



TÉCNICO
LISBOA

Implementation of a workflow for the processing and analysis of genome interaction datasets

Anandashankar Anil

Thesis to obtain the Master of Science Degree in

Biotechnology

Supervisor(s): Prof. Pelin Sahlén
Prof. Isabel Maria de Sá Correia Leite de Almeida

Examination Committee

Chairperson: Prof. Miguel Nobre Parreira Cacho Teixeira
Supervisor: Prof. Isabel Maria de Sá Correia Leite de Almeida
Member of the Committee: Prof. Sara Alexandra Cordeiro Madeira

October 2016

Dedicated to my family

Acknowledgments

This work could not have been completed without the aid and support of many individuals and organizations.

I would like express my deep gratitude to my supervisor Prof. Pelin Sahlén for the patient guidance of my work. I am deeply indebted to Science for Life Laboratory, Stockholm and the KTH Royal Institute of Technology, Stockholm for allowing me to use their resources in the completion of this work.

I extend my gratitude to my supervisor at Instituto Superior Técnico, Prof. Isabel Sá-Correia for granting me the opportunity to pursue this work.

I would also like to thank the euSYSBIO Erasmus Mundus programme for presenting me with the opportunity and funding to work on this project.

I also thank Prashanth, Saravanan, Hari, Anjan, Ron and Guillaume for having my back. I want to thank my family without whom I would not be where I am.

Resumo

Os activadores e as suas interações com promotores específicos desempenham um papel importante na transcrição de genes e consequentemente na expressão fenotípica. Recentemente, descobriu-se que a regulação distal de genes por activadores tem um papel em doenças como o cancro, com quase 80% das sequências de variantes associadas à doença localizadas dentro de activadores. O método HiCap combina uma enzima de restrição Hi-C 4-cortadora com a captura de sequência de regiões promotoras, permitindo que as interações de activadores ancorados a um promotor podem ser facilmente identificadas. Tendências inerentes a Hi-C podem ser propagadas no método HiCap, entretanto este comportamento necessita de investigação, uma vez que o HiCap possui uma maior seletividade e resolução. Além disso, as interações estruturais e funcionais não são diferenciadas no resultado do HiCap. Este projeto investiga a presença de tendências no HiCap de uma linhagem de células THP-1 estimuladas com um Lipopolissacarídeo. Algumas ferramentas de aprendizagem automática foram usadas para tentar diferenciar entre as interações estruturais e funcionais nos resultados do HiCap, usando dados de ChIP-seq de estudos sobre activadores como referência. Verificou-se que as tendências em Hi-C não se propagaram para o HiCap. Não obstante, os resultados dos algoritmos de aprendizagem automática sugerem que mais parâmetros de dados podem ser necessários para se distinguir claramente entre interações estruturais e funcionais.

Palavras-chave: Acentuadores, HiCap, Identificação de tendência, Aprendizagem Automática, Classificação de Classe Única, Modelo de Mistura de Gaussianas

Abstract

Enhancers and their interactions with specific promoters play an important role in gene transcription and consequently, phenotype expression. The distal regulation of genes by enhancers has recently been identified to play a role in diseases such as cancer with almost 80% of disease-associated sequence variants located within enhancers. With the HiCap method, which combines a 4-cutter restriction enzyme Hi-C with sequence capture of promoter regions, promoter-anchored enhancer interactions can be easily identified. Biases inherent in Hi-C may carry over into HiCap but bears investigation as HiCap has a higher selectivity and resolution. Also, structural and functional interactions are not differentiated in the HiCap output. In this project, the HiCap output from a line of THP-1 cells with Lipopolysaccharide stimulation was evaluated for inherent biases. Certain Machine Learning tools were used to try to differentiate between structural and functional interactions in HiCap output using datasets from ChIP-Seq studies on enhancers as reference. It was found that the biases in Hi-C were not carried over to the HiCap output. The results from the machine learning techniques suggest that more data parameters may be required to definitively distinguish structural and functional interactions.

Keywords: Enhancers, HiCap, Bias identification, Machine Learning, One Class Classification, Gaussian Mixture Models

Contents

| | |
|---|--------------|
| Acknowledgments | v |
| Resumo | vii |
| Abstract | ix |
| List of Tables | xiii |
| List of Figures | xv |
| Glossary | xvii |
| Glossary | xviii |
| 1 Introduction | 1 |
| 1.1 Introduction | 1 |
| 1.2 Problem Statement | 2 |
| 1.3 Scope | 2 |
| 1.4 Purpose | 2 |
| 1.5 Thesis Outline | 3 |
| 2 Background | 5 |
| 2.1 DNA, Chromosome and gene | 5 |
| 2.1.1 DNA and chromosome | 5 |
| 2.1.2 Genes and Gene Regulation | 6 |
| 2.2 Chromosome conformation capture | 8 |
| 2.2.1 Hi-C | 10 |
| 2.2.2 HiCap | 10 |
| 2.2.3 Biases in Hi-C and HiCap | 12 |
| 2.3 Machine Learning Techniques | 12 |
| 2.3.1 Theory | 13 |

| | | |
|----------|--|-----------|
| 2.3.2 | Expectation Maximization and Gaussian Mixture Models | 13 |
| 2.3.3 | One class classification | 16 |
| 3 | Design and Methodology | 19 |
| 3.1 | Workflow | 19 |
| 3.2 | Input | 20 |
| 3.2.1 | Data | 20 |
| 3.2.2 | Database Design | 22 |
| 3.3 | Methodology | 23 |
| 3.3.1 | Objective 1: Discovering Biases | 23 |
| 3.3.2 | Objective 2: Improving Interaction Calling | 24 |
| 4 | Results and Discussion | 29 |
| 4.1 | Objective 1: Discovering Biases | 29 |
| 4.1.1 | Repeat Overlap Bias | 29 |
| 4.1.2 | Restriction Enzyme Site Bias | 29 |
| 4.1.3 | GC Content Bias | 30 |
| 4.2 | Objective 2: Improving Interaction Calling | 31 |
| 4.2.1 | Approach 1: Modeling with Gaussian Mixtures | 32 |
| 4.2.2 | Approach 2: Classification with One Class SVM | 39 |
| 5 | Conclusions | 41 |
| 5.1 | Future Work | 42 |
| | Bibliography | 43 |

List of Tables

| | | |
|-----|--|----|
| 2.1 | Confusion Matrix for a classifier. | 17 |
| 4.1 | Format of presentation of results of clustering and classification | 33 |
| 4.2 | Gaussian Mixture Model fitted for training data with no regularization | 34 |
| 4.3 | Gaussian Mixture Model fitted for training data with regularization=0.0001 | 35 |
| 4.4 | Gaussian Mixture Model fitted for training data with regularization=0.001 | 36 |
| 4.5 | Gaussian Mixture Model fitted for training data with regularization=0.01 | 38 |
| 4.6 | One Class Support Vector Machine (SVM) fitted for training data with the rbf kernel | 39 |

List of Figures

| | | |
|-----|---|----|
| 2.1 | Theorised different levels of packing in a highly condensed mitotic chromosome. | 7 |
| 2.2 | Action of enhancers by looping of chromatin. | 8 |
| 2.3 | Simplified overview of Chromosome Conformation Capture(3C). | 9 |
| 2.4 | Simplified overview of Hi-C. | 10 |
| 2.5 | Simplified overview of HiCap. | 11 |
| 2.6 | Overview of overfitting and underfitting in machine learning. | 14 |
| 3.1 | The proposed workflow for processing data from HiCap experimental output and subsequent bias evaluation and model generation using machine learning algorithms. | 19 |
| 3.2 | Database Schema implemented in a relational database system. | 22 |
| 4.1 | Plot of Repeat Overlaps around distal interacting regions for different counts of supporting pairs | 30 |
| 4.2 | Plot of Restriction Enzyme sites 10 Kb around distal interacting regions for different counts of supporting pairs | 31 |
| 4.3 | Normalized GC Content measure binned into ten equal sized bins | 32 |
| 4.4 | Visualisation of GMM clusters with no regularization | 35 |
| 4.5 | Visualisation of GMM clusters with regularization=0.0001 | 36 |
| 4.6 | Visualisation of GMM clusters with regularization=0.001 | 37 |
| 4.7 | Visualisation of GMM clusters with regularization=0.01 | 38 |
| 4.8 | Visualisation of learning boundary with training and test data in one class classification | 40 |

Glossary

3C Chromosome Conformation Capture.

bp Base Pair.

ChIP-seq chromatin immunoprecipitation followed by high-throughput DNA sequencing.

DNA Deoxyribonucleic acid.

EM Expectation Maximization.

FPKM Fragments per kilobase of exon per million uniquely mapped reads.

GMM Gaussian Mixture Model.

kb Kilo Base Pair.

LPS Lipopolysaccharides.

OCC One Class Classification.

PCR Polymerase Chain Reaction.

RNA Ribonucleic acid.

SVM Support Vector Machine.

TF Transcription Factor.

TSS Transcription Start Site.

Chapter 1

Introduction

1.1 Introduction

The complete sequencing of the Human genome in 2003 led to an explosion of genomic data and techniques to process and extract information from this data which ushered in a new era in genetics and genomics called the genomic era[1]. The advances in genomics has helped with the rapid identification of newly discovered pathogens, gene-expression profiling to assess risk of disease and guide therapy, improve understanding of the role of specific genes in the causation of common conditions and so on. A part of studies in genomics focuses on the transcriptional control of genes. The type and amount of Ribonucleic acid (RNA) transcribed from genes control the phenotypic expression of the cell. Transcriptional regulation thus becomes important in studying disease phenotypes. A kind of regulatory element is cis-regulatory modules called enhancers. These short Deoxyribonucleic acid (DNA) segments, which may be situated many thousands of bases away from the genes they act on, can boost the transcription from the promoter of the target gene to a great degree[2]. The number of enhancers in eukaryotic genomes correlates with the complexity of the organism.

HiCap is a technique that has been formulated recently in[3] which is based on Hi-C and consequently on Chromosome Conformation Capture (3C), which can be considered as a part of the advances made in the genomic era. HiCap selects for promoter sequences and generates genome-wide maps of chromatin interactions where one of the interactors is a promoter sequence. It can generate fragments short enough for single-enhancer resolution[3]. Due to the novelty of the technique, a lot of scope in

the interpretation of the data generated by it exists. This thesis explores some of the facets of the HiCap output including biases which may be inherited from Hi-C and the question of classifying different types of chromatin interactions captured by HiCap.

1.2 Problem Statement

The problems tackled in this thesis are two fold -

- investigate and evaluate the output of the HiCap technique for the biases inherent in Hi-C which may have been inherited by HiCap
- investigate methods to distinguish between structural and functional interactions in HiCap output, mainly using Machine Learning algorithms.

1.3 Scope

This thesis will investigate how clustering with Gaussian Mixture Model (GMM) and classification with One Class SVM performs in distinguishing structural and functional interactions in the HiCap output from a line of THP-1 cells with LPS stimulation. Different variables inherent in the HiCap data were used as parameters in the machine learning techniques. The variables used are not exhaustive and further variables could be included in the future. The investigation is limited by the quality of the enhancer information in the reference datasets as well as by the assumptions made about data in HiCap output, like that it follows a Gaussian distribution.

Biases in Hi-C as evaluated in [4] will also be investigated if they exist in HiCap data generated from the line of THP-1 cells with LPS stimulation.

1.4 Purpose

Discovering the existence of biases in HiCap data serves to improve confidence in the results of studies based on data generated by this technique. By distinguishing functional interactions from structural interactions, the confidence in the identification of putative promoter - enhancer interactions by using HiCap can be increased. The

identification of promoter- enhancer interactions serve to improve the knowledge of transcriptional regulation in different kinds of cells and may help in determining the genetic underpinnings of disease.

1.5 Thesis Outline

This thesis is structured in the following manner. The first section introduces the concepts necessary to fully understand our problem statement and methods of choice. The second section focuses on the explanation of the methodology and tools used. The third section presents the results and discusses their implications. The final section includes the concluding remarks and suggestions on how the work can be carried forward.

Chapter 2

Background

2.1 DNA, Chromosome and gene

2.1.1 DNA and chromosome

DNA is a double stranded polymeric molecule which acts as the reservoir of heritable information in most multicellular organisms. This information is encoded by the order in which the monomers in DNA (which are the nucleotides adenine, thymine, guanine and cytosine) occur in the polymeric chain [5]. The information needed to encode an organism in its entirety is quite large and increases with increase in complexity of the organism. The human genome contains approximately 3.2×10^9 nucleotides[5] which if laid out linearly would be little bit more than 1 metre in length. In eukaryotic cells, the DNA is contained within a nucleus which is 5-8 μm in diameter[5]. In order to fit such long structures into a small enclosed space, the DNA in eukaryotic organisms is packaged into structures called chromosomes.

