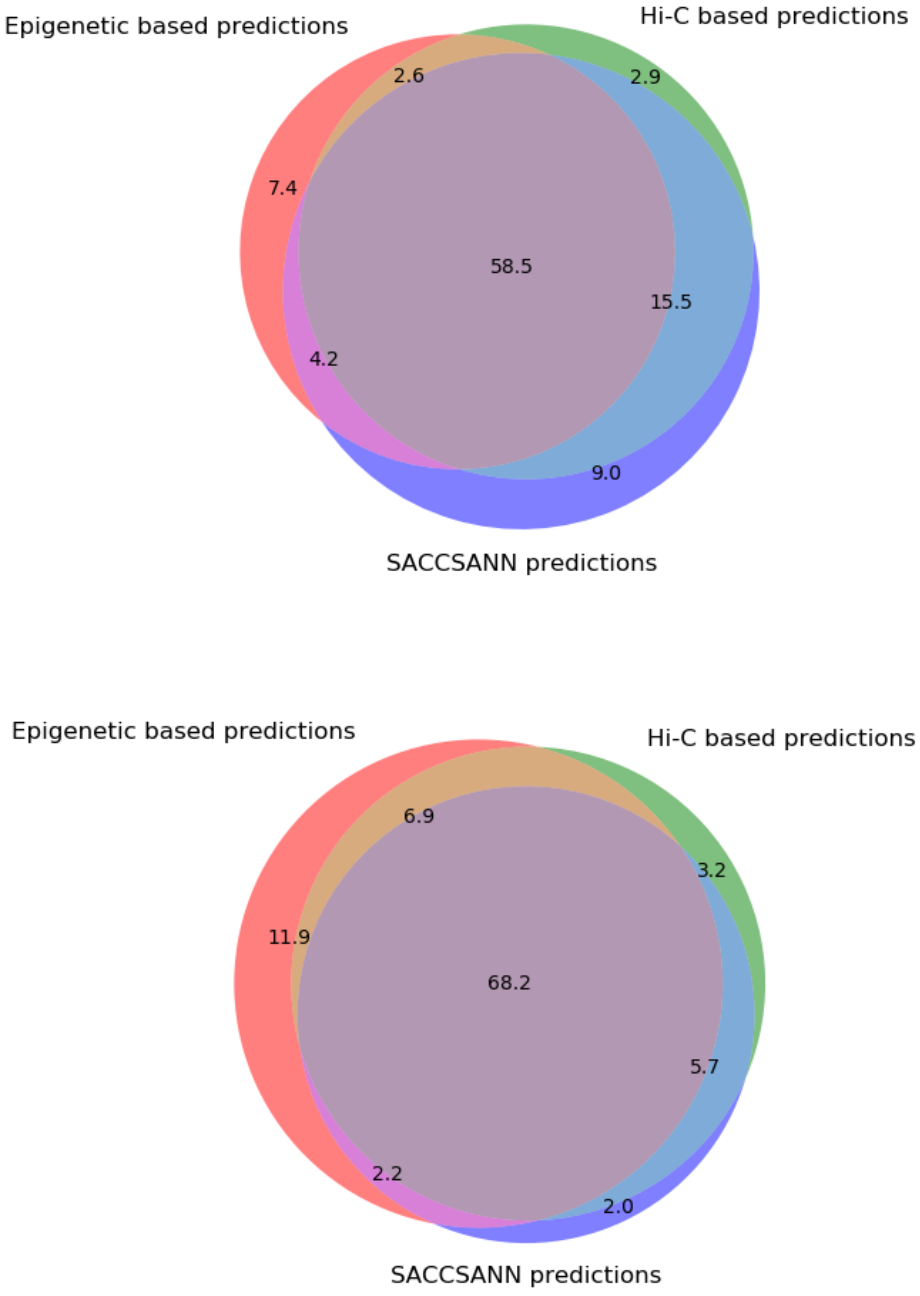

Figure 2.4: Venn diagram for the three compartment annotation methods (epigenetic state vectors clustering, Hi-C based annotations and SACCSANN annotations), produced with the Python matplotlib-venn module. Top diagram: each region corresponds to the percentage of bins classified as A by the corresponding combination of compartment annotators with respect to the total number of bins annotated as A by at least one method. Bottom diagram: same diagram for the bins annotated as being part of a B compartment.

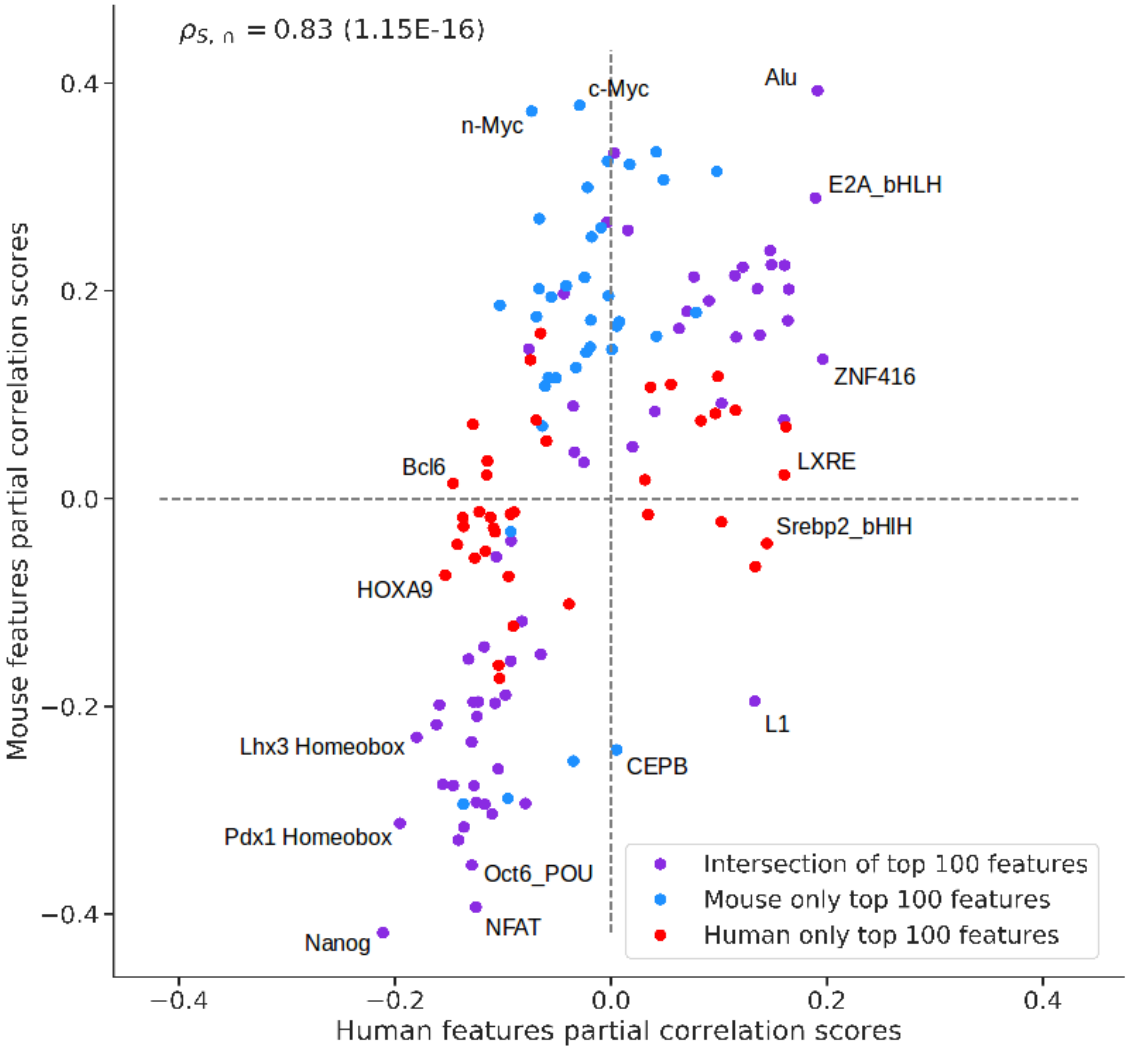

Figure 2.5: For each species, the partial correlation score is calculated between each top 100 features and A compartment predictions by SACCSANN while controlling for GC content. The correlation score at the top is the Spearman coefficient for the intersection of the top 100 features in human and mouse.

In human, Natale et al. (2018) [Natale et al., 2018] show that given their distribution in the genome, L1 and Alu elements represent chromatin regions with opposing features. Indeed, L1 elements tend to be inserted into AT rich regions of the genome while Alus prefer gene rich regions. However, since the partial correlation coefficient is calculated by controlling for GC content, this observation is not necessarily contradictory to Natale et al.'s findings.

It remains surprising that L1 behaves differently in the two species, even more with it being the only feature doing so at this magnitude.

SACCSANN also relies heavily on the presence of some transcription factor's predicted binding sites. In particular, homeobox transcription factors, including Nanog, Oct6, Pdx1 and Lhx3, are found to be negatively correlated with A compartment predictions in both mouse and human. Interestingly, Nanog was found to be enriched in B compartments [Stevens et al., 2017] and Lopes Novo et al. (2016) [Novo et al., 2016] showed that it is an important regulator of heterochromatin in mouse embryonic stem cells.

2.2.2 Compartment establishment rules are transferable

... across species The cross-species comparison of feature importance presented in the previous section prompted us to study the extent to which compartment predictive models trained on data from one species could be used to make predictions in another. Training SACCSANN to predict mouse ESC compartments yields a predictor that is accurate on both human (AUC = 80.8%) and mouse (AUC = 90.0%). Moreover, SACCSANN trained on human compartment data also performs well on the two species (AUC = 85.8% on mouse and AUC = 80.2% on human). Overall, this suggests that compartment formation rules are at least partially shared between mouse and human for ES cells and confirms the high correlation between mouse and human features found in Figure 2.5. It can also be noted that the human-trained predictor performs better on mouse (AUC = 85.8%) than on human (AUC = 80.2%) which reinforce the idea that compartments are easier to predict in mouse than in human.

Cell type specificity Several studies showed that compartments are cell-type specific [Bonev et al., 2017, Dixon et al., 2015, Lieberman-Aiden et al., 2009]. Focusing on the neural differentiation data set of Bonev et al. [Bonev et al., 2017], which mapped chromosome conformation in mouse ESC, neural progenitor cells (NPC) and cortical neurons (CN), we studied the extent to which a model trained on one cell type is applicable to an-

other (see Methods section for more details). Figure 2.6 shows that SACCSANN is able to learn cell-type specific compartment properties. Indeed, the highest AUC score of each row in Figure 2.6 is reached for the diagonal value, that is to say when the predictor is tested on the same cell-type than the one it was trained on. However, this observation does not hold for column values (ie. the predictor trained on NPC data performs better to annotate CN data than the CN trained predictor). This might be due to noise in the training CN dataset, which could prevent the model from learning the optimal set of parameters to annotate compartments. However, it does not prevent the CN predictor from extracting cell type specific information about compartments, as its best performance can be observed for the CN cell type.

