COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

Predicting RBP Sequence Specificities for Myelodysplastic Syndrome (MDS) Using the DeepBind Method

Final report for ML Functional Genomics

Elisa Bergomi, Eileen Choi, Zijie Zhu

December 20th, 2022

1 Modifications from the previous submission

From our last submission, there were major changes in our results (section 5) as we added more interpretable figures. Comparison plots of modifying architectural properties and callback functions were added and analyzed. Also, Section 4.3 was also changed as we included plots. Lastly, our next steps (section 6.2) also have been modified since we made progress on our project.

2 Abstract

Serine-rich splicing factors make up the SR family of proteins that are involved in alternative and mRNA splicing. SR proteins contain one or two RNA binding domains (RBD). More specifically, SRSF2 promotes exon recognition during splicing and dose so by binding to exonic splicing enhancer motifs in pre-mRNA through its RBD. Mutations in the SRSF2 genes are often associated with myelodysplastic syndromes (MDS), which are diseases of hematopoietic stem cells. Mutations in splicing factors represent a strong portion of driver mutations in human cancers (up to 50% of patients with myelodysplasia), and recent evidence shows that RNA-binding proteins (RBPs) are key players in the pathogenic events of MDS. Studies have brought researchers to the conclusion that MDS pathogenesis is probably caused by abnormal pre-mRNA splicing, and that the mutations are clustered in certain amino acid residues of SRSF2. We aim to apply a deep learning model based on DeepBind, a tool that analyzes how proteins bind to DNA and RNA and could be used to detect mutations to SRSF2 that could disrupt cellular processes and cause disease. We hope to understand the components of the model and improve it.

3 Introduction

Myelodysplastic syndrome is a type of myeloid neoplasm characterized by ineffective hematopoiesis, morphological dysplasia, and cytopenias, and it has a high risk of progression to acute myeloid leukemia (AML). There has been recent evidence that the post-transcriptional control of gene expression mediated by RNA Binding Proteins (RBPs) are the key components in the pathogenetic events of MDS. RBPs play a significant role in gene regulation, such as alternative splicing. Its significance in aberrant splicing in hematologic malignancies has been underlined by the comprehensive use of next-generation sequencing technologies. For MDS patients, mutations in SRSF2 genes are frequently reported (the most common SRSF2 mutants are P95H, P95L, and P95R) [4].

Accordingly, our project will try to answer the question of how to identify and predict RBP sequence specificities in Myelodysplastic Syndrome (MDS) based on deep convolutional

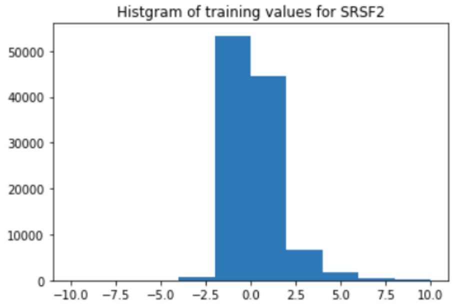


Figure 1: Histogram of the pulldown values in the training set (truncated from -10 to 10)

neural networks. To get more insights into the biological properties of MDS, the DeepBind method is applied to the sequence data of RNACompete to explore how variations affect RNA binding within a specific sequence.

4 Methods

4.1 The data

We focus on SRSF2 binding in the context of MDS, and primarily work with the RNACompete dataset. The dataset was produced by a pulldown of RNAs labeled with Cy5, a red–fluorescent label for protein and nucleic acid conjugates commonly used for imaging [2]. The preprocessing of the data involved the filtration of low-quality spots due to spatial trends or image analysis, as well as the removal of the background signal. The final data contains a batch of experiments represented with a matrix where rows correspond to probes and columns are the pulldown intensities of the RBPs. The probes are 41 bases long, and the entire RNACompete dataset contains the RBP intensities for 244 proteins, but we restrain our scope of study specifically to the data applicable to SRSF2 binding. Our sample size is 107738, and the original data values are pulldown intensities, so they are not binary values and take a range of positive and negative values1. As the first step of data processing, we set negative values to 0 and positive values to 1. As a result, the distribution of sequences that are associated with $y = 1$ is nearly 50 %.

4.2 ML methods

Our baseline method is DeepBind, and we explored different implementations of DeepBind, starting with the original DeepBind source code and data from the Nature website[1]. However, the website of the genes lab at the University of Toronto was not maintained well and has gone dark for months, as reported by other users. In addition, the original code was released in early 2015 and predates popular deep-learning frameworks such as Tensorflow and PyTorch. The researchers customized everything for their neural networks in C++, thus making their code very hard to work with.