In humans, the DNA is divided into 24 different kinds of chromosomes - 22 somatic (or body) chromosomes which are numbered 1 through 22 and 2 sex chromosomes which are labeled X and Y. Each somatic cell has 2 pairs of body chromosomes and an XX or XY pair depending on whether the individual is female or male, respectively. The chromosomes have different levels of packing and organization.

The regions in the DNA which encode for amino acids (which are the monomers of proteins) are called genes. The process of transcribing the nucleotide sequence of a gene into the nucleotide sequence of an RNA molecule and then translating that into

the amino acid sequence of a protein is called gene expression[5]. Proteins perform various functions in the cell and are vital to cell metabolism.

Throughout the cell cycle, the chromosomes in the nucleus may take two forms - interphase chromosomes and mitotic chromosomes. During interphase, the chromosomes exist in the nucleus as long, thin, tangled threads of DNA [5]. They are organized in various ways with each chromosome tending to occupy a particular region of the nucleus and not extensively tangling with one another. Specific regions of chromosomes are also attached to sites on the nuclear envelope or the nuclear lamina. During the M phase of the cell cycle, the chromosomes adopt a more compact structure, and form highly condensed mitotic chromosomes.

Certain proteins bind to DNA and are called chromosomal proteins, some of which are of a class called histones. The complex of chromosomal proteins with DNA is called chromatin[5]. The long strands of DNA are packed in order for it to fit into the nucleus. The first level of packing involves the winding of DNA around a core of proteins formed from histones to form a bead-like unit called a nucleosome. The nucleosome can be considered as the basic unit of eukaryotic chromosome structure. The next level of packing builds upon this basic 'bead-on-a-string' organization of DNA and nucleosome, in which the nucleosomes are packed upon one another with mediation from histones to form a 30-nm fiber. The different levels of packing in a mitotic chromosome can be seen in Figure 2.1.

The local structure of chromatin is adjusted by the cell in various ways to allow access to specific proteins involved in gene expression and in DNA replication and repair. These include chromatin-remodeling complexes and reversible chemical modification of the histones. Thus, due to this localized alteration of chromatin packing, the chromatin in interphase chromosomes is not uniformly packed. The regions of the chromosome that contain genes that are being actively expressed are generally more extended, while those that contain currently inactive genes are more compact.

2.1.2 Genes and Gene Regulation

While traditionally defined as a sequence of the genome which contributes to a phenotype, in view of the intricacies involved with associated related sequences which control its expression, the gene can be said to be *a union of genomic sequences encoding a*

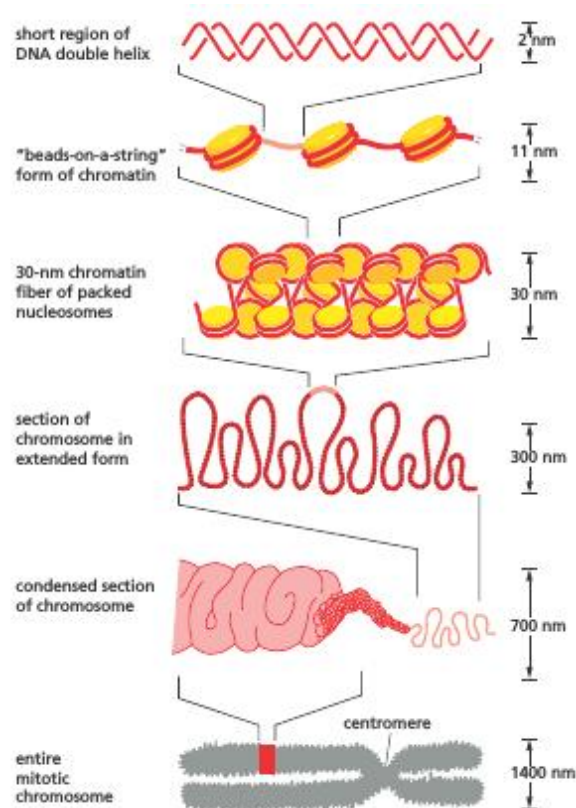


Figure 2.1: Theorised different levels of packing in a highly condensed mitotic chromosome. Adapted from [5].

coherent set of potentially overlapping functional products[6]. Regulatory sequences like enhancers, promoters and untranslated regions are considered gene associated. The promoter sequence is where the RNA polymerase binds to begin transcription of the gene, so by definition is situated close to the transcription start sites (TSS) of genes.

Enhancers, which are also called activators[7] or cis-regulatory modules[8] are a kind of genomic regulatory element which influences the intensity of genomic expression. They are distinct regions in the genome which contain binding site sequences for transcription factors (TFs). Transcription factors are proteins that bind to specific DNA sequence motifs. By forming complexes with TFs and gene promoters, enhancers can up or down regulate the transcription of a target gene[8]. Enhancers can be located at any distance from the target gene in the linear DNA sequence and come into spatial proximity by the looping of chromatin[8]. Enhancers can be found both upstream and downstream of their target genes and may regulate multiple genes[9]. Multiple enhancers may also regulate the same gene. The mechanism of enhancer action is

illustrated in Figure 2.2.

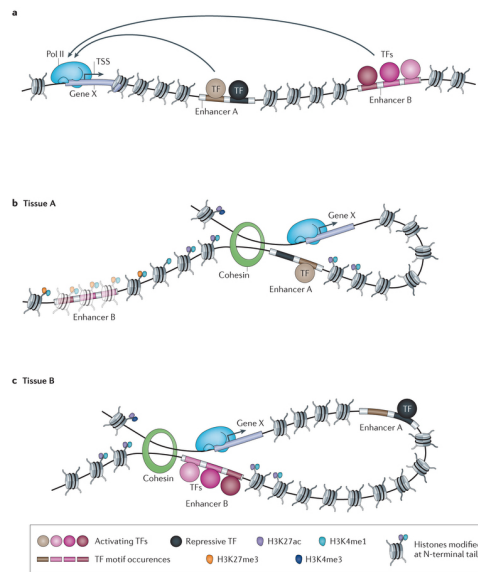


Figure 2.2: Action of enhancers by looping of chromatin. Adapted from [8].

The location and spatiotemporal activities of most of the enhancers are either not known with confidence or are unknown[10]. Also, predicting enhancers and their activity states from their DNA sequences is difficult as the TF binding motifs may be many and varied[8]. The activity of enhancers are also influenced by the openness of chromatin, i.e. chromatin where nucleosomal histone proteins are modified (notably by monomethylation of lysine 4 and acetylation of lysine 27 in histone H3)[2]. As recent studies have confirmed a role of mutations in distant cis-regulatory elements underlying various human diseases[9], the importance of identifying enhancers and their target genes can be understood. Out of the several computational and experimental approaches that have been developed to determine enhancer elements[8], a few of the methods used in these approaches are explained in more detail in later sections.

2.2 Chromosome conformation capture

Chromosome conformation capture (3C) is a technique used to detect the spatial organization of chromosomal DNA[11]. It can be used to detect long-range intrachromosomal or interchromosomal interactions. This method has several steps, in the first of which proximally located chromatin fragments are fixed with formaldehyde and then di-

gested with restriction enzymes. The nucleus is then lysed, the cross-linked complexes diluted, and the linked fragments ligated[11]. The crosslinks are then removed and the DNA purified. The DNA fragments are then amplified using a Polymerase chain reaction (PCR) with primers specific for pairs of DNA sequences under investigation. An illustration of this process can be seen in Figure 2.3.

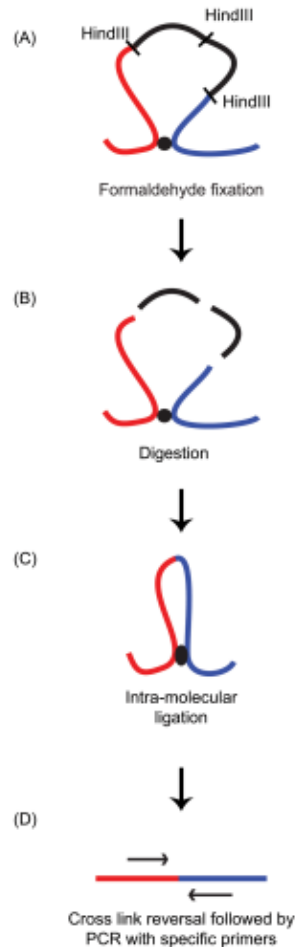


Figure 2.3: Simplified overview of Chromosome Conformation Capture(3C). Adapted from [11].

As enhancer elements may be located several megabases away from their target genes and their regulatory effect is mediated through looping interactions in which they become spatially adjacent to each other, 3C (and variants based on 3C) turns out to be an appropriate tool to study enhancers[12]. Various adaptations and extensions of 3C allow the user to choose the depth and breadth of enhancer interactions to be studied across the genome.

2.2.1 Hi-C

One of the variants of 3C is Hi-C[13]. Hi-C can identify chromatin interactions across an entire genome in an unbiased manner. In this method, the first step involves crosslinking the cellular DNA with formaldehyde; the DNA is then digested with a restriction enzyme to leave a 5'-overhang. In the next step, the 5'-overhang is filled to include a biotinylated residue. The resulting fragments are diluted and ligated under conditions that favor ligation between the cross-linked DNA fragments[13]. The resulting ligation products consist of fragments that were originally proximal in the nucleus and are also marked with biotin at the ligation junction. A Hi-C library is then created by shearing the DNA and selecting the biotin-containing fragments with streptavidin beads. The library can then be analyzed using massively parallel DNA sequencing to produce a catalog of interacting fragments[13]. An illustration of the process is given in Figure 2.4.

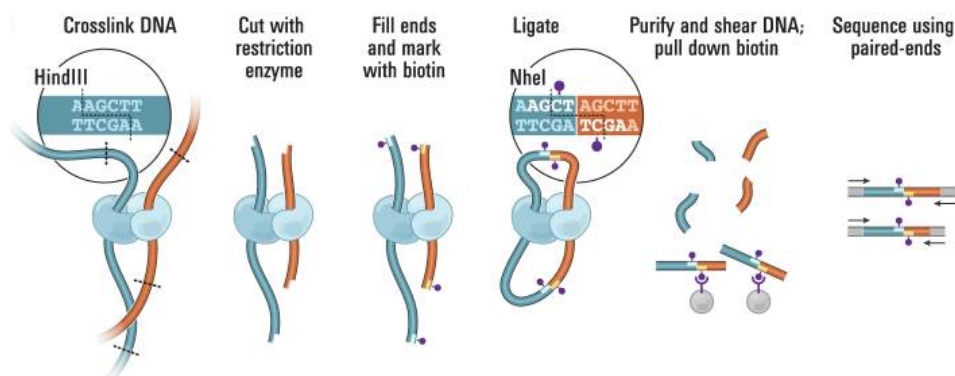


Figure 2.4: Simplified overview of Hi-C. Adapted from [13].

2.2.2 HiCap

HiCap is an extension of Hi-C. It substituted a 4-cutter restriction enzyme instead of the 6-cutter usually used in Hi-C and introduced the sequence capture of promoter regions[3]. Similar to Hi-C, HiCap also generates genome-wide maps of chromatin interactions with the added functionality of selecting for promoter-anchored interactions. The mean fragment size in HiCap is around 699 bp which gives it close to single-enhancer resolution. By fixing one interaction partner through sequence capture of promoter regions, HiCap has a higher sensitivity than Hi-C. This means that HiCap gives higher sensitivity with lower sequencing depth [3]. This method follows the same

steps as Hi-C till ligation and purification of fragments using the beads. Then labelled capture probes are added to further selectively purify the hybridised fragments. The fragments captured by hybridisation are then analysed and identified. These steps are illustrated in Figure 2.5.

A large fraction of distal regions were connected to the closest gene(around 65%)[3] and the rest were long range interactions. HiCap also identified promoter-promoter and distal-distal interactions. The distal regions map to putative enhancer elements as they interact with promoters and were found to be usually occupied by enhancer associated TFs[3].