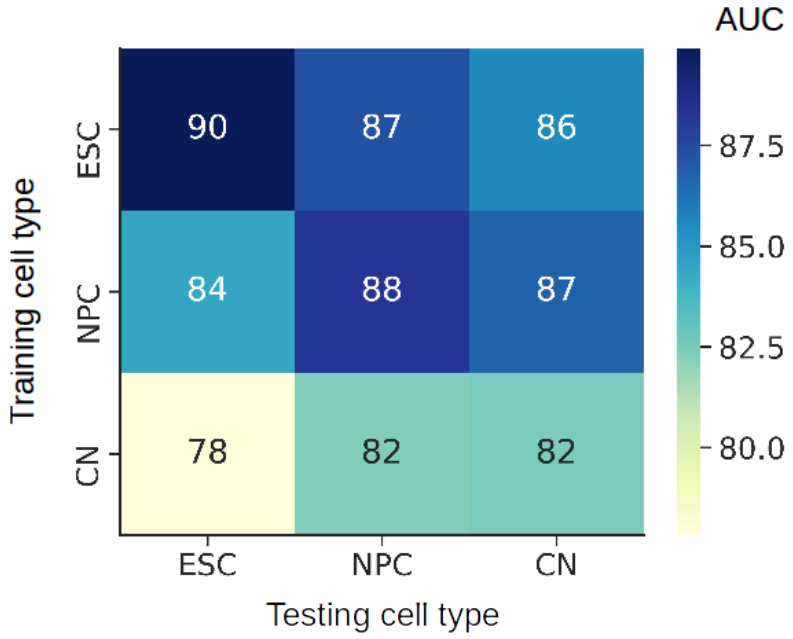

Figure 2.6: SACCSANN was trained and evaluated with chromosome-wise leave-one-out cross-validation on three cell types from Bonev neural differentiation [Bonev et al., 2017]. As expected, SACCSANN reaches the highest AUC score for he cell type it was trained on.

The cell type specificity of each predictor is also highlighted in Figure 2.7. The partial correlation score of each feature with predicted outputs can be seen to evolve across

differentiation. The heatmap also provides insights about how sequence determinants correlate with compartment predictions across differentation. For instance, the transcription factor binding sites of Nanog, Oct4 and Sox2 can be seen to either decrease or remain constant with differentiation, their highest negative correlation occurring at the embryonic stem cell stage. These three transcription factors were found to be critical for the maintenance of pluripotency in embryonic stem cells and to interact in regulatory networks [Pan and Thomson, 2007, Loh et al., 2006, J Rodda et al., 2005, Chambers et al., 2003]. The fact that Nanog and Oct4 loose importance over differentiation is consistent with these results. On the other hand, Sox2 remains highly negatively correlated across differentiation, especially in neuron progenitors, which joins studies showing it as determinant in neuron progenitors [Bani-Yaghoub et al., 2006]. Then, other features see their partial correlation score increase across differentiation. This is the case for the L1 transposable element and the TEAD transcription factor, which becomes more and more negatively correlated with A compartment annotations. On the contrary, TF binding sites like HIF2a, bHLHL E40 and Arnt become more highly positively correlated as differentiation occurs. Finally, features like the Nuclear Factor of Activated T cells (NFAT) binding sites and the E2A TF binding sites remain almost constant in their correlation score, which is consistent with their important role in brain development [Graef et al., 2003, Mackenzie and Oteiza, 2007, Hashimoto et al., 2011].

On Figure 2.6, we also observe that the overall AUC score decreases with the level of cell differentiation. Predicting compartment annotations for Cortical Neurons is harder for SACCSANN than predicting them for Embryonic Stem cells. This might be due to the fact that as differentiation takes place, epigenetic data becomes more important for the determination of compartments. As a result, it would become harder to infer compartments from sequence features alone. We further analyzed the evolution of compartment types across differentiation in Figure 2.13. As can be expected, the majority of bins remain in the same compartment type over the course of differentiation with only 24% of bins changing at least once of compartment type. Then, the performance of SACCSANN on bins evolving from B to A type compartments are the worst while the performance on those going from A to B are better. This can be related to the observation by Bonev et al. (2017) [Bonev

2.2 Results

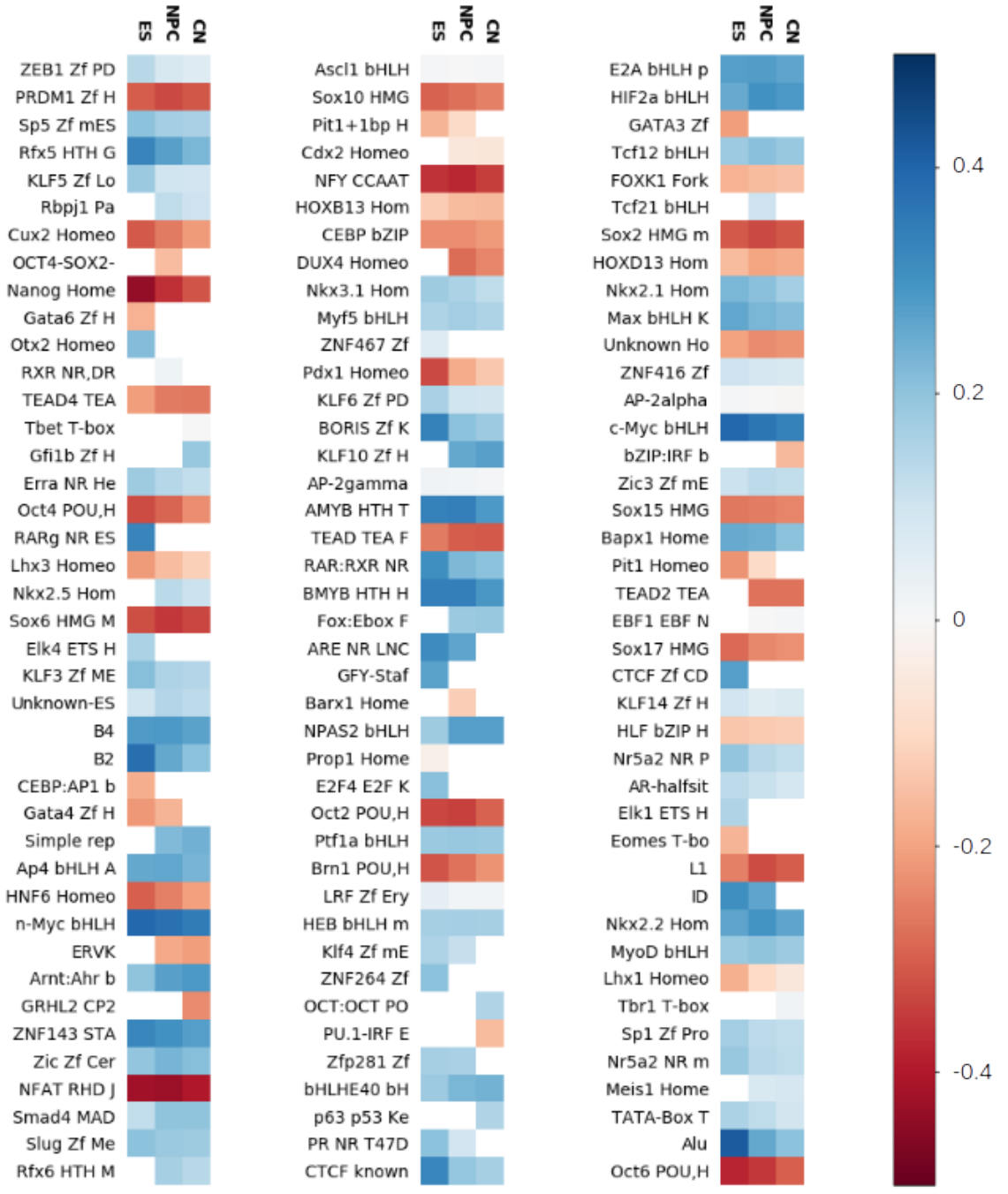

Figure 2.7: Partial correlation score of each of the top 100 features selected in each cell type for mouse ESC, NPC and CN. White entries correspond to features that were not selected in the top 100 of the corresponding cell type.

et al., 2017] that interaction strength between A type compartments decreased with differentiation while the interaction strength between B type compartments increases. Another element of explanation is that each of these categories contains very few examples compared to the bins that do not change annotation, making it harder for the predictor to pick up the patterns. Moreover, the corresponding PCA values tend to be smaller in the changing bins, implying that the confidence in the compartment annotations by HOMER is also low. It is important to note that the overall AUC score still remains higher than 80% for all cell types experiments were performed on.

... and across chromosomes Another interesting question is whether the organizational principles that guide compartment formation are the same across chromosomes. So far, SACCSANN has been trained on all-but-one chromosomes for predictions on the left-out chromosome. Here, we trained SACCSANN on individual chromosomes for annotations on all the others. Despite an important decrease in training data, the resulting AUC scores are surprisingly high (89.8% for mESC, 86.8% for NPC and 76.0% for CN in the Bonev dataset [Bonev et al., 2017]). The first insight from this experiment is that a set of compartment formation rules can be learned at the chromosome scale, i.e that the rule set is mainly encompassed within single chromosomes.

We represent the results of this experiment as a heatmap (see Figure 2.8). Similarly to what was observed in Figure 2.6, compartments become harder to predict over the course of mouse neural differentiation.

