# Understanding gene regulatory mechanisms of mouse immune cells using a convolutional neural network

by

Alexandra Maslova

B.Sc. Physics, The University of British Columbia, 2013

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

The Faculty of Graduate Studies

(Bioinformatics)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

April 2020

The following individuals certify that they have read, and recommend to the Faculty of Graduate and Postdoctoral Studies for acceptance, the thesis entitled:

**Understanding gene regulatory mechanisms of mouse immune cells using a convolutional neural network**

submitted by **Alexandra Maslova** in partial fulfillment of the requirements for the degree of **Master of Science** in **Bioinformatics**.

**Examining Committee:**

Sara Mostafavi, Statistics and Medical Genetics
*Supervisor*

Maxwell Libbrecht, Computing Science (Simon Fraser University)
*Supervisory Committee Member*

Elodie Portales-Casamar, Pediatrics
*Supervisory Committee Member*

Martin Hirst, Microbiology and Immunology
*Committee Chair*

# Abstract

Cell differentiation is controlled via complex interactions of genomic regulatory sites such as promoters and enhancers that lead to precise cell type-specific patterns of gene expression through a process that is not yet well understood. Local chromatin accessibility at these sites is a requirement of regulatory activity, and is therefore an important component of the gene regulation machinery. To understand how DNA sequence drives local chromatin accessibility within the context of immune cell differentiation, we examined a dataset of open chromatin regions (OCRs) derived with the ATAC-seq assay from 81 closely related mouse immune cell types. We trained a model that predicts local chromatin accessibility in each cell type based on DNA sequence alone, then analyzed the model to extract informative sequence features. We selected and optimized a convolutional neural network (CNN), which we named AI-TAC, that takes as input a 250bp DNA sequence of a potential OCR and predicts the relative chromatin activity at that OCR across the 81 different immune cell types in our dataset. Test dataset results showed that for many OCRs, AI-TAC is able to predict chromatin state with a high degree of accuracy, even differentiating between closely related cell types. Using CNN interpretability methods we were able to identify sequence motifs that are used by the model to make its predictions, many of which match closely to known transcription factor (TF) binding sites. The cell type - specific influence assigned to each motif by AI-TAC in many instances recapitulates prior biological knowledge about the role of these TFs in immune cell differentiation, lending credibility to our model and interpretation methods. Finally, we attempt to discern if the model detected any combinatorial activity between TFs that is predictive of chromatin accessibility. In summary, we showed that a CNN can be trained to discern the chromatin accessibility among even highly similar cell types, and that biologically relevant features can be extracted from the model using deep learning interpretation methods.

# Lay Summary

All cells in an organism contain an identical genetic blueprint (in the form of DNA sequence) that encodes all the information necessary for the organism to develop and function. However, different cell types look and behave in highly variable ways; for example, skin cells are very different from blood cells. This is possible because genes can be selectively switched on or off in different cells to establish distinct forms and functions. How genes are activated and deactivated at the right time during the process of cell differentiation - the formation of many different specialized cell types from a single stem cell - is not yet well understood.

This thesis aims to improve the understanding of the differentiation process of mouse immune cells by examining one particular regulatory "switch" that controls which genes are on in a given cell. At any one time most of the DNA in each cell is tightly packaged, but regions required to activate genes need to be open and accessible to function. Which regions of DNA are open can therefore determine which genes are switched on. In this work, we study how DNA accessibility in different immune cells is determined locally by the DNA sequence itself. Insight into this process would provide one piece of the puzzle of understanding how the DNA sequence is interpreted differently by each cell to produce distinctive cell types.

# Preface

This work was done in collaboration with members of the Mostafavi lab and the Benoist lab at Harvard Medical School. In particular, the following parts of the analysis were not performed by me:

- The mouse ATAC-seq data generation and processing was performed by members of the ImmGen consortium
- Data normalization was performed by Dr. Sara Mostafavi
- Bayesian optimization of the AI-TAC model hyperparameters was performed by Ke Ma from the Mostafavi lab
- Processing of the human ATAC-seq data was performed by Caleb Lareau from the Benoist lab
- Clustering of filter PWMs (mentioned in section 3.4.1) was performed by Ricardo Ramirez from the Benoist lab

Parts of this thesis, in particular Chapter 2 and 3, appear in a preprint article available on bioRxiv as Alexandra Maslova, Ricardo N. Ramirez, Ke Ma, Hugo Schmutz, Chendi Wang, Curtis Fox, Bernard Ng, Christophe Benoist, Sara Mostafavi, and the Immunological Genome Project. (2019) Learning Immune Cell Differentiation [34]. In particular, Figures 2.7, 2.4, 2.5, 2.6, 3.5, and 3.8a appear in Maslova et al. (2019) with some minor modifications. Additionally, portions of the text in sections 2.3, 3.2, 3.3, and 3.4 were taken from Maslova et al. (2019).

# Table of Contents

## Appendices

# List of Tables

# List of Figures

# Glossary

**CNN** Convolutional neural network

**DNN** Deep neural network

**IC** Information content

**OCR** Open chromatin region

**PWM** Position weight matrix

**TF** Transcription factor

# Acknowledgements

I would like to thank my supervisor, Dr. Sara Mostafavi, for her extensive support over the course of my graduate studies. I would also like to thank the members of my supervisory committee, Dr. Maxwell Libbrecht and Dr. Elodie Portales-Casamar, for their guidance over the course of this project.

I would like to acknowledge all the members of the Mostafavi lab who collaborated on this project with me, in particular: Ke Ma, Curtis Fox, Chendi Wang, Bernard Ng, and Hugo Schmutz. A special thank you to Dr. Christophe Benoist for all his insights and support, and Benoist lab members Ricardo Ramirez and Caleb Lareau for their contributions to the project.

I would also like to thank all the members of the Mostafavi lab, past and present, for their thoughtful feedback and overall support during the course of this project.

Finally, a big thank you to my friends and family, and especially my partner Dylan Reviczky, for being there during the challenging times.

To my dad, Igor, who's the reason I got this far.

# Chapter 1

# Introduction and Background

## 1.1 Introduction

Although all cells in multicellular organisms share the same genetic blueprint, they exhibit widely varying morphologies and perform very different functions. These differences are in large part owed to cell-specific patterns of gene expression, which are established during cell differentiation via complex epigenetic mechanisms. However, there are currently gaps in our understanding of the mechanisms by which these sequences coordinate precise programs of gene expression[41].

It is established that regulatory regions can enhance or suppress transcription of individual genes via the activity of proteins called transcription factors (TFs) that bind DNA in a sequence-specific manner. Although the sequence preference of many individual TFs has been identified, overall enhancer activity is more challenging to predict because TFs act in a cooperative manner that has not been well characterized [41]. Additionally, the frequency of these TF binding events is mediated by many epigenetic mechanisms such as DNA accessibility, the expression levels of the TF itself and other factors, and DNA methylation within the binding sequence [45].

Our work aims to help elucidate the regulatory mechanisms of noncoding genomic regions by examining cell type-specific effects of sequence on local DNA accessibility, which serves as an important mediator of regulatory activity. We analyze a dataset of open chromatin regions (OCRs) from dozens of isolated mouse immune cell types across the hematopoietic differentiation trajectory, created by the ImmGen consortium [24]. We approach this task by building a model that can predict cell-specific chromatin accessibility at potential OCR sites based on their sequence alone and then deriving the sequence features weighted most heavily by the model when making its predictions.

The remainder of this chapter is primarily dedicated to an overview

of relevant background information. We first describe what is currently known about the role of chromatin state in gene regulation as well as the mechanisms by which chromatin accessibility is established and maintained. Next, we provide an overview of methods that have been previously used to understand chromatin accessibility as a function of DNA sequence. The final section is a summary of the contributions of this thesis.

## 1.2 Biological Background

### 1.2.1 Immunological Genome Project

The Immunological Genome Project (ImmGen) is a collaboration between immunology and computational biology research groups that aims to thoroughly characterize the gene regulatory networks of the entire mouse immune system. By generating a large number of genome-wide datasets across a wide range of immune cell types, the consortium intends to build a comprehensive understanding of cell type-specific and condition-specific regulatory mechanisms. The consortium has established rigorously standardized protocols for animal care, cell isolation and data generation for all participating laboratories in order to produce consistent, high quality datasets[23].

### 1.2.2 Transcriptional Regulation

Regulatory regions of eukaryotic genomes are broadly classified into transcription start site-proximal promoters and cis-regulatory elements such as enhancers, silencers and insulators[35]. The core promoter is located in the immediate vicinity of a gene's transcription start site (TSS). The promoter sequence is sufficient to recruit general transcription factors, RNA Polymerase II and other proteins that form the transcriptional machinery that copies the DNA sequence of the gene into RNA[2]. However, transcription of a gene is often weak without additional regulatory activity at distal enhancer sites[41].

Enhancer sequences are characterized by their ability to affect transcription rates at large distances to their target TSS and independently of their relative orientation[41]. Enhancers function by recruiting TFs that recognize and bind short (typically 6-10bp) sequence motifs[41]. These TFs can then increase transcription rates of the target gene by recruiting transcription complexes to the gene promoter, either directly or via their binding partners. These long-range interactions are enabled by DNA looping which brings active enhancers into spatial proximity of their target promoter[41].

Some TFs act instead to repress transcription rates by interfering with the recruitment of transcriptional machinery, thus allowing certain enhancers to act as silencers under specific conditions[35]. Transcription at a single TSS is regulated via the coordinated activity of multiple enhancer and silencer sites[45].

Because TF motifs are short and abundant throughout the genome, the binding affinity of individual TFs alone does not account for precise cellular programs such as differentiation. Instead, the activity at a single enhancer site is the cumulative outcome of the binding of many different TFs. In the simplest case, the observed effect of multiple TFs is additive - the activity at a given enhancer sequence is proportional to the concentrations of the individual TFs for which binding sites are present. However, more precise regulatory "switches" require cooperativity between TFs. In some cases, TF cooperativity results from physical protein-protein interaction between adjacently bound TFs that enhances their binding affinity to their respective binding sequences. Alternatively, two TFs bound to the same enhancer may be responsible for recruiting the same cofactor or different components of a multi-protein complex. TFs may also facilitate the binding of other factors by triggering local DNA bending, or even by changing the sequence specificity of another TF through protein-protein interactions[45].

There are multiple models for how the sequence architecture of an enhancer enables its function, with evidence that each model may apply to some enhancers but not others. The enhanceosome model proposes that a specific, ordered protein interface is necessary for the full activation of an enhancer, requiring strict motif positioning within its sequence[45]. Most enhancers, however, do not seem to exhibit such strict motif grammar, instead containing variable motif subsets and spatial arrangements. The billboard model accounts for this flexible motif grammar by proposing that although some TFs bind cooperatively, others may act on an enhancer in an additive or independent way, thus allowing for the relative order of some motifs to change without significantly impacting enhancer activity[45]. Another proposed model is the "TF collective" that suggests that protein-protein interactions can recruit necessary TFs in cases where their motifs are not themselves present in the enhancer sequence. Observations show that different TF composition and ordering can lead to very similar enhancer activity patterns in some cases, but in other cases these activity patterns are sensitive to changes in motif positioning even for the same TF set, providing evidence that none of the described models apply universally to all enhancer sites[45].

Insulators are specific types of regulatory elements that are responsi-

ble for forming and maintaining higher order DNA structures that enable enhancer-promoter interactions. They are broadly categorized into two types: barrier and enhancer-blocking[16]. Barrier insulators maintain active chromatin regions by blocking the spread of heterochromatin formation, thus ensuring certain genomic regions are not transcriptionally silenced. Enhancer-blocking insulators were so named because of their ability to block enhancer-promoter interactions when placed between the two. They function by anchoring themselves to nuclear structures or to other insulators via TF interactions, thus establishing DNA loop domains. In vertebrates, CTCF is an especially prevalent TF at insulator sites that is able to form bonds with itself and other nuclear proteins. The formation of these loop domains brings some enhancers and promoters within spatial proximity of each other while blocking others from interacting[16].

### 1.2.3 Chromatin Accessibility

At any given time, the majority of eukaryotic DNA is packaged into a highly condensed chromatin structure that makes it inaccessible for binding by most TFs[31]. The basic unit of chromatin is called a nucleosome, and is composed of approximately 150bp of DNA wrapped around a protein octamer comprised of four types of core histones. Nucleosomes are formed along the DNA string like beads, allowing the DNA to be further condensed by linker histones that bind inter-nucleosomal DNA and interact with the core histones. For processes that require DNA-protein interactions, such as DNA transcription, replication and repair, the DNA must be made accessible[4]. Regulatory genomic regions that operate via TF binding likewise require chromatin accessibility to fulfill their regulatory functions. Unsurprisingly, active enhancers, promoters and insulators coincide with nucleosome-depleted regions of the genome, which are fully accessible stretches of DNA typically the length of one nucleosome (150-250bp)[31].

These open chromatin regions (OCRs) are established and maintained through several different mechanisms. Targeted nucleosome repositioning can be accomplished by a special class of TFs called pioneer factors that are able to bind nucleosomal DNA in a sequence-specific manner, and then displace nucleosomes either independently or by recruiting active chromatin remodelers. Alternatively, some TFs may bind inter-nucleosomal DNA and then initiate processes that destabilize and displace neighboring nucleosomes. Another proposed mechanism is that TFs bound to an accessible enhancer may recruit chromatin remodelers that evict nucleosomes at distal regulatory sites. Finally, nucleosomes in regulatory regions have relatively

high turn-over rates and TFs present in high enough concentrations may be able to passively out-compete histone proteins for DNA binding, thus ensuring local chromatin accessibility[31].

There is additionally evidence that CpG islands, which are approximately 1kb long regions of the genome with high GC content, are predisposed to low nucleosome occupancy[11]. These regions have high overlap with a subset of gene promoters, and may be responsible for maintaining chromatin accessibility for this set of regulatory sites[11].

### 1.2.4 Measuring Chromatin Accessibility

Here we describe some of the more popular methods for assaying genome-wide chromatin accessibility. Although each of these assays has different kinds of sequence bias, the accessible chromatin regions identified via all these methods are generally well correlated[31].

The earliest method for detecting large-scale chromatin accessibility patterns, published in 2006, used DNase I proteins to preferentially cleave only accessible DNA, then hybridize the DNA fragments onto tiled microarrays. Because microarrays are limited in throughput, this method is inadequate for measuring accessibility genome-wide. DNase I hypersensitive site sequencing (DNase-seq), developed several years later, overcomes this problem by using high-throughput short-read sequencing to characterize the accessible DNA fragments produced by DNase I cleavage[31].

Assay for transposase accessible chromatin with high-throughput sequencing (ATAC-seq) is a more recent method for profiling chromatin state genome-wide[6]. It uses hyperactive Tn5 transposase proteins that preferentially cut nucleosome-free DNA and simultaneously insert sequencing adaptors into the ligated DNA fragments. The tagged DNA fragments are then amplified and sequenced. After the sequencing reads are aligned to the reference genome, OCRs can be identified by finding areas with high numbers of aligned reads[6]. This protocol is simple and much faster than DNase-seq, taking hours rather than days to generate the sequencing libraries. Additionally it can profile samples with much smaller quantities of cells, on the order of thousands of cells rather than hundreds of thousands required for DNase-seq[31].

MNase-seq and NOMe-seq are another two recently developed methods for characterizing chromatin state[31]. MNase-seq or micrococcal nuclease sequencing uses MNase endonuclease/exonuclease proteins to digest internucleosomal DNA. The remaining DNA fragments, that were protected by histones during digestion, are sequenced and mapped to the reference genome

to identify DNA regions occupied by nucleosomes[31].

NOMe-seq stands for nucleosome occupancy and methylome sequencing and can be used to profile both methylation and chromatin state simultaneously. It utilizes a viral methyltransferase that methylates GC dinucleotides rather than the CG dinucleotides that are normally methylated in humans and mice. Because only accessible DNA is methylated, whole-genome bisulfite sequencing can then be used to identify OCRs as well as methylation at CpG sites. A much higher number of reads is required for NOMe-seq than the other methods described above because it involves sequencing the whole genome rather than selected fragments. However, this process also eliminates enrichment bias which is present in the other methods, thus allowing for a better quantification of chromatin accessibility[31].

## 1.3 Related Work

Here we provide an overview of methods that have been used to understand the effect of DNA sequence on local chromatin accessibility. This is by no means an exhaustive list, but we attempt to mention all the main categories of methods used for this task. Notably, we do not detail any of the approaches that exploit multi-omics data or genome annotation (rather than sequence information alone) to understand the mechanisms of local chromatin accessibility.

We start by describing approaches that rely on identifying individual TF binding sites within open chromatin regions, either by using known motifs or discovering them *de novo*. Next, we delve into machine learning methods designed to classify the chromatin state at putative regulatory regions as either accessible or inaccessible based on their sequence. The features most heavily weighted by these models can then be used to understand which motifs within the sequence are biologically meaningful predictors of chromatin state. Finally, we provide an overview of deep learning models designed to predict chromatin state from DNA sequence and the interpretation methods that have been applied to these models to understand how the input features are weighted by the model when making predictions.

### 1.3.1 Identification of Transcription Factor Binding Sites

One way of understanding regulatory DNA sequences is by identifying the presence of TF binding sites. For example, the over-representation or under-representation of certain motifs in OCRs can be used to infer which TFs are important for the regulatory activity of a given cell type. TF binding sites

can be identified using their known motifs, or they can be discovered *de novo* via methods that find over-represented k-mers within the input data.