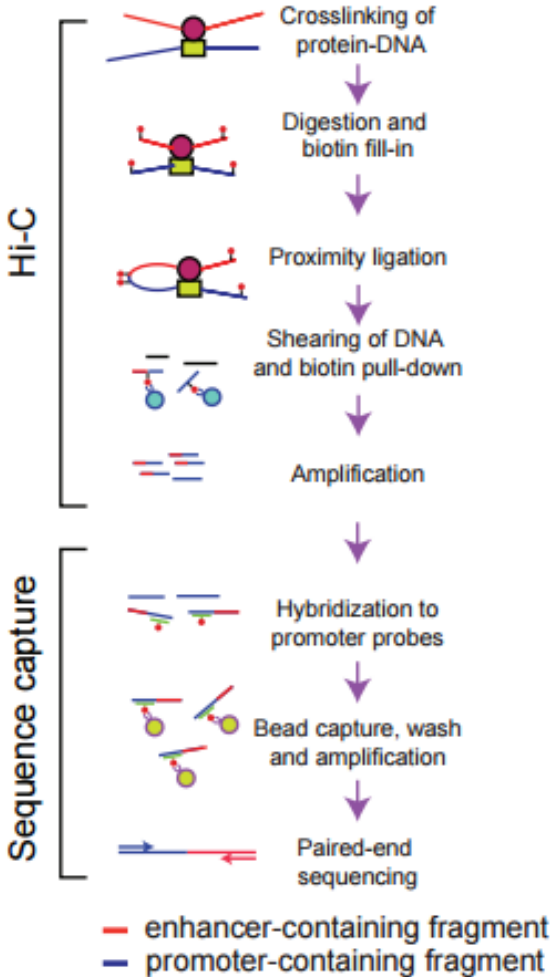


Figure 2.5: Simplified overview of HiCap. Adapted from [3].

2.2.3 Biases in Hi-C and HiCap

Studies have found various systematic biases in Hi-C read counts[4, 14]. These include biases due to GC content of fragment ends, distance between restriction enzyme cut sites and uniqueness/mappability of fragment ends of short sequence reads. These results were obtained with a six-cutter restriction enzyme in the genome cutting step of Hi-C[4]. It needs to be examined if the same biases are carried over unchanged into HiCap as it uses a four cutter restriction enzyme[3]. Another difference between Hi-C and HiCap is the targeted capturing of promoter sequences. As a PCR step exists in both Hi-C and HiCap, there is also a possibility of inherent PCR product bias.

2.3 Machine Learning Techniques

Machine learning is an approach to data-driven programming which uses the past performance of an algorithm or its performance on an example dataset to optimize itself[15]. It includes the study of algorithms that learn from data, construct models based on them and make predictions for new, unseen data based on the model constructed. Machine learning has a wide range of applications in various fields including and not limited to optical character recognition, search engines, data mining and bioinformatics.

Machine learning techniques can be divided into many categories depending on the criteria of the data used. If the techniques are classified by the nature of the learning algorithm used then the techniques can be divided into supervised, unsupervised, semi-supervised and reinforcement learning. In supervised learning, the algorithm is presented with inputs and desired outputs and aims to learn a rule which maps input to output. In unsupervised learning, desired outputs are not given or are unknown, leaving it up to the algorithm to assign a structure or pattern to the input data. The data given to semi-supervised learning techniques lies between supervised and unsupervised techniques - the inputs may or may not have designated outputs available. In reinforcement learning, the output of the algorithm can be said to be the sequence of actions that it takes and the sequence of correct actions it took to reach a goal[15]. This has applications in game playing and automated driving systems.

Another categorisation of machine learning techniques is based upon the desired

output of the algorithm. These include classification, regression, clustering, density estimation, dimensionality reduction and so on. Classification techniques are usually supervised in which example inputs are pre-divided into two or more classes and novel inputs are assigned to one of the classes as defined by the rules learned by the model using the example input. Regression is another supervised technique which uses a set of parameters to assign the data and label. Clustering is usually an unsupervised technique where the aim is to find clusters or groupings of input data[15]. In dimensionality reduction, data with high dimensionality is mapped to lower dimensional space by obtaining new variables which capture most of the variance in the input data.

2.3.1 Theory

In most machine learning techniques, the set of all data is usually divided into three - the training set, the test set and the validation set. The initial model is constructed with data from the training set to discover relationships among the data. The test set and validation set data are used to evaluate whether the relationships in the model hold true for data that is different from which the model was constructed. This is done so as to avoid overfitting the model. What this means is that if all the available data is used to construct the model, the machine learning technique used may identify and predict relationships among the data where none exist. For example noise in the data may cause the model to identify spurious relationships. This is illustrated in Figure 2.6. This figure shows the effect of model selection and data selection on overfitting and underfitting of data.

2.3.2 Expectation Maximization and Gaussian Mixture Models

A gaussian (or normal) distribution is a continuous probability distribution is informally called a bell curve. The probability density function of this distribution for a single variable x is as given in Equation 2.1 [17].

$$f(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\sigma^2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2.1)$$

where:

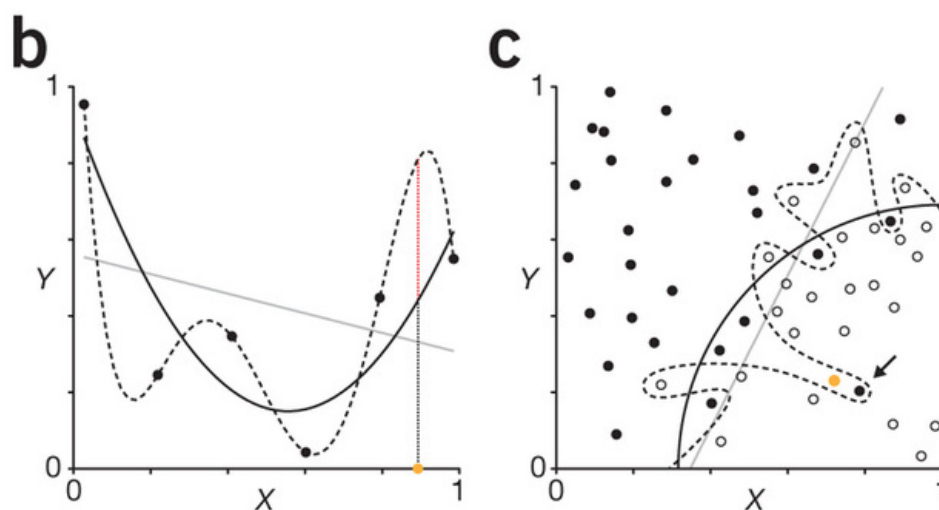


Figure 2.6: Overview of overfitting and underfitting in machine learning. In (b) shows a polynomial fit to data simulated from a third-order polynomial underlying a model with normally distributed noise. The fits shown exemplify underfitting (gray diagonal line, linear fit), reasonable fitting (black curve, third-order polynomial) and overfitting (dashed curve, fifth-order polynomial). In (c) shows two-class classification (open and solid circles) with underfitted (gray diagonal line), reasonable (black curve) and overfitted (dashed curve) decision boundaries. Adapted from [16].

μ = mean or expectation of the distribution (and also its median and mode)

σ = standard deviation

σ^2 = variance

For a D-dimensional vector x , the multivariate Gaussian distribution takes the form as in Equation 2.2

$$f(x|\mu, \Sigma) = \frac{1}{2\pi^{D/2}\sqrt{|\Sigma|}} e^{-\frac{(x-\mu)^T \Sigma^{-1} (x-\mu)}{2}} \quad (2.2)$$

where:

μ = D-dimensional mean vector

Σ = D x D covariance matrix

$|\Sigma|$ = determinant of Σ

The gaussian distribution is useful due to the central limit theorem which says that, subject to a few mild conditions, the sum of a set of random variables (which is itself a random variable), has a distribution that becomes more and more Gaussian like as the number of terms in the sum increases[17]. This means that any variable picked from a large enough dataset follows the gaussian distribution.

Mixture of Gaussians

In a machine learning technique utilizing a large dataset, the dataset can be modeled as a mixture of gaussian distributions. The centers and variances of the Gaussian components, as well as the mixing coefficients, will be considered as adjustable parameters to be determined as part of the learning process[17]. This can be used as a way to cluster data. A linear combination of Gaussians can give rise to very complex densities. A superposition of K Gaussian densities is of the form as in Equation 2.3.

$$p(x) = \sum_{k=1}^K \pi_k f(x|\mu_k, \Sigma_k) \quad (2.3)$$

where:

π_k = mixing coefficients

The term π_k can be considered as the prior probability of picking the k^{th} gaussian density[17]. The form of the Gaussian mixture distribution is governed by the parameters π , μ and Σ and a way to set the values of these parameters is to use maximum likelihood. A method of finding the solution for the maximum likelihood is the expectation maximization algorithm.

Expectation Maximization Algorithm

The Expectation Maximization or EM algorithm is an iterative method for finding maximum likelihood of parameters in distributions having latent variables. The algorithm is named after its two steps - the expectation step and the maximization step. The steps for EM for a mixture of Gaussians is as given below[17].

1. Initialize the means μ_k , covariances Σ_k and mixing coefficients π_k , and evaluate the initial value of the log likelihood.
2. **E** step. Evaluate the conditional probabilities using the current parameter values

$$\gamma(z_{nk}) = \frac{\pi_k f(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j f(x_n|\mu_j, \Sigma_j)} \quad (2.4)$$

3. **M** step. Re-estimate the parameters using the current conditional probabilities

$$\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) x_n \quad (2.5)$$

$$\Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (x_n - \mu_k^{new})(x_n - \mu_k^{new})^T \quad (2.6)$$

$$\pi_k^{new} = \frac{N_k}{N} \quad (2.7)$$

where

$$N_k = \sum_{n=1}^N \gamma(z_{nk}). \quad (2.8)$$

4. Evaluate the log likelihood

$$\ln p(X|\mu, \Sigma, \pi) = \sum_{n=1}^N \ln \left\{ \sum_{k=1}^K \pi_k f(x_n|\mu_k, \Sigma_k) \right\} \quad (2.9)$$

and check for convergence of either the parameters or the log likelihood. If the convergence criterion is not satisfied return to step 2.

The algorithm is deemed to have converged when the change in the log likelihood function, or alternatively in the parameters, falls below some threshold[17]. The EM algorithm is not guaranteed to find the largest maxima out of the various local maxima of the log likelihood function. This means that the EM may give different results on different runs and it may be a good practice to run a few repetitions to discover the optimal solution.

2.3.3 One class classification

Classification is traditionally a supervised machine learning technique where the categories that the input dataset are to be classified into are usually predefined. This creates a problem when data that cannot be categorized into one of these predefined categories occur in the dataset. One class classification algorithms are methods that are used when class information for one of the classes is absent, poorly-defined or not well defined[18]. One class classification techniques are inherently harder than binary or multiclass classification techniques. In recognition-based one-class learning, a sys-

tem is modeled with only examples of the target class in the absence of the counter examples[19]. This means that the algorithm tries to create a boundary surrounding the target class and tries to classify new data to one side of the boundary depending on the amount of similarity to target class with a threshold similarity. An effective threshold is crucial as a too strict threshold will exclude positive data, while a too loose threshold will include a large number of negative samples[19].

| | Object from target class | Object from outlier class |
|-----------------------|--------------------------|---------------------------|
| Classified as target | True positive, T^+ | False positive, F^+ |
| Classified as outlier | False negative, F^- | True negative, T^- |

Table 2.1: Confusion Matrix for a classifier.

Confusion matrices (as shown in Table 2.1) are usually used to compute the classification performance of multiclass(in this case binary) classifiers. However, to estimate the true error, the complete probability density of both the classes should be known[18]. As the probability density of only one class is known in one-class classification systems, the number of false positives and false negatives cannot be estimated with any confidence. Other methods have to be chosen to measure classification performance[18].

One of the algorithms included in OCC is the One class Support vector machine (OSVM). There exist many approaches to establish the classification boundary, one of which is constructing a hyper-plane around the data such that this hyper-plane is maximally distant from the origin and can separate the regions that contain no data[20]. Different kernels can be used although the algorithm performs best when the Gaussian kernel is used[18].

Chapter 3

Design and Methodology

3.1 Workflow

The workflow proposed for the processing of data and subsequent bias evaluation and model generation using machine learning algorithms is as illustrated in Figure 3.1.

