# Epigenetic Imputation

*Alexander Ku*

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2018-71
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2018/EECS-2018-71.html

May 17, 2018

# Epigenetic Imputation

Alexander Ku

(Joint research with Gunjan Baid)

*UC Berkeley*

May 2018

**Abstract**

Understanding binding affinities of transcription factor (TF) proteins to DNA sequence is crucial to the identification of regulatory regions that control differential gene expression across cell types. Recent advancements in ChIP-sequencing (ChIP-seq) allow us to accurately identify binding sites for a specific TF in a cellular context of interest. However, running a separate assay for each of the thousands of known TFs for a new cell type of interest is time and cost-intensive, thus motivating the need for an efficient computational method to infer experimental results of unknown experiments using prior information gathered from experiments on robustly annotated cell types. We propose an attention-based deep learning approach for learning the minimal set of epigenetic experiments required to accurately quantify transcription factor (TF) binding sites from DNA sequence.

# Contents

# 1 Introduction

## 1.1 Motivation

Understanding binding affinities of transcription factor (TF) proteins to DNA sequence is crucial to the identification of regulatory regions that control differential gene expression across cell types. TFs recognize and bind to short DNA motifs, regulating the rate at which nearby genes are transcribed into RNA. TF binding specificity is not just determined by DNA sequence: TF cooperation and other epigenetic factors also dictate where a given TF binds. TF cooperation occurs when multiple TFs bind to nearby motifs, co-regulating expression of nearby genes [6]. TF binding also depends on chromatin accessibility, whose local and long range 3-dimensional interactions dictate the ability of TFs to access regulatory regions of the genome [5].

Recent advancements in ChIP-sequencing (ChIP-seq) allow us to accurately identify binding sites for a specific TF in a cellular context of interest [13]. However, running a separate assay for each of the thousands of known TFs for a new cell type of interest is time and cost-intensive, thus motivating the need for an efficient computational method to infer (or impute) experimental results of unknown experiments using prior information gathered from experiments on robustly annotated cell types.

We propose an attention-based deep learning model based on the Transformer [15] for learning the minimal set of epigenetic experiments required to accurately quantify transcription factor (TF) binding sites from DNA sequence. Our method combines DNA sequence and partial epigenetic information to learn the most informative set of biological experiments that can be leveraged to impute missing experiments on unseen cell types. We frame this problem as a sequence transformation problem, whereby partially observed labels are transformed to have their missing values inferred, allowing for better predictions as more epigenetic data is provided to the model.

We train and evaluate this model on four cell lines from the ENCODE consortium [4], and though our preliminary experiments were unable to achieve state-of-the-art in predicting TF binding sites, we do find that the information gain through incremental inclusion of experimental data improves prediction accuracy. We see this work-in-progress as a promising argument for imputation as a valid framework for transfer learner and a step towards cell type agnostic models.

## 1.2 Outline

The remaining sections in this report are structured as follows. Section 2 reviews background information on the process of transcription, biological assays, and the deep neural networks. Section 3 covers both related works in transcription factor binding site prediction and other examples of deep learning in genomics. Section 4 goes into detail on our methods and model architecture. Results are

presented in Section 5. Section 6 includes analysis and discussion of the results, and concludes with potential areas of future exploration.

# 2 Background

## 2.1 Regulation of Gene Expression

The first step of gene expression is the transcription of DNA into mRNA, which begins with the recruitment of RNA polymerase to the gene. One or more TFs may serve to either activate or block the recruitment RNA polymerase, thereby directly upregulating or downregulating the given gene. TFs can act by binding to the promoter site, which is proximal to the gene, or to distal enhancer sites. While enhancers are linearly far from the promoter, they are brought into close spatial contact through folding of the DNA.

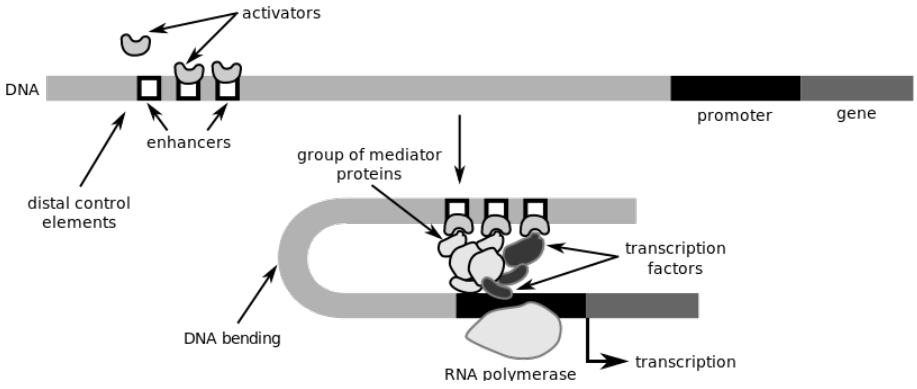


Figure 1: Sketch of gene expression regulation, from Wikimedia Commons, the free media repository [3].

