Modeling Gene Regulation by Integrating 1D and 3D Epigenomic Data with Graph Attention Networks

Alireza Karbalayghareh
Sloan Kettering Institute
karbalaa@mskcc.org

Christina Leslie
Sloan Kettering Institute
cleslie@cbio.mskcc.org

Abstract

Enhancers are distal elements that interact with each other and with promoters through DNA looping in order to regulate gene expression. Linking enhancers to their target genes – or more precisely, determining which genomic elements have a functional impact on the expression of each gene – is a critical and unsolved problem in regulatory genomics. Since enhancers can regulate target genes at a linear distance of 1Mb or more while "skipping over" nearby genes, it is unlikely that this problem can be solved using only 1D epigenomic data alone. Instead, incorporating data on 3D genomic architecture from Hi-C or HiChIP experiments may provide the way forward. Here we propose a model called Gene Regulation Model with 3D interactions (GRM-3D) that integrates both 1D epigenomic data, including chromatin accessibility and histone modifications, and 3D interaction data in order to predict the gene expression (promoter activity) at each genomic position. GRM-3D first uses convolutional neural networks (CNN) to find local representations of 1D epigenomic data and then integrates these extracted features across distal genomic regions using graphs extracted from H3K27ac HiChIP via graph attention networks to predict gene expression as measured by CAGE-seq. GRM-3D achieves better gene expression prediction compared to CNN models that do not exploit 3D data. Furthermore, feature attribution applied to GRM-3D more accurately identifies functional enhancers, as validated by CRISPRi-FlowFISH, than the recently published Activity-by-Contact model.

1 Introduction

Transcriptional gene regulation involves the collaboration of transcription factors binding both at the promoter and enhancer regions and the physical interaction of these bound complexes in 3D via DNA looping. Thanks to technological advances we now have access to genome-wide 3D interaction data, such as Hi-C [6] and HiChIP [8], in addition to traditional 1D epigenomic data, such as chromatin accessibility (DNase-seq and ATAC-seq [1]) and histone modifications (ChIP-seq and CUT&RUN [12]). So far, machine learning (ML) "regulatory models" for genome-wide prediction of expression have largely relied on 1D epigenomic data or DNA sequence [4, 11]. However, these methods only consider local features, such as promoters and at most nearby enhancers, and do not have the capability of capturing the impact of distal regulatory elements, which can be 1Mb or farther away from the promoters of the genes. Based on this insight, we believe that an effective and more reasonable ML model should take into account the 3D structure of the genome.

In this paper, we propose Gene Regulation Model with 3D interactions (GRM-3D), a model that integrates both 1D epigenomic and 3D interaction data to predict gene expression. The 1D data can include any standard epigenomic assays such as histone modification ChIP-seq, transcription factor ChIP-seq, or chromatin accessibility from DNase-seq or ATAC-seq. The 3D genome architecture data can be derived from Hi-C or HiChIP experiments. GRM-3D has two blocks: (1) the first block receives 1D data as an input and learns local representations using convolutional neural networks (CNN); (2) the second block receives a graph extracted from 3D data and the local representations from the first block and uses graph attention networks (GAT) to predict gene expression (CAGE-seq) across genomic positions (bins). We use CAGE-seq [9] since it is a tag-based protocol for measuring

Preprint. Under review.

gene expression and mapping the transcription start site (TSS), and therefore the read coverage at a TSS does not depend on the transcript length. Our motivation for proposing the GRM-3D model is two-fold. First we aim to improve gene expression prediction by leveraging 3D genomic architecture to incorporate distal enhancer elements. Second, we want to intepret the model to assess the importance of distal enhancers for regulation of specific target genes. To this end, we use feature attribution (FA) methods and show that FA of the proposed GRM-3D model can better rank the CRISPRi-FlowFISH validated enhancers of genes compared to baseline CNN models and recent enhancer-finding methods such as the Activity-by-Contact (ABC) model [3].