TFs bind short motifs, typically 6-10bp, that can vary by 1 or 2 bases from a consensus binding sequence[47]. It's hypothesized that this flexibility in binding preference, which leads to a range of affinities between a TF and its binding sites, enables precise control of transcription rates rather than acting as a binary on/off switch. To capture this variability in binding sites, TF motifs are typically represented as position weight matrices (PWMs) with 4 entries for every position along the length of the motif, one for each nucleotide. The entries commonly used are observed nucleotide counts at each position (these are also called position frequency matrices), the probabilities of observing each nucleotide at each position (also called position probability matrices), or log-odds scores for each nucleotide at the given position, defined as:

$$\log_2(p_{ij}/b_j)$$

where $p_{ij}$ is the probability of observing nucleotide $j$ at position $i$, and $b_j$ is the probability of observing nucleotide $j$ in the background model[19, 47].

There are a number of methods designed to identify the presence of known TF binding sites within a set of input sequences, for example sequences of OCRs. FIMO[18] (Find Individual Motif Occurrences) is one example of this method type that scans a database of sequences with a set of known TF PWMs, which can be obtained from databases such as CIS-BP[50] or JASPAR[14]. For each PWM, FIMO computes a similarity score to every position within the input sequence, then converts it to a p-value reflecting the probability of obtaining a similarity score at least as high on a random sequence. These p-values can then be used to filter for high-confidence motif occurrences.

To understand how these known motif occurrences relate to chromatin accessibility additional statistical analysis is required. ChromVAR[39] is a method designed to correlate motif instances identified with methods such as FIMO to sample-specific chromatin state. It takes as input a set of motif occurrences, along with aligned sequencing reads from the chromatin accessibility assay and locations of open chromatin peaks. For each motif a deviation score is computed corresponding to the read counts of all OCRs containing that motif minus the expected read counts at these OCRs (based on accessibility across all cells), and divided by the expected read counts. A high positive score indicates that the TF motif is highly correlated with accessible chromatin in a particular cell type, while a large negative score suggests a correlation with non-active regions. A deviation score can even

be computed for pairs of motifs to detect TF cooperativity, although there are computational limitations on the number of TF pairs that can be tested.

Because PWM-scanning approaches are limited to identifying instances of known motifs only, it can be advantageous to instead detect over-represented sequences within a dataset of interest and assemble those into novel motifs. HOMER[22] is a software package that identifies k-mers that are enriched in a target sequence set compared to a background sequence set and assembles them into *de novo* motifs. It then optimizes the PWMs of these motifs to be maximally enriched in the target sequence database using the cumulative hypergeometric distibution function to measure enrichment. This method can be applied to find motifs associated with chromatin accessibility by comparing OCR sequences to regions of non-accessible DNA.

### 1.3.2 Machine Learning Approaches

Machine learning models trained to classify accessible versus inaccessible genomic regions are able to simultaneously learn many motifs that are predictive of regulatory activity in the observed data. This class of methods typically requires the transformation of the input sequences into a k-mer feature representation. Here, we describe three such methods along with the advantages and disadvantages of each.

SeqGL[40] represents input sequences using k-mers with wildcard characters to train a logistic regression model for classifying regions as accessible or inaccessible. It uses a sparse group lasso constraint on k-mers clustered by similarity to impose similar weights on k-mers that are likely to belong to the same motif. This model is easily interpretable as the weights corresponding to each k-mer represent the importance of that k-mer to the prediction task at hand. Similar k-mers can furthermore be assembled into novel motifs.

A popular method called gkm-SVM[17] constructs a similarity kernel between all the input sequences using gapped k-mers similar to the wildcard k-mers of SeqGL. The kernel then serves as input into a support vector machine that is trained to classify genomic regions as accessible or inaccessible. Although gkm-SVM has excellent classification performance, the kernel feature representation makes the interpretation stage more challenging.

The Synergistic Chromatin Model (SCM)[21], another k-mer based approach, is an L1-regularized generalized linear model that makes per-base pair predictions of chromatin accessibility based on the input sequence. It represents its inputs as matrices indicating the presence of all possible k-mers at every position in the sequence. Unlike gkm-SVM and SeqGL, which

give binary predictions of chromatin accessibility for pre-defined sequences of interest, SCM outputs a quantitative prediction of chromatin state at a single base-pair resolution. Because SCM retains k-mer positions within its input feature space, it can reveal the importance of motifs at specific orientations to the output position. SCM is additionally easily interpretable, since weights assigned to each k-mer can be directly examined and important k-mers can be assembled into PWMs. However, due to the large size of the input, SCM is somewhat computationally intensive to train.

Although these approaches model additive effects of multiple motifs within the same sequence, they do not utilize information about the relative positioning of motifs to make predictions. Motif spacing and orientation is known to be an important factor in cooperative TF interactions, and the importance of some motifs may be missed entirely if synergistic effects with other TF binding sites are not considered.

### 1.3.3 Deep Neural Networks in Genomics

Deep neural networks(DNNs) are a powerful class of predictive models that have become very widespread in genomics data analysis. DNNs consists of multiple layers of "neurons", with the output of one layer serving as the input to the consecutive layer. Each neuron is a linear function of the inputs to that particular layer with a non-linear transformation (called an activation function) applied to the result (Figure 1.1). These activation functions enable DNNs to approximate complex non-linear functions rather than simply being linear transformations of the input.

DNNs are hugely flexible with regards to the output they can produce. They can be used as binary classifiers or predictors of continuous values, and the output can be of arbitrary length. All the internal parameters of a DNN are learned during a supervised training phase. Due to the typically large number of parameters in one network, a large set of labeled data is required to train deep models.

For tasks requiring predictions based on DNA sequence, convolutional neural networks(CNNs) are the most widely used deep learning model. CNNs are able to model complex, non-linear relationships between the features of the input sequence while accounting for their relative spacing. They have been successfully used to predict genomic data such as chromatin state[29, 53], TF binding[1], and gene expression[52] on the basis of DNA sequence alone.

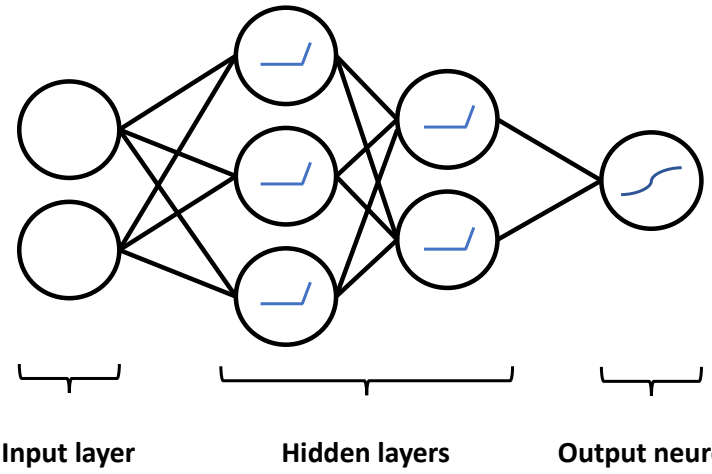



Figure 1.1: Schematic of a basic DNN. Hidden fully connected layers with the ReLU activation function, and an output neuron with a sigmoid activation function.

### Convolutional Neural Networks for Predicting Chromatin State

CNNs are a special class of DNNs containing one or more layer of convolutional operators, typically at the start of the network. A convolutional operator (or filter) is a small matrix of weights that scans the input and computes the sum of element-wise multiplication between the input and the filter. During model training, these convolutional filters are optimized to recognize low-level features of the input that are informative for the particular classification task. In the case of CNNs trained on DNA sequence data, the first layer convolutional filters correspond to short sequence motifs.

A benefit of these models is the lack of pre-processing required on the input - a CNN can process a DNA sequence of any length with no manual input as to the relevant features. In contrast, the methods previously described require either a pre-specified set of features (in the case of PWM-scanning methods or "bag of k-mers" methods) or an assumption on the length of relevant features (in the case of *de novo* motif discovery methods). CNNs, on the other hand, can detect relevant features of arbitrary length within the input DNA sequence.

In this section, we describe several CNN model architectures that have been used to predict chromatin accessibility. The first three examples,

DeepSea[53], OrbWeaver[3] and Basset[29], are basic CNNs consisting of only convolutional and fully connected layers. Then we describe two models, DanQ[38] and Basenji[28], that incorporate additional features into their architectures to improve on the performance of vanilla CNNs.

DeepSea[53], developed in 2015, was the first CNN model trained to predict local chromatin accessibility from DNA sequence. DeepSea was trained on data from hundreds of different cell lines, containing assays of DNase I-hypersensitive sites, histones marks, and TF binding events. Training a model to predict multiple outputs at once, called multi-task learning, provides more statistical power for optimizing parameters corresponding to lower-level features (for example, TF motifs that are relevant for both TF binding and chromatin accessibility predictions). Each model input is a 1000bp sequence centered on a TF binding event and corresponding binarized labels of TF binding, chromatin accessibility and histone marks across the different cell types. DeepSea consists of 3 convolutional layers, that encode sequence motifs and their interactions, followed by a hidden fully connected layer. The final output layer has a sigmoid function applied to it, which normalizes all the values to be between 0 and 1. These values can then be interpreted as probabilities - in this case, probabilities of TF binding events and open chromatin peaks in the given sequence.

The OrbWeaver[3] model, trained on ATAC-seq, DNA methylation, and mRNA expression data from induced pluripotent stem cells, also utilizes a very similar architecture to DeepSea to predict chromatin accessibility. However, rather than learning the first layer filters during training the model is initialized with PWMs of 1320 known TF motifs. Each first layer filter has a direct correspondence to a TF, and the relative importance of each filter can be derived using one of the deep learning interpretation methods, which are described in more detail in the next section.

The Basset[29] model is very similar to DeepSea, but was trained to predict only DNA accessibility based on 600bp sequences at DNase I hypersensitive sites. The model was trained on 164 different human cell types simultaneously, also utilizing the multi-task learning approach. It's architecture varies slightly from DeepSea, with different sized convolutional filters and an additional hidden fully connected layer before the final output.

The DanQ[38] model performs significantly better than DeepSea on the same dataset by integrating a CNN with a recurrent neural network. Recurrent neural networks(RNNs) process the elements of an input sequence in consecutive order while maintaining a memory of the previous elements, enabling RNNs to model spatial dependencies between sequence features. DanQ consists of one convolutional layer used for learning sequence motifs

followed by a bi-directional long short-term memory network, a type of RNN that combines the output of two networks that process the sequence from opposite ends.

Although these methods are significantly better at making predictions than non-deep learning approaches, they are limited to short input sequences and therefore cannot account for long-range interactions between different enhancer and promoter sites. Basenji[28] can process input sequences of 131 kilobases to predict, along with other datatypes, the average DNase-seq read depth in 128bp bin segments across the entire input sequence. This model is thus able to incorporate information from a much larger sequence context than the CNNs described above and make predictions that are quantitative rather than binary. Basenji is able to process such long genomic regions due to the addition of dilated convolutions after the standard convolutional layers. Dilated convolutions are very similar to the standard convolution operator but they contain gaps, so rather than processing adjacent features they convolve features at pre-defined spaced intervals. This enables them to share information across long distances within the sequence that would otherwise require a much deeper network.

### Interpretation of Deep CNNs

Although CNNs are highly effective predictive models, they are typically considered "black boxes" because understanding how they utilize the input features to make predictions is nontrivial. However, there is a growing field dedicated to developing methods for interpreting deep neural networks. Here we describe some examples of methods that have been used for CNN interpretation in genomics. They are broadly split into three categories: perturbation-based, backpropagation-based and reference-based approaches[13].

Perturbation-based approaches are based on perturbing the model input and measuring the effect on the prediction. A common example of this is *in-silico* mutagenesis, where each base pair of the input sequence is changed one at a time and the effects on model predictions are quantified. This approach was used by the authors of DeepSea[53] and Basset[29] to identify the most important bases in the sequence and the most impactful nucleotide substitutions.

Another approach based on perturbation is the analysis of first layer filters of the Basset model[29]. Because each first layer filter of a CNN learns a weight matrix of a short motif, these filters can be directly interpreted as PWMs of TF binding sites that are informative to the model. To measure

the relative importance of these PWMs, each filter is removed from the model and the change in predictions serves as a motif influence score[29].

These perturbation-based approaches, particularly *in-silico* mutagenesis, are very computationally intensive since hundreds or thousands of model predictions need to be calculated to characterize each input sequence. More importantly, these methods do not account for any redundancies in the features - for example, if a sequence contains two motifs that have the same predictive value for the model, neither will be assigned importance by these methods. The input features may also have saturated their contribution to the output, in which case importance of those features may be underestimated.

Backpropagation-based approaches such as saliency maps[43] and guided backpropagation[46] essentially use the gradient of the model output with respect to the input features as a measure of importance. These methods are significantly faster than perturbation-based approaches as the gradient for all inputs can be computed in a single pass through the network. However, they do not address the issue of redundant features or the feature saturation problem, as the gradient would be zero in this realm. Additionally, the ReLU operator makes it difficult to estimate negative feature contributions when using gradients.

DeepLIFT[42] and Integrated Gradients[48] attempt to solve these issues by computing the contribution of all the input elements with respect to some reference. Integrated gradients computes the integral of the model gradients as the input features are scaled from some reference (for example 0) to their current values. DeepLIFT, on the other hand, computes how much the change from reference in each input feature contributes to the change in the output neurons. These values can be calculated in a highly computationally efficient manner similar to the backpropagation approaches. These reference-based methods are able to mitigate the issue of redundant contributions of input features and the neuron saturation problem, with the caveat that they require a somewhat arbitrary choice of reference.

## 1.4 Thesis Contributions

Because DNA accessibility is a pre-requisite for activity at regulatory genomic regions and transcription start sites, context-specific chromatin state plays an important role in the regulation of gene expression. Understanding how DNA sequence specifies local chromatin state can therefore provide insights into the mechanisms of sequence-directed transcriptional regulation.

We focused our efforts on a unique dataset of genome-wide chromatin accessibility profiles of 81 different mouse immune cell types generated using ATAC-seq by the ImmGen consortium. Unlike most previous efforts to profile chromatin state, this data was generated from isolated cell types rather than bulk tissue samples or cultured cell lines. It profiles a large number of different cell types spanning the differentiation trajectory of the immune system, allowing us to analyze subtle differences in chromatin accessibility between closely related cells.

To understand the link between DNA sequence and accessibility, we trained a model that predicts chromatin state from DNA sequence at putative OCRs. We then extracted the most predictive sequence features used by the model, which we expect to reflect the biologically important components of regulatory sequences. Due to their recent widespread success in the field of genomics we chose to use a convolutional neural network (CNN) as the predictive model. We trained a CNN (named AI-TAC) to predict, for a given OCR, the relative chromatin state for all 81 cell types using only the DNA sequence at that OCR.

To understand which relevant sequence features are learned by the model during training, we extracted the PWMs associated with each convolutional filter in the first layer of AI-TAC and computed several metrics to characterize the importance of each filter. We found that most of the informative filters matched closely to motifs of known TFs, many of which are known to have important roles in immune cell differentiation. Lastly, we tried to understand how combinations of first layer filters are used by the model to incorporate sequence context and motif cooperativity into its predictions.

The following is an outline of the content in the remaining chapters of this thesis:

- Chapter 2 details our selected model and the dataset used to train it. We first describe the steps of generating, processing and normalizing the ATAC-seq data used for training the model. We then outline, in detail, the CNN architecture we chose for predicting cell type-specific chromatin state. The final section shows a number of experiments validating the predictive performance of the model.

- Chapter 3 focuses on model interpretation with the goal of identifying predictive sequence features that might help shed light on regulatory mechanisms of chromatin accessibility. We first describe how we extract and characterize the first layer filter motifs of AI-TAC, and their relation to known TF motifs. Next, we show that using our defined

metrics of filter importance, we can replicate the models performance with only 1/3 of the first layer filters. Finally, we attempt to extract the combinatorial logic used by the model that could provide insight into TF cooperativity *in vivo*.

- Chapter 4 discusses our findings and proposes some directions for future work that may improve on AI-TACs predictive performance and interpretability.

# Chapter 2

# Part I: Model Architecture and Performance

In this chapter we describe AI-TAC, a CNN model trained on ATAC-seq data to predict cell type-specific local chromatin state from short DNA sequences. Section 2.1 provides an overview of the steps used to generate, process and normalize the dataset used to optimize AI-TAC. In section 2.2 we describe the details of the different components of AI-TAC, the overall model architecture, and the model optimization procedure. In section 2.3 we show the results of several experiments designed to test the predictive performance of AI-TAC. We compare AI-TACs performance on real data versus several simulated "null" datasets. Additionally, we show the robustness of the model optimization step with respect to the choice of training and test set split. Finally, we test the ability of AI-TAC to make meaningful predictions on a human cell ATAC-seq dataset not used in training the model.

## 2.1 Data

The dataset consists of bulk ATAC-seq assays performed on 81 mouse immune cell types belonging to six different immune lineages: $\alpha\beta$T, $\gamma\delta$T, B, lymphoid, myeloid and stem cells.

### 2.1.1 Data Generation

Each cell type was isolated from genetically identical C57BL/6 mice from the Jackson Laboratory by one of the ImmGen consortium immunology groups with flow cytometry using standardized procedures. All library construction and sequencing was performed at the core ImmGen lab using Illumina NextSeq 500 instruments and paired-end reads. A more detailed description of the data generation process can be found in Yoshida et al, 2019[24].

### 2.1.2 Processing and Normalization

The ATAC-seq peaks for all 81 cell types were obtained from Yoshida et al, 2019[24]. The processing steps, as described in the paper, are as follows. The sequenced reads were mapped to the mm10 mouse genome using bowtie, and non-unique, ChrM mapping and duplicate reads were filtered using samtools and Picard Tools. Peak calling was performed with MACS2 software using paired-end reads spanning less than 120 bp. Significant peaks were used to determine OCRs of length 250 bp centered on each peak summit. OCRs located in blacklisted genomic regions and ChrM homologous regions were then filtered out. The "peak height" of each OCR corresponds to the number of reads mapping to it.