The heatmap's structure highlights differences as well as similarities in behaviour between chromosomes. In ESC, models trained on individual chrosomomes are all performing similarly well, although we observe that certain chromosomes (e.g. 1 and 13) are slightly harder to predict on. As differentiation proceeds toward NPC and CN, overall prediction accuracies decrease, and certain chromosomes produce results that stand out. For instance, in CN, chromosome 13 becomes both a poor training data set and is poorly predicted from models trained on other chromosomes. However, looking a bit deeper into

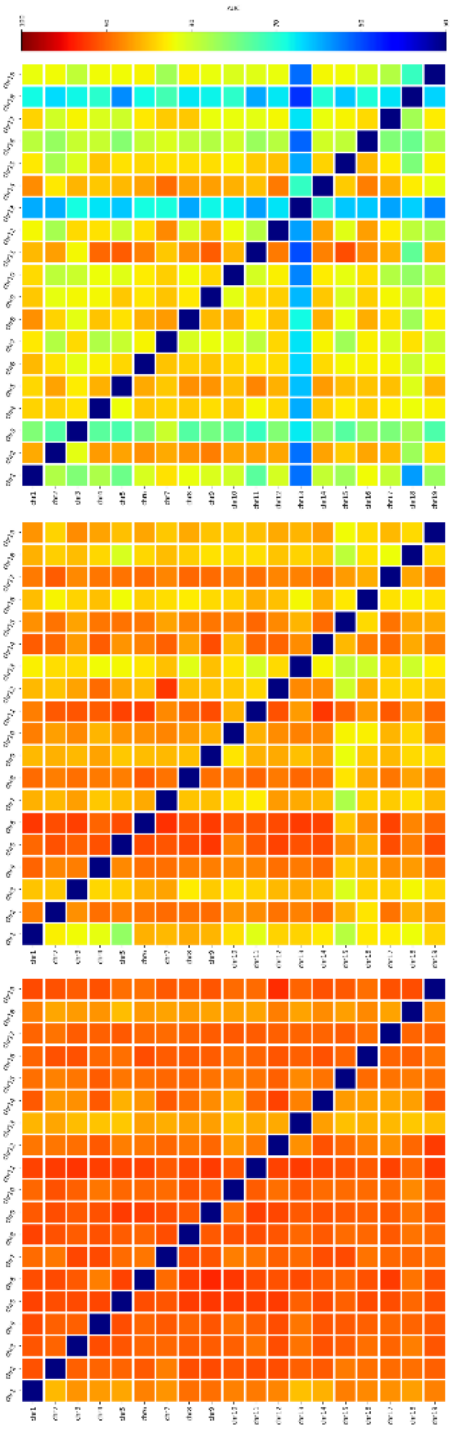

Figure 2.8: From left to right: heatmap for ESC, Neuron Progenitors (NPC) and Cortical Neurons (CN). Each entry is the AUC score achieved by SACCSANN trained on the corresponding row chromosome and tested on the corresponding column chromosome. The diagonal is left blank.

this chromosome did not highlight distinct characteristics in terms of compartment lengths and number. Interestingly, other chromosomes such as 3 and 18 become harder to predict but are still relatively useful as training data. One hypothesis for this observation is that SACCSANN learns a set of useful patterns for compartments formation in these chromosomes but that other important pattern present in these chromosomes are not found in the remaining ones. We can relate this to the finding of a sixth sub-compartment by Rao et al. (2014) [Rao et al., 2014] present only on human chromosome 19 and the observation that the corresponding human heatmap (see Figure 2.14) shows that chromosome 19 is poorly predicted. The results of Zhang et al. (2018) [Zhang et al., 2018b], who found chromosome 19 to be the poorest chromosome predictor of Hi-C contacts in five distinct human cell types, also go in that direction.

2.3 Discussion and Conclusions

In this work, we show that chromatin A and B compartments can be accurately predicted from sequence-level features alone. Our model is robust across different cell types and species and allows us to derive key sequence determinants defining A/B compartments, such as Alu transposable elements and the Nanog transcription factor binding sites in embryonic stem cells. A recent study by Roychowdhury et al. (2019) [Roychowdhury and Abyzov, 2019] hypothesizes that the enrichment of Alus in A compartments is linked to a stabilization role of Alus for DNA repair in open chromatin, attributing to this transposable element a key link with compartment establishment. Moreover, we found that compartment annotations could be inferred between human and mouse with only minor losses in performance, which points toward an evolutionary conservation of compartment establishment rules. This result is encouraging for the application of our method to evolutionary studies where only the inferred DNA sequence of ancestral species is available. Indeed, since our method only takes sequence features as input, it is possible to apply it to computationally inferred ancestral genomes. Moreover, the observed similarities in mouse and humans points out that in addition to being possible, such a study would probably be insightful given the similar feature behaviours, at least up to a certain point in the phylogenetic tree. Finally, we observe that SACCSANN can produce accurate compartment annotations with

training on individual chromosomes, implying that partial compartment formation rules can be derived at the chromosome scale.

While SACCSANN produces accurate compartment annotations for a broad range of data sets, further understanding of the biological processes underlying compartment formation would require a more in-depth study of the emerging key features of our current model. Then, it might also be interesting to augment our model with the use of Convolutional Neural Networks (CNNs). CNNs were primarily designed for image recognition and classification but their application to sequence classification problems is growing. They intuitively consist of the application and combination of several filters to an image or a sequence, where each filter can be interpreted as a detector for a specific feature. Although more computationally expensive, this type of network could take as input the raw string of nucleotides making up DNA sequence instead of pre-determined sequence pattern counts. By not constraining the initial pool of features, such an approach might lead to the discovery of new relevant sequence determinants. Similar approaches have been used successfully to predict noncoding-variant effect with the DeapSEA software [Zhou and G Troyanskaya, 2015] or DNA accessibility in Basset [Kelley et al., 2016] for instance. In both models, the authors stack convolutional layers to analyze the genome.

Improving the cell-type specificity of our model is also a point for future work. Indeed, compartment annotations on bins that change compartment type over the course of mouse neural differentiation were not always very accurate. One hypothesis to account for this result is that the bins that change compartment during differentiation have specific characteristics compared to the bins that remain unchanged. To describe these bins, a more complex model might be needed along with more training data. This could be achieved by combining data from different species for a given cell type, even more so given that we showed that different species (here mouse and human) share compartment establishment rules.

Establishing cell-type specific models of A/B compartments enables predictions for sequenced genomes without available such as computationally reconstructed ancestral genomes,

leading to a valuable insight into their gene expression profile. This work paves the way for applications to disease states such as cancer or to ancestral genomics where only the DNA sequence is available. In cancer for instance, we could hope that the changes in the the DNA sequences could be linked to the three-dimensional organization of these genomes.

2.4 Methods

Data sources

The mm10 genome assembly was used for experiments on mouse and the hg19 assembly for experiments on human cell types. Computational transcription factor binding site (TFBS) prediction was performed by HOMER [Heinz S., 2010] using HOMER's 'known_motifs' collection of position weight matrices for vertebrates. Repeat Masker (http://repeatmasker.org/) TEs annotations were obtained from the UCSC Genome Browser [W. James Kent and Haussler, 2002].

Table 2.1 lists the different Hi-C data sets used in this study:

Table 2.1: Hi-C Data sets. ESC: Embryonic Stem Cells, NP: Neuron Progenitor, CN: Cortical Neurons.

Reference	Cell type	Restriction enzyme	GEO
[Dixon et al., 2012]	mESC, hESC	HindIII	GSE35156
[Fraser et al., 2015]	mESC, NPC, Neurons	HindIII, NcoI	GSE59027
[Bonev et al., 2017]	mESC, NPC, CN	DpnII	GSE96107

The raw sequencing reads obtained from published Hi-C datasets were processed using the publicly available Hi-C User Pipeline HiCUP [Wingett et al., 2015].

For compartment annotations, we used both published annotations and annotations obtained with the specialized Hi-C programs in HOMER [Heinz S., 2010] on the aforementioned Hi-C contact maps. For the annotations by Bonev et al. (2017) [Bonev et al., 2017],

the entire genome was not annotated so we were not able to use it for fair evaluation of our model, which is why we did not include the results on this data set in this study. For the HOMER annotations, we used the following scripts successively (unless otherwise stated, the default parameters of the software were used):

- makeTagDirectory with the parameters -mapq 30 (keep a read pair if there is a single best alignment based on mapq with minimum 30 value), -tbp 1 (maximum tags per base pair, 1 is the advised value in the HOMER documentation which means only keeping read-pairs with the exact same ends once, the others most likely being clonal). A filtered directory of high-quality reads is then created, restricting for the restriction enzyme used in the experiment and using the default settings indicated in the HOMER documentation.
- analyzeHiC to create a background model at the wanted resolution (here 100kb) for normalization of the Hi-C data.
- runHiCpca to perform PCA on the normalized Hi-C matrix, while specifying the reference genome (here mm10 or hg19 depending on the species).
- findHiCCompartments to output the coordinates of compartments according to the previous PCA analysis of the Hi-C contact map.

For the production of Figure 2.3, we used the peak ChIP-seq tracks from ENCODE [ENCODE Project Consortium, 2012] and data sources listed in Table 2.2. When the alignment was to a different genome assembly than mm10, the liftOver tool (https://genome-store.ucsc.edu/) was used to map the peaks accordingly.

2.4.1 Predictive Model

The model we propose, SACCSANN (Sequence-based Annotator of Chromosomal A/B Compartments by Stacked Artificial Neural Networks), is a combination of two fully-connected artificial neural networks (ANNs). The first ANN assigns the probability of a given n-sized block (n is also called the resolution) of the genome of belonging to the A compartment. This network takes as input features representing GC content, computationally predicted transcription factors binding site (TFBS) counts and transposable elements

counts (TEs) found within the block. Prior to entering the network, all features are normalized with centering of the mean and scaling of the variance to unit value using the scikit learn preprocessing package [Pedregosa et al., 2011]. Cell-type specific A/B compartment labels are computationally generated from Hi-C data (see Data Sources section) and used to train the model. Since compartments have an average size of over 1Mb and that we are performing experiments at a resolution of 100kb, most compartments span over several neighbouring genomic bins. This is why a second ANN is then applied to the output of the first, in order to smooth compartment predictions. More precisely, the second network takes as input the predictions of the first network for the current bin and for a fixed number of its previous and next neighbors.

The two ANNs were implemented with the scikit-learn neural_network package [Pedregosa et al., 2011]. The sigmoid function was used as activation function for the hidden layers, the softmax function as output function and cross-entropy as the loss function for each network. The hyper-parameter values of the networks are detailed in Table 2.3.

The final architecture takes 100 features as input and the smoothing network looks at $1,000$kb on each side of the current bin to make the final prediction. The actual number of neighbors w then depends on the resolution used in the experiment (10 on each side for a resolution of 100kb for instance).

2.4.2 Training

For each experiment, the data is separated into a training and a testing set. Except for the individual chromosome training experiments, the architecture is trained with chromosome-wise leave-one-out cross-validation, i.e the architecture is trained on all chromosomes but one and tested on the left-out chromosome and this process is repeated such that each chromosome is the test chromosome once. In the case of individual chromosome training, the model is trained on a single chromosome and predictions are made on all the others. As the number of A and B bins is not balanced in all cell types we experimented on, the

Table 2.2: Data sources.

Name	GEO	Source
H3K36me3 ChIP-seq	GSM1000109	ENCODE
H3K9ac ChIP-seq	GSM1000127	ENCODE
H3K27ac ChIP-seq	GSM1000099	ENCODE
H3K4me3 ChIP-seq	GSM769008	ENCODE
H3K4me1 ChIP-seq	GSM769009	ENCODE
CTCF TFBS ChIP-seq	GSM918748	ENCODE
DNAse Hypersensitivity	GSM1014154	ENCODE
RNA-Seq	GSE96107	[Bonev et al., 2017]

Table 2.3: Final hyper-parameters for SACCSANN. IN: Intermediate Network, SN: Smoothing Network. α: regularization rate. The initial learning rate is the one passed to the Adam optimizer for training.

Parameter	IN value	SN value
Nb features	100	.
Nb neighbors	.	1000kb
Nb hidden layers	1	2
Nb nodes per layer	256	64
α	0.001	0.001
Initial learning rate	0.0001	0.01

training data set is balanced by random downsampling prior to training SACCSANN. The two networks composing SACCSANN are trained separately, each by backpropagation to minimize the loss function and using the Adam optimizer [Kingma and Ba, 2014] and L2 regularization to avoid overfitting.