We define two variants of GRM-3D model: a cell-type-agnostic (CTA) and a cell-type-specific (CTS) model, depending on the type of 1D inputs. In this paper we have used a minimal set of 1D data relevant to gene regulation: DNase-seq as a measure for chromatin accessibility, H3K4me3 ChIP-seq for promoter activity, and H3K27ac ChIP-seq for enhancer activity. For 3D data, we found that H3K27ac HiChIP data provided an advantage over Hi-C since it contains more regulatory interactions, such as enhancer-enhancer (E-E) and enhancer-promoter (E-P) loops, rather than general structural (CTCF- or cohesin-mediated) interactions seen in Hi-C data. For GRM-3D/CTA, we used the three aforementioned 1D epigenomic signals – all from a given cell type – as input. We call the model cell-type-agnostic because the inputs are cell-type-specific, and thus the trained models can be applied to other cell types given the corresponding inputs. For GRM-3D/CTS, we use DNA sequence as input and take a multi-task learning approach, predicting DNase, H3K4me3, and H3K27ac in the CNN block and predicting CAGE-seq after the GAT block. We call this model cell-type-specific because it learns the transcription factor (TF) binding motifs specific to the trained cell type and consequently does not have the ability to generalize to another cell type. The aim of GRM-3D/CTS is to study the TF binding motifs in enhancer and promoter regions of the genes in a given cell type and to be able to predict the impact of nucleotide-level interventions in DNA sequence on target gene expression. We have shown in experiments that GRM-3D models in both CTA and CTS settings demonstrate superior performance than the baseline CNN models for both prediction performance and for validation of functional enhancers of genes.

2 Method

2.1 CNN Layers for Local Representations of 1D Data

The first input to our model is 1D data (epigenomic signals or DNA sequence). Regardless of the types of 1D inputs, we use several CNN layers to find local representations of 5Kb bins of the genome. We consider genomic regions of 6Mb in length as our input; hence, we have vectors of size $N = 1200$ after the CNN layers, each containing local representations of the 5Kb bins in the 6Mb region. We define the set of local representations by $H = \{h_1, h_2, \cdots, h_N\}$, where $h_i \in \mathbb{R}^F$ and $F = 64$ is the number of CNN filters at the last layer of CNN block. Then H is given to the next block, which is a graph neural network (GNN) [5, 15, 14], where h_i is the node feature of node i.

2.2 Graph Attention Layers for Integration via 3D Data

We employ the graphs extracted from H3K27ac HiChIP data in order to predict gene expression (CAGE-seq) and capture gene regulatory mechanisms. We processed the HiChIP data by HiC-DC [2], which provides significance scores (q-values) for all interactions by fitting a background model based on genomic distance and other sources of systematic bias. We have filtered the processed HiChIP contact matrix by keeping only the interactions having less than a (permissive) q-value of 0.1. As we use the 5Kb resolution of the HiChIP data, we define a graph in which there is an edge between two 5Kb genomic bins if there is a significant interaction between them. The graph corresponding to an input of 6Mb genomic region therefore has $N = 1200$ nodes whose features $H = \{h_1, h_2, \cdots, h_N\}$ have been learned in the previous CNN block.

The graph attention layer receives a graph $G = (V, E)$ and a set of node features $H^t = \{h_1^t, h_2^t, \cdots, h_N^t\}$ from the previous layer t and outputs an updated set of node features $H^{t+1} = \{h_1^{t+1}, h_2^{t+1}, \cdots, h_N^{t+1}\}$. In each GAT layer we define two weight matrices: $W_p^t \in \mathbb{R}^{F' \times F}$ for promoters (or self nodes) and $W_e^t \in \mathbb{R}^{F' \times F}$ for enhancers (or neighbor nodes). Note that here, unlike previous graph neural networks [14], we do not include self-loops in the graph G. We have decoupled self-loops and neighbor-loops because their functions are different in our problem, representing the role of the promoters and enhancers, respectively. By only including the neighbor-loops in the graph G, we aim to benefit from enhancers of genes, in particular distal enhancers that cannot be