The raw read counts were log2-transformed after adding a pseudocount of 2, and normalized by quantile normalization across the cell types by Dr. Sara Mostafavi. For the purposes of our analysis we additionally filtered out all OCRs on the X and Y chromosomes (because samples came from both male and female mice) and any OCR sequence with undetermined bases in the reference. This resulted in a total of 327,927 OCRs where a peak was identified in at least one of the 81 cell types.

## 2.2 The AI-TAC Model

### 2.2.1 Model Architecture

Bayesian optimization of model hyperparameters was performed by Mark Ma using the skopt package[44] to determine the optimal model architecture for our problem. We found models with architectures similar to Basset[29] performed best, and chose a very similar model with three convolutional and two fully connected hidden layers. The details of AI-TACs architecture are described below and are shown in figure 2.1.

Each model input is a 251bp OCR sequence transformed via one-hot-encoding into a 4x251 binary matrix such that each row corresponds to a nucleotide (A, T, G, C) and each column corresponds to a position along the sequence, as demonstrated in Figure 2.2. For example, if the sequence starts with a C, the first column of the one-hot-encoded sequence would contain a 1 in the 4th row and 0s in rows 1 through 3. We additionally pad the input sequence with 0's on either end to a total length of 271 to ensure every position is properly scanned by the first layer convolutional filters.

The first three hidden layers of the network are composed of convolutional filters. Each filter is a matrix that "scans" the length of the sequence

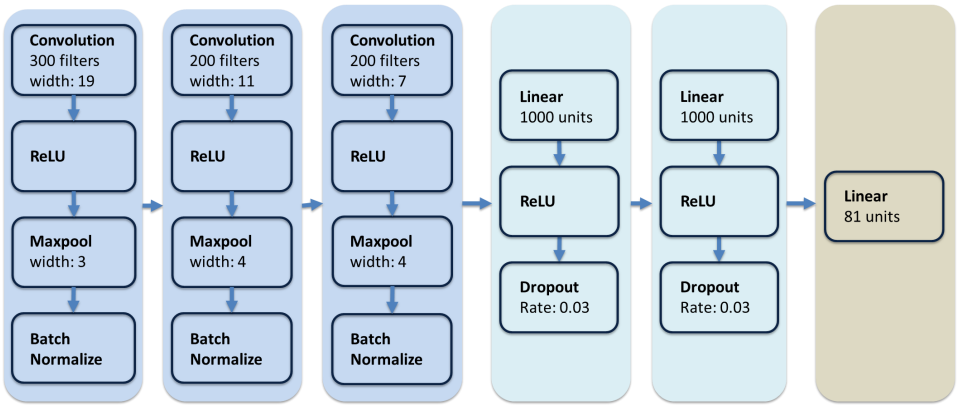



Figure 2.1: The architecture of the AI-TAC model.

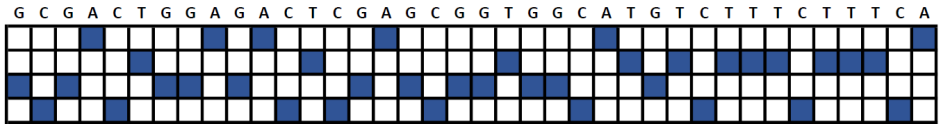



Figure 2.2: Conversion of sequence into one-hot encoded matrix.

to detect a specific pattern. More formally, for input $X$ of length $L$ with $N$ input channels ($N$x$L$ matrix) and a convolutional filter $W$ of width $M$ ($N$x$M$ matrix) the output for that filter is a vector of length $L - M$ where the entry at position $i$ is:

$$\text{convolution}(X)_i = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} W_{nm} X_{n,i+m} + b \tag{2.1}$$

where $b$ is a bias term that is optimized for each filter.

The first layer of AI-TAC consists of 300 convolutional filters of dimension 4x19 that scan the input for a particular sequence (i.e. motif). The second layer is 200 filters with dimensions 300x11 that detects relationships between each of the first layer filters, and the third layer is 200 filters with

dimensions 200x7.

The output of each convolutional layer is passed through an activation function which introduces non-linearity to the model. AI-TAC utilizes the rectified linear activation function (ReLU) which thresholds values such that all inputs below 0 are set to 0:

$$\text{ReLU}(x) = \begin{cases} x \text{ if } x \geq 0 \\ 0 \text{ if } x < 0 \end{cases}$$

The output of each convolutional layer is condensed using the maxpooling operation, which computes the maximum value within a contiguous window of the convolutional layer output. For a maxpool operator of width $w$ and stride $w$ (equal to it's width) applied to a sequence $X$, the output at position $i$ is computed as:

$$\text{maxpool}(X)_i = \max\{X_{wi}, X_{wi+1}, ..., X_{wi+w}\} \tag{2.2}$$

The maxpool layer reduces the intermediate output of the layer which reduces the number of parameters in the next layer making the model optimization more robust. It also allows for more variability in the spacing of input features thus accounting for any biological variation in motif spacing. The convolutional layers of AI-TAC have maxpooling of width 3, 4, and 4, respectively, with stride equal to the width of the maxpool operator.

Next, we applied batch normalization, which enables faster model optimization and acts as an implicit regularizer of the model weights by normalizing the outputs of hidden layers to reduce their variance [27]. Each activation unit fed into the batch normalization layer is mapped to a corresponding output $Y_i$ via the following equation:

$$Y_i = \frac{X_i - E[X_i]}{\sqrt{Var[X_i] + \epsilon}} \cdot \gamma + \beta \tag{2.3}$$

$E[X_i]$ and $Var[X_i]$ are the mean and variance, respectively, of activation $X_i$ over a training mini-batch. During model evaluation (as opposed to model training) these values get substituted with the overall mean and variance computed over the training set. Parameters $\gamma$ and $\beta$ are learned during the training phase.

The output of the convolutional layers is then fed into a traditional fully connected network with 2 hidden layers. As the name implies, in a fully connected layer each input neuron is connected to each output neuron via a

weight. The output vector $Y$ of a fully connected layer consists of neurons $Y_j$ computed in the following way:

$$Y_j = \sum_{i=i}^{I} w_{ij} X_i \tag{2.4}$$

where $X$ is the layer input. AI-TAC contains two hidden fully connected layers each with 1000 output units. The second of these layers maps to the model's final 81-unit output(corresponding to each of the 81 cell types) via a linear transformation.

### 2.2.2 Model Training

The model was trained by minimizing the loss function below with respect to the model weights using the Adam optimizer [30]. We chose cosine distance as the objective function:

$$f(x) = 1 - \frac{\hat{Y} \cdot Y}{|\hat{Y}||Y|} \tag{2.5}$$

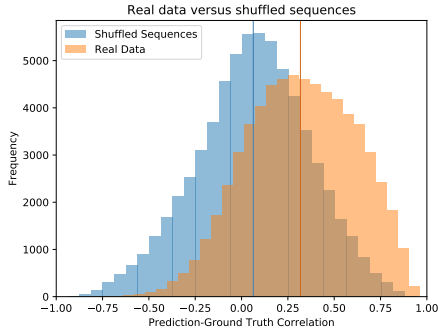
This loss maximizes the Pearson correlation between the predicted and observed chromatin activity state across the 81 cell types for the training set OCRs. We chose this loss to emphasize accurate prediction of OCRs with differential activity profiles across the cell types, to aid in identifying sequence features correlated with differential activity during the model interpretation stage.

AI-TAC was implemented in PyTorch version 0.4.0. The training was performed using the Adam optimizer [30] for 10 epochs with a learning rate of 0.001 and a mini-batch size of 100. As an additional form of regularization we implement drop-out of 3% on the two hidden fully connected layers. Drop-out sets a random portion (in our case 3%) of the input neurons to zero for each training sample, which helps prevent over-fitting of the model to the training set [25].

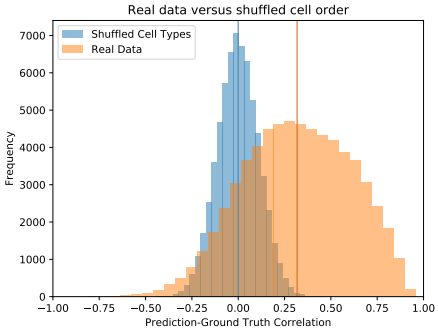
## 2.3 Model Performance

### 2.3.1 Comparing to Randomized Null

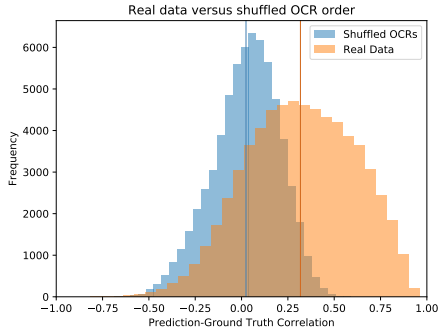
We first trained AI-TAC on a randomly selected subset of OCRs consisting of 90% of the 327,927 OCRs in our dataset. We benchmarked the performance of AI-TAC on the remaining 10% of our dataset against simulated




(a)




(b)




(c)

Figure 2.3: Performance of the model (measured by Pearson correlation) on real and shuffled mouse ATAC-seq data. (a) randomization by shuffling the sequences, (b) by permuting the chromatin accessibility profiles, and (c) by shuffling the assignment of each OCR to its accessibility profile

"null" datasets to ensure it had learned some meaningful predictive features. We compared AI-TAC to three different models, each trained on data randomized in one of the following ways:

- Randomly shuffled 251 base pair input sequences
- Randomly shuffled ATAC-seq activity vectors of length 81
- Randomly permuted order of input sequence and output vectors in the dataset

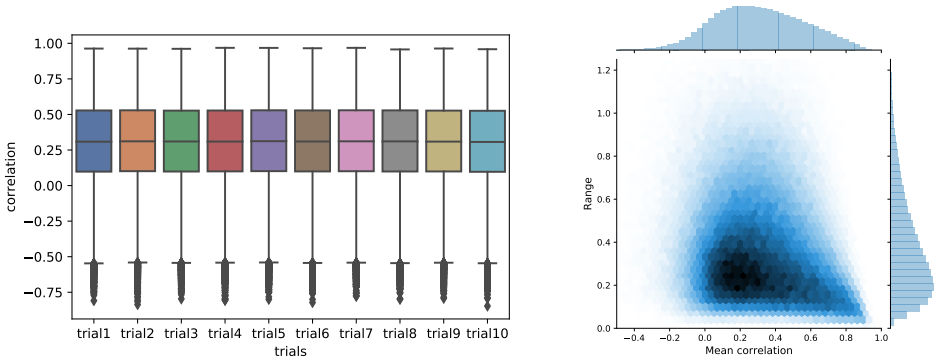
Figure 2.7 shows the correlations between model predictions and ATAC-seq derived activity values for each of the test set samples, comparing the

AI-TAC results to each of the three "null" models. We observed that the average correlation of the "null" model predictions is close to zero, as expected since there shouldn't be any detectable patterns in the randomized data. Notably, the shuffled sequence model has the highest average correlation compared to the other "null" models, reflecting the fact that GC content on its own is somewhat predictive of chromatin state because CpG islands are generally ubiquitously accessible[11]. In contrast, the average AI-TAC prediction correlation is 0.32, indicating that there is real signal and structure in the dataset that the model is able to exploit to make predictions.

### 2.3.2 10x10 Cross-validation

To ensure the results we obtained with AI-TAC were not highly sensitive to training set selection and model initialization, we performed a 10-fold cross-validation of the entire dataset a total of 10 times, thus obtaining 100 different models. In this way, each of the 327,927 OCRs was considered as a test OCR by ten different trained models.

We observed very stable test set distributions across all the models (Figure 2.4a). Additionally, our results show that OCRs that were well predicted had the most stable predictions across the 10x10 cross-validation trials (Figure 2.4b), indicating that the model is able to consistently learn highly predictive features for that subset of OCRs regardless of training set selection.




(a) (b)

Figure 2.4: (a) The prediction correlation for each OCR in the dataset when it was part of the test set, for all 10 cross-validation trial. (b) The mean test prediction correlation for each OCR across 10 independently trained models on the x-axis versus the range of correlation values on the y-axis.

### 2.3.3 Chromosome Leave-out

Enhancers regulating the activity of the same gene have been shown to have phenotypic redundancy[37]. To ensure our results were not overly optimistic due to the presence of highly similar OCR sequences (at functionally redundant enhacers) in both the training and test set we performed a chromosome leave-out validation experiment. We tested the robustness of our model by leaving each of the 19 mouse autosomes as a test set and training the model on the remainder of the data. The results in figure 2.5 show that all 19 of these models have very similar test set prediction correlation distributions to AI-TAC, indicating that our model is not significantly impacted by the choice of training and test sets.

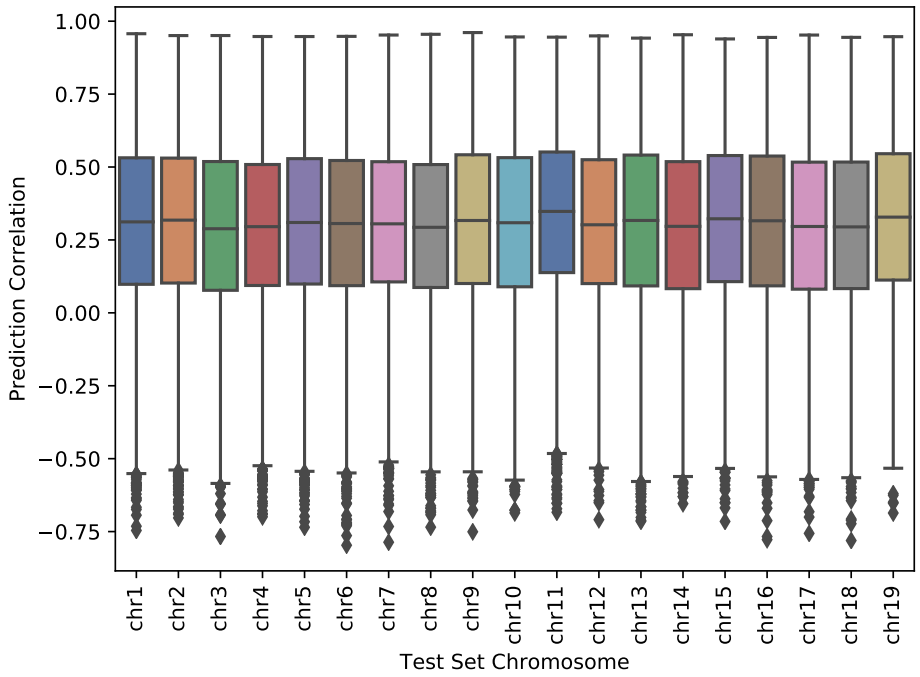



Figure 2.5: Boxplot showing the test performance of 19 separate models, trained by leaving each of the 19 autosomes out once.

### 2.3.4 Predictions Vary by OCR Type

We were further interested in characterizing the differences between OCRs that were predicted well versus those that were predicted poorly. When we looked at AI-TACs prediction accuracy versus the variance of the ATAC-seq signal across the 81 cell types at each OCR, we observed that the highly well-predicted peaks were most likely to also have high variance (Figure 2.6). This is unsurprising considering the Pearson correlation loss used to optimize AI-TAC emphasizes the accurate prediction of high-variance OCRs.

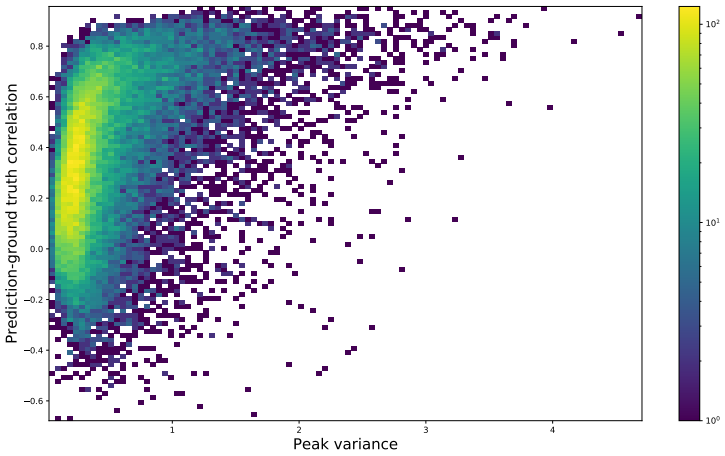



Figure 2.6: Variance of peak heights of each OCR on x-axis versus AI-TAC prediction accuracy on the y-axis.

### 2.3.5 Model Performance on Human Data

There is a high degree of similarity between mouse and human regulatory sequences, and previous work has shown that deep learning models can generalize from mouse to human data[8]. We therefore decided to validate the performance of AI-TAC on an ATAC-seq dataset of human primary immune cell types from Corces et al, 2016[9].

Data processing and normalization steps identical to those described for the mouse data were performed by Caleb Lareau, providing us with 539,611 OCRs of 250bp each with chromatin activity values for 18 different cell types. For eight of the human immune cell types for which ATAC-seq data

was available closely matching equivalents were present in our mouse dataset (Table 2.1). We compared the predictions of the AI-TAC model trained on mouse data to the observed chromatin activity in the corresponding human cell type, averaging over the prediction values when appropriate to obtain lineage-level predictions.