TF binding specificity depends not just on binding site DNA sequence, but

also on chromatin accessibility. Nuclear DNA is packed tightly in chromatin complexes, which means that most of the genome is inaccessible to the proteins that regulate transcription. A region of DNA is referred to as being in an "open" state if it is accessible, and in a "closed" state otherwise. Open chromatin is associated with active transcription, as the loose structure allows TFs, RNA polymerase, and other proteins to access sites of interest.

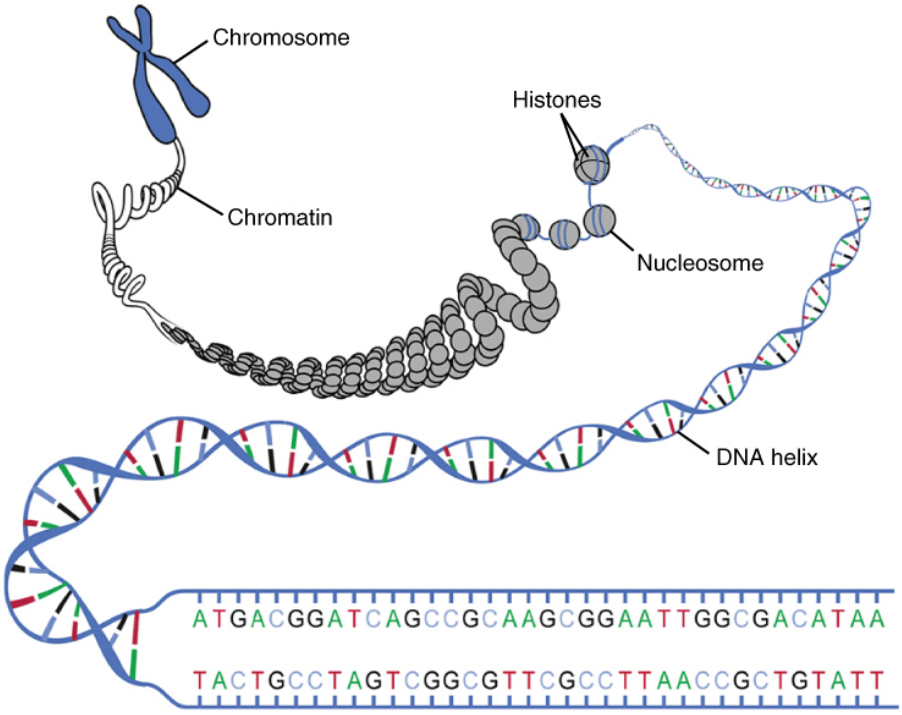


Figure 2: Sketch of chromatin macrostructure, from Wikimedia Commons, the free media repository [2].

## 2.2 Deep Neural Networks

Deep neural networks (DNNs) are modern renditions of the classic multilayer perceptrons from the sixties, and represent a powerful class of machine learning models that are able to fully leverage modern compute. DNNs can be thought of as abstract function approximators that can be trained efficiently through stochastic gradient descent when given enough data and structurally specialized for particular domains [10]. For instance, deep convolutional neural networks (CNNs; [11]) excel in tasks where the input has strong spatial locality, such as natural images and audio signals. CNNs exploit spatial locality by learning a hierarchical set of translation-invariant filters. These translation-invariant properties are enjoyed in many biological applications, such as scanning the genome to call variants, which is only feasible when parameters are shared across genomic regions.

Recurrent neural networks (RNNs; [12]) share parameters along a temporal axis. They differ from CNNs in that they maintain an internal state which gets updated as the input is processed sequentially. RNNs excel in tasks where long-range relationships in the sequence need to be captured, such as natural language parsing. One pitfall of RNNs is that updating the internal state at each timestep requires the updated state from the previous timestep, a property which earns the RNN its "recurrent" namesake. The sequential nature of this computation make RNNs difficult to parallelize, and as a consequence slow to train.

Until recently, the state-of-the-art in sequence-to-sequence modeling has

been dominated by hybrid architectures that add an attention mechanism to an RNN [9]. Recent work on the Transformer [15] mitigates this computational constraint imposed by RNNs by relying solely on the attention component of the hybrid architecture, which can be trivially parallelized as a series of dot-products. The main component of the Transformer is the self-attention layer which computes representations at each position in parallel with scaled dot-product attention, relating different positions within a single sequence to compute the succeeding representation, capturing the entire scope of a sequence in a single layer. Scaled dot-product attention is computed as follows, where $K, V, Q$ are the keys, values, and queries, and $d$ is the queries and keys dimension:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d}})V \tag{1}$$

The authors found it beneficial to compute several attention functions by linearly projecting the keys, values, and queries into multiple representation spaces. A modification they call multi-headed attention.