captured in local models. We define the self-attention mechanism for the nodes i and j at layer t as $\beta_{i,j}^t = \frac{1}{|\mathcal{N}_i|}\sigma\left((a_p^t)^T W_p^t h_i^t + (a_e^t)^T W_e^t h_j^t\right)$ where $a_p^t \in \mathbb{R}^{F'}$ and $a_e^t \in \mathbb{R}^{F'}$ are two weight vectors, $\sigma(.)$ is the sigmoid function, $\mathcal{N}_i$ is the set of neighbors of node i (not including itself), $|\mathcal{N}_i|$ is the number of neighbors of node i, and $\beta_{i,j}^t \in \mathbb{R}$ is the attention weight from node j to node i at layer t. By using a sigmoid function instead of a conventional softmax function, we give extra freedom to the model to discard edges that are not related to enhancers. Therefore, here we have $\sum_{j\in\mathcal{N}_i}\beta_{i,j}^t \in [0,1]$ (as opposed to $\sum_{j\in\mathcal{N}_i}\beta_{i,j}^t = 1$ when using softmax function). We also account for the cardinality of the nodes by defining $\alpha_i^t = \sigma\left(a^t\sqrt{|\mathcal{N}_i|} + b^t\right)$, where $a^t \in \mathbb{R}^{F'}$ and $b^t \in \mathbb{R}^{F'}$ are two weight vectors. Finally, we define the updates of the node features at the next layer as $h_i^{t+1} = f\left(\alpha_i^t \circ \left(W_p^t h_i^t + \sum_{j\in\mathcal{N}_i}\beta_{i,j}^t W_e^t h_j^t\right)\right)$ where f is a nonlinearity function and $\circ$ is the element-wise product. We use an Exponential Linear Unit (ELU) for f. As in [14], we use K heads and concatenate the features as $h_i^{t+1} = \|_{k=1}^K f\left(\alpha_i^{t,k} \circ \left(W_p^{t,k} h_i^t + \sum_{j\in\mathcal{N}_i}\beta_{i,j}^{t,k} W_e^{t,k} h_j^t\right)\right)$ where $\|_{k=1}^K$ means concatenation of K independent heads. Therefore, h_i^{t+1} will have KF' features.

2.3 Poisson Regression

After the GAT layers, the last layer is a CNN layer with exponential nonlinearity in order to predict the CAGE-seq data in each bin. As the CAGE data are counts, we used Poisson regression, meaning that the expected value of each (CAGE-seq TSS) output given the inputs is the mean of a Poisson distribution. Suppose X is the 1D input for a 6Mb region, G is its corresponding graph, $Y = [y_1, \cdots, y_N]$ is the observed CAGE signal across 5Kb bins, and $f_i^\theta(X, G)$ is the predicted CAGE signal for the bin i, where θ is the parameters of the model. Now we assume that $y_i|X, G \sim \text{Poisson}(\lambda_i)$, where $\lambda_i = f_i^\theta(X, G)$ and $E(Y|X, G) = f^\theta(X, G)$. Hence, the loss function is the negative log-likelihood of the Poisson distribution $L_\theta = \frac{3}{N}\sum_{i=N/3}^{2N/3}\left(\log\Gamma(y_i + 1) + f_i^\theta(X, G) - y_i \log f_i^\theta(X, G)\right)$, where $\Gamma(.)$ is the gamma function. Note that we train our model for the middle one third region (2Mb or middle $N/3$ bins) instead of the whole 6Mb region (all N bins) in each batch; the reason is that we want to capture the effects of distal enhancers for all genes. As we have processed HiChIP data up to 2Mb, all the genes in the middle 2Mb regions can see the effects of their distal enhancers.

3 Experiments

We consider two ENCODE human cell lines, GM12878 and K562, for which complete 1D and 3D data are available. As a baseline model, we kept the first CNN layers intact and replaced the GAT layers with dilated CNN layers whose dilation rate is multiplied by two in each layer. By using 8 dilated CNN layers, we increased the model's receptive field up to 2.5Mb. Note that this technique is the best we can do for capturing distal elements without using 3D information and has also been employed in several methods like Basenji [4]. By comparing GRM-3D with this baseline CNN model, we can see how much using 3D interactions can help in gene expression prediction and also in determining functional enhancers. We used $K = 4$ and $F' = 16$ for GRM-3D models.