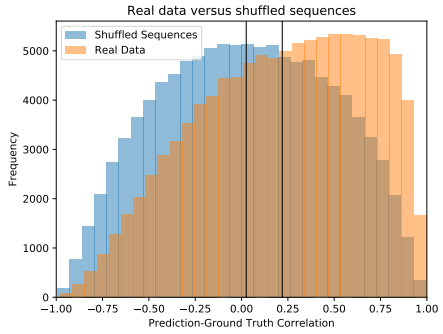
Figure 2.7 shows the results on real data versus three different types of randomized data, analogous to the test we performed on the mouse data:

- Randomly shuffled 251 base pair input sequences
- Randomly shuffled ATAC-seq activity vectors of length 81
- Randomly permuted order of input sequence and output vectors in the dataset

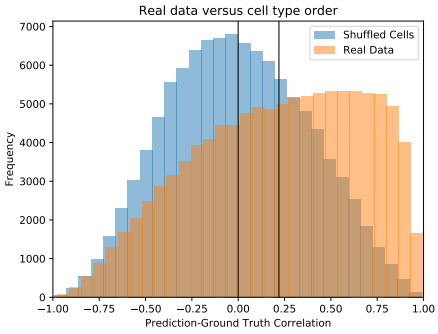
Despite never seeing human data in it's training set, the average correlation between AI-TACs predictions and the measured peak heights over the OCRs in the human dataset is 0.22, much higher than the models performance on randomized data. The fact that AI-TAC generalizes enough to make meaningful predictions on another species indicates that it learned real biological signal in our dataset.

## 2.4 Summary

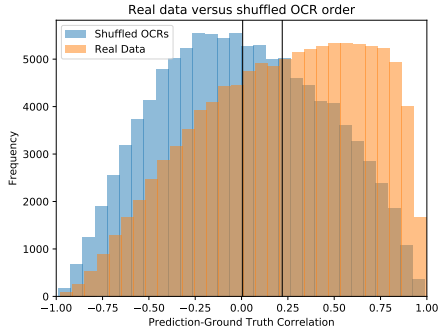
In summary, we trained a CNN with 3 convolutional and 2 fully connected layers to predict relative chromatin state at putative OCRs for 81 cell types simultaneously on the basis of 251bp DNA sequences. The model was trained using ATAC-seq data generated from mouse immune cells from six different lineages. The model was tested on a held-out set of OCRs from our mouse dataset, and compared to simulated "null" data. We also confirmed that the model performance is not sensitive to training set selection, by doing cross-validation and chromosome leave-out experiments. Finally, we showed that AI-TAC generalizes to human data by testing it on a human ATAC-seq dataset of closely matching immune cell types.




(a)




(b)




(c)

Figure 2.7: Performance of model trained on mouse ATAC-seq data on human DNA sequences. Performance is measured by Pearson correlation between model predictions and ATAC-seq data for the closest corresponding cell type. Results on real data are compared to data randomized in three ways: (a) randomization by shuffling the sequences, (b) by permutting the chromatin accessibility profiles, and (c) by shuffling the assignment of each OCR to its accessibility profile

| Mouse cell types | Human cell types |
|---|---|
| B.Fo.Sp | B |
| B.GC.CB.Sp | B |
| B.GC.CC.Sp | B |
| B.mem.Sp | B |
| B.MZ.Sp | B |
| B.PB.Sp | B |
| B.Sp | B |
| T.4.Nve.Fem.Sp | CD4 |
| T.4.Nve.Sp | CD4 |
| Treg.4.25hi.Sp | CD4 |
| NKT.Sp | CD4 |
| T.8.Nve.Sp | CD8 |
| T8.Tcm.LCMV.d180.Sp | CD8 |
| T8.Tem.LCMV.d180.Sp | CD8 |
| T8.TN.P14.Sp | CD8 |
| LTHSC.34-.BM | HSC |
| LTHSC.34+.BM | HSC |
| DC.4+.Sp | mDC |
| DC.8+.Sp | mDC |
| Mo.6C-II-.Bl | Mono |
| Mo.6C+II-.Bl | Mono |
| NK.27-11b+.BM | NK |
| NK.27-11b+.Sp | NK |
| NK.27+11b-.BM | NK |
| NK.27+11b-.Sp | NK |
| NK.27+11b+.BM | NK |
| NK.27+11b+.Sp | NK |
| DC.pDC.Sp | pDC |

Table 2.1: The human immune cell types measured by Corces et al. [9], and their mouse counterparts. All mouse cell populations that map onto the same human cell type were averaged in comparative analyses.

# Chapter 3

# Part II: Model Interpretation

Given that we've shown that AI-TAC learned relevant sequence features that are predictive of chromatin state, we are interested in understanding what those features are. Identifying important motifs and combinations of motifs that are correlated with chromatin state could provide important biological insights into the mechanism of cell-specific transcriptional regulation.

Section 3.1 describes how we extract PWMs learned by the first layer convolutional filters of AI-TAC and compare them to known mouse TF motifs. In section 3.2 we provide the details of computing filter information content, reproducibility and influence values, which characterize the relative importance and complexity of the recovered PWMs. We then performed experiments quantifying the degradation in the models performance when the first layer filters are limited to subsets of the most important motifs defined via the above metrics, the results of which are shown in section 3.3. Finally, in section 3.4 we show our attempts to decipher the combinatorial logic used by AI-TAC to make its predictions using two different ways: by examining the second layer convolutional filter weights, and by computing filter pair influence values.

## 3.1 Interpreting AI-TAC with First Layer Filters

To understand which sequence features are used by the model to make predictions we examine the first convolutional layer. The first layer consists of 300 filters, each 19 units long, that scan the input sequence and learn to recognize a specific motif. These were analyzed by converting them to position weight matrices (PWMs). To do so, for each first layer filter, we first identified all 19bp sequences that activate the filter by at least 1/2 of the maximum activation for that filter across all 51,732 well-predicted (Pearson correlation greater than 0.75) OCRs [29]. Next, we constructed a position frequency matrix based on the prevalence of each of nucleotide along the 19bp long sequences, and finally we converted the position frequency matrix to a PWM by using a background uniform nucleotide frequency of 0.25. This analysis yielded 300 PWMs, each capturing the motif that is detected by a

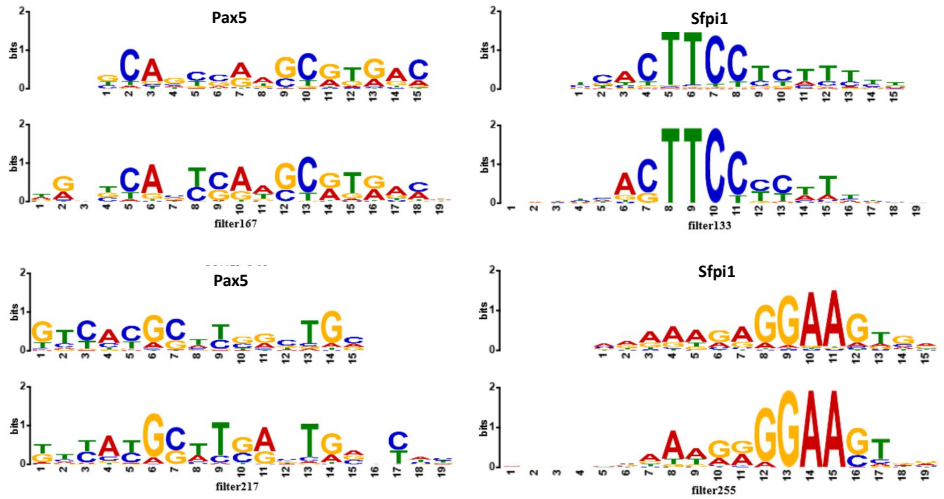



Figure 3.1: PWMs corresponding to 4 different convolutional filters in the first layer of AI-TAC. The filter motifs are shown aligned to known transcription factors in the CIS-BP database

first layer filter. The number of sequences comprising each of the PWMs is shown in figure 3.2, plotted against the IC of each filter. The (log scaled) number of sequences in each PWM is inversely proportional to the IC of each filter, which is unsurprising as we expect low IC motifs to occur more frequently by chance within the DNA sequence.

These PWMs were then compared to known mouse transcription factor motifs in the CIS-BP *Mus musculus* database [50] using the Tomtom motif comparison tool [20]. About a third of the first layer filters have a significant match to known TF motifs. Figure 3.1 shows four examples of filters that closely matched to known mouse TF binding motifs Pax5 and Sfpi1/PU.1. In the case of these TFs (and many others) the model learned both the forward and reverse compliment of the motif.

## 3.2 Filter Properties

To help characterize and prioritize the first layer filters of AI-TAC we computed the following metrics: the information content of each filter motif, the reproducibility of each filter across multiple training iterations of the model, and the influence of each filter on the model predictions. Appendix

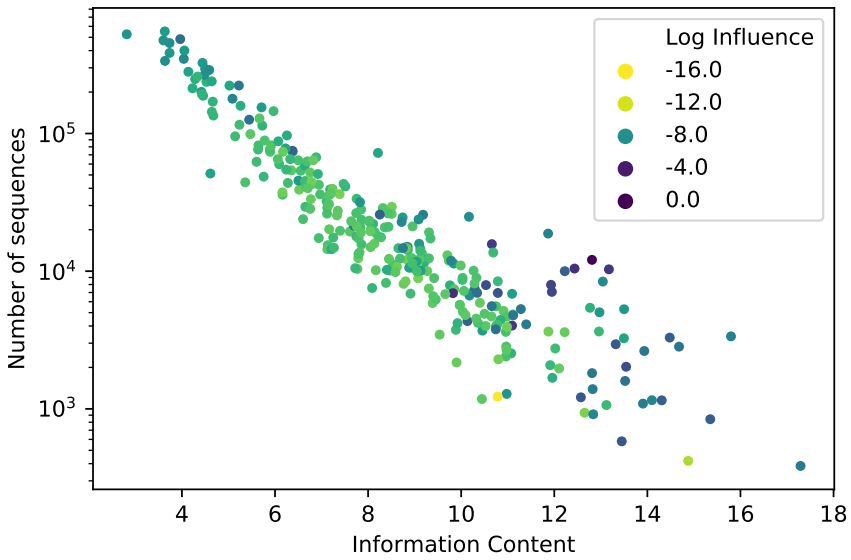



Figure 3.2: The number of sequences comprising each first layer filter PWM (log-scaled) versus the IC of each PWM, color coded by the log of the filter influence value.

A lists these metrics for each of the 300 filters, and the details of how they are obtained are described in the remainder of this section.

### 3.2.1 Information Content

We computed the information content of each motif using the following formula:

$$\mathrm{IC} = \sum_{i,j} p_{ij} \log_2(p_{ij}) - \sum_{i,j} b_j \log_2(b_j) \tag{3.1}$$

Where $p_{i,j}$ is the probability of observing nucleotide $j$ at position $i$ based on the observed frequencies and $b_j$ is the background frequency of nucleotide $j$ (set to 0.25 in our case).

### 3.2.2 Reproducibility

To understand how sensitive the convergence of the first layer filters is to the random training data split and model weight initializations we created a reproducibility metric. We trained 10 additional models using different random 90% subsets of the ATAC-seq dataset, then extracted the 300 filter PWMs from all 10 models. We measured the similarity between the filters of AI-TAC and those from each of the other models using the Tomtom PWM comparison tool[20]. We then defined a reproducibility score for each of the AI-TAC filter motifs as the number of models with at least one matching motif using an FDR q-value cut-off of 0.05 on the Tomtom results.

About a third of the filters are highly consistent between different training iterations of the model (Figure 3.3). The highly reproducible filters are much more likely to match a known TF binding motif than less reproducible ones. Additionally, the high information content filters are more likely to be reproducible, perhaps because obtaining a high IC motif is less likely during the optimization stage and therefore these are learned only if they are highly predictive.

### 3.2.3 Influence

The influence of each filter was computed by effectively removing the filter from AI-TAC and quantifying the impact on the models prediction. Specifically, we replaced all activation values for the given filter with its average activation value across all samples in the batch, then fed the output through the remaining layers of the model to obtain the altered prediction vector[29]. The overall influence value for a given filter was computed as the average (across OCRs) of the squared difference between the correlation in prediction of accessibility profiles (loss) of the altered and un-altered model.

Figure 3.4 shows the importance of each filter as a function of the information content of its PWM. Notably, high IC filters tend to have a bigger impact on model predictions. This is likely for the same reason that reproducibility values correlate with IC - these complex motifs are more difficult to learn, and therefore are learned by the model only if they are highly predictive of chromatin state.

Additionally, there is a strong correlation between filter influence values and whether or not the filter motif matches to a known mouse TF motif, indicating that for the most part motifs that are strongly correlated with regulatory activity have already been characterized. Reassuringly, many of the high influence filters correspond to motifs of known pioneer factors

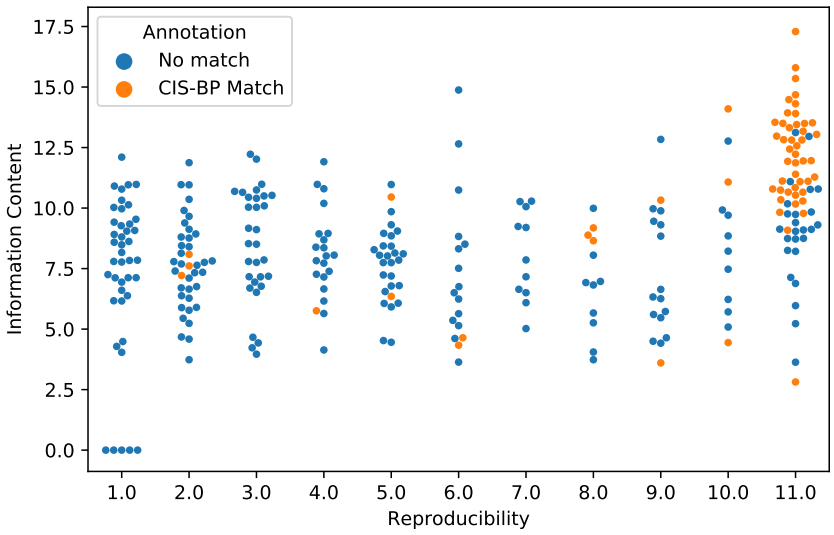



Figure 3.3: The reproducibility value of each AI-TAC first layer filter across 10 independent training iterations on the x-axis versus the information content of each filter on the y-axis. Each filter is color-coded by whether or not it matched a TF binding motif in the CIS-BP database with Tomtom q-value less than 0.05.

that are responsible for establishing chromatin accessibility, for example Sfpi1(PU.1), Pax5 and Cebp. The model also recovered the high information content motif of Ctcf, which plays an important role in the function of insulators.

To understand the cell type-specific impact of each filter we also computed an influence profile per cell population, as the average (across OCRs) of the squared difference between the original and modified prediction values for each output neuron (corresponding to a given cell population). We additionally computed a signed version of the influence profile, shown in Figure 3.5, by taking the difference between altered and un-altered predictions for each neuron, which shows whether the presence of each filter is predictive of higher or lower chromatin accessibility in each cell population.

All influence values were computed using 51,732 OCRs for which the AI-TAC prediction has greater than 0.75 correlation with the ground-truth chromatin accessibility. To ensure that including well-predicted OCRs from

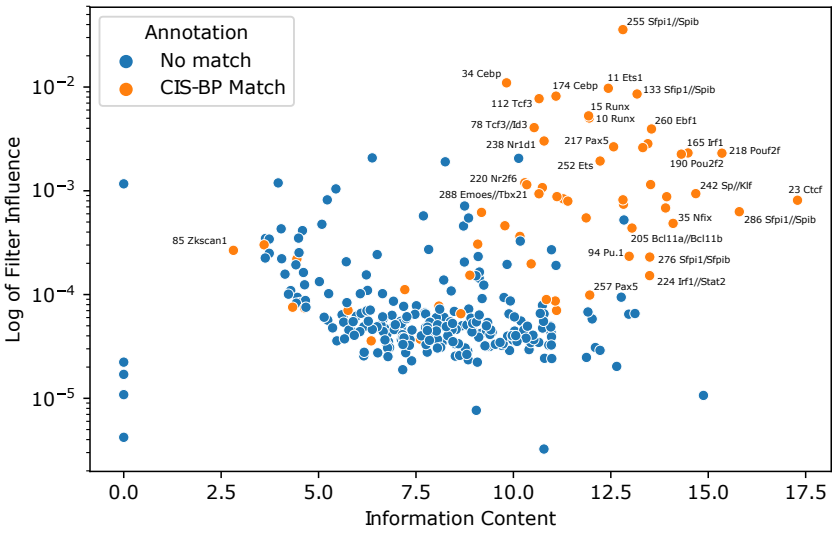



Figure 3.4: Information content (x-axis) versus the log of influence (y-axis) for all 300 first layer filters. Filters with PWMs matching to known TF motifs in the CIS-BP Mus musculus database are indicated in orange.

the training set would not bias our results, we compared the influence values obtained from well-predicted test and training set OCRs. The influence values computed on these two sets were strongly correlated. Additionally, the filter PWMs from the same model obtained from the test versus training set OCRs were virtually identical, with a Tomtom q-value of $2.9\text{x}10^{-27}$ or less for all but one pair of test set and training set derived PWMs.

The per-cell type influence values shown in Figure 3.5 indicate that some TF motifs have highly cell type specific predictive value, while others have more general importance across all lineages. There are many instances where the importance placed by AI-TAC on certain TF motifs recapitulates known biology. For example, filters corresponding to Pax5 and Ebf1, which are necessary for the commitment to and maintenance of the B-cell lineage [33, 36], have high influence values exclusively among the B-cells. The influence values of filters matching the motif recognized by both Eomes and Tbx21(T-bet) proteins recapitulates their importance in establishing the NK and CD8+ T-cell lineages[26, 51]. The Runx and Ets TF families play an important role in many different stages of haematopoiesis and are

expressed across the immune cell lineages [15, 49], which is reflected in the importance of their motifs in Figure 3.5. The Sfpi1/PU.1 motif has a high level of redundancy in the model and the highest influence overall, and has particularly high influence values for the stem, B-cell and myeloid lineages. Sfpi1 is responsible for determining immune cell fate, with high concentrations promoting myeloid differentiation while low concentrations promote B-cell differentiation [12]. Interestingly, the model produces high Sfpi1 influence values for both the B-cell and myeloid lineages, even though it is expected to be highly expressed only in the myeloid cells.