Like most sequence-to-sequence architectures, the Transformer is built from an encoder-decoder schema, where the encoder maps an input sequence to an intermediary continuous representation, which is then decoded to an output sequence by a decoder. In an encoder layer, one or two sub-layers of self-attention is followed by a fully-connected feed-forward network. In a decoder layer, one or two sub-layers of self-attention is followed by one or two sub-layers of encoder-decoder attention, which integrates information from the encoder, and finally by a fully-connected feed-forward network. In self-attention layers,

the keys, values, and queries all come from either the encoder or decoder. For the encoder-decoder attention, the keys and values come from the encoder while the queries come from the decoder.

# 3 Related Work

## 3.1 DeepBind

Several works have used convolutional neural networks to predict TF binding sites from DNA sequence. DeepBind [16] uses a single convolutional layer followed by a fully connected layer to classify sequences on variable length regions. Though this simple architecture is unable to achieve state-of-the-art in predicting TF binding sites, the interpretability of the filters it learns are surprisingly insightful and valuable to the biological community, illustrated in Figure 3.

## 3.2 DeepSEA

DeepSEA [1], uses a 3-layer convolutional neural network to predict binding behavior of 200bp regions of DNA, optionally flanked by up to 800bp of "context". This model is hosted online to be applied by biologists in predicting the local effects of single nucleotide polymorphisms (SNPs). SNPs are small genetic mutations which often turn genes on and off, alter protein structures in a cell, and cause genetic diseases including cancer. The model's accuracy increases as more contextual information is given, suggesting that there is value in considering larger genomic regions. DeepSEA serves as our state-of-the-art point of

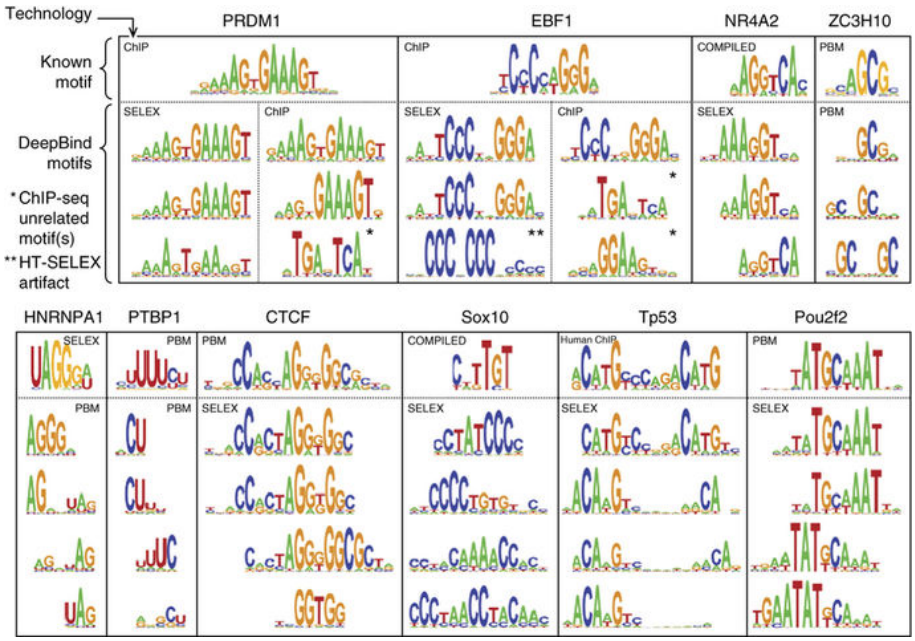


Figure 3: Example motif detectors learned by DeepBind models, along with known motifs, from [16].

comparison.

# 4 Approach

## 4.1 Imputation

While the neural network architectures in Section 3 have had considerable success in predicting TF binding sites from DNA sequence, they have two limitations that we seek to address. First, these methods predict TF binding events independently, and fail to share information regarding cooperative binding across model parameters. Second, because binding events are predicted independently,

when partial experimental information from a new cell type is available, the model is unable to leverage prior knowledge to bootstrap the prediction of unknown experiments.

To overcome these two limitations, we present a neural network architecture based on the Transformer that aims to improve TF binding site prediction by imputing a set of unknown binding events from a set of known experiments. This imputation model can be applied on a held out cell type of interest, removing the cell type specific constraints of previous models [1]. We frame this problem as a sequence-to-sequence task, whereby partially observed labels are transformed to have their missing values imputed. The method, which we call Transcription Factor Transformer Imputation (TFTI), incorporates cell type specific information into the model, allowing for transfer learning to unseen cell types.

## 4.2 Model Architecture

We modify several components of the original transformer architecture. On the encoder side, our imputation model takes in as input an $n$-gram embedded DNA sequence. We choose to work specifically with 4-grams to ensure a balance between expressive power and model complexity. Following the multi-headed attention, the encoder uses convolutional layers, instead of position-wise fully-connected layers. We found that switching from fully-connected to convolutional layers improved model performance.