Thanks to the suggestions of Professor Knowles, we were able to set up the Kipoi module on our Virtual Machine (VM) in the Google Cloud Platform (GCP). Conveniently, Kipoi has re-implemented all DeepBind modules in Keras and PyTorch, and since we are specifically predicting SRSF2 bindings, we work mostly with the DeepBind model trained on SRSF2 data, “DeepBind/Homo_sapiens/RBP/D00153.001_RNAcompete_SRSF2”.

4.2.1 DeepBind architecture

For any given sequence $s = (s_1, s_2, \dots, 2_n)$ where $s_i \in \{A, C, G, T, N\}$, DeepBind computes the binding score $f(s) = net_W\Bigg(pool\bigg(rect_b\Big(conv_M(s)\Big)\bigg)\Bigg)$, which represents four stages of computation, namely convolution, rectification, pooling, and neural network. More details are discussed as follows (Fig 2):

- The convolution stage trains the motif detector M_k, which is similar to a position weight matrix except its elements are not necessarily associated with probabilities (M_k is a $4 \times m$ matrix). Mathematically, the entries do not need to sum to one or be non-negative. Intuitively, this stage captures the cross-correlation between the sequence and the motif detectors.

- The rectification stage isolates positions with a good pattern match. It shifts the response of detector M_k by a value b_k, and clamps negative values to zeros (like a Relu).

- The pooling stage computes the maximum and average of each motif detector's rectified response. As the name suggests, this stage pools the previous outputs and reduces the outcome dimension. As DeepBind authors noted, max pooling performs well for DNA-binding proteins, while RNA-binding proteins benefit from both maximum and average pooling.

- The final neural network stage combines the responses to produce a final score. This stage could have no hidden layers or have one hidden layer with ReLU activation, and the setting with the higher validation performance is chosen for each new training dataset. For better generalization performance, the network nodes have a dropout probability following a Bernoulli distribution when the network is trained.

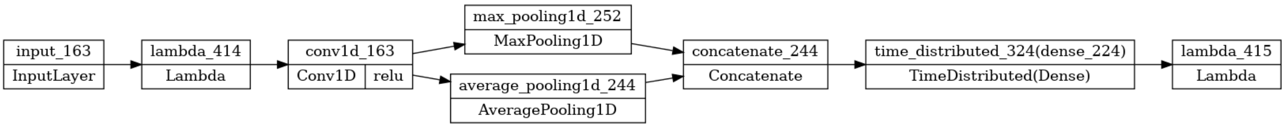
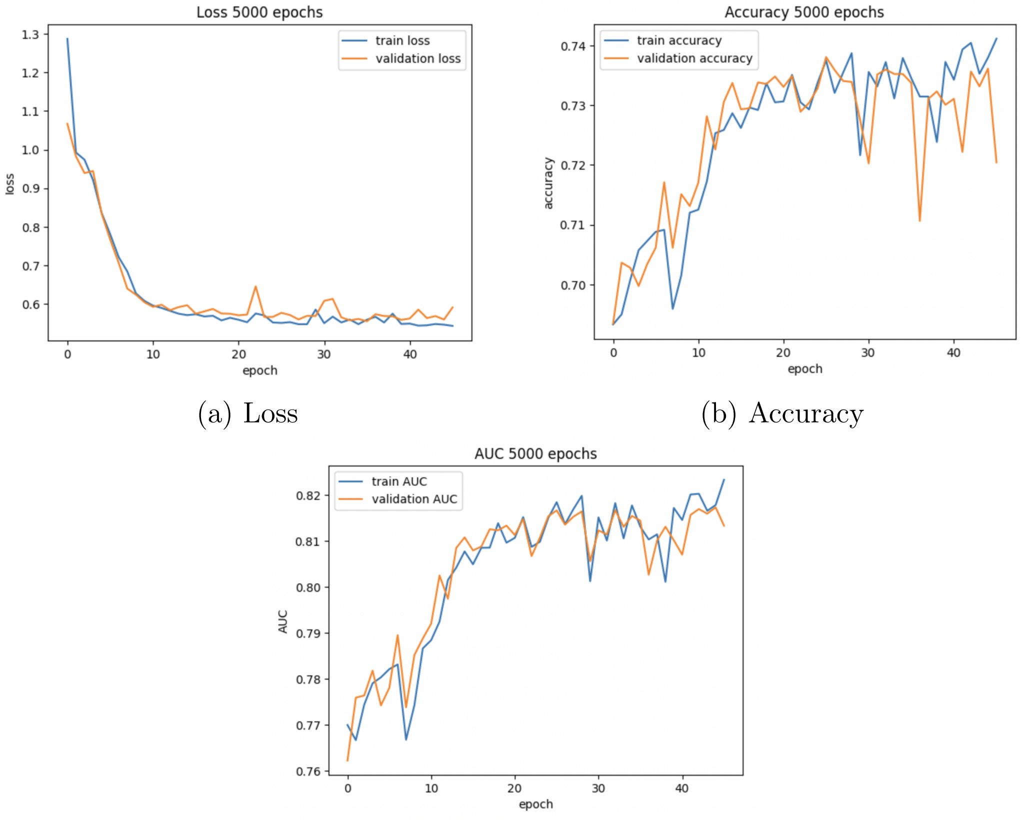