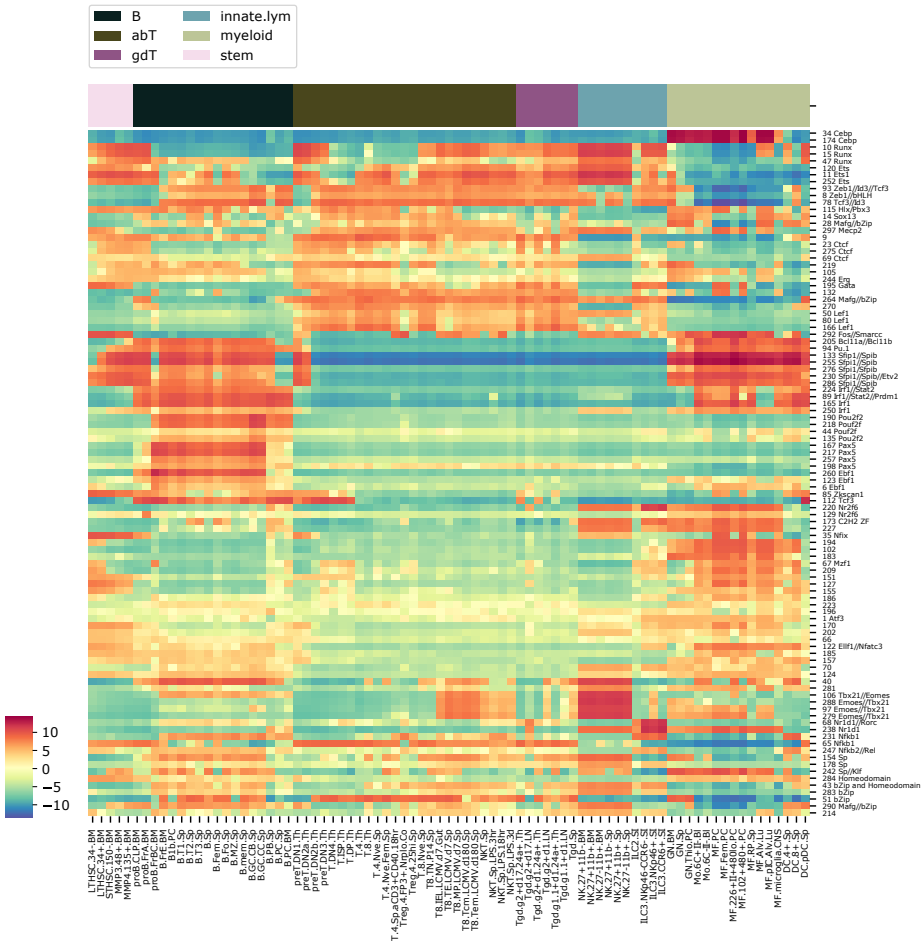



Figure 3.5: Cell type-specific log2-scaled influence values for the 99 reproducible filters found in at least 8 out of 10 different models.

## 3.3 Fine-tuning Model with Filter Subset

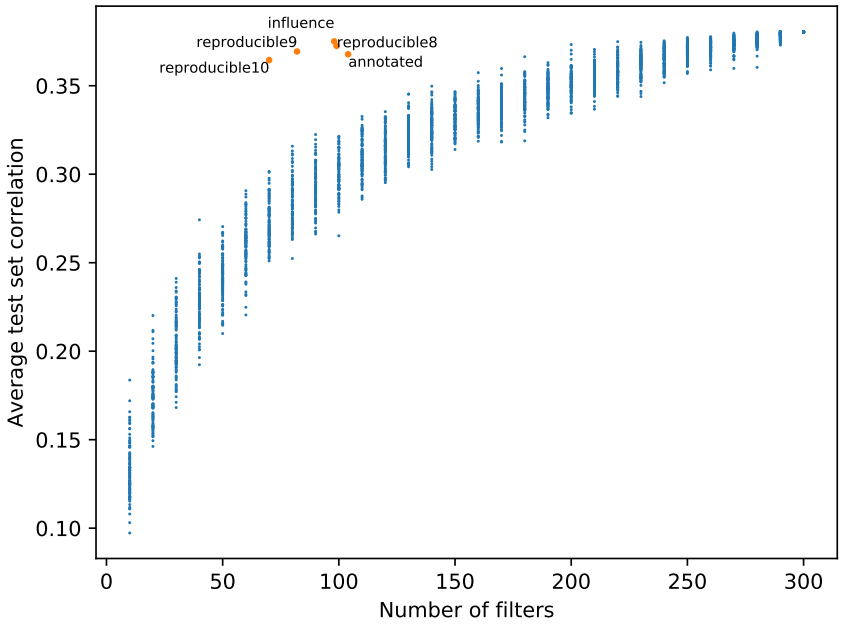



Figure 3.6: Model performance on test set using subsets of first layer motifs. In blue are experiments with randomly selected filter subsets, repeated 100 times for subsets of size 10 to 300 filters (in increments of 10). In orange, results for manually defined filter subsets selected based on filter properties: reproducible8, reproducible9. and reproducible10 correspond to all filters with reproducibility metric of at least 8, 9, or 10, respectively; annotated - all filters with CIS-BP TF motif match with q-value 0.05 or less; influence - 98 highest influence filters.

Because we expect the reproducible filters to be the most critical for good model performance, we decided to test how well the model predictions can be recovered with only the 99 filters found in 8 out of 10 additional models. We performed the experiment by removing all first layer weights that do not belong to the set of 99 reproducible filters, as well as all second layer weights corresponding to these filters. The first layer weights are then frozen, and the model is fine-tuned (with an adjusted learning rate of 0.0001) on the training set OCRs. The fine-tuning was performed for 10 additional

epochs, and the model with the best performance on the validation set was retained (typically found after 2-5 epochs).

The model used for these experiments was obtained by copying AI-TAC first layer weights into a randomly initialized model, freezing the first layer filters, and training the model for 10 epochs. The best model, obtained after 3 epochs, was selected based on performance on a validation set. This yielded a model with identical filters to AI-TAC and very similar performance on the test set OCRs.

As a null control, we compared these results to models for which a random subset of first layer filters was retained. For subsets of size 10 to 300 filters (in increments of 10) we repeated the fine-tuning procedure 100 times with random subsets selected (results in Figure 3.6). Additionally, we tested the model performance when only including filters reproduced in 9 out of 10 trials (82 filters), 10 out of 10 trials (70 filters), all filters matching a CIS-BP TF motif with a Tomtom q-value less than 0.05 (61 filters), and filters with influence greater than 0.0001 (98 filters). We found all of these subsets were sufficient to obtain average performance almost as high as the unaltered model, in contrast to the randomly selected filter sets. Subsets selected by reproducibility, influence and annotation status all have very similar performance, which is unsurprising since those metrics are highly correlated, and produced very similar filter subsets.

Figure 3.7 shows the performance of the 99 reproducible filter model at the OCR level. The correlations between the truncated model predictions and observed OCR activity profiles are virtually unchanged compared to the predictions of the full AI-TAC model.

The fact that AI-TACs predictive power can be almost entirely recovered with only a third of the first layer filters indicates that the majority of these filters are not meaningful feature detectors. This is consistent with previous findings that the number of first layer filters required for a CNN to learn the entire set of relevant motifs for a particular classification problem is larger than the set itself[32]. This necessary over-parameterization of the network is likely due to the difficulty of converging to a meaningful PWM from a random filter initialization during the model optimization stage.

## 3.4 Detecting TF Cooperativity with AI-TAC

The model predictions do not depend on the detection of individual motifs alone, but rather on patterns of motif combinations and their surrounding sequence context. The way AI-TAC models these relationships between

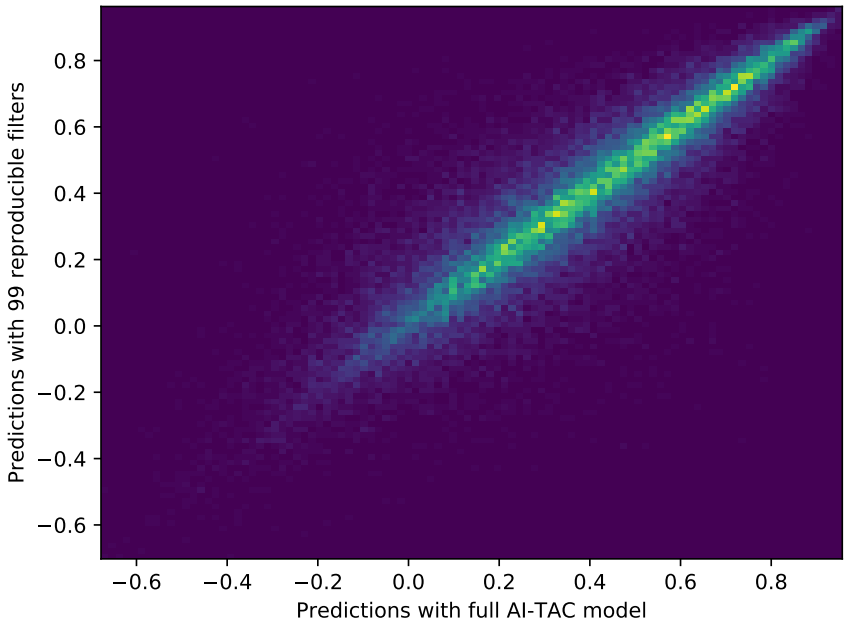



Figure 3.7: Correlations between predictions and ground truth peak heights for test set OCRs for the full AI-TAC model versus the model with only the 99 most reproducible filters.

sequence features should reflect, to some degree, the underlying biological mechanisms that drive cell type-specific chromatin accessibility patterns. We attempted to understand this combinatorial logic of the model in two ways: by analyzing the weights of the second layer convolutional filters and by computing combined influence values for select filter pairs.

### 3.4.1 Second Layer Filters

Because higher order relationships between the first layer motifs are encoded in the deeper layers of the network, an obvious first attempt at identifying important filter combinations is to look for combinations of motifs assembled by the second layer convolutional filters[5]. Due to the maxpooling applied to the first layer output, constructing clear motifs from second layer activations was not possible and we instead examined the second layer filter weights directly.

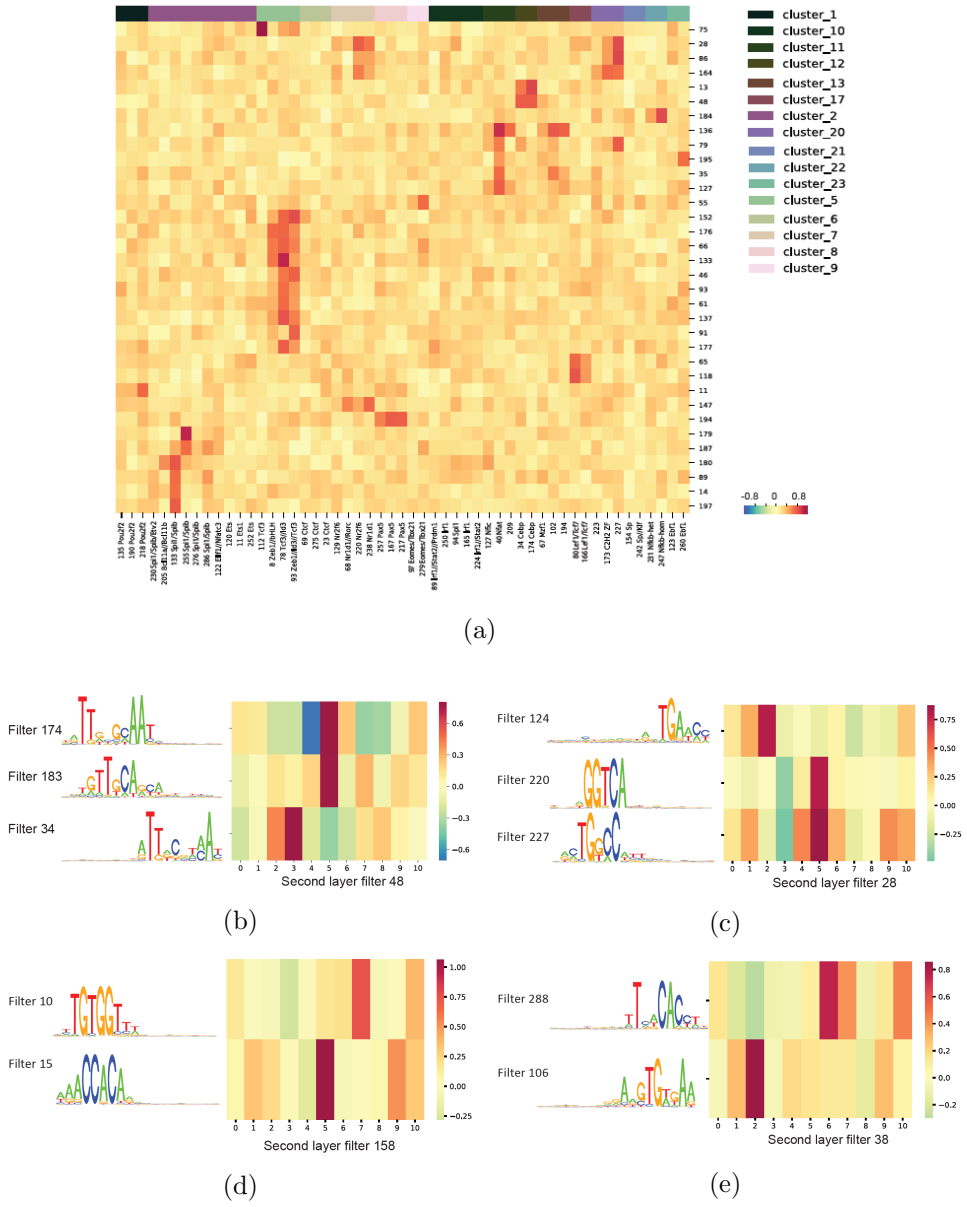



Figure 3.8: (a) Maximum weights between all reproducible first layer filters belonging to a cluster and a subset of second layer filters. Second layer filters were selected based on a weight threshold of 0.7 for at least one first layer motif. For a few examples, we visualized second layer filter weights for the most heavily weighted first layer motifs along with the corresponding PWMs. (b-c) Two instances of a second layer motif aggregating similar first layer motifs. (d-e) Two examples of a second layer filter recognizing reverse compliments first layer motifs.

We found that in a large number of cases the second layer filters recognized similar (Figures 3.8b, 3.8c) or reverse complement (Figures 3.8d, 3.8e) first layer motifs. This indicates that the second layer convolutional filters are assembling "cleaner" versions of first layer motifs rather than learning the combinatorial logic between them. Figure 3.8a shows the magnitude of the weights placed on each first layer convolutional filter by each second layer filter, with the first layer PWMs clustered by similarity. Clustering was performed by Ricardo Ramirez using RSAT[7]; more details can be found in Maslova et al, 2019 [34].

It shows on a more global scale the trend of second layer filters agglomerating similar motifs from the first layer. This is consistent with the findings of Koo and Eddy, 2019[32] for convolutional filters with smaller maxpooling windows. Their study suggests that increasing the maxpooling size in the first layer can force the first layer weights to converge to more complete motifs during training. The second layer filters then correspond to the interactions between these motifs.

### 3.4.2 Filter Pair Influence

To see whether AI-TAC was able to detect any instances of cooperative activity between TFs, we looked for evidence of non-additive effects on the model predictions when particular motif pairs are present in the input sequence. To do this, we computed filter pair influence values in a similar way as single filters, by removing both filters from the first layer of the model at once and quantifying the change in AI-TACs predictions. To make this task computationally tractable we selected the 40 most important filters by influence and reproducibility metrics and computed pair influence values for all 1600 possible pairs.

Figure 3.9 shows these pair influence values against the sum of individual influence values for the filters in the pair. For filter pairs that correspond to TFs that act independently, we expect the effect of removing both filters at once to be equal to the sum of their individual influence values (additive effects). However, if both filters are required to accurately predict the chromatin activity profile of an OCR, as would be expected for TFs that cooperate *in vivo*, removing either filter individually should produce a similar impact as removing both filters at once. In cases where two different TFs have redundant function, and thus the presence of either motif alone is sufficient to accurately predict the OCR activity profile, the impact of removing both TF filters will be greater than the sum of their individual filter values.

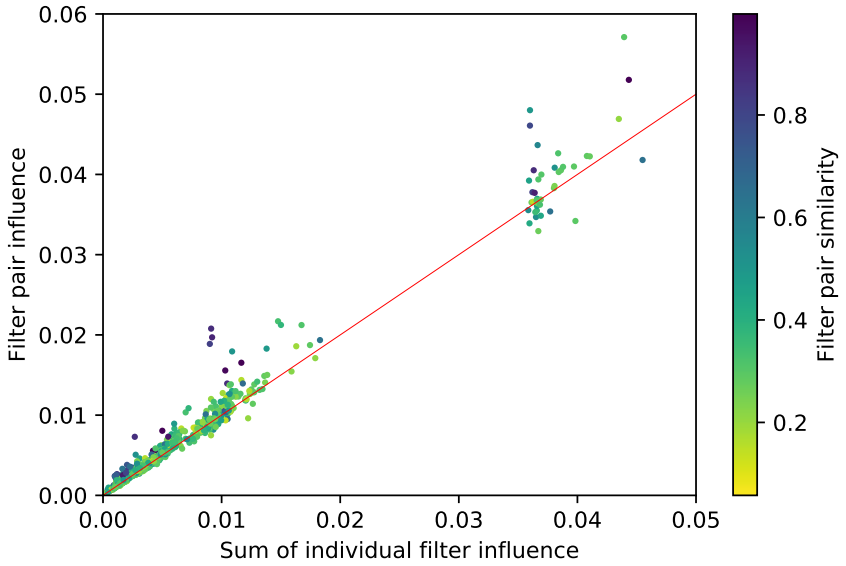



Figure 3.9: Sum of individual filter influence values of given filter pair on the x-axis versus the computed influence value when removing both filters in the pair from the model simultaneously (y-axis).