On the decoder side, the network takes in as input a embedded sequence of

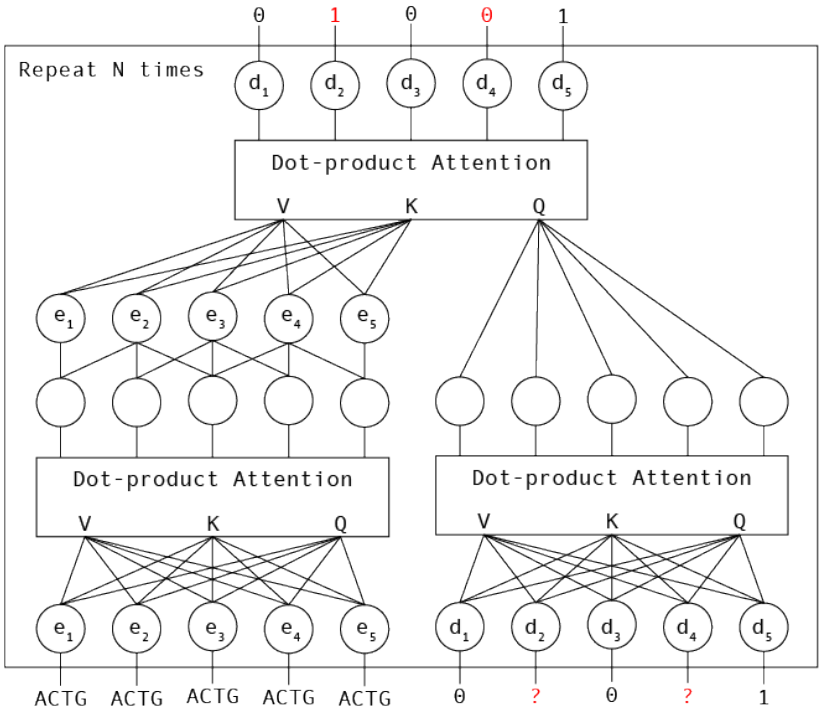


Figure 4: The TFTI Transformer network.

partially observed labels, where missing labels get a unique embedding, and outputs a sequence of binary labels corresponding to negative and positive values. All missing labels are imputed in the final output. Since there is no underlying order in the label space, we remove the positional embedding and causal masking used in the decoder self-attention. The decoder generates outputs using the latent input, so we want to allow each position to attend to all other positions, including subsequent ones. The total size of our network is ∼50 million parameters, which is comparable to DeepSEA's ∼60 million parameters. The TFTI model architecture is illustrated in Figure 4.

## 4.3 Input and Label Embedding

For the DNA sequence input, we use the DNA encoder in the Tensor2Tensor [**tensor2tensor**] library to represent each 4-gram with a unique token. We embed each of the 4-grams using a learned embedding matrix and use sinusoidal positional encoding, which was introduced by [15]. For the labels, we use position-wise learned embeddings, as a binding event for a particular transcription factor should be interpreted differently as a binding event for another transcription factor. Thus, for each label, we learn an embedding for the positive, negative, and unknown values.

## 4.4 Training Procedure

To train a model that can perform imputation of missing data at test time, we introduce stochasticity into the training procedure by randomly discarding a subset of the labels. The proportion of the labels feed to the model is a hyperparameter, $p$, where any particular label is kept with probability $p$, and discarded labels are replaced with an unknown token. We implement this procedure by generating a random boolean mask for each batch of training examples. When evaluating the model, we modify this procedure to mask out the same label across all examples in a batch. The $p$ used at inference time does not have to match the $p$ used at test time, we found that setting $p = 0.1$ at training time yields the best results.

# 5 Preliminary Results

## 5.1 Dataset

Both training and evaluation use the dataset published by the authors of DeepSEA, which contains 4.2 million training examples from the ENCODE consortium [**encode-1**, **encode-2**]. Each training example is DNA sequence of length 1000, one hot encoded, and a corresponding binary label vector for 919 chromatin features. Each value in the label applies only the middle 200bp of the DNA sequence, and the 400bp flange on each side provides additional context. The chromatin features contain not only transcription factors, but also DNase I-hypersensitivity sites and histone marks. The DNA sequences from the hg37 reference genome and only those sequences with at least one transcription factor binding event are included in the dataset. The dataset suffers from class imbalance as most of the labels are negative.

We consider only a subset of the labels present in the DeepSEA dataset. Specifically, we look at the HeLa-S3, GM12878, HepG2, and K562 cell types. We train on the HeLa-S3, GM12878, HepG2, and K562 cell types and validate on held out chromosomes from the same cell types. For testing, we use the H1-hESC cell type, which is never seen during training, to evaluate the transfer learning capabilities of our model. This setup is similar to the zero-shot learning problem, as the model is trained without any data from the test cell type. There is biological motivation for such a setup, as we seek to apply our model to newly sequenced cell populations.