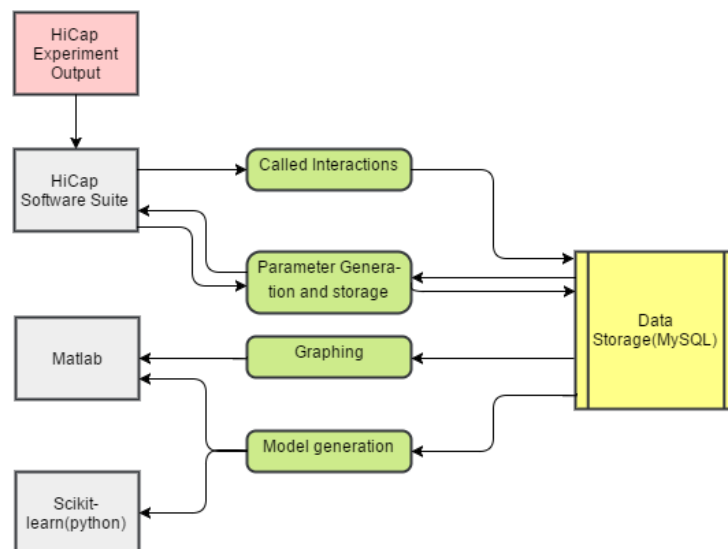


Figure 3.1: The proposed workflow for processing data from HiCap experimental output and subsequent bias evaluation and model generation using machine learning algorithms.

The input is the experimental output that is obtained from the HiCap method. This HiCap output is then passed to the HiCap software suite modules which call the interactions, parse and store them in the data repository. The various further parameters needed for bias evaluation and model generation are then generated using the requisite modules of the HiCap Software Suite and stored in the data repository. The plots

and models are then generated using MATLAB and scikit-learn by drawing the data from the repository. The various parts of the workflow will be explained in more detail in the following sections.

3.2 Input

3.2.1 Data

There are two different kind of datasets used in this project. First is the input dataset obtained from the HiCap procedure on a THP-1 cell line and calling the resulting interactions using the HiCap interaction-calling developed in association to [3]. The second dataset is a combined reference dataset of ChIP-Seq results of the enhancer landscape from various published studies in [21, 22, 23, 24, 25, 26].

THP-1 is a human leukemia monocytic cell line, and is a common model to estimate modulation of monocyte and macrophage activities[27]. The input dataset contains data from two experiments with THP-1 cell lines - one in which the cells were stimulated with inflammatory lipopolysaccharides(LPS) and the other which was not stimulated with LPS. LPS evokes strong immune responses in animal cells and is usually found in the outer membrane of gram-negative bacteria.

In each experiment, a number of interactions are 'called'(or taken to be true interactions) depending on whether the interaction had at least three supporting read pairs in each biological replicate[3]. An interaction is a ligated pair of probe-selected promoter fragment and a distal interacting fragment. Each interaction has different measured parameters which are as given below.

1. Probe related data: A unique identifier for each probe designed against promoters in the experiment. Each probe would therefore have an associated gene, chromosome number and other chromosome related information.
2. Interactor Chromosome related data: Data on which chromosome the interacting distal region(the putative enhancer) is situated and the start and end positions of the fragment on the chromosome.
3. Interactor Expression data: Data related to interactor expression which are as

below

- FPKM: FPKM (or Fragments per kilobase of exon per million fragments mapped) value for the interaction.
- Distance: The distance between the restriction enzyme fragments involved in the interaction which is the distance from the promoter with which the distal region is interacting. The distance can be a positive or negative value depending on whether the distal region is situated upstream or downstream of the promoter.
- Number of Supporting Pairs: The number of supporting pairs in the which support that particular interaction.
- p-value: The p-value based on background frequency of the observed interaction.
- Strand Combination: The number of supporting pairs for each combination in which ligation of the two fragments in the interaction can take place. Ligation can occur in the forward-forward, forward-reverse, reverse-forward or reverse-reverse manner. This parameter is presented in the '*x_y_z_a*' format where x, y, z and a represent the number of supporting pairs for each combination respectively. This parameter gives a measure of entropy in the interaction.

The reference dataset contains called peak information from ChIP-Seq, which includes the chromosome number, start and end positions on the chromosome in the BED format.

The total number of interactions in the HiCap output for THP-1 with LPS stimulation dataset number a total of 9,264,115. This dataset was then filtered using the number of supporting pairs higher than 3 and the p-value for the interactions lesser than 0.05 as filters down to 809,520 interactions. Out of these filtered interactions, 593,781 interactions overlapped with ChIP-Seq peaks in the reference dataset, which are all assumed to be functional interactions. The rest of the interactions numbering 215,739 do not overlap with the reference dataset and may contain a mixture of structural and functional interactions.

3.2.2 Database Design

A relational database was created to store the input data. The database schema is as shown in Figure 3.2.

The Probe table includes all the Probe related information. A unique integer identifier for each probe called probeID was used as index to the table.

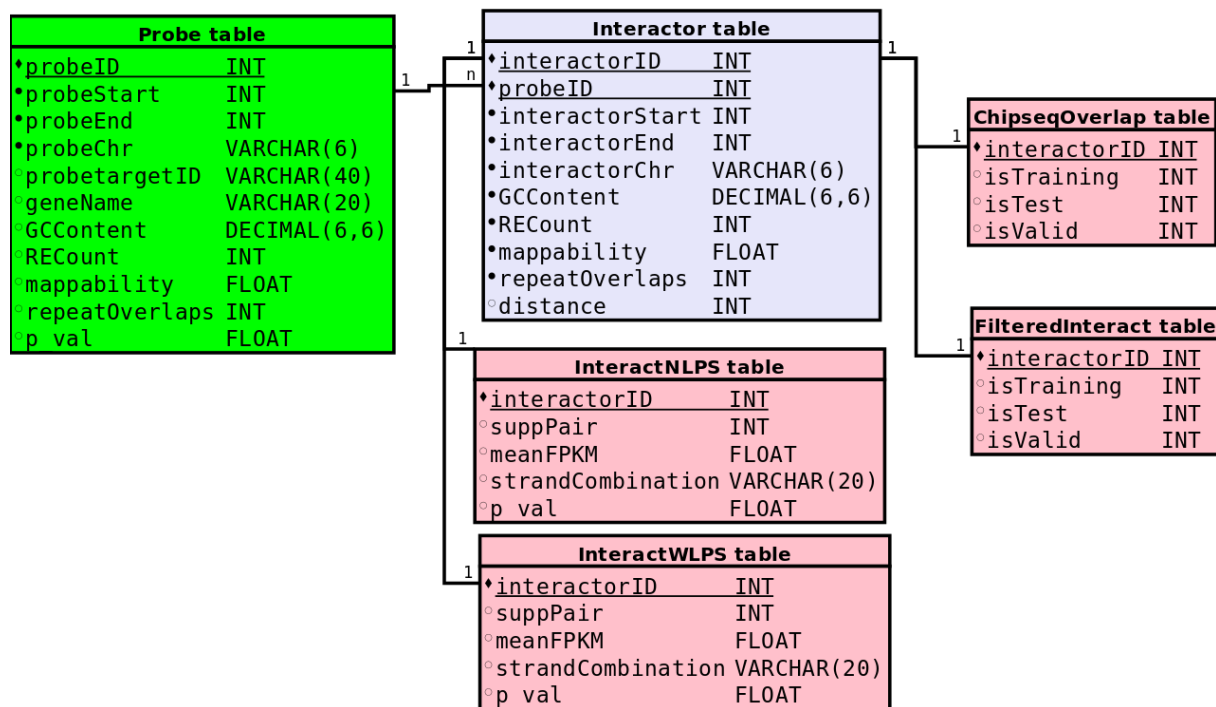


Figure 3.2: Database Schema implemented in a relational database system used to store the Probe and Interactor information to be used in this project.

The Interactor table stores all the information related to the distal interacting fragments that is common to both experiments - with LPS and without. A unique integer identifier for each distal fragment called interactID was used as index to the table. The probeID from the Probe table is used to connect which Probe is connected to which distal fragment.

The InteractWLPS and InteractNLPS tables stores the interacting fragment information specific to the experiments with LPS and without respectively. The unique interactID identifier is used to connect these tables to the main Interactor table.

The ChipseqOverlap table stores the information about the distal interacting fragments which overlap with ChIP-Seq peaks from the reference dataset. The flags isTraining, isTest and isValid indicate whether the data is used in Training sets, Test Sets or Validation sets respectively.

The FilteredInteract table stores the information about the distal interacting fragments which do not overlap with ChIP-Seq peaks from the reference dataset. The flags isTraining, isTest and isValid indicate whether the data is used in Training sets, Test Sets or Validation sets respectively.

The information in the fields GCContent, RECount, mappability and repeatOverlaps in the Probe and Interactor tables were generated from the respective start and end positions of the chromosomes as part of this project. This information was used to calculate bias information.

3.3 Methodology

The objectives of this project were two-fold:

1. to check whether the biases from the Hi-C method, on which HiCap is based, is carried over to HiCap.
2. to find a way to improve the calling of interactions and see if structural interactions can be differentiated from functional interactions.

3.3.1 Objective 1: Discovering Biases

Hi-C has number of documented systematic biases including biases due to GC Content, distance between restriction enzyme cut sites, mappability of fragments. The repeat overlap bias of fragments is a measure similar to mappability that was evaluated in the HiCap output data. The repeat overlap is something that arises when a sequence pattern repeats in two different distal fragments which do not interact with the same promoter sequence. This means that there is an ambiguity in mapping the fragments uniquely to a promoter.

For finding the GC content bias, a normalisation process was undertaken. The percentage GC content of probe fragments was binned, normalised by dividing with the total number of probes and then plotted. The probes themselves are only 120 Base Pair (bp) long, so the restriction fragments which the probes select for were chosen. The number of interactions with a certain number of supporting pairs falling into specific percentage GC bins divided by the total number of interactions with that number of

supporting pairs was also plotted. The equation used in the normalisation process is as shown in Equation 3.1.

$$\begin{aligned}x &= a/b \\ y &= c/d\end{aligned}\tag{3.1}$$

where:

n = a percentage GC bin range

a = Number of probes with GC_n

b = Total number of probes total

c = Number of interactions with Supporting pair =(1, 2, 3, 4, ...) and GC_n

d = Total number of Interactions with Supporting pair =(1, 2, 3, 4, ...)

For the Restriction enzyme cut sites, the chromosome sequence around a 10 kilo-base region of the distal interacting fragments were searched for the recognition site of the 4-cutter restriction enzyme used in HiCap(DpnII).

Tools Used

A modified version of the software used in [3] implemented in C++ was compiled on gcc version 4.8.4 and run on Ubuntu 14.04. The results were stored on and retrieved from a relational database implemented on mysql Version 14.14 Distribution 5.7.15 for Linux. Matlab(R2016a) was used to aggregate data and generate plots to visualise the results.

3.3.2 Objective 2: Improving Interaction Calling

In the current HiCap procedure, interactions are a ligated pair of two fragments. One fragment contains the promoter and is selected for by a probe designed for it and the other fragment contains a sequence which putatively interacts with the promoter on the first fragment. A particular pair of fragments ligate only if they are in spatial proximity at the time of DNA crosslinking. As this spatial proximity may not mean that the 2 fragments actually interact, the interactions called in HiCap can be divided into 2 cases - Functional and Structural interactions.

Functional interactions are those in which the two fragments on the ligated pair actually interact. Structural interactions are those in which the two fragments on the ligated pair do not interact and were simply spatially adjacent at the time of DNA cross linking. Discriminating between structural and functional interactions based on just the number of supporting pairs for the interactions, as is currently done, might lead to the exclusion of functional interactions which have a low number of supporting pairs. As the DNA crosslinking captures a temporal snapshot of the cell(s) in the experiment, certain structural interaction might have supporting pairs higher than an arbitrarily fixed threshold.

A way to discriminate between structural and functional interactions could be to use more parameters than just the number of supporting pairs of each interaction. From the experiment, the distance, FPKM, p value and strand combination can also be included as parameters to decide whether an interaction is structural or functional.

A way to find patterns in the interactions is to use machine learning techniques. From the nature of the data, two approaches can be used. As a sample of what structural or functional interactions does not exist, the first approach is to use an unsupervised clustering technique that uses all the parameters as input.

The second approach is to use the reference dataset of enhancers identified from ChIP-Seq peak studies as a model to verify what actual functional interactions look like. The reference dataset is intersected with the input dataset which yields a subset of interactions in the input dataset which are assumed to be functional interactions. As the rest of the interactions may be of either type, a negative class of structural interactions can not be defined. This means that the conventional supervised binary classification techniques can not be used. In this case, a one class classification technique can be used in which a model is constructed where only the target class is defined. The rest of the data is classified as either of the target class or not.