Figure 2: Visualization of an example DeepBind model (Homosapiens-RBP-SRSF2) created in Keras

The entire framework is then trained with mini-batch stochastic gradient descent. The loss function is the negative log-likelihood of mean-squared error or cross-entropy, depending on whether the response variable is continuous (e.g. PBM data) or binary (e.g. CHiP data) in the training dataset. The loss function also has $L1$ regularization penalty terms on all weights and intercepts terms in the system.

4.3 Training and tuning

We first find the motif ID for the SRSF2 protein (RNCMPT00072) and select the data associated with this motif. We then divide the data into training set, validation set, and test set. For the first round of training, we follow the authors' choice of training parameters, which are published as supplementary material to the original paper. The authors do provide certain justifications for their choice of hyperparameters, such as the batch size being 64 and the learning rate being either 0.0005 or 0.05. Next, we explore ways of improving the model



(a) Loss

(b) Accuracy

(c) AUC

Figure 3: DeepBind with Adam optimizer: Training and validation loss, accuracy, and AUC

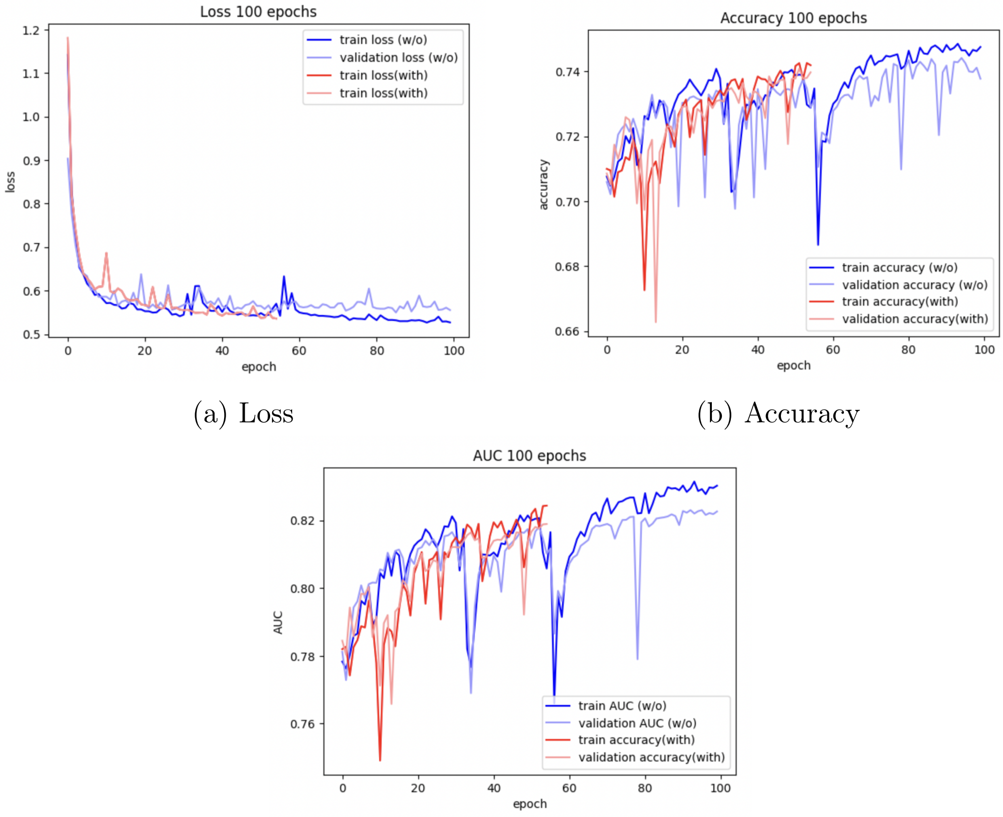
with respect to binary accuracy and Area Under the Curve (AUC), such as changing the model dimensions and adding early stopping based on validation loss. We discuss the changes and their corresponding results in the Results section below.