As seen in Figure 3.9, filter pairs composed of highly similar motifs have, for the most part, higher than expected pair influence values. It's likely that these similar filters serve a redundant purpose in the model, recognizing the same motifs within the input sequence, and therefore their individual influence values are underestimated. The more interesting cases are the highly dissimilar pairs that have an unexpectedly high pair influence values, as these may be indicative of a biological redundancy rather than a technical one. There are six such filter pairs (corresponding to four different TF combinations) listed in Table 3.1, which are interesting candidates for further investigation.

There are also several filter pairs, listed in Table 3.2 that have lower than expected pair influence values, implying that the detection of both motifs is necessary for a correct model prediction and perhaps indicative of cooperativity between the TFs. Our results indicate that the model is detecting possible cooperativity between pioneer factors Sfpi1 and Cebp and Tcf3, a TF critical for normal B and T-cell development[10].

The efficacy of this method for determining combinatorial logic within

| | |
|---|---|
| filter174 - Cebp | filter255 - Sfpi1 |
| filter133 - Sfpi1 | filter174 - Cebp |
| filter133 - Sfpi1 | filter165 - Irf1 |
| filter10 - Runx | filter11 - Ets1 |
| filter15 - Runx | filter11 - Ets1 |
| filter11 - Ets1 | filter78 - Tcf3/Id3 |

Table 3.1: Filter pairs with higher than expected pair influence values.

| | |
|---|---|
| filter255 - Sfpi1 | filter78 - Tcf3/Id3 |
| filter255 - Sfpi1 | filter11 - Tcf3 |
| filter255 - Sfpi1 | filter93 - Zeb1/Tcf3/Id3 |
| filter174 - Cebp | filter78 - Tcf3/Id3 |

Table 3.2: Filter pairs with lower than expected pair influence values.

the model is likely hampered by the redundancy present among the first layer motifs of AI-TAC. This approach may prove more fruitful if applied to a CNN with unique PWMs in the first layer, for example a model that is initialized with frozen PWMs of known TF motifs such as OrbWeaver[3].

## 3.5 Summary

In conclusion, we extracted the PWMs of all 300 first layer filters of AI-TAC to understand which sequence features are informative of local chromatin accessibility. We compared these filters to a database of known motifs of mouse TFs, and computed metrics of IC, reproducibility and influence for each filter. A table containing the above information for all 300 first layer filters can be found in Appendix A.

We found that about a third of the filter PWMs matched closely to known TF motifs, and this subset also has high filter reproducibility and influence values. We then performed an experiment to see how much of the model accuracy would be retained when using only the top third of filters determined with our importance metrics. We found that the model performance remains almost unchanged, and is considerably higher than performance for a randomly selected subset of filters.

Finally, we looked for evidence of TF cooperativity in our model. We examined the second convolutional layer of AI-TAC to determine which combinations of first layer filters are weighted highly when detected together.

We found that the second layer filters tend to group similar or reverse complement first layer motifs. We then computed influence values for selected pairs of first layer filters, and found several pairs that may be of interest for further investigation.

# Chapter 4

# Conclusions

## 4.1 Summary

We trained a CNN model using ATAC-seq data to predict chromatin accessibility across a large set of related immune cell types from sequence alone. We used Pearson correlation as the loss function for training our model, which prioritized the accurate prediction of regions with variable activity across different cell populations. Based on comparisons of model predictions on held-out test examples and human DNA sequences to results on simulated "null" data, we concluded that the model learned some biologically meaningful features. We additionally showed that the predictive performance of AI-TAC is stable with regards to the choice of training and test set splits.

The second part of the thesis was dedicated to extracting predictive sequence features from the model. We first identified the PWMs learned by the first layer convolutional filters of AI-TAC and found that many of them closely resemble binding motifs of known TFs. We additionally assigned two metrics of importance to every filter: influence and reproducibility. Influence values correspond to the impact of removing the filter on the model predictions, while reproducibility is the number of independent training iterations in which the filter was recovered. We validated these metrics by testing the model while using just a third of the most important filters and showing that the performance is almost as good as using the unaltered AI-TAC model.

In the last part of this thesis we attempted to understand the combinatorial logic within AI-TAC that may be representative of TF interactions *in vivo*. We tried two different approaches for this task: examining the second layer convolutional filter weights to understand how the model combines the first layer filters, and computing influence values for selected pairs of filters. The latter analysis yielded a handful of candidate motif pairs for further investigation. However, a full understanding of the combinatorial effects of motifs remains a challenge.

## 4.2 Discussion

Notably, the vast majority of the filters identified in this study with high influence and reproducibility values matched closely to known mouse TF motifs. In addition, the cell type-specific influence values of these filters were largely in agreement with the known roles of the corresponding TFs within immune cell differentiation. While it's comforting that our results coincide closely with prior biological knowledge, the question arises of why we are not seeing many novel motifs in our model. One possibility is that the motifs learned by both AI-TAC and through the many individual biological experiments aimed at understanding immune cell differentiation represent the lowest hanging fruit of relevant TFs - i.e. the most prevalent and statistically significant motif instances within the data. Alternatively, it's possible that the vast majority of TFs that are relevant to chromatin state within immune cells have already been characterized by the immunology field. Fully explaining cell type specific chromatin state would then require understanding the interactions between these TFs and the role of other epigenetic mechanisms in chromatin accessibility.

There are a number of epigenetic mechanisms that impact chromatin state and are not directly reflected in the DNA code that may limit how well the model can predict chromatin state from short, local DNA sequence alone. These mechanisms include:

- DNA methylation within regulatory sequences, which can affect binding affinity between TFs and their motifs[45]
- Modifications of histones within nucleosomes which can alter nucleosome stability[31]
- Indirect binding of important TFs at regulatory sites via protein-protein interactions with other TFs [45]
- Removal of nucleosomes via interaction with distal regulatory sites[31]

The results of repeated cross-validation experiments, depicted in Figure 2.4b, do validate the notion that for some OCR the model fails to make consistently good predictions. Although a subset of OCRs are uniformly very well-predicted across different training iterations of the model, many OCRs were on average predicted poorly. This can be partially explained in terms of the variance of chromatin state at each OCR - we can expect that high prediction correlation values would be more difficult to obtain for

very uniform activity vectors. We do, in fact, observe that peak variance is correlated with the prediction accuracy of AI-TAC (Figure 2.6). However, it is also likely that some of the epigenetic mechanisms listed above are impacting chromatin accessibility in ways that are undetectable to AI-TAC.

Cooperative interaction between TFs may be the other key to understanding how chromatin accessibility is controlled with high precision during differentiation. We already noted some of the potential technical reasons why our approaches to understanding TF cooperativity did not yield many candidate TF interactions, namely that our model architecture allowed for large amounts of redundancy among the first layer filters as well as dispersed representations of motifs. However, there may be a low amount of evidence for TF cooperativity in our dataset for biological reasons. It's possible that chromatin accessibility within this biological system can be largely predicted based on motifs of lineage-specific pioneer factors and the additive effects of differentially-expressed TFs that contribute to accessibility via passive competition with nucleosomes for DNA binding. Improved methods for extracting the combinatorial logic used by the model would help clarify these questions.

## 4.3 Future Work

Understanding how various epigenetic mechanisms contribute to chromatin state can be explored by integrating additional genomic data within a CNN framework. For example, DNA methylation and histone modification assays can be added to the sequence information to provide better predictions of chromatin accessibility. The role of long-range interactions between regulatory elements can be examined by incorporating data from a DNA conformation assay such as Hi-C.

The analysis of TF cooperativity can be improved by adjusting the architecture of the model to make our approaches more informative. For example, the maxpooling after the first convolutional layer can be increased to make the second layer filters more interpretable. Additionally, eliminating redundancies within the first convolutional layer would make the influence values for both individual filters and filter pairs more informative. Finally, increasing the resolution of the model output, for example by changing it to per-base pair predictions of ATAC-seq read depth, could provide better information about the impact of the spatial organization of motifs on chromatin state.

# Bibliography


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

[2] Robin Andersson and Albin Sandelin. Determinants of enhancer and promoter activities of regulatory elements. *Nature Reviews Genetics*, 21(February):71–87, 2020.

[3] Nicholas E. Banovich, Yang I. Li, Anil Raj, Michelle C. Ward, Peyton Greenside, Diego Calderon, Po Yuan Tung, Jonathan E. Burnett, Marsha Myrthil, Samantha M. Thomas, Courtney K. Burrows, Irene Gallego Romero, Bryan J. Pavlovic, Anshul Kundaje, Jonathan K. Pritchard, and Yoav Gilad. Impact of regulatory variation across human iPSCs and differentiated cells. *Genome Research*, 28(1):122–131, 2018.

[4] Oliver Bell, Vijay K Tiwari, Nicolas H Thomä, and Dirk Schübeler. Determinants and dynamics of genome accessibility. *Nature Publishing Group*, 2011.

[5] Nicholas Bogard, Johannes Linder, Alexander B. Rosenberg, and Georg Seelig. A Deep Neural Network for Predicting and Engineering Alternative Polyadenylation. *Cell*, 178(1):91–106.e23, 2019.

[6] Jason D. Buenrostro, Beijing Wu, Howard Y. Chang, and William J. Greenleaf. ATAC-seq: A method for assaying chromatin accessibility genome-wide. *Current Protocols in Molecular Biology*, 109(1):21.29.1–21.29.9, 2015.

[7] Jaime Abraham Castro-Mondragon, Sébastien Jaeger, Denis Thieffry, Morgane Thomas-Chollier, and Jacques Van Helden. RSAT matrix-clustering: Dynamic exploration and redundancy reduction of transcription factor binding motif collections. *Nucleic Acids Research*, 45(13):1–13, 2017.

[8] Ling Chen, Alexandra E. Fish, and John A. Capra. Prediction of gene regulatory enhancers across species reveals evolutionarily conserved sequence properties. *PLoS Computational Biology*, 14(10), 2018.

[9] M. Ryan Corces, Jason D. Buenrostro, Beijing Wu, Peyton G. Greenside, Steven M. Chan, Julie L. Koenig, Michael P. Snyder, Jonathan K. Pritchard, Anshul Kundaje, William J. Greenleaf, Ravindra Majeti, and Howard Y. Chang. Lineage-specific and single-cell chromatin accessibility charts human hematopoiesis and leukemia evolution. *Nature Genetics*, 48(10):1193–1203, 2016.

[10] Renée F. De Pooter and Barbara L. Kee. E proteins and the regulation of early lymphocyte development. *Immunological Reviews*, 238(1):93–109, 2010.

[11] Aimée M. Deaton and Adrian Bird. CpG islands and the regulation of transcription. *Genes and Development*, 25(10):1010–1022, 2011.

[12] Rodney P. DeKoter and Harinder Singh. Regulation of B lymphocyte and macrophage development by graded expression of PU.1. *Science*, 288(5470):1439–1442, 2000.

[13] Gökcen Eraslan, Žiga Avsec, Julien Gagneur, and Fabian J. Theis. Deep learning: new computational modelling techniques for genomics. *Nature Reviews Genetics*, 20(7):389–403, 2019.

[14] Oriol Fornes, Jaime A. Castro-Mondragon, Aziz Khan, Robin van der Lee, Xi Zhang, Phillip A. Richmond, Bhavi P. Modi, Solenne Correard, Marius Gheorghe, Damir Baranašić, Walter Santana-Garcia, Ge Tan, Jeanne Chèneby, Benoit Ballester, François Parcy, Albin Sandelin, Boris Lenhard, Wyeth W. Wasserman, and Anthony Mathelier. JASPAR 2020: update of the open-access database of transcription factor binding profiles. *Nucleic acids research*, 48(D1):D87–D92, 2020.

[15] Lee Ann Garrett-Sinha. Review of Ets1 structure, function, and roles in immunity. *Cellular and Molecular Life Sciences*, 70(18):3375–3390, 2013.

[16] Miklos Gaszner and Gary Felsenfeld. Insulators: Exploiting transcriptional and epigenetic mechanisms. *Nature Reviews Genetics*, 7(9):703–713, 2006.

[17] Mahmoud Ghandi, Dongwon Lee, Morteza Mohammad-Noori, and Michael A. Beer. Enhanced Regulatory Sequence Prediction Using Gapped k-mer Features. *PLoS Computational Biology*, 10(7), 2014.

[18] Charles E. Grant, Timothy L. Bailey, and William Stafford Noble. FIMO: Scanning for occurrences of a given motif. *Bioinformatics*, 27(7):1017–1018, 2011.

[19] Michael Gribskov. Identification of sequence patterns, motifs and domains. In *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, volume 1-3, pages 332–340. 2018.

[20] Shobhit Gupta, John A. Stamatoyannopoulos, Timothy L. Bailey, and William Stafford Noble. Quantifying similarity between motifs. *Genome Biology*, 8(2):R24, 2007.

[21] Tatsunori Hashimoto, Richard I Sherwood, Daniel D Kang, Nisha Rajagopal, Amira A Barkal, Haoyang Zeng, Bart J M Emons, Sharanya Srinivasan, Tommi Jaakkola, and David K Gifford. A synergistic DNA logic predicts genome-wide chromatin accessibility. *Genome Research*, 26:1430–1440, 2016.

[22] Sven Heinz, Christopher Benner, Nathanael Spann, Eric Bertolino, Yin C Lin, Peter Laslo, Jason X Cheng, Cornelis Murre, Harinder Singh, and Christopher K Glass. Simple Combinations of Lineage-Determining Transcription Factors Prime cis-Regulatory Elements Required for Macrophage and B Cell Identities. *Molecular Cell*, 38:576–589, 2010.

[23] Tracy S.P. Heng, Michio W. Painter, Kutlu Elpek, Veronika Lukacs-Kornek, Nora Mauermann, Shannon J. Turley, Daphne Koller, Francis S. Kim, Amy J. Wagers, Natasha Asinovski, Scott Davis, Marlys Fassett, Markus Feuerer, Daniel H.D. Gray, Sokol Haxhinasto, Jonathan A. Hill, Gordon Hyatt, Catherine Laplace, Kristen Leatherbee, Diane Mathis, Christophe Benoist, Radu Jianu, David H. Laidlaw, J. Adam Best, Jamie Knell, Ananda W. Goldrath, Jessica Jarjoura, Joseph C. Sun, Yanan Zhu, Lewis L. Lanier, Ayla Ergun, Zheng Li, James J. Collins, Susan A. Shinton, Richard R. Hardy, Randall Friedline, Katelyn Sylvia, and Joonsoo Kang. The immunological genome project: Networks of gene expression in immune cells. *Nature Immunology*, 9(10):1091–1094, 2008.

[24] Authors Hideyuki Yoshida, Caleb A Lareau, Ricardo N Ramirez, Jason D Buenrostro, Christophe Benoist, Hideyuki Yoshida, Samuel A Rose, Barbara Maier, Aleksandra Wroblewska, Fiona Desland, Aleksey Chudnovskiy, Arthur Mortha, Claudia Dominguez, Julie Tellier, Edy Kim, Dan Dwyer, Susan Shinton, Tsukasa Nabekura, YiLin Qi, Bingfei Yu, Michelle Robinette, Ki-Wook Kim, Amy Wagers, Andrew Rhoads, Stephen L Nutt, Brian D Brown, Sara Mostafavi, and The Immunological Genome Project. The cis-Regulatory Atlas of the Mouse Immune System. *Cell*, 176:897–912, 2019.

[25] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors, 2012. arXiv:1207.0580.

[26] Andrew M. Intlekofer, Naofumi Takemoto, E. John Wherry, Sarah A. Longworth, John T. Northrup, Vikram R. Palanivel, Alan C. Mullen, Christopher R. Gasink, Susan M. Kaech, Joseph D. Miller, Laurent Gapin, Kenneth Ryan, Andreas P. Russ, Tullia Lindsten, Jordan S. Orange, Ananda W. Goldrath, Rafi Ahmed, and Steven L. Reiner. Effector and memory CD8+ T cell fate coupled by T-bet and eomesodermin. *Nature Immunology*, 6(12):1236–1244, 2005.

[27] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *32nd International Conference on Machine Learning, ICML 2015*, volume 1, pages 448–456. International Machine Learning Society (IMLS), 2015.

[28] David R. Kelley, Yakir A. Reshef, Maxwell Bileschi, David Belanger, Cory Y. McLean, and Jasper Snoek. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Research*, 28(5):739–750, 2018.

[29] David R. Kelley, Jasper Snoek, and John L. Rinn. Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Research*, 26(7):990–999, 2016.

[30] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR, 2015.

[31] Sandy L. Klemm, Zohar Shipony, and William J. Greenleaf. Chromatin accessibility and the regulatory epigenome. *Nature Reviews Genetics*, 20(April):207–220, 2019.

[32] Peter K. Koo and Sean R. Eddy. Representation learning of genomic sequence motifs with convolutional neural networks. *PLOS Computational Biology*, 15(12):e1007560, 2019.

[33] Elizabeth M. Mandel and Rudolf Grosschedl. Transcription control of early B cell differentiation. *Current Opinion in Immunology*, 22(2):161–167, 2010.