Strand Combination to interaction entropy

As the parameter 'strand combination' cannot be used as such in the input of algorithms of either of the approaches, a new derived parameter called 'interaction entropy' was defined. This was based on the 'tissue specificity index' as defined in [28] and expanded in [29]. The value of interaction entropy must give a sense of whether an

interaction prefers a specific strand combination. The equation for interaction entropy is defined as given in Equation 3.2

$$e = \frac{\sum_{j=1}^4 (1 - [(c_j)/(c_{max})])}{3} \quad (3.2)$$

where:

j = specific value of strand combination

c_j = number of supporting pairs of the j^{th} strand combination

c_{max} = maximum value of supporting pairs in a strand combination

Thus the value of interaction entropy ranges between 0 and 1. The closer the entropy value of an interaction is to 1, the likelier is it to favour a particular strand combination. If the value is 1, the interaction has supporting pairs of only one of forward-forward, forward-reverse, reverse-forward or reverse-reverse combinations. If the value of entropy is 0, the number of supporting pairs for all the four combinations are equal.

Tools Used - Approach 1

The input data was stored on and retrieved from a relational database implemented on mysql Version 14.14 Distribution 5.7.15 for Linux. The interaction entropy value was calculated using a program in C++ compiled on gcc version 4.8.4 and run on Ubuntu 14.04.

For clustering, a gaussian mixture model was used. The matlab(R2016a) implementation 'fitgmdist'[30] was used to model a gaussian mixture with two components. All parameters were scaled to a range of either 0 to 1 or -1 to 1 depending on whether they included negative values. A range of regularization values were used.

Tools Used - Approach 2

The reference dataset was intersected with the input data using the 'intersect' option of bedtools v2.17.0. The data was stored on and retrieved from a relational database implemented on mysql Version 14.14 Distribution 5.7.15 for Linux. The interaction entropy value was calculated using a program in C++ compiled on gcc version 4.8.4 and run on Ubuntu 14.04.

For one class classification, the implementation of The One-Class SVM (as in[20]) of scikit-learn was used with the RBF kernel. The language of implementation was python 2.7.6. All parameters were scaled to a range of either 0 to 1 or -1 to 1 depending on whether they included negative values.

Chapter 4

Results and Discussion

This chapter will present our results for the different objectives as defined in Chapter 3. The results for the first objective, which is the evaluation for biases in HiCap output data is discussed first, followed by the results of clustering using gaussian mixture models and then the classification using one-class SVM.

4.1 Objective 1: Discovering Biases

4.1.1 Repeat Overlap Bias

The distribution of repeat overlaps around the distal interacting regions(interactors) is as shown in Figure 4.1. The interactors were grouped on the number of supporting pairs its corresponding interaction had in HiCap output. Most of the interacting regions map to very low numbers of repeat overlaps. This means that the distal interacting regions can be uniquely mapped to the promoter regions and no bias with respect to repeat overlaps could be seen in the HiCap output data.

4.1.2 Restriction Enzyme Site Bias

The distribution of the cut site counts of the restriction enzyme used in HiCap(DpnII) in a 10Kilo Base Pair (kb) region around the distal interacting regions is as shown in Figure 4.2. As in the case in section 4.1.1, the interactors were grouped on their number of supporting pairs. The plots show an enrichment in the number of supporting pairs around 20 to 35 restriction enzyme cut sites. The recognition sequence of the

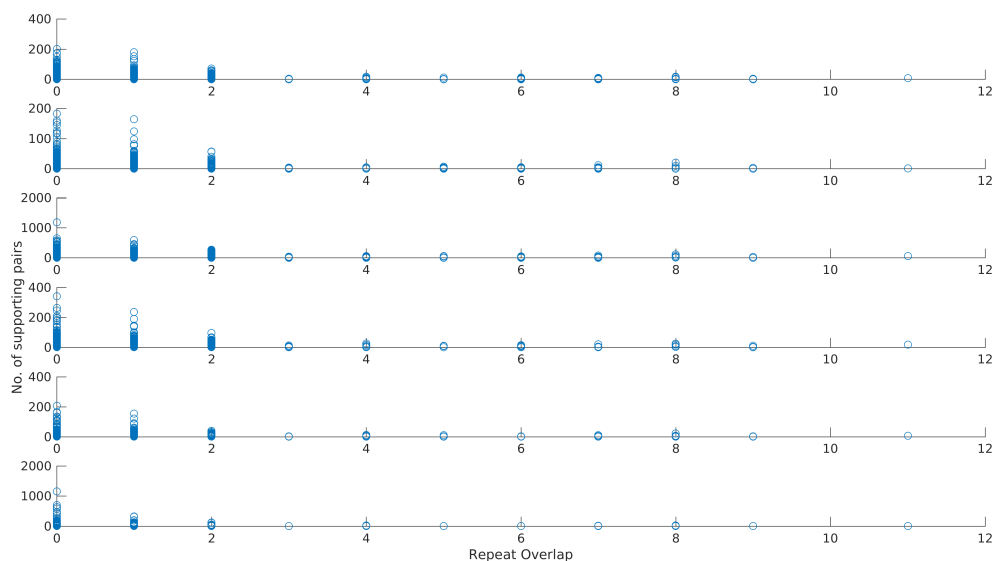


Figure 4.1: Plot of Repeat Overlaps around distal interacting regions for different counts of supporting pairs. From the top, the plots show repeat overlaps against interactions with 1 supporting pair, 2 supporting pairs, 3 supporting pairs, 4 supporting pairs, 5 supporting pairs, and more than 5 supporting pairs.

four-cutter DpnII is 'GATC'. Assuming that the bases in the genome are uniformly distributed, the expected number of occurrences of the 4-mer 'GATC' in a 10kb region is approximately 39. This shows that the enrichment found could be due to the normal density of restriction cut sites in the genome and no significant bias selecting for or against restriction enzyme cut sites in the HiCap method could be found.

4.1.3 GC Content Bias

A normalized mapping of GC content of interactors for each grouping of support pairs and of GC content of probe containing restriction enzyme fragments was done with respect to the Equation 3.1 and can be seen in Figure 4.3. The GC content was binned with bin size corresponding to 10 % GC content. As the probes themselves are only 120 bp in length, the GC content of the restriction enzyme fragment which the probe would select for was used instead. These fragments are variable sized. The figure shows a shift of the peak of the GC content of the distal interacting regions with respect to Probe fragment GC content to regions of lower GC. This could be a result of promoter regions having a higher GC content compared to the rest of the genome. As the probes select for promoters, consequently the probe containing regions have

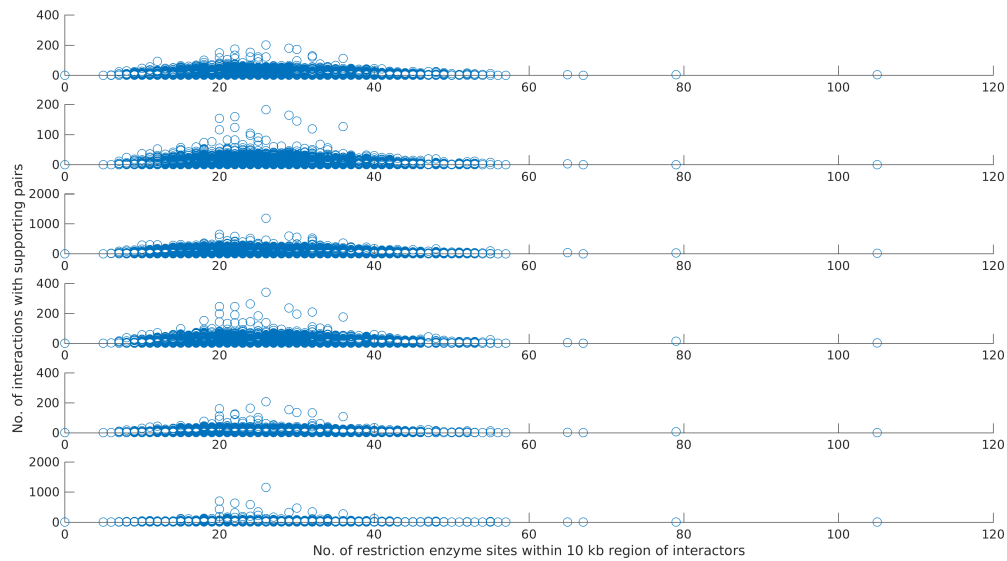


Figure 4.2: Plot of Restriction Enzyme sites 10 Kb around distal interacting regions for different counts of supporting pairs. From the top, the plots show restriction enzyme sites against interactions with 1 supporting pair, 2 supporting pairs, 3 supporting pairs, 4 supporting pairs, 5 supporting pairs, and more than 5 supporting pairs.

higher GC as well. In the case of the interactors, HiCap seems to pick fragments with GC content that is in the normal range of the human genome except for the fragments that have more than 5 supporting pairs which have a slight enrichment of GC content. This may be a natural consequence of the fact that cis-regulatory elements were found to be enriched in GC nucleotides[31].

GC Content Correlation

The GC content of probes and interacting distal regions showed very low correlation with Spearman's $\rho = 0.1948$ and Pearson's $r = 0.1989$. This also seems to indicate that the GC content of the probes may not affect the selection of distal interacting regions.

4.2 Objective 2: Improving Interaction Calling

The interactions in the HiCap output for THP-1 with LPS stimulation dataset were filtered down to 809,520 from a total of 9,264,115 interactions using the number of supporting pairs higher than 3 and the p-value for the interactions lesser than 0.05 as filters.

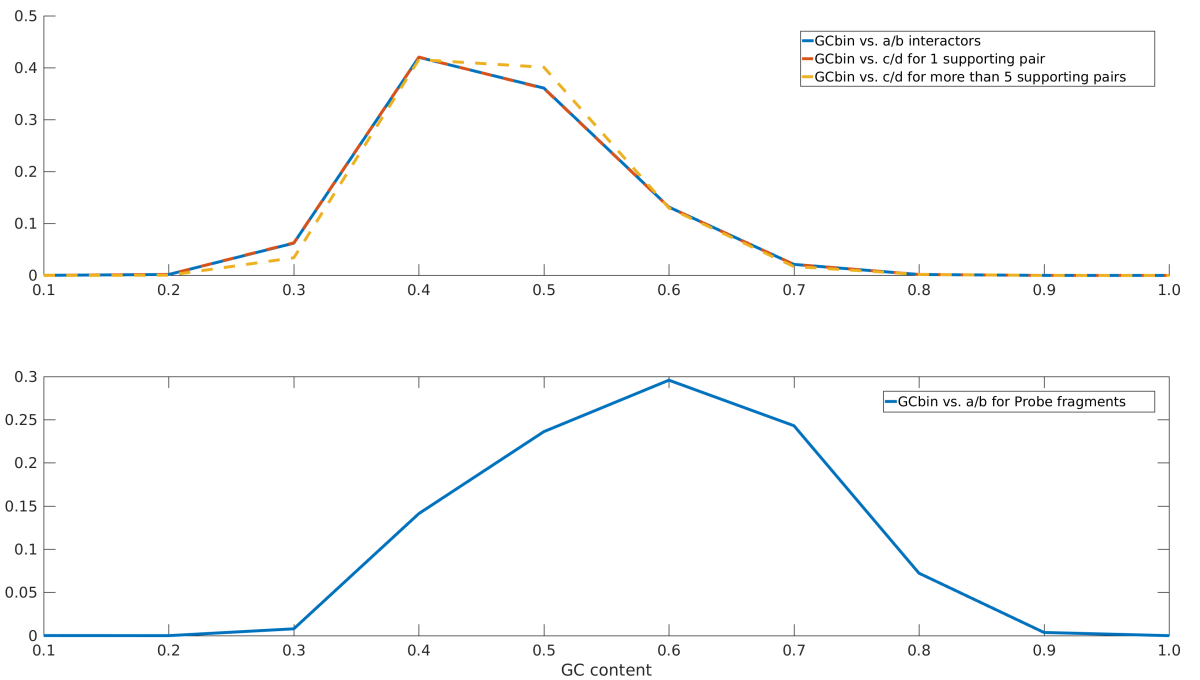


Figure 4.3: Normalized GC Content measure binned into ten equal sized bins. The first subplot shows the plot of normalized GC content of all interactors against the GC content bins and the plots of normalized GC content interactors with 1 and more than 5 supporting pairs against the GC content bins

4.2.1 Approach 1: Modeling with Gaussian Mixtures

The input dataset was labeled with 'ChIPSeqOverlapSet' or 'NegSet' depending on whether a particular interaction intersected with ChIP-Seq peaks in the reference dataset or not. It was assumed that the interactions overlapping with ChIP-Seq peaks are true functional interactions. The 'NegSet' may include structural and functional interactions. All parameters were normalised to the range [0, 1] if it did not include negative values or to the range [-1, 1] if it did. The normalisation is required as the values for certain parameters had very big ranges; for instance, 'distance' had a range of approximately $[-2^8, 2^8]$ which skews the fitted distribution if used without normalisation.