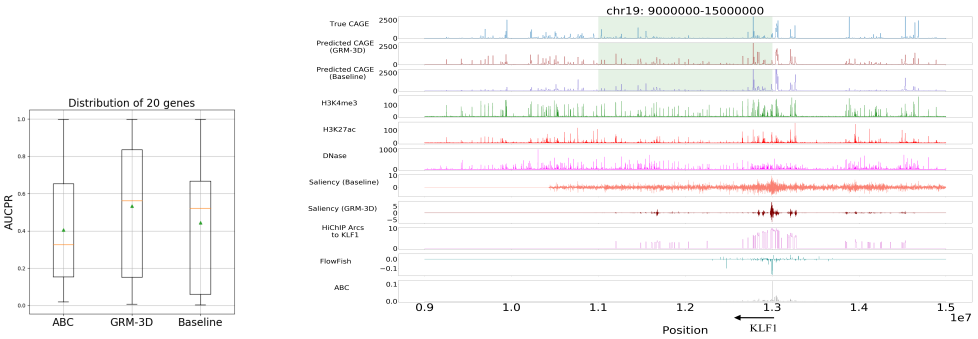
3.1 Prediction Performance

In each cell line, we held out chromosomes 3,8,12 for test and chromosomes 1,17,21 for validation and trained on all remaining chromosomes except X and Y. Although our model predicts the CAGE signal at all genomic bins, here we only looked at predictions in the GENCODE-annotated TSS bins because CAGE-seq signals, unlike RNA-seq, only appear at TSS bins. In order to look at the predictions for single genes, we restricted to predictions at bins with only a single TSS whose position is at least 500bp from the bin boundaries. Table 1 shows the prediction results for the two cell lines GM12878 and K562. We reported the loss values and Spearman correlation (SP) for three different sets of genes: $\mathcal{A}$ is the complete set of genes; $\mathcal{B}$ is the set of genes with non-zero expression ("expressed genes", based on noise in data, defined as those whose CAGE is more than 5), and $\mathcal{C}$ is the set of expressed genes having at least one neighbor in the graph.

We produced results for GRM-3D in both the CTA and CTS scenarios and their equivalent baseline models. We see in Table 1 that using GRM-3D in both cell lines leads to better loss and SP than the baseline models. We also observe that after restricting to expressed genes or to expressed genes with neighbors in the graph, which is equivalent to having distal enhancer candidate elements, the problem gets harder and the difference between GRM-3D and the baseline models increases. In the

Table 1: Prediction performance on held-out test chromosomes 3,8,12 and cell lines GM12878 and K562. The bold font shows the best result. In both frameworks, cell-type-agnostic (CTA) and cell-type-specific (CTS), GRM-3D outperforms the baseline model. $\mathcal{A}$ is the set of all genes, $\mathcal{B}$ is the set of expressed genes, and $\mathcal{C}$ is the set of expressed genes with at least one neighbor in the graph. The last column shows the log2 fold change in gene expression between GM12878 and K562. The loss is the negative log-likelihood averaged over genes, SP is Spearman correlation, R is Pearson correlation, and MSE is mean squared error.

	GM12878						K562						Log FC	
Gene sets	$\mathcal{A}: \lvert\mathcal{A}\rvert = 1868$		$\mathcal{B}: \lvert\mathcal{B}\rvert = 970$		$\mathcal{C}: \lvert\mathcal{C}\rvert = 731$		$\mathcal{A}: \lvert\mathcal{A}\rvert = 1868$		$\mathcal{B}: \lvert\mathcal{B}\rvert = 894$		$\mathcal{C}: \lvert\mathcal{C}\rvert = 737$		$\mathcal{A}: \lvert\mathcal{A}\rvert = 1868$	
Metric	Loss	SP	Loss	SP	Loss	SP	Loss	SP	Loss	SP	Loss	SP	R	MSE
GRM-3D/CTA	**142.80**	**0.8547**	**265.97**	**0.6336**	**296.50**	**0.5540**	**153.92**	**0.8528**	**312.43**	**0.6438**	**351.97**	**0.5998**	**0.7114**	**2.88**
Baseline	179.52	0.8407	331.82	0.5755	359.36	0.48039	172.43	0.8477	351.88	0.6018	396.47	0.5579	0.6800	3.309
GRM-3D/CTS	**186.55**	**0.7762**	**329.15**	**0.5402**	**369.27**	**0.4445**	**185.62**	**0.7830**	**366.32**	**0.5346**	**401.78**	**0.5003**	**0.5044**	**4.406**
Baseline	220.08	0.7351	397.54	0.49658	452.96	0.4240	234.50	0.7064	454.99	0.4702	518.77	0.4294	0.2861	6.023



(a) (b)

Figure 1: Feature attribution applied to the GRM-3D model discovers the functional enhancers of genes. (a) Distributions of AUCPR for identifying validated enhancer regions of 20 genes in K562 cells using CRISPRi-FlowFISH. GRM-3D outperforms the baseline model and the previously proposed ABC model [3] in ranking the true enhancers among candidate enhancers for genes. Green triangles show the mean values. (b) Prediction tracks of the GRM-3D and baseline models and saliency scores for the gene *KLF1*, for which the AUCPR values for GRM-3D, baseline, and ABC models are 0.8089, 0.6945, and 0.3199, respectively. Middle green-shaded region is where we do predictions. Saliency track of the baseline is noisy and non-informative, while saliency track of GRM-3D is sorting the importance of enhancers because it only attends to relevant regions determined by HiChIP graphs. HiChIP arcs here show the intensity of graph edges to the KLF1 gene.

last column of Table 1, we report the Pearson correlation (R) and mean squared error (MSE) for the true log2 fold change and the predicted log2 fold change between the cell lines GM12878 and K562 (set $\mathcal{A}$). This again confirms the better performance of the GRM-3D compared to the corresponding baseline models. Overall, these results validate our hypothesis that using 3D genomic interactions and our proposed GRM-3D models can lead to improved gene expression prediction.