## 5.2 Comparison to DeepSEA

We compare our model to DeepSEA [16], a state-of-the-art convolutional model for predicting TF binding sites from DNA sequence. We trained our model using the same training and validation sets as DeepSEA, but considered only a subset of their labels corresponding to cell type GM12878 to allow for transferability between cell types. In comparing these models we consider a set of 35 epigenetic marks, including 34 transcription factor binding events and a binary DNase label. Each of these marks represents a separate but correlated binary prediction task. Each model is evaluated on the average Area Under Receiver Operating Characteristic (avgAUC) across these 35 marks. In this comparison, methods are evaluated on held out chromosomes of the same cell type. As shown in Figure 2, our model does not outperform DeepSEA, even when additional epigenetic information is provided. Although it is important to note that our model is cell type agnostic and generalizes to unseen cells, as shown in Figure 1.

## 5.3 Transfer Learning

While restraining models to a single cell type makes comparing to existing models easier, the most exciting use cases of our model involve application to a new cell type absent in the training set. To this end, we train a model on 19 epigenetic marks (18 TFs and DNase) across 4 cell types (HeLa-S3, GM12878, HepG2, K562) and test on a fifth (H1-hESC). Cell types and transcription factors were chosen to have a large intersection of marks over a sufficient number of cell types in the DeepSEA dataset. In this context we expect imputation to

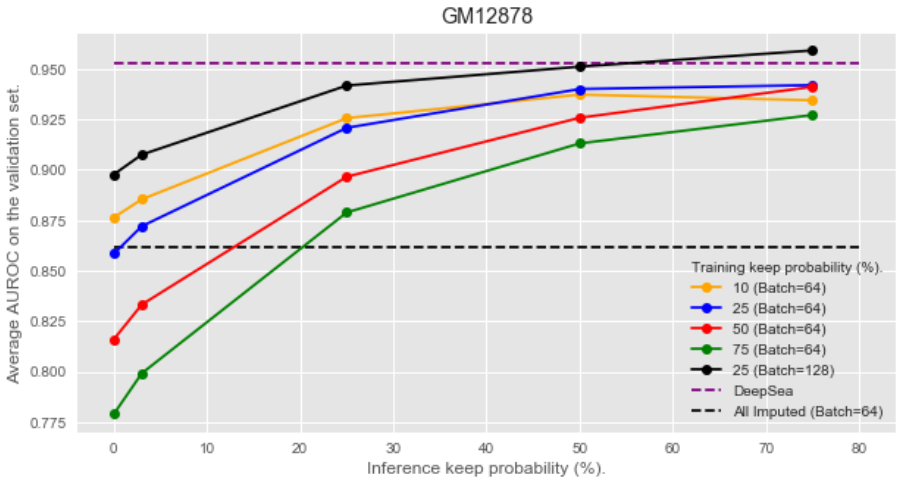


Figure 5: Imputation curves of models keeping different percents of the marks.

be valuable, since despite all cell types having the same genetic sequence, the behavior of TFs differ and offers a window of insight into epigenetic behavior.

| Probability of keeping a mark | Average AUC on held out chromosome in test cell | Average AUC on held out chromosome in training cells |
|---|---|---|
| 0.00 | 0.8211206 | 0.874447 |
| 0.03 | 0.8250943 | 0.883317 |
| 0.25 | 0.8473549 | 0.918758 |
| 0.50 | 0.8641222 | 0.933156 |
| 0.75 | 0.8725221 | 0.937384 |

Table 1: Multi-cell model trained with 25% on held out chromosomes.

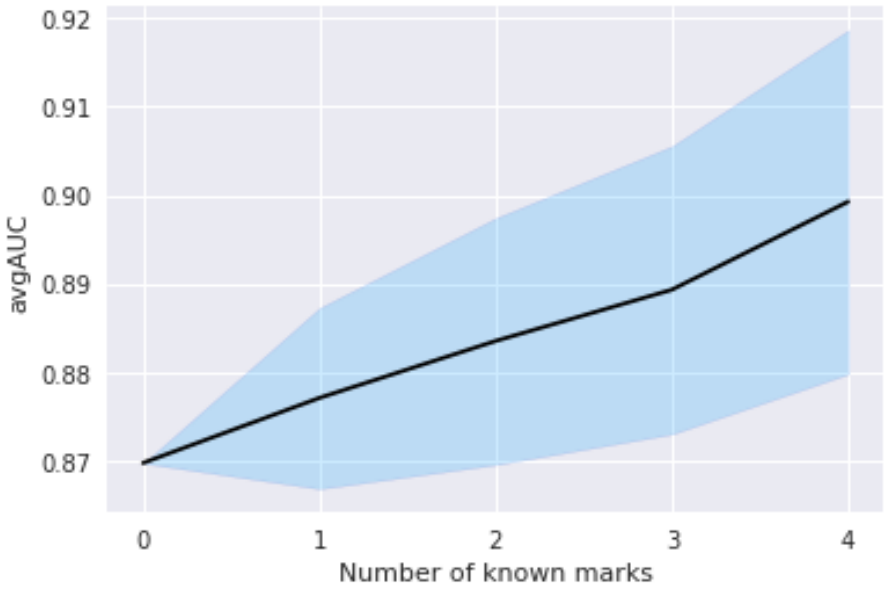


Figure 6: Average AUC as the number of known marks is increased. The blue bands represent lower and upper bounds of the Average AUC and the black line is the Expected Average AUC taken across all possible subsets of a given size.