[34] Alexandra Maslova, Ricardo N. Ramirez, Ke Ma, Hugo Schmutz, Chendi Wang, Curtis Fox, Bernard Ng, Christophe Benoist, Sara Mostafavi, and the Immunological Genome Project. Learning immune cell differentiation. *bioRxiv*, page 2019.12.21.885814, dec 2019.

[35] James P Noonan and Andrew S Mccallion. Genomics of Long-Range Regulatory Elements. *Annu. Rev. Genomics Hum. Genet*, 11:1–23, 2010.

[36] Stephen L. Nutt, Barry Heavey, Antonius G. Rolink, and Meinrad Busslinger. Commitment to the B-lymphoid lineage depends on the transcription factor Pax5. *Nature*, 402(S6763):14–20, 1999.

[37] Marco Osterwalder, Iros Barozzi, Virginie Tissiéres, Yoko Fukuda-Yuzawa, Brandon J. Mannion, Sarah Y. Afzal, Elizabeth A. Lee, Yiwen Zhu, Ingrid Plajzer-Frick, Catherine S. Pickle, Momoe Kato, Tyler H. Garvin, Quan T. Pham, Anne N. Harrington, Jennifer A. Akiyama, Veena Afzal, Javier Lopez-Rios, Diane E. Dickel, Axel Visel, and Len A. Pennacchio. Enhancer redundancy provides phenotypic robustness in mammalian development. *Nature*, 554(7691):239–243, 2018.

[38] Daniel Quang and Xiaohui Xie. DanQ: A hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Research*, 44(11), 2016.

[39] Alicia N. Schep, Beijing Wu, Jason D. Buenrostro, and William J. Greenleaf. ChromVAR: Inferring transcription-factor-associated accessibility from single-cell epigenomic data. *Nature Methods*, 14(10):975–978, 2017.

[40] Manu Setty and Christina S. Leslie. SeqGL Identifies Context-Dependent Binding Signals in Genome-Wide Regulatory Element Maps. *PLoS Computational Biology*, 11(5):1–21, 2015.

[41] Daria Shlyueva, Gerald Stampfel, and Alexander Stark. Transcriptional enhancers: From properties to genome-wide predictions. *Nature Reviews Genetics*, 15:272–286, 2014.

[42] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *34th International Conference on Machine Learning, ICML 2017*, volume 7, pages 4844–4866, 2017.

[43] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, 2014. arXiv:1312.6034v2.

[44] Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, volume 4, pages 2951–2959, 2012.

[45] François Spitz and Eileen E M Furlong. Transcription factors : from enhancer binding to developmental control. *Nature Publishing Group*, 13:613–626, 2012.

[46] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for Simplicity: The All Convolutional Net, 2015. arXiv:1412.6806v3.

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

[48] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic Attribution for Deep Networks, 2017. arXiv:1703.01365v2.

[49] Dominic Chih Cheng Voon, Yit Teng Hor, and Yoshiaki Ito. The RUNX complex: Reaching beyond haematopoiesis into immunity. *Immunology*, 146(4):523–536, 2015.

[50] Matthew T. Weirauch, Ally Yang, Mihai Albu, Atina G. Cote, Alejandro Montenegro-Montero, Philipp Drewe, Hamed S. Najafabadi, Samuel A. Lambert, Ishminder Mann, Kate Cook, Hong Zheng, Alejandra Goity, Harm van Bakel, Jean Claude Lozano, Mary Galli,

Mathew G. Lewsey, Eryong Huang, Tuhin Mukherjee, Xiaoting Chen, John S. Reece-Hoyes, Sridhar Govindarajan, Gad Shaulsky, Albertha J.M. Walhout, François Yves Bouget, Gunnar Ratsch, Luis F. Larrondo, Joseph R. Ecker, and Timothy R. Hughes. Determination and inference of eukaryotic transcription factor sequence specificity. *Cell*, 158(6):1431–1443, 2014.

[51] Jiang Zhang, Marie Marotel, Sébastien Fauteux-Daniel, Anne Laure Mathieu, Sébastien Viel, Antoine Marçais, and Thierry Walzer. European journal of immunology. 48(5):738–750, 2018.

[52] Jian Zhou, Chandra L. Theesfeld, Kevin Yao, Kathleen M. Chen, Aaron K. Wong, and Olga G. Troyanskaya. Deep learning sequence-based ab initio prediction of variant effects on expression and disease risk. *Nature Genetics*, 50(8):1171–1179, 2018.

[53] Jian Zhou and Olga G. Troyanskaya. Predicting effects of noncoding variants with deep learning-based sequence model. *Nature Methods*, 12(10):931–934, 2015.

# Appendix A

# Filter motif information

Properties of all 300 first layer filters of AI-TAC. The columns correspond to:

- Influence - overall filter influence
- IC - information content of the filter PWM
- Reprod. - reproducibility, i.e. the number of runs in which this filter was recovered
- Top TF matches - TFs corresponding to best matching motif in the CIS-BP database

,

| Filter | Influence | IC | Reprod. | Top TF Matches |
|---|---|---|---|---|
| filter255 | 0.0357675 | 12.81011043 | 10 | Sfpi1/ Spib/ Spib/ Bcl11a/ Bcl11b/ Etv2/ Ets2/ Erg/ Erf/ ENSMUSG00000044690 |
| filter34 | 0.010975358 | 9.823882667 | 10 | Cebpb/ Cebpe/ Cebpa/ Nfil3/ Cebpg/ Cebpd/ Dbp/ Tef/ Hlf/ Atf5 |
| filter11 | 0.009727289 | 12.43567913 | 10 | Etv2/ Ets2/ Ets1/ Erf/ ENSMUSG00000044690/ Fev/ Etv2/ Ets2/ Erg/ Erf |
| filter133 | 0.008568606 | 13.17366462 | 10 | Sfpi1/ Spib/ Spib/ Bcl11a/ Bcl11b/ Irf1/ Etv2/ Ets2/ Ets1/ Erf |
| filter174 | 0.008169892 | 11.09285091 | 10 | Cebpe/ Cebpb/ Cebpa/ Cebpg/ Hlf/ Cebpd/ Nfil3 |

| | | | | |
|---|---|---|---|---|
| filter112 | 0.007724161 | 10.6590297 | 10 | Myf5/ Myod1/ Ascl2/ Ascl1/ Myf5/ Myog/ Tcf12/ Tcf3/ Tal1/ Lyl1 |
| filter15 | 0.005275883 | 11.93004972 | 10 | Runx2/ Runx3/ Runx1/ Runx2/ Runx3/ Runx2/ Runx3 |
| filter10 | 0.005030511 | 11.94800097 | 10 | Runx1/ Runx2/ Runx3/ Runx2/ Runx3/ Runx2/ Runx3 |
| filter78 | 0.004063911 | 10.53256981 | 10 | Id3/ Id4/ Id1/ Snai2/ Zeb1/ Myf5/ Myod1/ Mesp2/ Mesp1/ Tcf4 |
| filter260 | 0.003942707 | 13.54372185 | 10 | nan |
| filter238 | 0.00301885 | 10.78727984 | 10 | Rorc/ Rorb/ Rarb/ Esr1/ Esr2/ Rxrb/ Nr2f6/ Rxrg/ Nr2f2/ Esrrb |
| filter167 | 0.002844589 | 13.44976933 | 10 | Pax9/ Pax5/ Pax8/ Pax1/ Pax9/ Pax5/ Pax8/ Pax1/ Pax9/ Pax5 |
| filter217 | 0.002654829 | 12.57158572 | 10 | Pax9/ Pax5/ Pax8/ Pax1/ Pax9/ Pax5/ Pax8/ Pax1/ Pax9/ Pax5 |
| filter292 | 0.002608087 | 13.3200093 | 10 | Smarcc2/ Smarcc1/ Fosb/ Fos/ Batf3/ Batf/ Fosb/ Fos/ Bach1/ Bach2 |
| filter165 | 0.002315825 | 14.4805664 | 10 | Irf1/ Bcl11a/ Bcl11b/ Stat2/ Prdm1 |
| filter218 | 0.002303334 | 15.35124635 | 10 | Tbpl2 |
| filter190 | 0.002254576 | 14.3090253 | 10 | Tbpl2 |
| filter245 | 0.002075286 | 6.377445222 | 1 | |
| filter121 | 0.002055486 | 10.13494298 | 0 | |

| | | | | |
|---|---|---|---|---|
| filter252 | 0.001935959 | 12.22508784 | 10 | Etv2/ Ets2/ Ets1/ Erf/ ENSMUSG00000044690/ Fev/ Etv2/ Ets2/ Erg/ Erf |
| filter40 | 0.001902964 | 8.250938104 | 10 | |
| filter220 | 0.001194611 | 10.29211886 | 10 | Esr1/ Esr2/ Rorc/ Rorb/ Rarb/ Rxrb/ Nr2f6/ Rxrg/ Nr2f2/ Nr2c2 |
| filter295 | 0.001191121 | 3.963877444 | 2 | |
| filter271 | 0.001167049 | | | |
| filter166 | 0.001148039 | 13.52171258 | 10 | Lef1/ Tcf7l2 |
| filter288 | 0.001144482 | 10.343384 | 10 | Tbx1/ Tbx10/ Tbx20/ Tbx21/ Eomes/ Mga/ Tbx5/ Tbx4/ Tbx21/ Tbr1 |
| filter68 | 0.001073696 | 10.74340634 | 10 | Nr1d1/ Nr1d2/ Rorc/ Rorc/ Rorb/ Pparg/ Ppard/ Ppara/ Pparg/ Ppard |
| filter57 | 0.00104311 | 5.444640633 | 1 | |
| filter242 | 0.000939029 | 14.6802471 | 10 | Sp2/ Sp3/ Sp6/ Sp8/ Sp7/ Sp9/ Sp5/ Sp2/ Sp3/ Sp6 |
| filter93 | 0.000936151 | 10.65612167 | 10 | Id3/ Id4/ Id1/ Zeb1/ Snai2/ Mesp2/ Mesp1/ Tcf4/ Figla/ Atoh8 |
| filter279 | 0.000878933 | 11.1171853 | 10 | Klf6/ Klf5/ Klf3/ Klf1/ Klf2/ Mga/ Tbx21/ Eomes/ Tbx4/ Tbx1 |
| filter89 | 0.000877186 | 13.9338102 | 10 | Stat2/ Irf1/ Prdm1/ Bcl11a/ Bcl11b/ Irf2 |
| filter106 | 0.000836286 | 11.28191316 | 10 | Tbx20/ Tbx1/ Tbx10/ Mga/ Tbx21/ Eomes/ Tbx5/ Tbx21/ Tbr1/ Bcl11a |
| filter231 | 0.000818648 | 12.81160589 | 10 | Nfkb1/ Rel/ Relb/ Rel/ Rela/ Hivep2/ Hivep1/ Hivep3/ Nfkb2/ Sp110 |

| | | | | |
|---|---|---|---|---|
| filter51 | 0.000818111 | 5.223610074 | 10 | Mafk/ Mafg |
| filter23 | 0.000810822 | 17.29094559 | 10 | Ctcf/ Ctcfl |
| filter97 | 0.000794982 | 11.39666997 | 10 | Tbx4/ Tbx5/ Mga/ Tbx20/ Rhox8/ Tbx1/ Tbx10/ Tbx19/ T/ Tbx21 |
| filter120 | 0.000742252 | 12.8251035 | 10 | Etv2/ Ets2/ Ets1/ Erf/ ENSMUSG00000044690/ Fev/ Etv2/ Gabpa/ Ets2/ Erf |
| filter9 | 0.000713634 | 8.747762923 | 10 | |
| filter275 | 0.000683147 | 13.90513714 | 10 | Ctcf/ Ctcfl |
| filter286 | 0.00062891 | 15.79696475 | 10 | Spib/ Sfpi1/ Spib/ Irf1/ Bcl11a/ Bcl11b/ Stat2/ Prdm1/ Etv2/ Ets2 |
| filter240 | 0.000619261 | 9.181616067 | 7 | Sp2/ Sp3/ Sp6/ Sp8/ Sp7/ Sp9/ Sp5/ Plagl1/ Zbtb7a/ Zbtb7c |
| filter236 | 0.000573209 | 7.69064873 | 1 | |
| filter230 | 0.000548002 | 11.86735023 | 10 | Sfpi1/ Spib/ Etv2/ Ets2/ Erg/ Erf/ ENSMUSG00000044690/ Fev/ Spib/ Etv2 |
| filter195 | 0.00054669 | 8.849514119 | 9 | Gata2/ Gata3/ Gata6/ Gata2/ Gata2 |
| filter65 | 0.000523359 | 12.83675477 | 8 | Nfkb1 |
| filter35 | 0.000485219 | 14.09783964 | 9 | Nfix/ Nfib/ Nfia/ Nfic/ Nfix/ Nfib/ Nfia/ Nfic/ Nfix/ Nfib |
| filter264 | 0.000475761 | 5.085780637 | 9 | Mafg |
| filter8 | 0.00046047 | 9.781212087 | 10 | Zeb1/ Snai2/ Id3/ Id4/ Id1/ Mesp2/ Mesp1/ Myf5/ Myod1/ Tcf4 |
| filter173 | 0.000457876 | 8.719776746 | 10 | Zfp711/ Zfa/ Zfy1/ Zfx |
| filter205 | 0.000437324 | 13.043296 | 10 | Bcl11a/ Bcl11b/ Sfpi1/ Spib/ Etv2/ Ets2/ Erg/ Erf/ ENSMUSG00000044690/ Fev |

| | | | | |
|---|---|---|---|---|
| filter179 | 0.00043018 | 4.040188655 | 0 | Zkscan1/ Srf/ Atf1/ Sp100/ Sp140/ IRC900814/ Dnajc21 |
| filter172 | 0.000413453 | 4.586823362 | 1 | Prkrir |
| filter154 | 0.000363637 | 10.16701761 | 10 | Sp2/ Sp3/ Sp6/ Sp8/ Sp7/ Sp9/ Sp5/ Sp2/ Sp3/ Sp6 |
| filter274 | 0.000349809 | 4.486219114 | 0 | Dmrt3/ Dmrta1 |
| filter114 | 0.000346892 | 3.63658211 | 5 | |
| filter128 | 0.000342323 | 3.734457583 | 1 | |
| filter227 | 0.000327319 | 10.17649143 | 10 | |
| filter47 | 0.000306001 | 9.084441796 | 10 | Runx2/ Runx3/ Runx1/ Runx2/ Runx3 |
| filter219 | 0.000301505 | 3.602551329 | 8 | Setbp1/ Atf1/ Mafk/ Dnajc21/ Hbp1/ Zkscan1/ Tcf7/ Homez/ Hhex/ Mecp2 |
| filter201 | 0.000272047 | 7.825621662 | 3 | ENSMUSG00000079994 |
| filter211 | 0.000270309 | 10.97696139 | 0 | |
| filter85 | 0.000266754 | 2.814624733 | 10 | Atf1/ Homez/ Pbx4/ Pbx2/ Pbx3/ Pbx1/ Setbp1/ Dnajc21/ Atf5/ Atf4 |
| filter132 | 0.000253488 | 4.498443623 | 8 | |
| filter294 | 0.000249725 | 3.734606292 | 7 | Mafk/ Setbp1/ Hmga1/ Hmga2/ Hmga1-rs1/ Atf1/ Zkscan1/ Hhex/ Prkrir/ Dnajc21 |
| filter273 | 0.000242837 | 6.504206603 | 6 | |
| filter94 | 0.000233849 | 12.97110591 | 10 | Irf1/ Stat2/ Spib/ Bcl11a/ Bcl11b/ Sfpi1/ Spib |
| filter209 | 0.000232634 | 9.095851542 | 10 | |
| filter276 | 0.000229846 | 13.49932331 | 10 | Spib/ Sfpi1/ Spib/ Bcl11a/ Bcl11b/ Ehf/ Elf5/ Etv6/ Spic/ Spib |
| filter297 | 0.000225054 | 3.631780121 | 10 | Mecp2 |
| filter58 | 0.000221362 | 4.053184804 | 7 | Setbp1 |

| | | | | |
|---|---|---|---|---|
| filter115 | 0.000220052 | 4.44280022 | 9 | Setbp1/ Hbp1/ Tcf7/ Hhex/ Hmga1/ Hmga2/ Hmga1-rs1/ Tcf7l1/ Nanog/ Hoxb9 |
| filter290 | 0.000207083 | 5.710840446 | 9 | Mafg |
| filter102 | 0.000205864 | 8.741785902 | 10 | |
| filter148 | 0.000197608 | 10.45587484 | 4 | Nfe2l2/ Nfe2 |
| filter194 | 0.000195206 | 9.837238958 | 10 | |
| filter270 | 0.000193527 | 4.41762007 | 8 | |
| filter80 | 0.000190898 | 11.09343066 | 10 | Tcf7l2/ Lef1 |
| filter50 | 0.000164866 | 9.128977137 | 10 | Tcf7l2/ Lef1 |
| filter162 | 0.000163157 | 4.611822441 | 5 | |
| filter30 | 0.000157237 | 4.138480327 | 3 | Homez |
| filter28 | 0.000154734 | 6.226353937 | 9 | Mafg |
| filter73 | 0.000153573 | 8.882060357 | 7 | nan/ Lhx5/ Lhx1/ Gsx1/ Vsx2/ Arx/ Prrxl1/ Vsx2/ Rax/ Prrxl1 |
| filter224 | 0.000152219 | 13.49567136 | 10 | Stat2/ Irf1/ Bcl11a/ Bcl11b |
| filter67 | 0.000151678 | 9.039821323 | 10 | Mzf1 |
| filter127 | 0.000143981 | 9.121384419 | 10 | |
| filter122 | 0.000138046 | 8.212007369 | 10 | Etv6/ Elf5/ Ehf/ Elf1/ Elf2/ Etv2/ Ets2/ Erg/ Erf/ ENSMUSG00000044690 |
| filter207 | 0.00013326 | 5.02135361 | 6 | |
| filter43 | 0.000124214 | 4.641292484 | 8 | Pbx4/ Pbx2/ Pbx3/ Pbx1/ Mafk |
| filter147 | 0.00012326 | 9.240360208 | 6 | |
| filter191 | 0.000111593 | 7.212592448 | 1 | Atf1 |
| filter14 | 0.000109722 | 6.256898425 | 8 | |
| filter164 | 0.000109263 | 4.284922921 | 0 | |
| filter139 | 0.000108369 | 8.403904967 | 1 | Six1 |
| filter151 | 0.000102033 | 6.640858201 | 8 | |
| filter296 | 0.000101989 | 5.25963005 | 7 | |
| filter95 | 0.000101887 | 6.070796759 | 4 | |
| filter268 | 0.000100834 | 4.227078213 | 2 | |