As there were five parameters in the input data as defined in section 3.3.2, the 'fitgmdist' MATLAB procedure was run once with all five parameters and a second time after using Principal Component Analysis (PCA) to convert the five parameters to two with the added advantage that the clusters can also be visualised. In each run of the 'fitgmdist' procedure, the EM algorithm was repeated 20 times, and the largest loglikelihood is chosen from all the repeats. The runs were also repeated with different

values of regularization. The regularization term controls for the complexity of the fitted model and is a means to reduce overfitting in case of noisy data.

The results of clustering will be presented in the format as in Table 4.1. The table does not correspond to a traditional confusion matrix for classifiers, nor do the calculated values correspond to sensitivity of the classification. This is because a properly defined example negative class to exemplify structural interactions does not exist. The variable 'n' indicates the number of interactions clustered into a specific cluster given by the superscript which is either C1 for Cluster 1 and C2 for Cluster 2.

| Label | No PCA | | with PCA | |
|----------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|
| | Cluster I | Cluster II | Cluster I | Cluster II |
| <i>Training data</i> | | | | |
| ChIP-Seq overlap | $\frac{n_{CS}^{C1}}{N_{CS}}$ | $\frac{n_{CS}^{C2}}{N_{CS}}$ | $\frac{n_{CS}^{C1}}{N_{CS}}$ | $\frac{n_{CS}^{C2}}{N_{CS}}$ |
| No overlap | $\frac{n_{NCS}^{C1}}{N_{NCS}}$ | $\frac{n_{NCS}^{C2}}{N_{NCS}}$ | $\frac{n_{NCS}^{C1}}{N_{NCS}}$ | $\frac{n_{NCS}^{C2}}{N_{NCS}}$ |
| <i>Test Data</i> | | | | |
| ChIP-Seq overlap | $\frac{n_{CS}^{C1}}{N_{CS}}$ | $\frac{n_{CS}^{C2}}{N_{CS}}$ | $\frac{n_{CS}^{C1}}{N_{CS}}$ | $\frac{n_{CS}^{C2}}{N_{CS}}$ |
| No overlap | $\frac{n_{NCS}^{C1}}{N_{NCS}}$ | $\frac{n_{NCS}^{C2}}{N_{NCS}}$ | $\frac{n_{NCS}^{C1}}{N_{NCS}}$ | $\frac{n_{NCS}^{C2}}{N_{NCS}}$ |

Table 4.1: Format of presentation of results of clustering and classification.

The variable 'N' indicates the total number of interactions in various cases as given by the subscript - CS gives total number of ChIP-Seq overlapping interactions in the procedure input, NCS gives the total number of non-ChIP-Seq overlapping interactions in the procedure input, C1 and C2 gives the total number of interactions clustered into Cluster 1 and Cluster 2. It is important to note that the clustering of interactions into clusters 1 and 2 are not connected with the labels in any manner.

The training and test set each contain 269,840 interactions which is composed of 197,927 interactions which overlap with the reference ChIP-Seq dataset and 71,913 interactions which do not overlap with the reference dataset.

Case 1: No Regularization

In this case, the 'fitgmdist' procedure was run without any regularization. The results obtained are as shown in Table 4.2.

| Label | No PCA | | with PCA | | |
|----------------------|-----------|------------|-----------|------------|--|
| | Cluster I | Cluster II | Cluster I | Cluster II | |
| <i>Training Data</i> | | | | | |
| ChIP-Seq overlap | 0.6774 | 0.3226 | 0.7585 | 0.2415 | |
| No overlap | 0.6534 | 0.3466 | 0.7047 | 0.2953 | |
| <i>Test Data</i> | | | | | |
| ChIP-Seq overlap | 0.6756 | 0.3244 | 0.7561 | 0.2439 | |
| No overlap | 0.6527 | 0.3473 | 0.7052 | 0.2948 | |

Table 4.2: Gaussian Mixture Model fitted for training data with no regularization

The visualisation of the clustering after the five parameter input was converted to 2 parameters with PCA is as shown in Figure 4.4. The green region contains the datapoints clustered into Cluster 1 and the red region those clustered into Cluster 2. The contour curves of one of the fitted gaussians can also be seen as dashed lines. The datapoints of the interactions that intersect with ChIP-Seq peaks are shown as red dots and the datapoints of the interactions that do not intersect with ChIP-Seq peaks are shown as blue dots.

It can be seen from the Table 4.1 that without PCA approximately 68% of the ChIP-Seq overlapping interactions are clustered into one cluster and with PCA, it rises to 75%. As can be seen from the figure, the dataset is very grainy.

Case 2: With Regularization=0.0001

The 'fitgmdist' procedure was run with a regularization value of 0.0001. The results are as shown in Table 4.3.

The visualisation of the clustering after the five parameter input was converted to 2 parameters with PCA is as shown in Figure 4.5. The green region contains the datapoints clustered into Cluster 1 and the red region those clustered into Cluster 2. The contour curves of both of the fitted gaussians can also be seen as dashed lines, one fully and the other partially. It can be seen from the Table 4.3 that without PCA

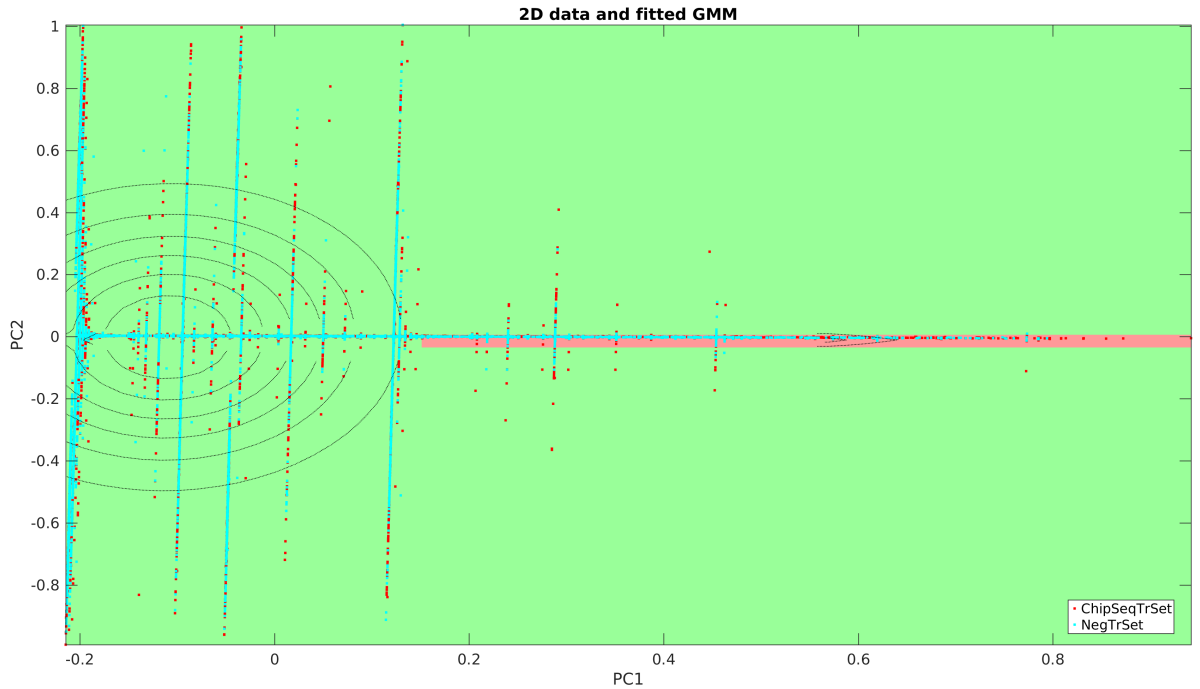


Figure 4.4: Visualisation of GMM clusters with no regularization of the with PCA run as in Table 4.2. The green region contains Cluster 1 datapoints and the red region contains Cluster 2 datapoints

| Label | No PCA | | with PCA | |
|----------------------|-----------|------------|-----------|------------|
| | Cluster I | Cluster II | Cluster I | Cluster II |
| <i>Training Data</i> | | | | |
| ChIP-Seq overlap | 0.1798 | 0.8202 | 0.1617 | 0.8383 |
| No overlap | 0.2208 | 0.7792 | 0.2000 | 0.8000 |
| <i>Test Data</i> | | | | |
| ChIP-Seq overlap | 0.1829 | 0.8171 | 0.1646 | 0.8354 |
| No overlap | 0.2199 | 0.7801 | 0.1988 | 0.8012 |

Table 4.3: Gaussian Mixture Model fitted for training data with regularization=0.0001

approximately 82% of the ChIP-Seq overlapping interactions are clustered into one cluster and with PCA, it rises to 83%. This seems to indicate that the regularization is effective to smoothen out the noisy data. The corresponding values for the non ChIP-Seq overlapping set seems to rise as well.

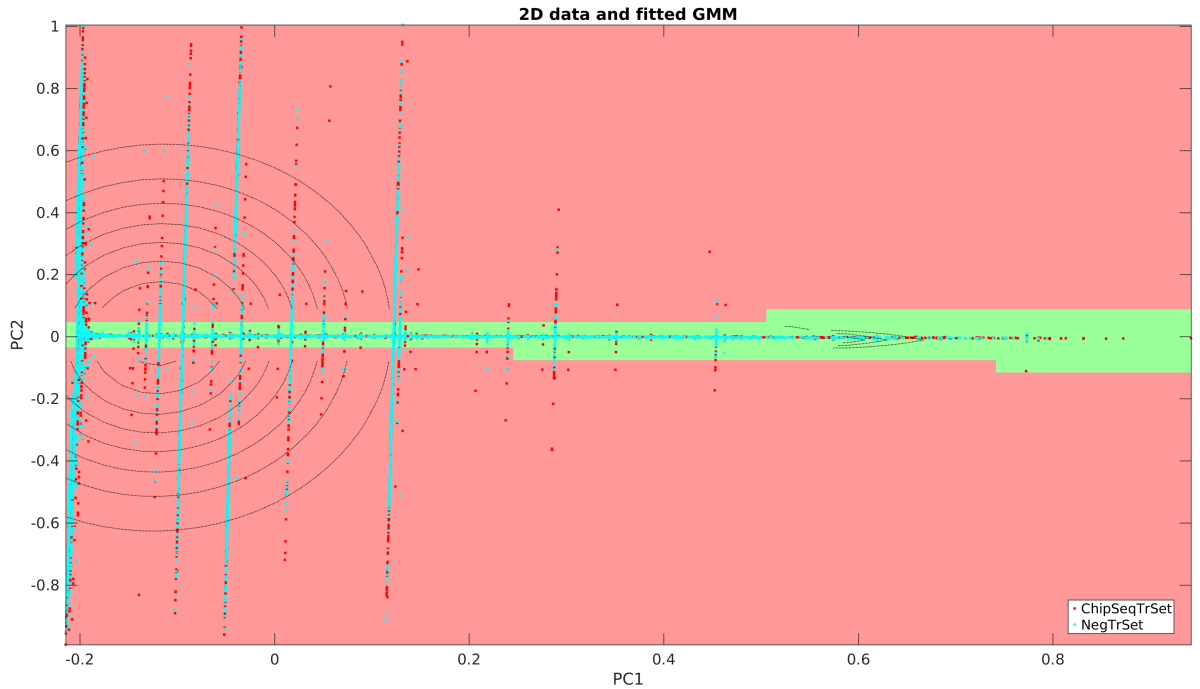


Figure 4.5: Visualisation of GMM clusters with regularization=0.0001 of the with PCA run as in Table 4.3. The green region contains Cluster 1 datapoints and the red region contains Cluster 2 datapoints.