## 5.4 Experimental Design

### 5.4.1 Beam Search

We perform greedy forward subset selection based on beam search over our imputation model to determine the set of experiments which maximizes prediction accuracy on imputed experiments. Beam search provides a tractable approach to exploring the space of all experimental subsets.

We define $E$ as the set of all experiments and $S \subseteq E$ as a subset of those experiments. Starting from the root of the search tree $\mathcal{B}_0 = \{\emptyset\}$, beam search computes candidate beams $\mathcal{B}'_i = \{S \cup \{e\} : S \in \mathcal{B}_{i-1}, e \in E, e \notin S\}$. Candidates

are ordered according to a heuristic value function and only the top $\beta$ experiment sets are kept, thus $\mathcal{B}_i \subseteq \mathcal{B}'_i$ and $|\mathcal{B}_i| = \beta$. For our heuristic, we use the average area under the curve (AUC) computed over all epigenetic marks when partial information from experiments in $S$ is provided to the model during inference. This heuristic is not sub-modular, we found that adding particular marks to a subset occasionally has a negative effect on AUC.

| Subset size | Experiments | Average AUC of imputed marks on held out chromosome |
|---|---|---|
| 0 | $\emptyset$ | 0.869723 |
| 1 | {CHD2} | 0.905142 |
| 2 | {DNase, CHD2} | 0.924376 |
| 3 | {RFX5, DNase, CHD2} | 0.934909 |
| 4 | {RFX5, EZH2, DNase, CHD2} | 0.944609 |

Table 2: Average AUC of optimal subsets up to size four.

Although beam search is not guaranteed to give the optimal subset of experiments, we can always trade computational efficiency for a better solution by increasing $\beta$. When $\beta$ is infinite, beam search is equivalent to an exhaustive breadth-first search. In terms of complexity, exhaustively searching the space of experiment subsets of size $k$ is $\mathcal{O}(2^k)$ in time complexity, while beam search runs in $\mathcal{O}(\beta k)$. We find that running beam search with $\beta = 4$ to find optimal subsets up to size $k = 4$ gets the same results as exhaustive search on the imputation task, shown in Table 2.

### 5.4.2 Shapley-value Analysis

Although Beam Search provides us with an optimal set of experiments, it does not provide us with insight into how specific assays affect the efficacy of our model. Multi-perturbation Shapley-value Analysis (MSA; [8]) gleans insight into the marginal and cooperative contributions of epigenetic marks learned from our imputation model, allowing us to determine the consequence of different assay pairs on the final accuracy of our model. MSA has been succsessfully been employed to identify the contributions of multiple genes to the success of specific biological pathways [7]. In our setting, we use the marginal contributions indentified by MSA to identify outliers that have synergistic or adversarial affects on prediction accuracy. Finally, two dimensional MSA allows us to identify cooperative contribution among marks.

The Shapley value computes the overall gain from a subset of a coalition of players, and can determine the most important players in the outcome of a game [14]. To determine marginal and cooperative contributions of epigenetic marks in the prediction accuracy of our model, we define a coalitional game as a set of experiments $N$ of size $n$ and a value function $v$ that measures the contribution of a subset of experiments in the coalition. We define the value function $v(S)$ as follows:

$$v(S) = \text{AverageAUC}(S) - \text{AverageAUC}(\emptyset) \tag{2}$$

Where $S$ is a subset of epigenetic experiments, and AverageAUC($S$) is the average area under the curve (AUC) computed over all epigenetic marks when

partial information from experiments in $S$ is provided to the model during inference. We use this value function to compute the generalized Shapley Value $\phi$ for a given subset of epigenetic marks $C$:

$$\phi_C(v) = \sum_{T \subseteq N \setminus C} \frac{(n - |T| - |C|)!|T|!}{(n - |C| + 1)!} \sum_{S \subseteq C} (-1)^{|C|-|S|} v(S \cup T) \tag{3}$$

Computing the Shapley value for each experiment requires inference to be run on all possible combinations of partial experiments from the 19 original experiments, resulting in 524,287 inferences to be run. To decrease the number of inference runs, we utilize Multi-perturbation Shapley Value Analysis (MSA) to predict the value function for combinations containing more than five experiments using projection pursuit regression [8].