2.4.3 Feature Selection

We applied our model to both the human and mouse genomes. As inputs, we used features derived from the DNA sequence of the species being studied: binding site counts for 334 Transcription Factors (TFs) for the human and mouse genomes, 35 Transposable Elements (TEs) for the mouse genome and 41 for the human genome, as well as the GC content of the sequence. This results in 370 features for the mouse genome and 376 for the human genome. To avoid the use of redundant features and limit overfitting, we ranked the features according to a Random Forest (RF) classifier and used the top ones as outputted by cross-validation on each data set as input to the first network. This step is performed using the feature_impotances_ method of the scikit learn ensemble Random Forest package [Pedregosa et al., 2011]. Briefly, this implementation combines the expected fraction of samples a feature contributes to classifying with the decrease in impurity reached by adding this feature to the tree, also known as the mean decrease in impurity method [Louppe, 2014], to get an estimate of the relative feature's importance in a tree. These feature importances are then averaged over all the decision trees composing the forest in order to obtain a more reliable estimate of a feature's predictive power. The number of selected features is one of the tuned hyper-parameters of the model. As an example, the AUC score goes from 88.2% when using all the features to 90% when only using the top 100 in mouse embryonic stem cells.

2.4.4 Hyper-parameter Tuning

We used the Bayesian optimization software Spearmint [Snoek et al., 2012] to tune the hyper-parameters of the model, ie the number of selected features, the initial learning rate, the L2 regularization rate, the number of hidden layers and the number of nodes per hid-

den layer in each ANN as well as the number of neighbors to take into account for the smoothing network. Since the number of parameters is high, we divided the tuning task into two separate optimization problems. We first tuned the parameters of the intermediate network and used the resulting parameters to then tune the smoothing network. For the first task, we ran Spearmint for 400 iterations with 80 random starts on two mouse data sets: ESC from Dixon et al. 2012 [Dixon et al., 2012] and Cortical Neurons from Bonev et al. (2017) [Bonev et al., 2017]. For the smoothing network tuning, we ran 200 iterations of Spearmint with 40 random starts on the same data sets. For each iteration, we used 5-fold cross-validation over the entire data set to estimate the performance of the current model. The results of Spearmint can be found in the additional file 3. For most parameters, the optimal value found by Spearmint was approximatively the same for both data sets (since the parameter space is large, not exactly the same parameter values were tested in each data set). For the number of nodes per layer, we opted for the nearest power of two. In the case where they differed, we found that these parameters did not influence the model's performance greatly and hence arbitrarily chose a consensus value. The resulting architecture was then applied to the other data sets to confirm their optimality. See Table 2.3 for final results.

2.4.5 Performance Metrics

We used the integral of the Receiving Operating Curve (ROC), also called Area Under the Curve (AUC) score as a performance measure. For binary classification, the ROC curve represents the rate of true positives against the rate of false positives (see Figure 2.9 for an example). A perfect AUC score is 100% while a random classifier would achiever a score of 50%.

2.4.6 Partial Correlation Scores

For Figure 2.5, we measured the degree of association between each input feature to SACC-SANN and the outputted prediction of the corresponding genomic bin to be in an A compartment. Since we believe that the features are driven by GC content, we calculated the partial correlation of these two variables while controlling for GC content.

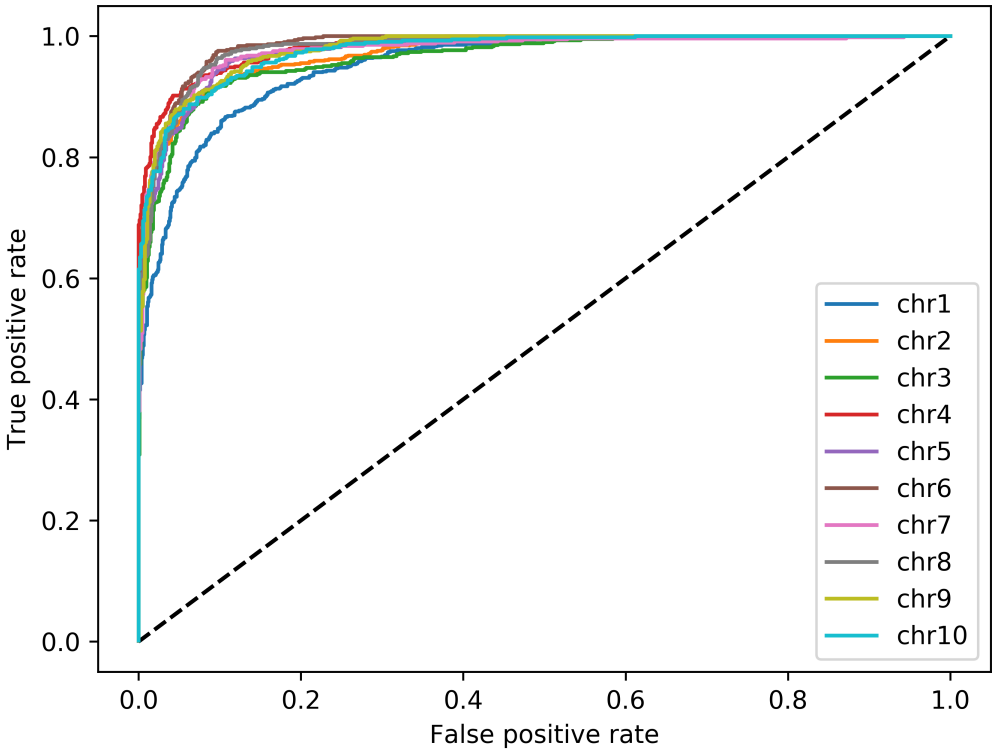

Figure 2.9: SACCSANN ROC curve for the first 10 chromosomes in mouse embryonic stem cells. Dashed line: expected performance of a random binary classifier.

The partial correlation score of two variables X and Y while controlling for variable Z is calculated by correlating the residuals of the regression between X and Z and the residuals of the regression between Y and Z. We used linear regression between the features and GC content to calculate the first residuals and logistic regression for those between A compartment predictions and GC content. Indeed, as A compartment predictions are probabilities of the bin belonging to an A type compartment, logistic regression is better able to capture the relationship between GC content and A compartment predictions than linear regression (see Figure 2.15). We then used Spearman coefficient to correlate the residuals.

Competing interests

The authors declare that they have no competing interests.

Author's contributions

J.A.P. contributed to the design of the study, implemented the computational analyses, and wrote the manuscript. C.J.F.C. conceived the study, contributed to the study design, helped with data generation and contributed to the manuscript. M.B. conceived the study, coordinated the computational analysis, and helped draft the manuscript. All authors have reviewed and approved the final manuscript.

Acknowledgements

We thank Jacek Majewski and Stefan Kremer for their valuable and constructive feedback on this study. For insightful conversations, we also thank Faizy Ahsan, Alexander Butyaev, Vincent Mallet, Samy Coulombe and ZiChao Yan.

2.5 Supplementary Figures

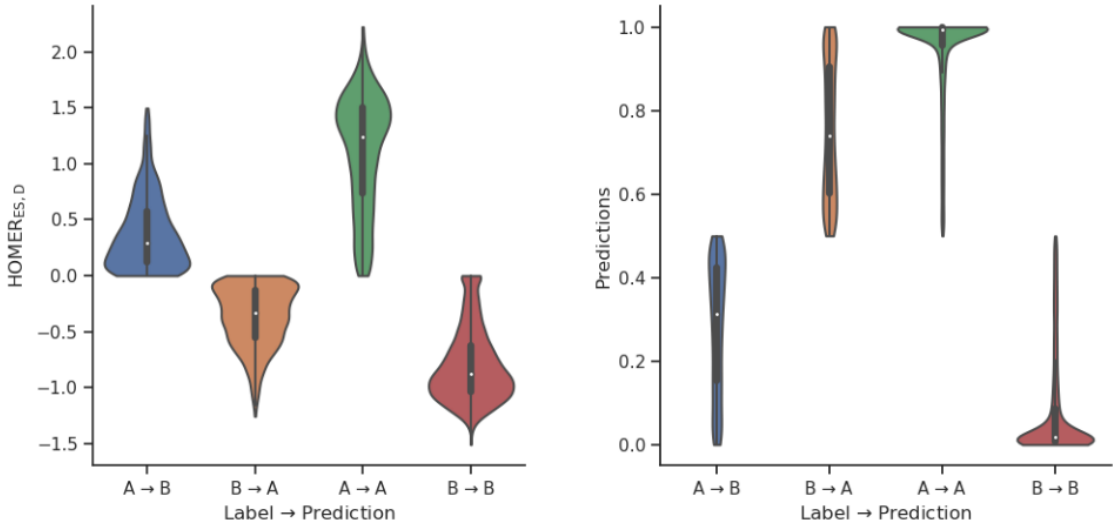

Figure 2.10: Error Analysis. On the x axis, each compartment is classified as belonging to one out four distinct categories: compartments annotated as A by both HOMER and SACCSANN ($A \rightarrow A$), B by both HOMER and SACCSANN ($B \rightarrow B$), A by HOMER and B by SACCSANN ($A \rightarrow B$) or B by HOMER and A by SACCSANN ($B \rightarrow A$). On the left: PCA value outputted by HOMER for each category. On the right: Probability of each compartment being A type according to SACCSANN.

2.5 Supplementary Figures

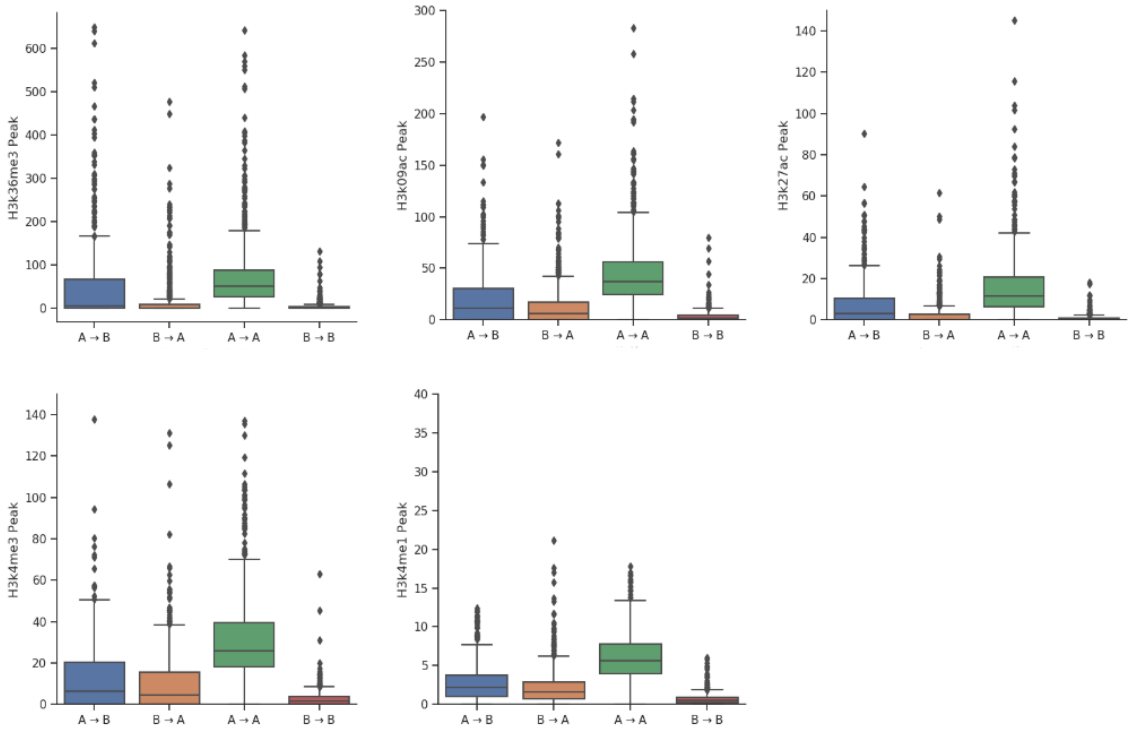

Figure 2.11: Histone marks repartition. Same x axis distribution than in Figure 2.10.