| | | | | |
|---|---|---|---|---|
| filter257 | 0.0000989 | 11.95876667 | 10 | Pax9/ Pax5/ Pax8/ Pax1/ Pax9/ Pax5/ Pax8/ Pax1/ Pax9/ Pax5 |
| filter105 | 0.0000943 | 12.76764281 | 9 | |
| filter180 | 0.000094 | 4.4561539 | 4 | |
| filter183 | 0.0000937 | 9.758769455 | 10 | |
| filter153 | 0.0000915 | 9.197231963 | 6 | |
| filter247 | 0.0000896 | 10.84583255 | 10 | Nfkb1/ Mzf1/ Rel/ Relb/ Nfkb2 |
| filter291 | 0.0000877 | 4.662654179 | 2 | Tcf7 |
| filter70 | 0.0000868 | 9.921290335 | 9 | |
| filter135 | 0.0000867 | 11.07461293 | 9 | Tbpl2 |
| filter210 | 0.0000862 | 6.922007029 | 7 | |
| filter241 | 0.0000839 | 4.526449324 | 4 | |
| filter281 | 0.0000836 | 5.726584818 | 8 | |
| filter298 | 0.0000826 | 4.429468168 | 2 | |
| filter100 | 0.0000787 | 10.74515578 | 5 | |
| filter239 | 0.000078 | 7.510870349 | 5 | |
| filter141 | 0.0000775 | 8.087321777 | 1 | Emx1/ Nfil3/ Nkx1-1/ Nkx1-1/ Meox2/ Tef/ nan/ Vsx2/ Dlx3/ Dlx6 |
| filter203 | 0.000077 | 7.183626561 | 2 | |
| filter72 | 0.0000755 | 4.334675782 | 5 | Setbp1/ Atf1/ Tcf7/ Homez/ Zkscan1/ Pbx4/ Pbx2/ Pbx3/ Pbx1/ Hhex |
| filter83 | 0.0000755 | 4.675820603 | 1 | Dnajc21 |
| filter3 | 0.0000742 | 4.643168512 | 5 | Zkscan1/ Mecp2/ Dnajc21/ Mafk/ Atf1/ Foxo6/ Pou3f3/ Irx5/ Irx1/ Hbp1 |
| filter117 | 0.0000725 | 8.853449517 | 4 | Arnt2 |
| filter213 | 0.0000722 | 8.108926847 | 4 | |
| filter284 | 0.0000708 | 6.326748192 | 8 | Irx2/ Irx1/ Rhox11/ Irx5/ Irx1/ Rfx5/ Rfx6/ Rfx4/ Rfx8/ Dmrt1 |

| | | | | |
|---|---|---|---|---|
| filter234 | 0.0000704 | 5.756526898 | 3 | Atf5/ Atf4/ Naif1/ Atf2/ Sp140/ IRC900814/ Atf1/ Tef/ Kdm2b |
| filter123 | 0.0000703 | 11.10732837 | 10 | nan |
| filter253 | 0.0000698 | 6.241630823 | 5 | |
| filter101 | 0.0000688 | 8.482753328 | 0 | |
| filter31 | 0.000068 | 11.91286781 | 3 | |
| filter136 | 0.0000664 | 7.723492504 | 3 | |
| filter131 | 0.0000663 | 7.754124606 | 2 | |
| filter229 | 0.000066 | 6.09104818 | 6 | Hhex/ Hoxd12/ NP_032300.2/ Hoxa10/ Hoxb9/ NP_032296.2/ Hoxc8/ Hoxb9/ Hoxd4/ NP_032296.2 |
| filter75 | 0.0000658 | 10.68908648 | 2 | |
| filter69 | 0.0000657 | 13.12038514 | 10 | Ctcf |
| filter49 | 0.0000654 | 8.650625448 | 7 | Atf3/ Jdp2/ Atf2/ Atf7 |
| filter198 | 0.0000652 | 9.886005642 | 8 | Pax9/ Pax5/ Pax8/ Pax1 |
| filter250 | 0.0000652 | 10.7869779 | 10 | Irf1 |
| filter143 | 0.000065 | 7.746473215 | 4 | |
| filter244 | 0.0000647 | 12.95999431 | 10 | Etv2/ Ets2/ Ets1/ Erf/ ENSMUSG00000044690/ Fev/ Etv2/ Ets2/ Erg/ Erf |
| filter283 | 0.0000645 | 7.472977411 | 9 | Nrl/ Mafa/ Maf |
| filter170 | 0.0000639 | 5.602994073 | 8 | |
| filter99 | 0.0000635 | 9.123393773 | 1 | |
| filter155 | 0.0000631 | 6.884448159 | 10 | |
| filter159 | 0.0000631 | 5.63674747 | 5 | Zfp300/ Zscan20/ Atf1 |
| filter214 | 0.0000619 | 5.968200513 | 10 | |
| filter91 | 0.0000612 | 7.265551141 | 3 | |
| filter55 | 0.000061 | 6.644188575 | 6 | |
| filter4 | 0.0000609 | 10.02617469 | 0 | |
| filter81 | 0.0000604 | 5.143625096 | 5 | Hhex |
| filter124 | 0.0000602 | 8.219548597 | 9 | |
| filter176 | 0.0000599 | 8.022155786 | 4 | |
| filter188 | 0.0000592 | 8.910855997 | 0 | |

| | | | | |
|---|---|---|---|---|
| filter249 | 0.0000586 | 7.252156283 | 0 | |
| filter259 | 0.0000582 | 12.02149639 | 2 | |
| filter146 | 0.0000579 | 6.651357662 | 1 | |
| filter287 | 0.0000566 | 5.23635501 | 1 | |
| filter12 | 0.0000556 | 10.19329729 | 3 | |
| filter24 | 0.0000554 | 6.276244049 | 1 | |
| filter6 | 0.0000552 | 10.77284828 | 10 | |
| filter144 | 0.0000552 | 7.10771801 | 1 | |
| filter41 | 0.0000548 | 5.887367592 | 1 | |
| filter160 | 0.0000546 | 8.384143074 | 3 | |
| filter98 | 0.0000545 | 9.05066866 | 4 | |
| filter184 | 0.0000542 | 5.644051942 | 3 | Dnajc21/ Setbp1/ Atf1/ Pbx4/ Pbx2/ Pbx3/ Pbx1 |
| filter20 | 0.0000541 | 7.857412384 | 2 | |
| filter182 | 0.0000541 | 7.857444219 | 6 | |
| filter39 | 0.0000538 | 9.34016601 | 0 | |
| filter82 | 0.0000529 | 8.952985455 | 3 | |
| filter60 | 0.0000527 | 8.437548134 | 4 | |
| filter149 | 0.0000526 | 7.123474357 | 0 | |
| filter269 | 0.0000512 | 6.968962403 | 7 | |
| filter299 | 0.0000511 | 8.056581304 | 3 | |
| filter171 | 0.0000509 | 8.807042867 | 0 | |
| filter2 | 0.0000508 | 6.602491545 | 0 | |
| filter111 | 0.0000504 | 6.551389637 | 4 | |
| filter140 | 0.0000503 | 6.940249007 | 0 | |
| filter130 | 0.0000501 | 8.686382883 | 3 | |
| filter66 | 0.0000499 | 9.306904407 | 8 | |
| filter145 | 0.0000499 | 7.835945138 | 0 | |
| filter152 | 0.0000496 | 10.96147533 | 0 | |
| filter272 | 0.0000495 | 10.36105545 | 1 | |
| filter233 | 0.0000494 | 8.142564571 | 4 | |
| filter42 | 0.0000491 | 8.278420741 | 4 | |
| filter1 | 0.000049 | 7.135086991 | 10 | Atf3 |
| filter142 | 0.000049 | 6.16214829 | 0 | |
| filter216 | 0.0000489 | 6.501634666 | 5 | |
| filter29 | 0.0000487 | 10.95965327 | 1 | |
| filter126 | 0.0000485 | 7.849950976 | 4 | |
| filter246 | 0.0000485 | 7.794905625 | 0 | |

| | | | | |
|---|---|---|---|---|
| filter178 | 0.0000481 | 10.327217 | 8 | Sp2/ Sp3/ Sp6/ Sp8/ Sp7/ Sp9/ Sp5/ Sp2/ Sp3/ Sp6 |
| filter199 | 0.0000481 | 7.744841132 | 4 | Hoxb1/ Hoxa1/ Hoxb2/ Hoxd4/ Hoxa6/ Hoxb2/ Pou6f1/ Pou2f3/ Meox1/ Hoxa2 |
| filter33 | 0.0000477 | 5.360230505 | 5 | |
| filter262 | 0.0000475 | 10.74862289 | 2 | |
| filter103 | 0.0000474 | 7.118303554 | 0 | |
| filter192 | 0.0000474 | 8.358581497 | 3 | |
| filter17 | 0.0000473 | 6.694936865 | 2 | |
| filter138 | 0.0000462 | 8.530819708 | 2 | |
| filter109 | 0.0000461 | 8.93098452 | 1 | |
| filter222 | 0.0000461 | 6.379401097 | 0 | |
| filter26 | 0.000046 | 8.427250934 | 4 | |
| filter232 | 0.000046 | 6.942583904 | 2 | |
| filter197 | 0.0000457 | 9.417900687 | 0 | |
| filter248 | 0.0000455 | 7.397851979 | 1 | |
| filter256 | 0.0000454 | 10.26651299 | 6 | |
| filter265 | 0.0000454 | 5.77818146 | 1 | |
| filter74 | 0.0000451 | 7.79247293 | 2 | |
| filter175 | 0.0000447 | 9.308963928 | 4 | |
| filter18 | 0.0000445 | 10.03106411 | 2 | |
| filter36 | 0.0000445 | 8.023095847 | 3 | |
| filter228 | 0.0000442 | 8.955259152 | 4 | |
| filter215 | 0.000044 | 6.658615306 | 3 | |
| filter5 | 0.0000439 | 6.060014083 | 4 | |
| filter61 | 0.0000437 | 8.93558998 | 3 | |
| filter13 | 0.0000434 | 5.896416221 | 1 | |
| filter157 | 0.0000433 | 9.454012595 | 8 | |
| filter125 | 0.0000431 | 10.96646503 | 1 | |
| filter137 | 0.0000423 | 9.166575155 | 2 | |
| filter19 | 0.0000418 | 6.755846361 | 1 | |
| filter186 | 0.0000412 | 9.399774999 | 10 | |
| filter267 | 0.000041 | 7.152791477 | 3 | |
| filter45 | 0.0000406 | 10.50199055 | 2 | |
| filter88 | 0.0000405 | 5.916821437 | 4 | |
| filter63 | 0.0000403 | 9.971185496 | 0 | |
| filter92 | 0.0000402 | 10.05949031 | 6 | |

| | | | | |
|---|---|---|---|---|
| filter263 | 0.0000401 | 8.316991382 | 5 | |
| filter285 | 0.00004 | 7.637366915 | 1 | |
| filter44 | 0.0000399 | 9.969803863 | 8 | Tbpl2 |
| filter280 | 0.0000398 | 9.85114045 | 4 | |
| filter21 | 0.0000396 | 7.816081845 | 1 | |
| filter200 | 0.0000391 | 10.97817101 | 3 | |
| filter261 | 0.0000388 | 9.110338959 | 2 | |
| filter254 | 0.0000385 | 10.28534523 | 6 | |
| filter278 | 0.0000385 | 7.162393957 | 6 | |
| filter48 | 0.0000382 | 7.192766847 | 4 | |
| filter54 | 0.0000379 | 10.32338372 | 0 | |
| filter113 | 0.0000379 | 7.770356573 | 0 | |
| filter84 | 0.0000374 | 5.666161006 | 7 | |
| filter118 | 0.0000371 | 7.609764226 | 1 | Junb/ Zic4/ Zic3/ Zic1/ Zic4/ Tbx3 |
| filter161 | 0.0000369 | 7.123738039 | 0 | |
| filter177 | 0.0000369 | 8.053713308 | 7 | |
| filter206 | 0.0000368 | 10.52473197 | 2 | |
| filter86 | 0.0000366 | 6.70425404 | 1 | |
| filter237 | 0.0000366 | 7.347821421 | 1 | |
| filter289 | 0.0000364 | 9.989819394 | 7 | |
| filter32 | 0.0000362 | 7.846011003 | 0 | |
| filter185 | 0.0000358 | 5.469848364 | 8 | Mecp2/ Hhex/ Nr2e1/ Xbp1/ Setbp1/ Cphx |
| filter243 | 0.0000358 | 6.350843954 | 4 | Cphx/ Pbx4/ Pbx2/ Pbx3/ Pbx1 |
| filter107 | 0.0000358 | 10.03660388 | 2 | |
| filter59 | 0.0000353 | 10.09228773 | 2 | |
| filter79 | 0.0000352 | 8.447790033 | 1 | |
| filter16 | 0.0000351 | 6.795651699 | 4 | |
| filter96 | 0.0000346 | 9.658079637 | 1 | |
| filter150 | 0.0000346 | 10.65068912 | 2 | |
| filter119 | 0.0000344 | 10.44550581 | 2 | Usf1/ Mitf/ Tcfe3/ Arntl/ Tcfec/ Arnt/ Bhlhe41/ Gmeb1/ Creb1/ Usf2 |
| filter104 | 0.0000341 | 8.166199904 | 0 | |
| filter202 | 0.0000336 | 9.735935293 | 10 | |
| filter46 | 0.0000336 | 8.58816777 | 0 | |
| filter193 | 0.0000336 | 8.507109625 | 2 | |

| | | | | |
|---|---|---|---|---|
| filter77 | 0.0000332 | 9.536742897 | 0 | |
| filter62 | 0.000033 | 7.164869314 | 2 | Gm239/ Gm98/ Zkscan1 |
| filter27 | 0.000033 | 7.756526575 | 1 | |
| filter189 | 0.000033 | 6.754216621 | 5 | |
| filter90 | 0.0000326 | 10.97267291 | 4 | |
| filter181 | 0.0000326 | 10.90853891 | 0 | |
| filter56 | 0.0000324 | 7.330371177 | 1 | |
| filter196 | 0.0000323 | 9.704030662 | 9 | |
| filter225 | 0.0000321 | 9.27022719 | 0 | |
| filter223 | 0.0000316 | 9.304710026 | 10 | |
| filter156 | 0.0000316 | 9.386572384 | 1 | |
| filter52 | 0.000031 | 8.134521812 | 1 | |
| filter226 | 0.0000309 | 12.10358919 | 0 | |
| filter212 | 0.0000308 | 6.822379603 | 7 | |
| filter64 | 0.0000306 | 6.770211089 | 2 | Gata2/ Gata5/ Gata2/ Gata3 |
| filter38 | 0.0000305 | 8.761957551 | 1 | |
| filter53 | 0.0000305 | 8.050241968 | 4 | |
| filter116 | 0.0000295 | 10.40140809 | 2 | |
| filter76 | 0.000029 | 12.22319326 | 2 | |
| filter110 | 0.0000289 | 9.901229468 | 1 | |
| filter266 | 0.0000286 | 8.827576915 | 5 | |
| filter22 | 0.0000279 | 7.788815142 | 1 | |
| filter282 | 0.0000278 | 6.17079111 | 0 | |
| filter293 | 0.0000275 | 7.235811417 | 4 | |
| filter204 | 0.0000274 | 6.518757601 | 2 | |
| filter7 | 0.0000265 | 8.805051502 | 1 | |
| filter221 | 0.0000259 | 8.620098067 | 0 | |
| filter277 | 0.0000256 | 6.159722079 | 3 | |
| filter108 | 0.0000255 | 8.508599291 | 5 | |
| filter71 | 0.0000252 | 6.780928854 | 4 | |
| filter134 | 0.0000248 | 11.87473453 | 1 | |
| filter251 | 0.0000243 | 10.79871508 | 3 | |
| filter37 | 0.0000242 | 10.9847853 | 2 | |
| filter129 | 0.0000235 | 8.840784607 | 8 | Rarg/ Rara/ Nr2c2/ Nr2c1/ Pparg/ Ppard/ Ppara/ Esr1/ Esr2/ Thrb |
| filter0 | 0.000023 | 7.387973909 | 3 | |

| | | | |
|---|---|---|---|
| filter158 | 0.0000223 | | |
| filter163 | 0.0000223 | 9.075127994 | 0 |
| filter25 | 0.0000203 | 12.65092678 | 5 |
| filter168 | 0.0000189 | 7.159924343 | 2 |
| filter208 | 0.000017 | | |
| filter169 | 0.0000108 | | |
| filter235 | 0.0000107 | 14.87710862 | 5 |
| filter187 | 0.00000766 | 9.047789275 | 0 |
| filter87 | 0.00000421 | | |
| filter258 | 0.00000324 | 10.78377334 | 0 |