3.2 Validation of Enhancers

Thanks to feature attribution methods [10, 7, 13] for ML models, it is possibe to derive the important input features for the prediction of a specific output. As we have built our model so that each gene can be influenced by its potential enhancers through connections in the H3K27ac HiChIP graph, we hypothesized that feature attribution analysis would allow us to identify distal enhancers that contribute to the regulation of target gene expression. Data on functionally validated enhancers is not abundant in the literature. CRISPRi-FlowFISH [3] is a recent method that interferes with candidate enhancer regions using KRAB-dCas9 and measures the extent of decrease in the expression level of a target gene. The developers of FlowFISH also defined a score called Activity-by-Contact (ABC) for finding and ranking enhancers of a gene and is considered the current state-of-the-art for this problem, which also uses 3D information (KR-normalized Hi-C).

We used two feature attribution methods: gradient by input in the CTS model and DeepSHAP [7] in the CTA model. Both approaches gave similar results. Figure 1a shows the distribution of AUCPR values for the 20 genes from the K562 FlowFISH data set with more than 10 candidate enhancers. There are around 2700 enhancer-gene (E-G) pairs. We observe in Figure 1a that GRM-3D feature attribution outperforms baseline models as well as ABC scores in ranking the importance of candidate enhancers. Figure 1b shows an example of saliency scores (log2-scaled) for the gene *KLF1*. For this gene there are 118 candidate enhancers, 5 of which are validated enhancers by FlowFISH (i.e. interference leads to a significant decrease in expression). For this gene, our GRM-3D/CTA achieves an AUCPR of 0.8089, while the AUCPR for the baseline model and ABC are 0.6945 and 0.3199, respectively. By looking at the saliency scores of the baseline models, all regions appear noisy, and it is hard to identify the real enhancers. However, the saliency scores of our GRM-3D models tend to have high values in the regions where there are contacts with gene promoters in the 3D interaction structure, giving the model opportunity to discover the functional enhancer regions.

References

[1] 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, 2015.

[2] Mark Carty, Lee Zamparo, Merve Sahin, Alvaro González, Raphael Pelossof, Olivier Elemento, and Christina S Leslie. An integrated model for detecting significant chromatin interactions from high-resolution hi-c data. *Nature communications*, 8(1):1–10, 2017.

[3] Charles P Fulco, Joseph Nasser, Thouis R Jones, Glen Munson, Drew T Bergman, Vidya Subramanian, Sharon R Grossman, Rockwell Anyoha, Benjamin R Doughty, Tejal A Patwardhan, et al. Activity-by-contact model of enhancer–promoter regulation from thousands of crispr perturbations. *Nature genetics*, 51(12):1664–1669, 2019.

[4] 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.

[5] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *ICLR*, 2017.

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

[7] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*, pages 4765–4774, 2017.

[8] Maxwell R Mumbach, Adam J Rubin, Ryan A Flynn, Chao Dai, Paul A Khavari, William J Greenleaf, and Howard Y Chang. Hichip: efficient and sensitive analysis of protein-directed genome architecture. *Nature methods*, 13(11):919–922, 2016.

[9] Toshiyuki Shiraki, Shinji Kondo, Shintaro Katayama, Kazunori Waki, Takeya Kasukawa, Hideya Kawaji, Rimantas Kodzius, Akira Watahiki, Mari Nakamura, Takahiro Arakawa, et al. Cap analysis gene expression for high-throughput analysis of transcriptional starting point and identification of promoter usage. *Proceedings of the National Academy of Sciences*, 100(26):15776–15781, 2003.

[10] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. *arXiv preprint arXiv:1704.02685*, 2017.

[11] Ritambhara Singh, Jack Lanchantin, Gabriel Robins, and Yanjun Qi. DeepChrome: deep-learning for predicting gene expression from histone modifications. *Bioinformatics*, 32(17):i639–i648, 08 2016.

[12] Peter J Skene and Steven Henikoff. An efficient targeted nuclease strategy for high-resolution mapping of dna binding sites. *Elife*, 6:e21856, 2017.

[13] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. *arXiv preprint arXiv:1703.01365*, 2017.

[14] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *ICLR*, 2018.

[15] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *ICLR*, 2019.