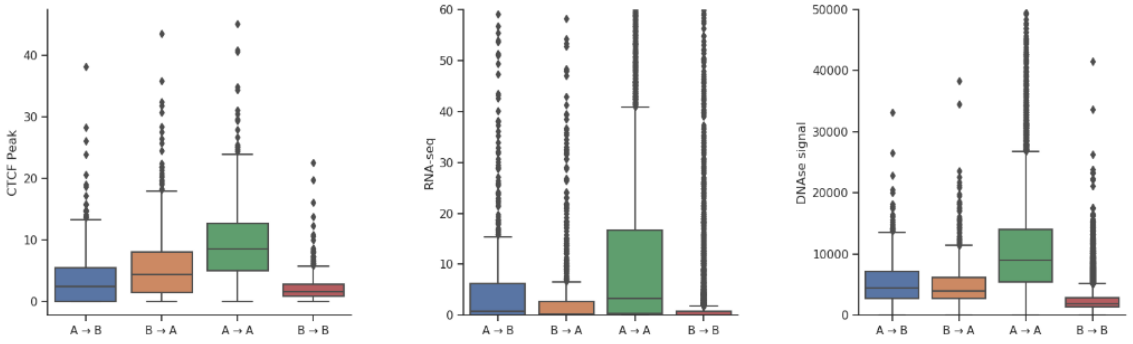

Figure 2.12: External data repartition. Same x axis distribution than in Figure 2.10.

2.5 Supplementary Figures

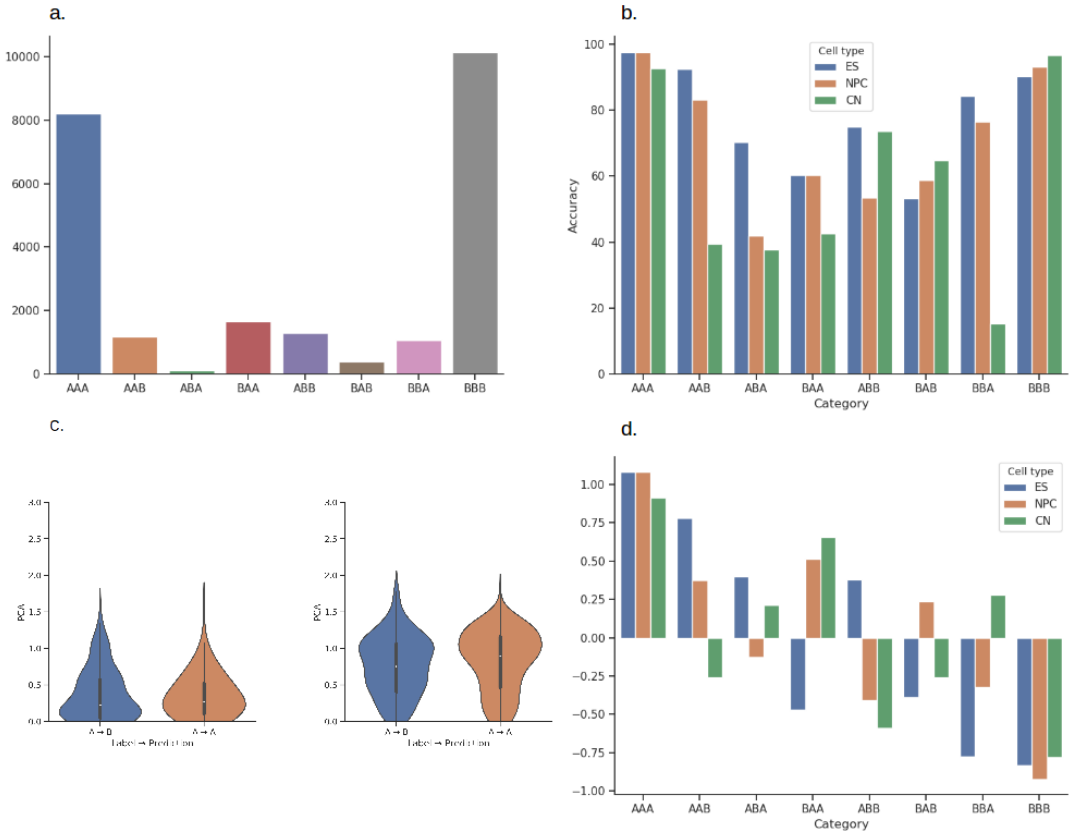

Figure 2.13: Compartment analysis across mouse neural differentiation. a: number of genomic bins as a function of compartment type where on the x-axis XYZ, X is the compartment type in ESC, Y in NPC and Z in CN. b, d. SACCSANN accuracy and HOMER PCA values as a function of compartment type and cell type. c: HOMER PCA value repartition for categories BBA (left) and BAA (right) in cortical neurons (CN). x axis: compartments annotated as A by both HOMER and SACCSANN ($A \rightarrow A$) and compartments annotated as A by HOMER and B by SACCSANN ($A \rightarrow B$).

2.5 Supplementary Figures

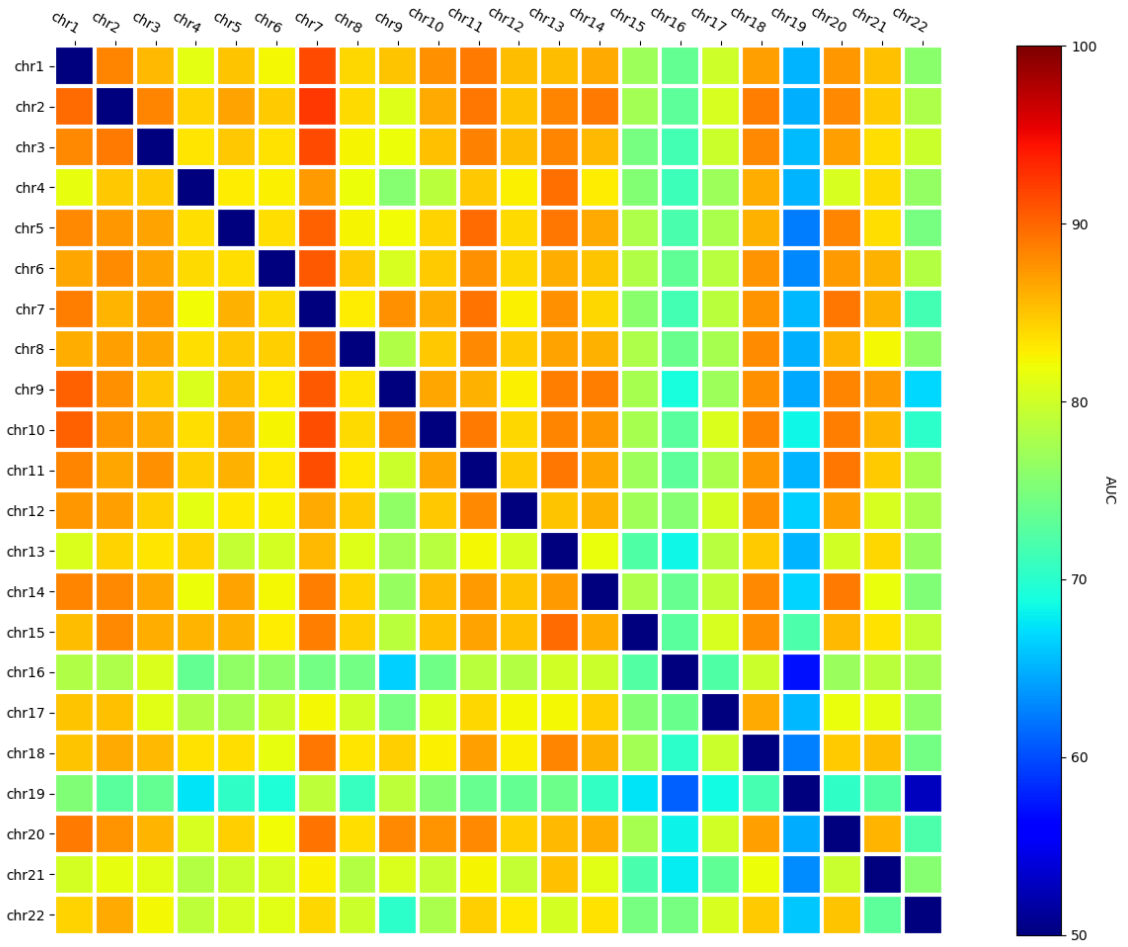

Figure 2.14: Individual chromosome training for human Embryonic Stem Cells. The row chromosomes are the training data while the column chromosomes are the testing data. The diagonal is intentionally left blue to avoid training and testing on the same chromosomes.