Case 3: With Regularization=0.001

The 'fitgmdist' procedure was run with a regularization value of 0.001. The results are as shown in Table 4.4.

| Label | No PCA | | with PCA | |
|----------------------|-----------|------------|-----------|------------|
| | Cluster I | Cluster II | Cluster I | Cluster II |
| <i>Training Data</i> | | | | |
| ChIP-Seq overlap | 0.867 | 0.133 | 0.1037 | 0.8963 |
| No overlap | 0.8577 | 0.1423 | 0.1271 | 0.8729 |
| <i>Test Data</i> | | | | |
| ChIP-Seq overlap | 0.8659 | 0.1341 | 0.1060 | 0.8940 |
| No overlap | 0.8547 | 0.1453 | 0.1289 | 0.8711 |

Table 4.4: Gaussian Mixture Model fitted for training data with regularization=0.001

The visualisation of the clustering after the five parameter input was converted to 2 parameters with PCA is as shown in Figure 4.6. The green region contains the datapoints clustered into Cluster 1 and the red region those clustered into Cluster 2. The

contour curves of both of the fitted gaussians can also be seen as dashed lines, one fully and the other partially. It can be seen from the Table 4.4 that without PCA approximately 86% of the ChIP-Seq overlapping interactions are clustered into one cluster and with PCA, it rises to 89%. As a very high percentage of ChIP-Seq overlapping interactions seem to be in a single cluster and assuming that these are all functional interactions, it can be seen that a high percentage of the non-overlapping interactions are clustered into the same cluster. This seems to indicate that the number of structural interactions in the dataset may be low.

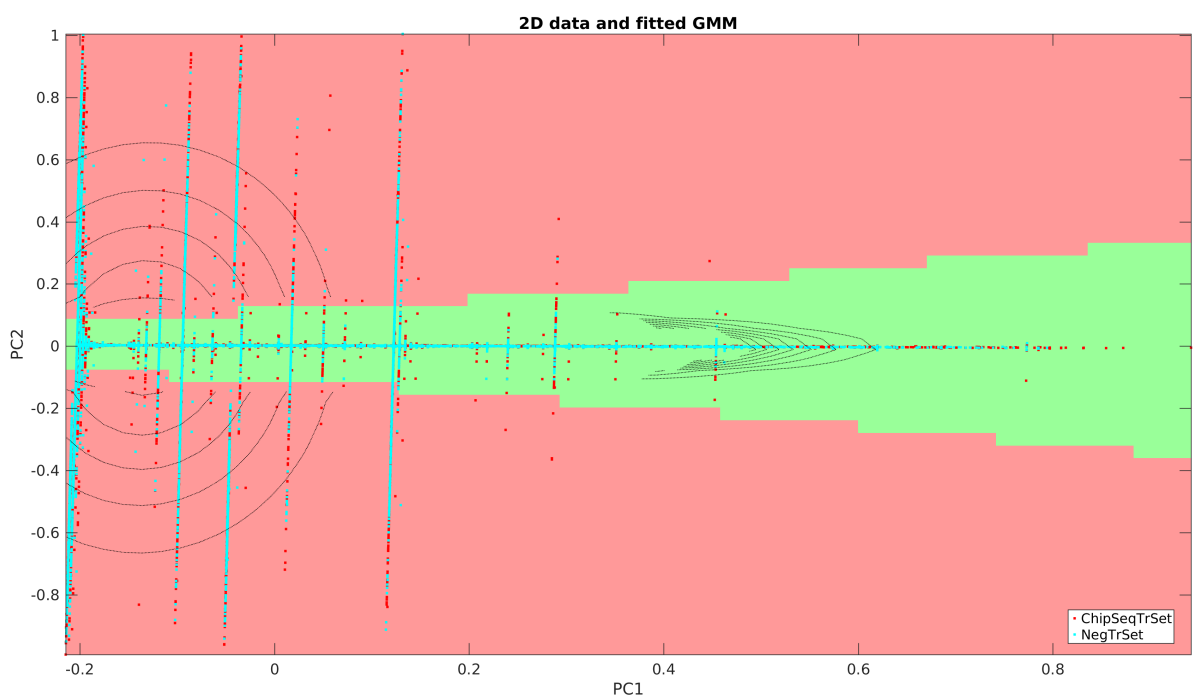


Figure 4.6: Visualisation of GMM clusters with regularization=0.001 of the 'with PCA' run as in Table 4.4. The green region contains Cluster 1 datapoints and the red region contains Cluster 2 datapoints

Case 4: With Regularization=0.01

The 'fitgmdist' procedure was run with a regularization value of 0.01. The results are as shown in Table 4.5.

The visualisation of the clustering after the five parameter input was converted to 2 parameters with PCA is as shown in Figure 4.7. The green region contains the datapoints clustered into Cluster 1 and the red region those clustered into Cluster 2. The

| Label | No PCA | | with PCA | |
|----------------------|-----------|------------|-----------|------------|
| | Cluster I | Cluster II | Cluster I | Cluster II |
| <i>Training Data</i> | | | | |
| ChIP-Seq overlap | 0.9418 | 0.0582 | 0.9441 | 0.0559 |
| No overlap | 0.9366 | 0.0634 | 0.9326 | 0.0674 |
| <i>Test Data</i> | | | | |
| ChIP-Seq overlap | 0.9403 | 0.0597 | 0.9428 | 0.0572 |
| No overlap | 0.9353 | 0.0647 | 0.9312 | 0.0688 |

Table 4.5: Gaussian Mixture Model fitted for training data with regularization=0.01

contour curves of both of the fitted gaussians can also be seen as dashed lines, one fully and the other partially. It can be seen from the Table 4.5 that without PCA approximately 94% of the ChIP-Seq overlapping interactions are clustered into one cluster and with PCA, it is also around 94%. As only very low number of non overlapping interactions are clustered into the second cluster, it may not be clear if the model becomes too simplified with this value of regularization.

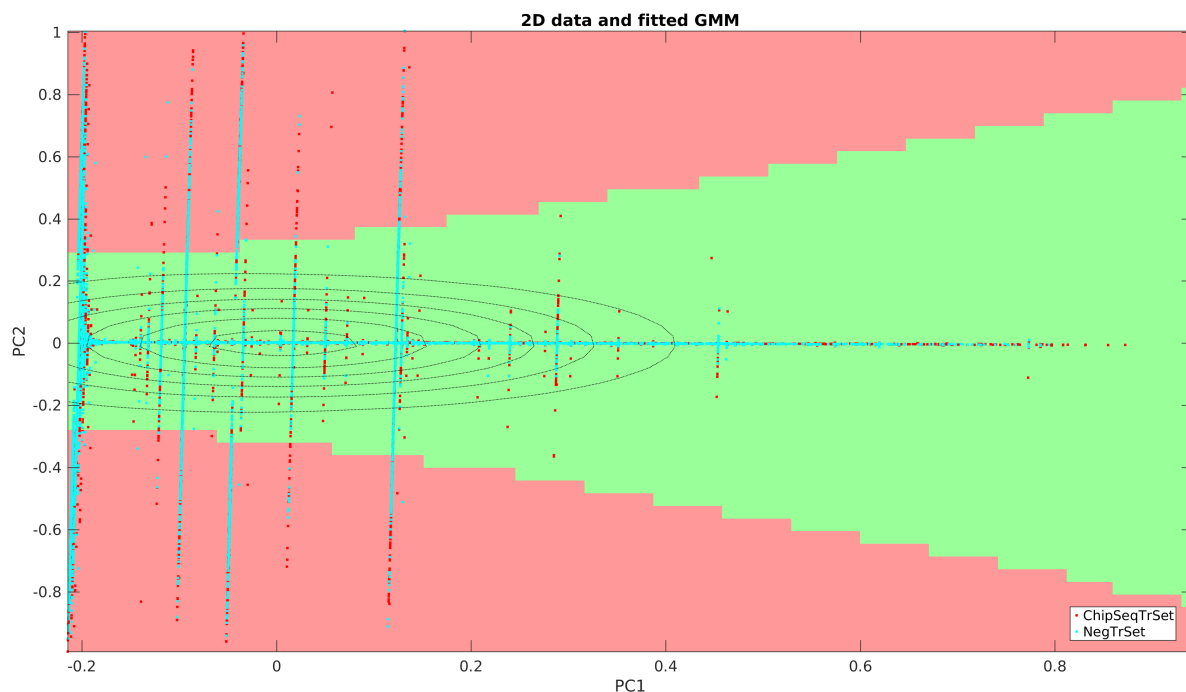


Figure 4.7: Visualisation of GMM clusters with regularization=0.01 of the 'with PCA' run as in Table 4.5. The green region contains Cluster 1 datapoints and the red region contains Cluster 2 datapoints

4.2.2 Approach 2: Classification with One Class SVM

The second approach is to use the reference dataset of enhancers identified from ChIP-Seq peak studies as a model to verify what actual functional interactions look like with only data from the ChIP-Seq overlapping interactions given in the training set to the One Class SVM, so that that it learns what a functional interaction is and can construct a learning boundary around them.

The OneClassSVM procedure of sklearn was run with the rbf kernel with a gamma of 0.1. The results are as shown in Table 4.6.

| Label | No PCA | | with PCA | |
|------------------|---------------|----------|---------------|----------|
| | Learned Class | Rejected | Learned Class | Rejected |
| <i>Test Data</i> | | | | |
| ChIP-Seq overlap | 0.8995 | 0.1006 | 0.9002 | 0.0998 |
| No overlap | 0.9005 | 0.0994 | 0.9024 | 0.0976 |

Table 4.6: One Class SVM fitted for training data with the rbf kernel

The visualisation of the clustering after the five parameter input was converted to 2 parameters with PCA is as shown in Figure 4.8. The thick red line indicates the class boundary learned by the algorithm. The yellow region inside the boundary contains the datapoints clustered into the Class and the region outside the boundary contains the datapoints rejected as being dissimilar to and not belonging with the datapoints in the learned class. From the Table 4.6, it can be seen that for both ChIP-Seq overlapping and non-overlapping interactions around 90% of the datapoints are accepted by the classifier as part of the learned class. From the figure 4.8, it can be seen that the data is very grainy and the datapoints of the ChIP-Seq overlapping interactions are widely distributed when compared to the non-overlapping set.

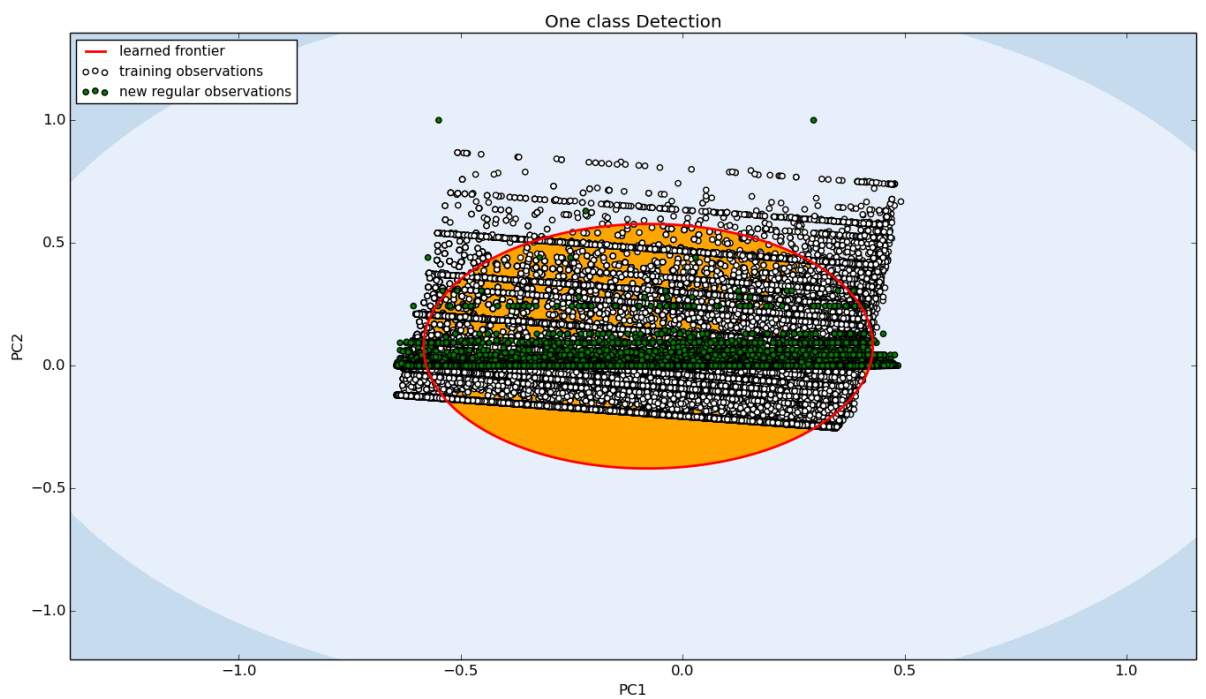


Figure 4.8: Visualisation of learning boundary with training and test data of the 'with PCA' run as in Table 4.6. The datapoints inside the learning boundary are accepted by the One class SVM, while the ones outside are rejected.