5 Results

We start by loading the pretrained Kipoi SRSF2 model and test it against the test set to get benchmark metrics values, such as binary accuracy and AUC. When DeepBind was published in 2015, the literature on deep learning optimizers was still rapidly growing and the authors used Stochastic Gradient Descent (SGD) with mini-batches. Therefore, our first model improvement is to re-train the model using the Adam optimizer with a similar initial learning rate of 0.0005. The results are promising and the evaluation metrics are already better than the benchmark values (*Fig.3*). In particular, our training yields an AUC of 81% (Fig 3).

5.1 Checkpoints and early stopping

After getting Kipoi SRSF2 model run, we added callback functions such as Earlystopping and ModelCheckPoints. More specifically, the model monitors the validation cross-entropy loss and terminates training if the validation loss keeps increasing for more than ten steps. The best model has been retrieved thanks to ModelCheckPoints callback continuously saving the model weights and metrics. In Figure 4, wee the effects of early stopping. The red curves terminate after the 50th epoch as the validation loss spikes up (and the accuracy and AUC plummet).



(a) Loss

(b) Accuracy

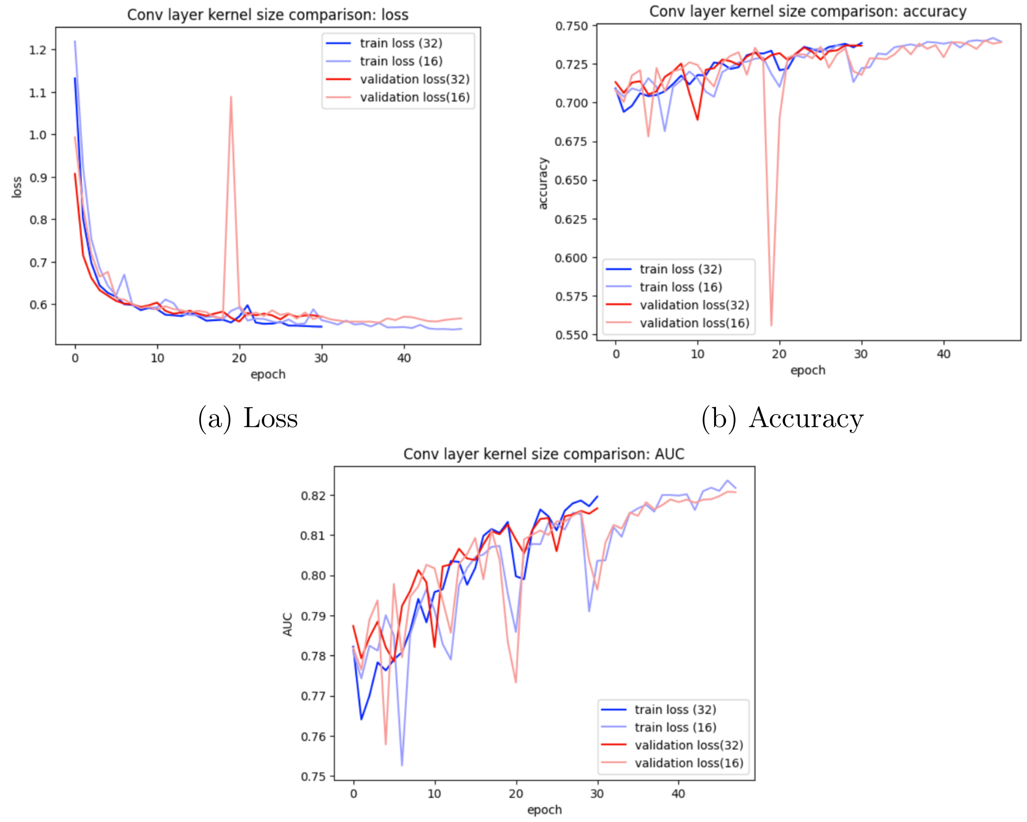
(c) AUC

Figure 4: Modifying the earlystopping: comparison of with and without

With looking ahead bias, we see the blue training loss decreases further after the early stopping point, but the validation loss stays relatively flat, indicating the early stopping point is probably reasonably chosen. However, validation accuracy and AUC in plots (b) and (c) do suggest room for further improvement had the model not earlystopped the training process.