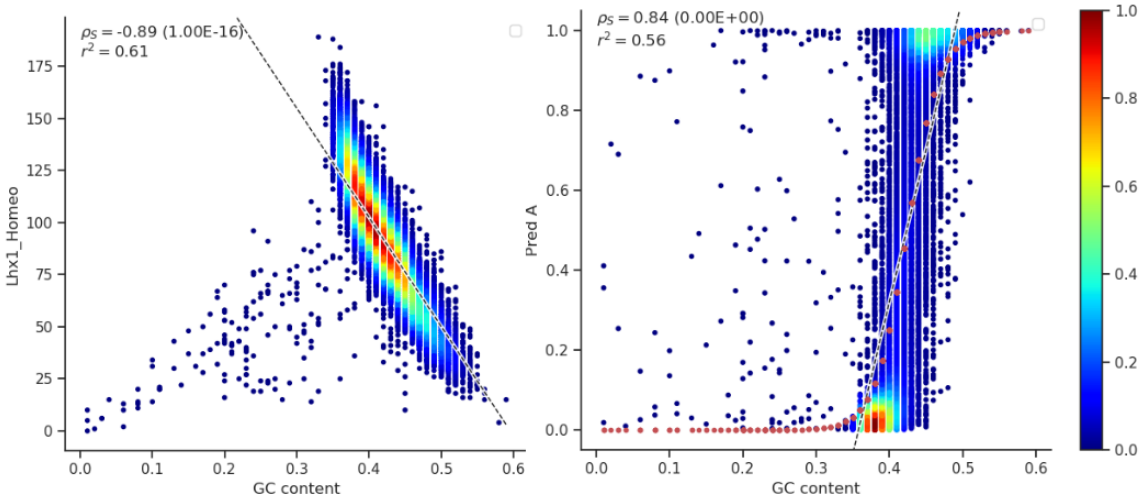

Figure 2.15: Variable correlations in mouse embryonic stem cells. Right: correlation between GC content and Lhx1 transcription factor binding site counts. Left: correlation between GC content and A compartment prediction. Dashed line: linear regression between the y-axis variable and GC content. Red dotted line: logistic regression between A compartment predictions and GC content.

2.6 Additional file 1: Sequence determinants partial correlation scores

Table 2.4: Model features partial correlation scores in mouse and human embryonic stem cells

Feature name	Score in mouse	Score in human
ZEB1_Zf_PD	0.17	0.16
KLF5_Zf_Lo	0.19	0.09
Cux2_Homeo	-0.28	-0.15
Nanog_Home	-0.42	-0.21
Gata6_Zf_H	-0.20	-0.13
Otx2_Homeo	0.20	-0.04
Erra_NR_He	0.21	0.08
Oct4_POU,H	-0.30	-0.11
Lhx3_Homeo	-0.23	-0.18
Sox6_HMG_M	-0.32	-0.14
KLF3_Zf_ME	0.20	0.14
CEBP:AP1_b	-0.21	-0.12
Gata4_Zf_H	-0.23	-0.13
HNF6_Homeo	-0.27	-0.16
NFAT_RHD_J	-0.39	-0.13
FOXK1_Fork	-0.19	-0.10
Ascl1_bHLH	0.05	0.02
Sox10_HMG	-0.29	-0.12
Pit1+1bp_H	-0.15	-0.13
NFY_CCAAT	-0.33	-0.14
HOXB13_Hom	-0.14	-0.12
Phox2a_Hom	-0.06	-0.11
Nkx3.1_Hom	0.16	0.12
Pdx1_Homeo	-0.31	-0.20

2.6 Additional file 1: Sequence determinants partial correlation scores

KLF6_Zf_PD	0.16	0.06
AP-2gamma	0.04	-0.03
AMYB_HTH_T	0.27	0.00
BMYB_HTH_H	0.26	0.02
Barx1_Home	-0.12	-0.08
Prop1_Home	-0.04	-0.09
Oct2_POU,H	-0.29	-0.08
Ptf1a_bHLH	0.22	0.12
Brn1_POU,H	-0.29	-0.12
LRF_Zf_Ery	0.08	0.04
E2A_bHLH	0.08	0.16
Klf4_Zf_mE	0.16	0.14
HEB_bHLH_m	0.20	0.16
CTCF_known	0.33	0.00
E2A_bHLH_p	0.29	0.19
HOXD13_Hom	-0.16	-0.09
Nkx2.1_Hom	0.21	0.11
Unknown_Ho	-0.20	-0.11
ZNF416_Zf	0.13	0.20
AP-2alpha	0.03	-0.03
Sp5_Zf_mES	0.18	0.07
Sox15_HMG	-0.26	-0.10
Bapx1_Home	0.22	0.16
Pit1_Homeo	-0.22	-0.16
Sox17_HMG	-0.28	-0.13
KLF14_Zf_H	0.09	0.10
HLF_bZIP_H	-0.15	-0.06
AR-halfsit	0.14	-0.08
Slug_Zf_Me	0.23	0.15
Eomes_T-bo	-0.20	-0.12
L1	-0.19	0.13
Nkx2.2_Hom	0.24	0.15

2.6 Additional file 1: Sequence determinants partial correlation scores

Lhx1_Homeo	-0.20	-0.16
TATA-Box_T	0.09	-0.04
Alu	0.39	0.19
Oct6_POU,H	-0.35	-0.13
Olig2_bHLH	-0.05	-0.12
OCT4-SOX2	-0.07	-0.09
Mef2c_MADS	0.08	0.10
Hoxc9_Home	-0.04	-0.14
Tbet_T-box	-0.10	-0.04
AP-1_bZIP	-0.03	-0.11
Znf263_Zf	-0.02	0.03
Nkx2.5_Hom	0.08	0.08
Klf9_Zf_GB	0.12	0.10
Simple_rep	0.07	-0.13
Pax8_Paire	0.09	0.12
Srebp2_bHL	-0.04	0.14
MIR	-0.06	-0.13
Atf3_bZIP	-0.02	-0.11
THRa_NR_C1	0.07	0.16
E2F6_E2F_H	0.11	0.04
Cdx2_Homeo	0.04	-0.11
ERVL-MaLR	-0.03	-0.14
Bcl6_Zf_Li	0.01	-0.15
STAT6_Stat	-0.16	-0.10
Foxa2_Fork	-0.03	-0.11
FoxL2_Fork	0.06	-0.06
Srebp1a_bH	-0.07	0.13
Mef2a_MADS	-0.02	0.10
E2F3_E2F_M	0.11	0.06
LXRE_NR_,D	0.02	0.16
OCT:OCT_PO	-0.01	-0.09
PU.1-IRF_E	-0.17	-0.10

2.6 Additional file 1: Sequence determinants partial correlation scores

Rbpj1_Pa	-0.02	-0.14
Foxf1_Fork	0.08	-0.07
FOXA1_Fork	0.16	-0.07
BATF_bZIP	-0.01	-0.12
ZNF322_Zf	0.02	0.03
FOXM1_Fork	0.13	-0.07
Six2_Homeo	-0.12	-0.09
HOXA9_Home	-0.07	-0.15
L2	-0.01	-0.09
CDX4_Homeo	0.02	-0.12
PRDM1_Zf_H	-0.29	-0.10
Rfx5_HTH_G	0.31	0.05
SF1_NR_H29	0.20	0.00
NeuroD1_bH	0.12	-0.05
RARg_NR_ES	0.33	0.04
Unknown-ES	0.11	-0.06
Ap4_bHLH_A	0.25	-0.02
n-Myc_bHLH	0.37	-0.07
Arnt:Ahr_b	0.17	0.01
ZNF143_STA	0.32	0.02
Zic_Zf_Cer	0.20	-0.07
Smad4_MAD	0.13	-0.03
Rfx6_HTH_M	0.14	-0.02
CEBP_bZIP	-0.24	0.00
Myf5_bHLH	0.14	0.00
ZNF467_Zf	0.07	-0.06
BORIS_Zf_K	0.33	0.00
TEAD_TEA_F	-0.25	-0.03
RAR:RXR_NR	0.32	0.10
ARE_NR_LNC	0.30	-0.02
NPAS2_bHLH	0.18	-0.07
E2F4_E2F_K	0.18	0.08

2.6 Additional file 1: Sequence determinants partial correlation scores

ZNF264_Zf	0.19	-0.06
Zfp281_Zf	0.15	-0.02
PR_NR_T47D	0.19	-0.10
Tcf12_bHLH	0.17	-0.02
Sox2_HMG_m	-0.29	-0.14
Max_bHLH_K	0.27	-0.07
c-Myc_bHLH	0.38	-0.03
Zic3_Zf_mE	0.12	-0.06
EBF1_EBF_N	-0.03	-0.09
CTCF_Zf_CD	0.26	-0.01
Nr5a2_NR_P	0.21	-0.02
MyoD_bHLH	0.17	0.01
Sp1_Zf_Pro	0.16	0.04
Nr5a2_NR_m	0.21	-0.04

2.7 Additional file 2: Hyperparameter Tuning

All the experiments were performed at 100kb. For the optimization of the intermediate network, each number of features was optimized during 100 iterations. For the smoothing network, 200 iterations were performed, we only report the most successful ones in terms of AUC.

Table 2.5: Spearmint summary on Dixon [Dixon et al., 2012] mouse embryonic stem cells for the intermediate network. α: L2 regularization parameter.

N features	N layers	N nodes	α	Learning rate	AUC (%)
20	2	23	0.001	0.01	86.8
40	2	294	0.001	0.0001	86.8
100	2	268	0.001	0.001	87.5
150	2	257	0.00001	0.0001	87.5

Table 2.6: Spearmint summary on Dixon [Dixon et al., 2012] mouse embryonic stem cells for the smoothing network.

N neighbors	N layers	N nodes	α	Learning rate	AUC (%)
20	2	55	0.001	0.001	89.5
10	2	17	0.00001	0.01	89.4

Table 2.7: Spearmint summary on Bonev [Bonev et al., 2017] mouse cortical neurons for the intermediate network.

N features	N layers	N nodes	α	Learning rate	AUC (%)
20	1	146	0.0001	0.01	77.6
40	1	265	0.0001	0.0001	78.2
100	1	285	0.001	0.0001	78.5
150	1	240	0.00001	0.0001	78.4

Table 2.8: Spearmint summary on Bonev [Bonev et al., 2017] mouse cortical neurons for the smoothing network.

N neighbors	N layers	N nodes	α	Learning rate	AUC (%)
12	2	65	0.001	0.01	80.7
14	2	10	0.00001	0.01	80.7

3

Summary and Conclusion

Spatial genomic organization plays a key role in gene regulation. The introduction of technologies such as Hi-C has allowed increasing insights into the 3D properties of genomes and cell types. For instance, chromatin A and B compartments were found to divide the genome into active and inactive regions. This new flow of large scale data enables many computational approaches to study and model the genome's organization in 3D, including machine learning ones. While different studies have been made to infer elements of chromatin spatial architecture, there is a lack of methods to make these inferences from DNA sequence data alone.

In this thesis, we show that chromatin A and B compartments can be accurately predicted from sequence-level features alone. Our model is robust across different cell types and species and allows us to derive key sequence determinants defining A/B compartments, such as Alu transposable elements and the Nanog transcription factor binding sites in mouse and human embryonic stem cells. We also show that A/B compartment annotations can be learned from human and mouse reference genome sequences. Trained models from one species can then be applied to another's (i.e., human to mouse or vice versa) with only minor losses in accuracy. This result suggests that compartment determinants could be evolutionary linked. Finally, we observe that our model can produce accurate compartment annotations with training on individual chromosomes, implying that partial compartment formation rules can be derived at the chromosome scale.