We perform one and two dimensional MSA to compute the marginal contributions of nineteen epigenetic marks, including eighteen TFs and one chromatin accessibility mark (DNase hypersensitivity). Figure 7 visualizes the marginal and shared contribution of each mark. One dimensional Shapley Analysis reveals that CHD2, DNase, and p300 have the highest marginal contribution in the system. These marginal contribution scores closely align with the average AUC calculated from each mark.

Two dimensional MSA characterizes pairwise interactions between marks. The two dimensional information between mark $i$ and $j$, denoted $I_{i,j}$, is specified in [8] as:

$$I_{i,j} = \phi(i,j) - \phi_C(i,\bar{j}) - \phi_C(\bar{i},j) \tag{4}$$

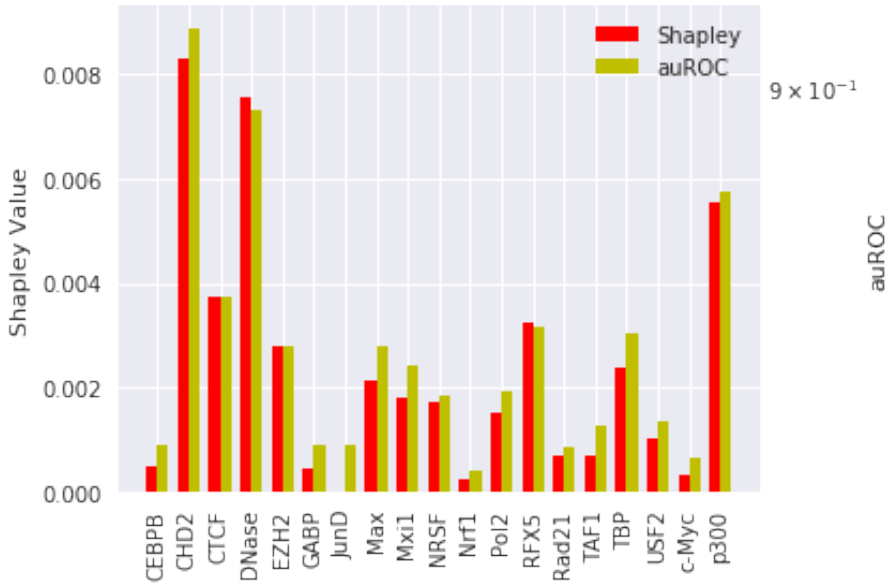


Figure 7: One dimensional MSA run on 19 epigenetic marks.

Where $\phi(i,j)$ is the Shapley value of $i$ and $j$, $\phi_C(i,\bar{j})$ is the Shapley value of the game containing $i$ but not $j$, and $\phi_C(j,\bar{i})$ is the Shapley value of the game containing $j$ but not $i$. The 2D information indicates whether pairwise interactions are neutral, synergistic or antagonistic. If $I_{i,j} = 0$ then the marginal contribution of marks $i$ and $j$ are additive, if $I_{i,j} > 0$ then the pairwise interaction is synergistic, otherwise the interaction is antagonistic. Figure 8 plots a heatmap of the two dimensional information computed from all pairwise combinations of the 19 marks. From this figure, we conject that the interaction between JunD and multiple other factors is antagonistic, which is a hypothesis that is supported by the one dimensional MSA analysis, as JunD has no marginal contribution.