Chapter 5

Conclusions

The goal of the project was to evaluate for biases in HiCap data which may have carried over from its parent technique Hi-C and to discover methods for differentiating between different kinds of interactions in the output data of HiCap. It was found that most of the biases in Hi-C data may not be carried over into HiCap as per the results of this work. It is hypothesized that this may be due to the higher selectivity and resolution inherent in the HiCap method. This does not discount the fact that this selectivity of HiCap may dispose it towards biases that are not found in Hi-C data. This is an area that bears more investigation.

The Machine Learning techniques of clustering using Gaussian Mixture models and classification using One-class SVM were used to investigate whether structural and functional interactions in HiCap data can be distinguished from each other. The various input parameters were normalized to similar ranges and a reference dataset with enhancers identified from ChIP-Seq peaks was used as a model for functional interactions. It was found that although the techniques separated the interactions into two, more work may be needed to increase the confidence in the classification. This may include filtering the reference datasets for high confidence enhancer peaks.

A major achievement of the present work is that the machine learning approach used in this project has not been used in similar multi-factorial datasets of genomic data. The datasets are incomplete in that a proper negative set consisting of structural interactions could not be defined. However, the techniques proved to be quite successful in classifying the known data, proving that such techniques can be applied on similar data with successful results.

5.1 Future Work

This work can be extended to the HiCap output of THP-1 cells with no LPS stimulation and a comparative analysis of which interactions were classified into similar classes in both datasets can be done to verify the validity of the results. More parameters can also be used in the classification/clustering which takes into account the enhancer RNA expression in the HiCap experiments[3].

As an improvement in clustering in the method of clustering with Gaussian Mixture models could be seen when regularization was applied, it is another indication that the data is grainy. So instead of clustering with just two classes, more clusters could be used. An estimate of how many clusters to use could be optimized using cross-validation methods. This could have been done by using more test datasets. A constraint on using more clusters is that more information on structural interactions to use as negative controls will be required to classify which clusters contain functional interactions and which contain structural.

One of the ways to improve the learning of a classification boundary in one class classification would be define a custom kernel as opposed to the predefined rbf kernel used in this work. Also as the one-class classification does not provide a clear cut boundary on the dataset set used, binary classification methods could be used if the dataset and the reference dataset cannot be further refined. This is due to the fact that as both reference dataset and input dataset may contain data from both classes, one-class classification may not give much improvement over binary or multiclass classification methods. So to give a clearer picture of the results, binary or multiclass classification may be carried out and the resulting model compared with the model generated by one-class classification.

Bibliography

- [1] A. E. Guttmacher and F. S. Collins. Welcome to the genomic era. *New England Journal of Medicine*, 349:996–998, Sept. 2003. doi:10.1056/NEJMe038132.
- [2] W. Schaffner. Enhancers, enhancers - from their discovery to today's universe of transcription enhancers. *Biological Chemistry*, 396(4):311–327, Feb. 2015. doi:10.1515/hsz-2014-0303.
- [3] P. Sahlén, I. Abdullayev, D. Ramsköld, L. Matskova, N. Rilakovic, B. Lötstedt, T. J. Albert, J. Lundeberg, and R. Sandberg. Genome-wide mapping of promoter-anchored interactions with close to single-enhancer resolution. *Genome Biology*, 16:156, Aug. 2015. doi: 10.1186/s13059-015-0727-9.
- [4] E. Yaffe and A. Tanay. Probabilistic modeling of hi-c contact maps eliminates systematic biases to characterize global chromosomal architecture. *Nature Genetics*, 43:1059–1065, Oct. 2011. doi:10.1038/ng.947.
- [5] B. Alberts, K. Hopkin, A. Johnson, J. Lewis, M. Raff, and P. Walter. *Essential Cell Biology*. Garland Science, 3rd edition, 2009. ISBN:978-0-8153-4130-7.
- [6] M. B. Gerstein, C. Bruce, J. S. Rozowsky, D. Zheng, J. Du, J. O. Korb, O. Emanuelsson, Z. D. Zhang, S. Weissman, and M. Snyder. What is a gene, post-encode? history and updated definition. *Genome Research*, 17:669–681, 2007. doi:10.1101/gr.6339607.
- [7] G. Khoury and P. Gruss. Enhancer elements. *Cell*, 33:313–314, June 1983. doi:10.1016/0092-8674(83)90410-5.
- [8] D. Shlyueva, G. Stampfel, and A. Stark. Transcriptional enhancers: from prop-

- erties to genome-wide predictions. *Nature Reviews Genetics*, 15:272–286, Mar. 2014. doi:10.1038/nrg3682.
- [9] L. A. Pennacchio, W. Bickmore, A. Dean, M. A. Nobrega, and G. Bejerano. Enhancers: five essential questions. *Nature Reviews Genetics*, 14:288–295, Apr. 2013. doi:10.1038/nrg3458.
- [10] J. O. Y. nez Cuna, E. Z. Kvon, and A. Stark. Deciphering the transcriptional cis-regulatory code. *Trends in Genetics*, 29:11–22, Jan. 2013. doi:10.1016/j.tig.2012.09.007.
- [11] N. F. Cope and P. Fraser. Chromosome conformation capture. *Cold Spring Harbour Protocols*, 29, Feb. 2009. doi: 10.1101/pdb.prot5137.
- [12] J. Dekker. The three ‘c’ s of chromosome conformation capture: controls, controls, controls. *Nature methods*, 3:17–21, Jan. 2006. doi:10.1038/nmeth823.
- [13] E. Lieberman-Aiden, N. L. van Berkum, L. Williams, M. Imakaev, T. Ragoczy, A. Telling, I. Amit, B. R. Lajoie, P. J. Sabo, M. O. Dorschner, R. Sandstrom, B. Bernstein, M. A. Bender, M. Groudine, A. Gnirke, J. Stamatoyannopoulos, L. A. Mirny, E. S. Lander, and J. Dekker. Comprehensive mapping of long range interactions reveals folding principles of the human genome. *Science*, 326(5950):289–293, Oct. 2009. doi:10.1126/science.1181369.
- [14] M. Hu, K. Deng, S. Selvaraj, Z. Qin, B. Ren, and J. S. Liu. Hicnorm: removing biases in hi-c data via poisson regression. *Bioinformatics*, 28:3131–3133, Sept. 2012. doi: 10.1093/bioinformatics/bts570.
- [15] E. Alpaydin. *Introduction to Machine Learning*, chapter 1, pages 1–19. The MIT Press, 2 edition, 2010.
- [16] J. Lever, M. Krzywinski, and N. Altman. Points of significance: Model selection and overfitting. *Nature Methods*, 13:703–704, Aug. 2016. doi:10.1038/nmeth.3968.
- [17] C. M. Bishop. *Pattern Recognition and Machine Learning*, chapter 2, pages 67–127, 423–455. Springer, 2006.

- [18] S. S. Khan and M. G. Madden. One-class classification: Taxonomy of study and review of techniques. *The Knowledge Engineering Review*, 29:345–374, June 2014. doi:10.1017/S026988891300043X.
- [19] Y. Sun, A. K. C. Wong, and M. S. Kamel. Classification of imbalanced data: a review. *International Journal of Pattern Recognition and Artificial Intelligence*, 23: 687–719, June 2009. doi: 10.1142/S0218001409007326.
- [20] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt. Support vector method for novelty detection. In *Advances in Neural Information Processing Systems 12*, page 582–588. MIT Press, 2000.
- [21] M. E. Reschen, K. J. Gaulton, D. Lin, E. J. Soilleux, A. J. Morris, S. S. Smyth, and C. A. O’Callaghan. Lipid-induced epigenomic changes in human macrophages identify a coronary artery disease-associated variant that regulates ppap2b expression through altered c/ebp-beta binding. *PLoS Genetics*, 11, Apr. 2015. doi: 10.1371/journal.pgen.1005061.
- [22] T.-H. Pham, J. Minderjahn, C. Schmidl, H. Hoffmeister, S. Schmidhofer, W. Chen, G. Längst, C. Benner, and M. Rehli. Mechanisms of in vivo binding site selection of the hematopoietic master transcription factor pu.1. *Nucleic Acids Research*, 41(13), May 2013. doi: 10.1093/nar/gkt355.
- [23] M. J. Iglesias, S.-J. Reilly, O. Emanuelsson, B. Sennblad, M. P. Najafabadi, L. Folkersen, A. Mälarstig, J. Lagergren, P. Eriksson, A. Hamsten, and J. Odeberg. Combined chromatin and expression analysis reveals specific regulatory mechanisms within cytokine genes in the macrophage early immune response. *PLoS One*, 7(2):91–100, Feb. 2012. doi: 10.1371/journal.pone.0032306.
- [24] M. B. Gerstein, A. Kundaje, M. Hariharan, S. G. Landt, K.-K. Yan, C. Cheng, X. J. Mu, E. Khurana, J. Rozowsky, R. Alexander, R. Min, P. Alves, A. Abyzov, N. Adleman, N. Bhardwaj, A. P. Boyle, P. Cayting, A. Charos, D. Z. Chen, Y. Cheng, D. Clarke, C. Eastman, G. Euskirchen, S. Fietze, Y. Fu, J. Gertz, F. Grubert, A. Harman, P. Jain, M. Kasowski, P. Lacroute, J. J. Leng, J. Lian, H. Monahan, H. O’Geen, Z. Ouyang, E. C. Partridge, D. Patacsil, F. Pauli, D. Raha, L. Ramirez, T. E. Reddy, B. Reed, M. Shi, T. Slifer, J. Wang, L. Wu, X. Yang,

- K. Y. Yip, G. Zilberman-Schapira, S. Batzoglou, A. Sidow, P. J. Farnham, R. M. Myers, S. M. Weissman, , and M. Snyder. Architecture of the human regulatory network derived from encode data. *Nature*, 489(7414):91–100, Sept. 2012. doi: 10.1038/nature11245.
- [25] J. Wang, J. Zhuang, S. Iyer, X. Lin, T. W. Whitfield, M. C. Greven, B. G. Pierce, X. Dong, A. Kundaje, Y. Cheng, O. J. Rando, E. Birney, R. M. Myers, W. S. Noble, M. Snyder, and Z. Weng. Sequence features and chromatin structure around the genomic regions bound by 119 human transcription factors. *Genome Research*, 22(9):1798–1812, Sept. 2012. doi:10.1101/gr.139105.112.
- [26] J. Wang, J. Zhuang, S. Iyer, X.-Y. Lin, M. C. Greven, B.-H. Kim, J. Moore, B. G. Pierce, X. Dong, D. Virgil, E. Birney, J.-H. Hung, and Z. Weng. Factorbook.org: a wiki-based database for transcription factor-binding data generated by the encode consortium. *Nucleic Acids Research*, 41:D171–D176, Jan. 2013. doi:10.1093/nar/gks1221.
- [27] W. Chanput, J. J. Mes, and H. J. Wichers. Thp-1 cell line: An in vitro cell model for immune modulation approach. *International Immunopharmacology*, 23:37–45, Nov. 2014. doi: 10.1016/j.intimp.2014.08.002.
- [28] I. Yanai, H. Benjamin, M. Shmoish, V. Chalifa-Caspi, M. Shklar, R. Ophir, A. Bar-Even, S. Horn-Saban, M. Safran, E. Domany, D. Lancet, and O. Shmueli1. Genome-wide midrange transcription profiles reveal expression level relationships in human tissue specification. *Bioinformatics*, 21(5):650–659, Sept. 2005. doi: 10.1093/bioinformatics/bti042.
- [29] B.-Y. Liao, N. M. Scott, and J. Zhang. Impacts of gene essentiality, expression pattern, and gene compactness on the evolutionary rate of mammalian proteins. *Molecular Biology and Evolution*, 23:2072–2080, Aug. 2006. doi: 10.1093/molbev/msl076.
- [30] G. McLachlan and D. Peel. *Finite Mixture Models*. John Wiley and Sons, 2000. doi: 10.1002/0471721182.
- [31] C. M. Koch, R. M. Andrews, P. Flicek, S. C. Dillon, U. Karaöz, G. K. Clelland, S. Wilcox, D. M. Beare, J. C. Fowler, P. Couttet, K. D. James, G. C. Lefebvre,

A. W. Bruce, O. M. Dovey, P. D. Ellis, P. Dhimi, C. F. Langford, Z. Weng, E. Birney, N. P. Carter, D. Vetric, and I. Dunham. The landscape of histone modifications across 1in five human cell lines. *Genome Research*, 17:691–707, June 2007. doi:10.1101/gr.5704207.