Establishing cell-type specific models of A/B compartments enables predictions for sequenced genomes without available Hi-C data or for genomes where it is difficult to perform Hi-C, leading to a valuable insight into their gene expression profile. Application to disease states such as cancer or to ancestral genomics could be interesting future work. Focusing on ancestral genomics, it would be interesting to investigate how much change in the reference DNA sequence is needed for our model to modify its compartment annotations. Indeed, only small changes in DNA sequences may be observed between ancestral species. Therefore, it would be of interest to investigate how much change in the DNA sequence is required to impact the prediction of A/B compartments. A first step in this direction could be to apply our model to different human genomes and analyze the differences in compartment annotations, if any.

Then, although our model produces accurate compartment annotations for a broad range of data sets, further understanding of the biological processes underlying compartment formation would require a more in-depth study of the emerging key features of our current model. An analysis of ChIP-seq experiments for the mentioned features could be a way to validate their contribution to the establishment of chromosomal compartments. Moreover, such an analysis could lead to the discovery of new links between sequence determinants and A/B compartments, through their enrichment relatively to one type of chromosomal compartment for instance.

It might also be interesting to improve our model using convolutional neural networks (CNNs) for instance. Although more computationally expensive, this type of network could take as input the raw string of nucleotides making up the DNA sequence instead of predetermined sequence pattern counts. By not constraining the initial pool of features, such an approach might lead to the discovery of new relevant sequence determinants. However, this network would have to scan the entire DNA sequence in order to extract such features, which is computationally much more expensive than summarizing 100kb bins with the count of specific sequence-motifs, which is the approach used in the current model.

By performing Hi-C analysis for four distinct mammalian species, a study by Rudan et al. (2015) [Rudan et al., 2015] showed that TADs were highly conserved in syntenic regions. Another way to improve our model for in-between species predictions could hence be to take synteny into account, in the features of the model for instance.

The question of the resolution at which the experiments presented in this thesis were performed is also a possible point of improvement. Preliminary work in this direction has been done, at up to 20kb resolution, with encouraging results. As the resolution of the experiments increases further, sparsity in the features might become an issue. However, compartments are on average multi-megabase long, so performing much higher resolution compartment annotations might not increase compartment knowledge by much.

Finally, different studies showed the existence of sub-compartments within compartments, each with their own genetic and epigenetic characteristics. In this thesis, we were not able to recover all these sub-compartments, which is probably the consequence of the relatively low resolution at which we worked (100kb against 1kb in one of these studies) and the small number of available biological data for the studied cell types. As was seen previously, applying our method at higher resolutions might create sparsity issues in the features. In the case of sub-compartments, it could however help in recovering their structures and gain additional insights into their specificities.

Bibliography

[A. Croft et al., 1999] A. Croft, J., Bridger, J., Boyle, S., Perry, P., Teague, P., and Bickmore, W. (1999). Differences in the localization and morphology of chromosomes in the human nucleus. *The Journal of cell biology*, 145:1119–31.

[Adriaens, 2018] Adriaens, Carmen, S. L. F. M. e. a. (2018). Blank spots on the map: some current questions on nuclear organization and genome architecture. *Histochemistry and Cell Biology*, 150(6):579–592.

[Alipanahi et al., 2015] Alipanahi, B., Delong, A., Weirauch, M., and J Frey, B. (2015). Predicting the sequence specificities of dna- and rna-binding proteins by deep learning. *Nature biotechnology*, 33.

[Baldi et al., 1999] Baldi, P., Brunak, S., Frasconi, P., Soda, G., and Pollastri, G. (1999). Exploiting the past and the future in protein secondary structure prediction. *Bioinformatics (Oxford, England)*, 15:937–46.

[Bani-Yaghoub et al., 2006] Bani-Yaghoub, M., Tremblay, R. G., Lei, J. X., Zhang, D., Zurakowski, B., Sandhu, J. K., Smith, B., Ribecco-Lutkiewicz, M., Kennedy, J., Walker, P. R., and Sikorska, M. (2006). Role of sox2 in the development of the mouse neocortex. *Developmental Biology*, 295(1):52 – 66.

[Bergstra and Bengio, 2012] Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*.

[Bickmore, 2013] Bickmore, W. A. (2013). The spatial organization of the human genome. *Annual Review of Genomics and Human Genetics*, 14(1):67–84. PMID: 23875797.

[Bonev et al., 2017] Bonev, B., Cohen, N. M., Szabo, Q., Fritsch, L., Papadopoulos, G. L., Lubling, Y., Xu, X., Lv, X., Hugnot, J.-P., Tanay, A., and Cavalli, G. (2017). Multiscale 3d genome rewiring during mouse neural development. *Cell*, 171(3):557 – 572.e24.

BIBLIOGRAPHY

[Brackley et al., 2016] Brackley, C. A., Brown, J. M., Waithe, D., Babbs, C., Davies, J., Hughes, J. R., Buckle, V. J., and Marenduzzo, D. (2016). Predicting the three-dimensional folding of cis-regulatory regions in mammalian genomes using bioinformatic data and polymer models. *Genome Biology*, 17(1):59.

[Breiman, 1984] Breiman, Leo; Friedman, J. O. R. S. C. (1984). *Classification and regression trees*. Wadsworth & Brooks/Cole Avanced Books & Software, Monterey, CA.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

[Bryson and Ho, 1969] Bryson, A. E. and Ho, Y.-C. (1969). Applied optimal control: optimization, estimation, and control.

[Cameron et al., 2018] Cameron, C. J., Dostie, J., and Blanchette, M. (2018). Estimating dna-dna interaction frequency from hi-c data at restriction-fragment resolution. *bioRxiv*.

[Cauchy, 1847] Cauchy, A.-L. (1847). *Méthode générale pour la résolution des systémes d'équations simultanées*.

[Chambers et al., 2003] Chambers, I., Colby, D., Robertson, M., Nichols, J., Lee, S., Tweedie, S., and Smith, A. (2003). Functional expression cloning of nanog, a pluripotency sustaining factor in embryonic stem cells. *Cell*, 113:643–55.

[Christopher J.F. Cameron and Dostie, 2016] Christopher J.F. Cameron, James Fraser, M. B. and Dostie, J. (2016). Mapping and visualizing spatial genome organization. In David P. Bazett-Jones, G. D., editor, *The Functional Nucleus*, pages 359–384. Springer Nature, Switzerland.

[Colah, 2015] Colah, C. (2015). Understanding lstm networks.

[Cordaux and Batzer, 2009] Cordaux, R. and Batzer, M. (2009). The impact of retrotransposons on human genome evolution. *Nat Rev Genet*, 10.

[Daniel et al., 2014] Daniel, C., Silberberg, G., Behm, M., and Öhman, M. (2014). Alu elements shape the primate transcriptome by cis-regulation of rna editing. *Genome Biology*, 15(2):R28.

[Di Pierro et al., 2017] Di Pierro, M., Cheng, R., Lieberman Aiden, E., G. Wolynes, P., and N. Onuchic, J. (2017). De novo prediction of human chromosome structures: Epigenetic marking patterns encode genome architecture. *Proceedings of the National Academy of Sciences*, 114:201714980.

[Dixon et al., 2015] Dixon, J., Jung, I., Selvaraj, S., Shen, Y., E Antosiewicz-Bourget, J., Young Lee, A., Ye, Z., Kim, A., Rajagopal, N., Xie, W., Diao, Y., Liang, J., Zhao, H., Lobanenkov, V., R Ecker, J., Thomson, J., and Ren, B. (2015). Chromatin architecture reorganization during stem cell differentiation. *Nature*, 518:331–6.

[Dixon et al., 2012] Dixon, J., Selvaraj, S., Yue, F., Kim, A., Li, Y., Shen, Y., Hu, M., Liu, J., and Ren, B. (2012). Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature*, 485(7398):376–380.

[E. Rumelhart et al., 1986] E. Rumelhart, D., E. Hinton, G., and J. Williams, R. (1986). Learning representations by back propagating errors. *Nature*, 323:533–536.

[ENCODE Project Consortium, 2012] ENCODE Project Consortium (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74.

[Farré et al., 2018] Farré, P., Heurteau, A., Cuvier, O., and Emberly, E. (2018). Dense neural networks for predicting chromatin conformation. *BMC Bioinformatics*, 19(1):372.

[Fortin and Hansen, 2015] Fortin, J.-P. and Hansen, K. D. (2015). Reconstructing a/b compartments as revealed by hi-c using long-range correlations in epigenetic data. *Genome Biology*, 16(1):180.

[Fraser et al., 2015] Fraser, J., Ferrai, C., M Chiariello, A., Schueler, M., Rito, T., Laudanno, G., Barbieri, M., L Moore, B., Kraemer, D., Aitken, S., Xie, S., J Morris, K., Itoh, M., Kawaji, H., Jaeger, I., Hayashizaki, Y., Carninci, P., Rr Forrest, A., Semple, C., and Nicodemi, M. (2015). Hierarchical folding and reorganization of chromosomes are linked to transcriptional changes in cellular differentiation. *Molecular Systems Biology*, 11:852.

[Gerstein Z. Wang, 2009] Gerstein Z. Wang, M. (2009). Rna-seq: a revolutionary tool for transcriptomics. *Nat Rev Genet*, 10:57–63.

[Gibcus and Dekker, 2013] Gibcus, J. and Dekker, J. (2013). The hierarchy of the 3d genome. *Molecular Cell*, 49(5):773 – 782.

[Gorkin et al., 2014] Gorkin, D., Leung, D., and Ren, B. (2014). The 3d genome in transcriptional regulation and pluripotency. *Cell Stem Cell*, 14(6):762 – 775.

[Graef et al., 2003] Graef, I. A., Wang, F., Charron, F., Chen, L., Neilson, J., Tessier-Lavigne, M., and Crabtree, G. R. (2003). Neurotrophins and netrins require calcineurin/nfat signaling to stimulate outgrowth of embryonic axons. *Cell*, 113(5):657 – 670.

[Hashimoto et al., 2011] Hashimoto, Y., Tsutsumi, M., Myojin, R., Maruta, K., Onoda, F., Tashiro, F., Ohtsu, M., and Murakami, Y. (2011). Interaction of hand2 and e2a is important for transcription of phox2b in sympathetic nervous system neuron differentiation. *Biochemical and Biophysical Research Communications*, 408(1):38 – 44.

[Heinz S., 2010] Heinz S., Benner C., S. N. B. E. e. a. (2010). Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and b cell identities. *Mol Cell*.

[Hochreiter and Schidhuber, 1997] Hochreiter, S. and Schidhuber, J. (1997). Long short-term memory. *Neural Computation*, pages 1735–1780.

[Huang et al., 2015] Huang, J., Marco, E., Pinello, L., and Yuan, G.-C. (2015). Predicting chromatin organization using histone marks. *Genome biology*, 16:162.

[Imakaev et al., 2012] Imakaev, M., Fudenberg, G., McCord, R., Naumova, N., Goloborodko, A., Lajoie, B., Dekker, J., and Mirny, L. (2012). Iterative Correction of Hi-C Data Reveals Hallmarks of Chromosome Organization. *Nat Methods*, 9(10):999–1003.