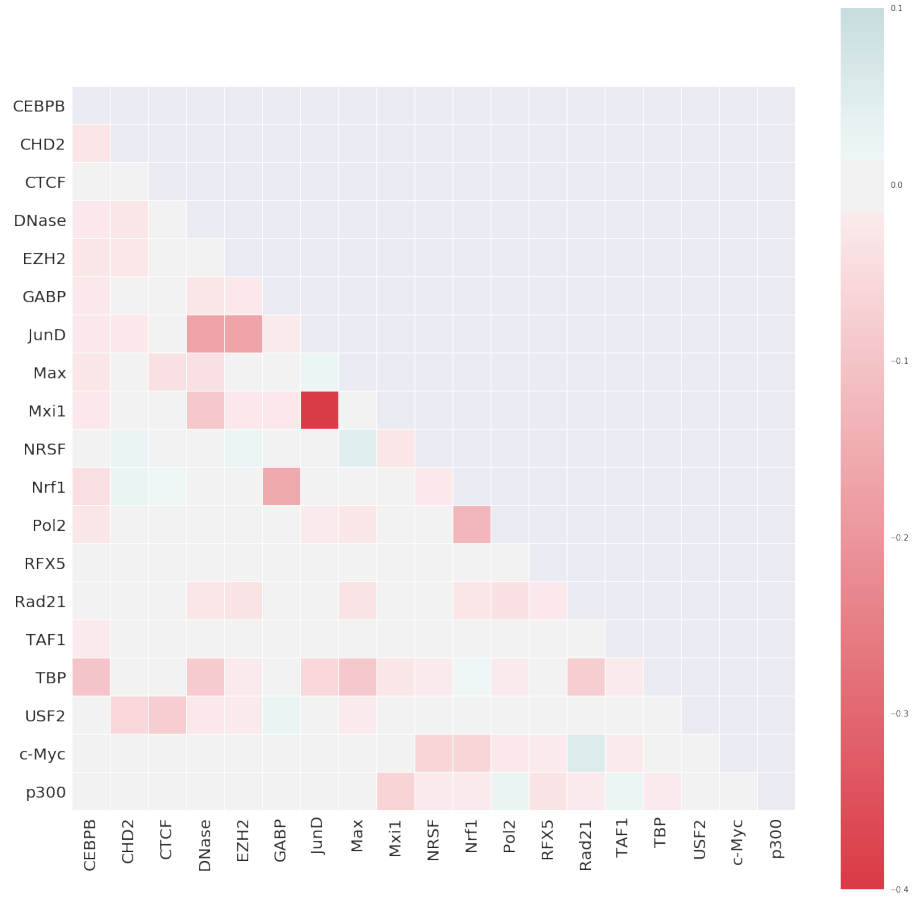


Figure 8: Two dimensional MSA run on 19 epigenetic marks.

# 6 Discussion

So far, in our work-in-progress, we were not able to achieve state-of-the-art results in predicting TF binding sites. However, this project is a continued effort and represents the iterative refinement process of research. We had a hypothesis, that providing epigenetic data to our model would improve model performance and enable cell type generalization, which we show to be true using

our imputation method. Furthermore, we show that our imputation model can be readily employed to determine the set of experiments which maximizes prediction accuracy on imputed experiments, a first step in using DNNs to inform experimental design in biology.

# References


[1] Babak Alipanahi et al. "Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning". In: *Nature biotechnology* (2015).

[2] Wikimedia Commons. *DNA Macrostructure*. 2016. URL: https://commons.wikimedia.org/wiki/File:0321_DNA_Macrostructure.jpg.

[3] Wikimedia Commons. *Role of transcription factor in gene expression regulation*. 2012. URL: https://commons.wikimedia.org/wiki/File:Role_of_transcription_factor_in_gene_expression_regulation.svg.

[4] Project *relaxTheENCODEConsortium*. "An Integrated Encyclopedia of DNA Elements in the Human Genome". In: *Nature* 489.7414 (2010), pp. 57–74. DOI: http://doi.org/10.1038/nature11247.

[5] N. Gheldof et al. "Cell-type-specific long-range looping interactions identify distant regulatory elements of the CFTR gene". In: (2010). URL: https://academic.oup.com/nar/article-abstract/38/13/4325/2409392.

[6] A. Jolma et al. “DNA-dependent formation of transcription factor pairs alters their binding specificity”. In: (2015). URL: https://www.nature.com/articles/nature15518.

[7] Alon Kaufman, Martin Kupiec, and Eytan Ruppin. “Multi-knockout genetic network analysis: the Rad6 example.” In: *Proceedings. IEEE Computational Systems Bioinformatics Conference* (2001), pp. 332–340. ISSN: 1551-7497. DOI: 10.1109/CSB.2004.1332446. URL: http://www.ncbi.nlm.nih.gov/pubmed/16448026.

[8] Alon Keinan et al. “Fair attribution of functional contribution in artificial and biological networks.” In: *Neural computation* 16.9 (Sept. 2004), pp. 1887–1915. ISSN: 0899-7667. DOI: 10.1162/0899766041336387. URL: http://dx.doi.org/10.1162/0899766041336387.

[9] Yoon Kim et al. “Structured Attention Networks”. In: *CoRR* abs/1702.00887 (2017). arXiv: 1702.00887. URL: http://arxiv.org/abs/1702.00887.

[10] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *Nature* 521.7553 (2015), pp. 436–444.

[11] Yann LeCun et al. “Backpropagation applied to handwritten zip code recognition”. In: *Neural Computation* 1.4 (1989), pp. 541–551.

[12] Zachary Chase Lipton. “A Critical Review of Recurrent Neural Networks for Sequence Learning”. In: *CoRR* abs/1506.00019 (2015). arXiv: 1506.00019. URL: http://arxiv.org/abs/1506.00019.

[13] D. Raha et al. “ChIP?Seq: A method for global identification of regulatory elements in the genome”. In: (2010). URL: http://onlinelibrary.wiley.com/doi/10.1002/0471142727.mb2119s91/full.

[14] Management G. Owen?- Science and 1972. “Multilinear extensions of games”. In: (1972). URL: https://pubsonline.informs.org/doi/abs/10.1287/mnsc.18.5.64.

[15] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems 30* (2017).

[16] Jian Zhou and Olga G Troyanskaya. “Predicting effects of noncoding variants with deep learning-based sequence model”. In: *Nature methods* 12.10 (2015), pp. 931–934.