5.2 Modifying architectural properties

We then try to understand and optimize the choice of motif detector length, which determines the number of unique positions in the motif detectors that can be learned by the DeepBind model. This parameter is present in the convolutional layer of the model as the kernel size. The original version of DeepBind uses motif length $m = 16$ due to the short length of RNA compete sequences (under 45 nt). However, we argue that constraining the motif length to 16 is somewhat arbitrary, and we aim to analyze the improvement (or degradation) of the model when modifying this parameter. Fig 5 shows the effect of changing the kernel size to 32 (versus the original 16) on the loss, accuracy, and AUC of the model. We notice that this change does not seem to have a great effect, and only slightly improves the model performance.



(a) Loss

(b) Accuracy

(c) AUC

Figure 5: Modifying the kernel size: comparison of 16 and 32

6 Discussion

6.1 Challenges

Although we initially planned on focusing on colorectal cancer, we chose to shift our focus towards Myelodysplastic Syndromes before the interim report for two reasons:

- We found more literature supporting and studying the role of RBPs in pathogenic events in MDS versus in colorectal cancer.
- The data we gathered included SRSF2, a gene commonly mutated in MDS patients, but excluded the most commonly-mutated genes for colorectal cancer.

Retrieving the data to train our model to the SRSF2 protein specifically was a challenge. Some articles which implemented DeepBind only tested it on a few RBPs, not including any SR proteins [3]. After an extensive search, we were finally able to download RNACompete data with SRSF2, after battling with various data formats. Unfortunately, we were not able to find the same data for the mutants, such as P95H, and could not evaluate the potential improvement in model robustness had we trained the model on both SRSF2 and its mutants.

Our work on interpretability plots of the model using mutagenesis and saliency maps was made difficult because of the technical restrictions of different versions of Keras. More specifically, the majority of Kipoi expects Keras 2. x with Tensorflow backend, while the Kipoi-Interpret and Kipoi-Veff modules are stale and under-maintained and still only accept Keras 1. x versions.

6.2 Next steps

We successfully improved the model by analyzing its architectural components and challenging some choices made in the original DeepBind framework (such as the kernel size of the convolutional layer).

In the context of MDS, further research could focus on applying this model to RNACompete data of the SRSF2 mutants. A current important direction in precision medicine is to use binding models to identify or visualize variants that could potentially change protein binding. To achieve this, interpretability plots would be crucial in understanding which portions of the RNA sequence are crucial for the RBP binding. Mutation maps illustrate the effect that every possible point mutation in a the sequence may have binding affinity and translate the binding importance of each base into the height of the base letter.

References

[1] Babak Alipanahi et al. "Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning". en. In: *Nature Biotechnology* 33.8 (Aug. 2015), pp. 831–838. ISSN: 1546-1696. DOI: `10.1038/nbt.3300`. URL: `https://www.nature.com/articles/nbt.3300` (visited on 10/23/2022).

[2] Debashish Ray et al. "A compendium of RNA-binding motifs for decoding gene regulation". en. In: *Nature* 499.7457 (July 2013), pp. 172–177. ISSN: 1476-4687. DOI: `10.1038/nature12311`. URL: `https://www.nature.com/articles/nature12311` (visited on 12/07/2022).

[3] Debashish Ray et al. "Rapid and systematic analysis of the RNA recognition specificities of RNA-binding proteins". eng. In: *Nature Biotechnology* 27.7 (July 2009), pp. 667–670. ISSN: 1546-1696. DOI: `10.1038/nbt.1550`.

[4] Xiaoxue Wang, Xiaomeng Song, and Xiaojing Yan. "Effect of RNA splicing machinery gene mutations on prognosis of patients with MDS". In: *Medicine* 98.21 (May 2019), e15743. ISSN: 0025-7974. DOI: `10.1097/MD.0000000000015743`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6571257/` (visited on 12/21/2022).