[J Rodda et al., 2005] J Rodda, D., Chew, J.-L., Lim, L.-H., Loh, Y.-H., Wang, B., Ng, H.-H., and Robson, P. (2005). Transcriptional regulation of nanog by oct4 & sox2. *The Journal of biological chemistry*, 280:24731–7.

[Jesse R. Dixon and Ren, 2012] Jesse R. Dixon, Siddarth Selvaraj, F. Y. A. K. Y. L. Y. S. M. H. J. S. L. and Ren, B. (2012). Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature*, 485:376 – 380.

[Jones, 2001] Jones, D. (2001). A taxonomy of global optimization methods based on response surfaces. *J. of Global Optimization*, 21:345–383.

[Jones et al., 2001] Jones, E., Oliphant, T., Peterson, P., et al. (2001). SciPy: Open source scientific tools for Python.

[Jost et al., 2014] Jost, D., Carrivain, P., Cavalli, G., and Vaillant, C. (2014). Modeling epigenome folding: Formation and dynamics of topologically associated chromatin domains. *Nucleic acids research*, 42.

[Karl Pearson, 1901] Karl Pearson, F. (1901). Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

[Kelley et al., 2016] Kelley, D. R., Snoek, J., and Rinn, J. (2016). Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Research*, 26:gr.200535.115.

[Kingma and Ba, 2014] Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

[Knight and Ruiz, 2007] Knight, P. and Ruiz, D. (2007). A fast algorithm for matrix balancing. *IMA J. Numer. Anal.*, 33.

[Lieberman-Aiden et al., 2009] Lieberman-Aiden, E., L van Berkum, N., Williams, L., Imakaev, M., Ragoczy, T., Telling, A., Amit, I., Lajoie, B., Sabo, P., Dorschner, M., Sandstrom, R., Bernstein, B., Bender, M., Groudine, M., Gnirke, A., A. Stamatoyannopoulos, J., Mirny, L., S Lander, E., and Dekker, J. (2009). Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science (New York, N.Y.)*, 326:289–93.

[Loh et al., 2006] Loh, Y.-H., Wu, Q., Chew, J.-L., Vega, B., Zhang, W., Chen, X., Bourque, G., George, J., Leong, B., Liu, J., Wong, K.-Y., W Sung, K., Lee, C., Zhao, X., Chiu, K. P., Lipovich, L., Kuznetsov, V., Robson, P., W Stanton, L., and Ng, H.-H. (2006). The oct4 and nanog transcription network regulates pluripotency in mouse embryonic stem cells. *Nature genetics*, 38:431–40.

[Louppe, 2014] Louppe, G. (2014). *Understanding Random Forests: From Theory to Practice*. PhD thesis, Universite de Liege.

[Ma et al., 2018] Ma, X., Ezer, D., Adryan, B., and Stevens, T. J. (2018). Canonical and single-cell hi-c reveal distinct chromatin interaction sub-networks of mammalian transcription factors. *Genome Biology*, 19(1):174.

[Mackenzie and Oteiza, 2007] Mackenzie, G. G. and Oteiza, P. I. (2007). Zinc and the cytoskeleton in the neuronal modulation of transcription factor nfat. *Journal of Cellular Physiology*, 210(1):246–256.

[Mockus et al., 2014] Mockus, J., Tiesis, V., and Zilinskas, A. (2014). The application of bayesian methods for seeking the extremum. *Towards Global Optimization*, 2:117–129.

[Müllner, 2011] Müllner, D. (2011). Modern hierarchical, agglomerative clustering algorithms.

[Nagano et al., 2017] Nagano, T., Lubling, Y., VÃąrnai, C., Dudley, C., Leung, W., Baran, Y., Mendelson Cohen, N., Wingett, S., Fraser, P., and Tanay, A. (2017). Cell-cycle dynamics of chromosomal organization at single-cell resolution. *Nature*, 547:61–67.

[Natale et al., 2018] Natale, F., Scholl, A., Rapp, A., Yu, W., Rausch, C., and Cardoso, M. C. (2018). Dna replication and repair kinetics of alu, line-1 and satellite iii genomic repetitive elements. *Epigenetics & Chromatin*, 11(1):61.

[Nikumbh and Pfeifer, 2017] Nikumbh, S. and Pfeifer, N. (2017). Genetic sequence-based prediction of long-range chromatin interactions suggests a potential role of short tandem repeat sequences in genome organization. *BMC Bioinformatics*, 18(1):218.

[Novo et al., 2016] Novo, C., Tang, C., Ahmed, K., Djuric, U., Fussner, E., P Mullin, N., P Morgan, N., Hayre, J., Sienerth, A., Elderkin, S., Nishinakamura, R., Chambers, I., Ellis, J., P Bazett-Jones, D., and J Rugg-Gunn, P. (2016). The pluripotency factor nanog regulates pericentromeric heterochromatin organization in mouse embryonic stem cells. *Genes & development*, 30.

[Oluwadare and Cheng, 2017] Oluwadare, O. and Cheng, J. (2017). Clustertad: An unsupervised machine learning approach to detecting topologically associated domains of chromosomes from hi-c data. *BMC Bioinformatics*, 18.

[Pan and Thomson, 2007] Pan, G. and Thomson, J. A. (2007). Nanog and transcriptional networks in embryonic stem cell pluripotency. *Cell Research*, 17:42–49.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[Quentin, 1994] Quentin, Y. (1994). A master sequence related to a free left Alu monomer (FLAM) at the origin of the B1 family in rodent genomes. *Nucleic Acids Research*, 22(12):2222–2227.

[Rao et al., 2014] Rao, S., Huntley, M., Durand, N., Stamenova, E., Bochkov, I., Robinson, J., Sanborn, A., Machol, I., Omer, A., Lander, E., and Aiden, E. (2014). A 3d map of the human genome at kilobase resolution reveals principles of chromatin looping. *Cell*, 159(7):1665 – 1680.

[Rennie et al., 2018] Rennie, S., Dalby, M., van Duin, L., and Andersson, R. (2018). Transcriptional decomposition reveals active chromatin architectures and cell specific regulatory interactions. *Nature Communications*, 9.

[Robins and Monro, 1951] Robins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407.

[Rosenblatt, 1962] Rosenblatt, F. (1962). *Principles of neurodynamics: perceptrons and the theory of brain mechanisms*. Report (Cornell Aeronautical Laboratory). Spartan Books.

[Roychowdhury and Abyzov, 2019] Roychowdhury, T. and Abyzov, A. (2019). Chromatin organization modulates the origin of heritable structural variations in human genome.

[Rudan et al., 2015] Rudan, M. V., Barrington, C., Henderson, S., Ernst, C., Odom, D., Tanay, A., and Hadjur, S. (2015). Comparative hi-c reveals that ctcf underlies evolution of chromosomal domain architecture. *Cell Reports*, 10(8):1297 – 1309.

[Ryba et al., 2010] Ryba, T., Hiratani, I., Lu, J., Itoh, M., Kulik, M., Zhang, J., C Schulz, T., Robins, A., Dalton, S., and Gilbert, D. (2010). Evolutionarily conserved replication timing profiles predict long-range chromatin interactions and distinguish closely related cell types. *Genome research*, 20:761–70.

[Salzberg, 1994] Salzberg, S. L. (1994). C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Machine Learning*, 16(3):235–240.

[Sanyal et al., 2012] Sanyal, A., Lajoie, B., Jain, G., and Dekker, J. (2012). The long-range interaction landscape of gene promoters. *Nature*, 489:109–13.

[Snoek et al., 2012] Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 2951–2959. Curran Associates, Inc.

[Song and E Crawford, 2010] Song, L. and E Crawford, G. (2010). Dnase-seq: A high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells. *Cold Spring Harbor protocols*, 2010:pdb.prot5384.

[Srinivas et al., 2010] Srinivas, N., Krause, A., Kakade, S., and Seeger, M. (2010). Gaussian process optimization in the bandit setting: No regret and experimental design. *ICML 2010 - Proceedings, 27th International Conference on Machine Learning*, pages 1015–1022.

[Srivastava et al., 2014] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

[Stevens et al., 2017] Stevens, T., Lando, D., Basu, S., Atkinson, L., Cao, Y., Lee, S., Leeb, M., Wohlfahrt, K., Boucher, W., OâĂŹShaughnessy-Kirwan, A., Cramard, J., Faure, A., Ralser, M., Blanco, E., Morey, L., Sanso, M., Palayret, M., Lehner, B., Di Croce, L., and

Laue, E. (2017). 3d structure of individual mammalian genomes studied by single cell hi-c. *Nature*.

[Tsompana and Buck, 2014] Tsompana, M. and Buck, M. J. (2014). Chromatin accessibility: a window into the genome. *Epigenetics & Chromatin*, 7(1):33.

[W. James Kent and Haussler, 2002] W. James Kent, Charles W. Sugnet, T. S. F. K. M. R. T. H. P. A. M. Z. and Haussler, D. (2002). The human genome browser at ucsc. *Genome Research*, 12.

[Whalen et al., 2016] Whalen, S., M Truty, R., and S Pollard, K. (2016). Enhancer-promoter interactions are encoded by complex genomic signatures on looping chromatin. *Nature genetics*, 48.

[Wingett et al., 2015] Wingett, S., Ewels, P., Furlan-Magaril, M., Nagano, T., Schoenfelder, S., Fraser, P., and Andrews, S. (2015). HiCUP: pipeline for mapping and processing Hi-C data. *F1000Res*, 4:1310.

[Yaffe and Tanay, 2011] Yaffe, E. and Tanay, A. (2011). Probabilistic modeling of hi-c contact maps eliminates systematic biases to characterize global chromosomal architecture. *Nature genetics*, 43:1059–65.

[Yoshua Bengio and Frasconi, 1994] Yoshua Bengio, P. S. and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2).

[Zhang et al., 2018a] Zhang, R., Yang, Y., Zhang, Y., Wang, Y., and Ma, J. (2018a). Predicting CTCF-mediated chromatin loops using CTCF-MP. *Bioinformatics*, 34(13):i133–i141.

[Zhang et al., 2018b] Zhang, S., Chasman, D., Knaack, S., and Roy, S. (2018b). In silico prediction of high-resolution hi-c interaction matrices. *bioRxiv*.

[Zhang et al., 2018c] Zhang, Y., An, L., Xu, J., Zhang, b., Jim Zheng, W., Hu, M., Tang, J., and Yue, F. (2018c). Enhancing hi-c data resolution with deep convolutional neural network hicplus. *Nature Communications*, 9.

[Zhou and G Troyanskaya, 2015] Zhou, J. and G Troyanskaya, O. (2015). Predicting effects of noncoding variants with deep learning-based sequence model. *Nature methods*, 12:931–934.

[Zhu et al., 2016] Zhu, Y., Chen, Z., Zhang, K., Wang, M., Medovoy, D., Whitaker, J., Ding, b., Li, N., Zheng, L., and Wang, W. (2016). Constructing 3d interaction maps from 1d epigenomes. *Nature Communications*, 7:10812.