Neural Networks in Genome Wide Association Studies

Sarah Wessel

DS 4440 Neural Networks

Fall 2020

Abstract

In the past five years, neural networks have begun to be applied to genome wide association studies (GWAS) to study complex relationships between individual genetic variants and observable traits or diseases. Recently, deep-learning-based GWAS techniques have also been applied to the study of gene expression, which requires more advanced tools able to represent complex non-linear relationships. While many methods for this have been documented, there has been little work comparing the tradeoffs between approaches. The following pages outline the results of an effort to compare some of the more prominent approaches including DeepSEA (2015) and DeepLIFT (2020) on their ability to learn gene expression and will discuss the tradeoffs of each.

A GitHub repository for this project is available at:
https://github.com/snwessel/NeuralNetworksForGWAS

Introduction and Motivation

Genome wide association studies (GWAS) are used in genomics research to study relationships between single nucleotide polymorphisms (SNPs) and observable traits or diseases. For example, in 2010, a GWAS was used to identify 40 SNPs which are strongly associated with the severity of sickle cell disease [4]. In relatively simple cases like this, a GWAS can be performed by identifying statistically significant linear relationships. However, in recent years, researchers have begun to shift their focus to applications which have more complicated genetic relationships. Schizophrenia, which has been associated with 3,069 different variants, is a major focus of advanced GWAS methods [5]. Over the past five years, new approaches using deep learning have begun to appear in literature, and they demonstrate major improvements in the ability to study non-linear relationships.

One major focus of these new approaches is the study of noncoding or regulatory SNPs, which do not directly encode instructions for protein production. For years, noncoding DNA was believed to have no use and was often referred to as "junk DNA," but it is now believed to control how and when other DNA elements are read. Many new machine learning approaches focus on understanding and modeling these complex relationships. One prominent method called DeepSEA studies the effects of noncoding DNA by observing transcription factor binding [3]. Transcription factor is a protein which binds to DNA and controls the extent to which genes are expressed in different cells. This means that each types of cell can read different information from the same DNA. The study of these bindings allows us to better understand how individual cell types are controlled by our genes. An ongoing project called the Human Cell Atlas which aims to aggregate data about gene expression in cells, published a paper in May of 2020 describing the use of gene expression data to identify the types of cells COVID-19 is able to bind to. By identifying that ocular surface epithelium cells expressed genes associated with receptors for COVID-19, the researchers were able to predict that COVID-19 would be able to enter the body through the eye.

While recent innovations with deep learning have pushed the research space forward, there is still little consensus on which methods perform best in any given context. This may be in part because, until recently, there were few tools for processing data and visualizing results. This meant that each group of researchers needed to configure their own tools for processing genomic data, set up infrastructure for training and testing, and build their own tools for visualizing the findings of the GWAS – making it difficult to use multiple data sources or re-train existing models. However, in April of 2019, researchers at the Flatiron Institute announced the arrival of a new open-source library called Selene which was built to "enable the application of deep learning in biology" [2]. The library provides a guide for cleaning and processing data, a framework for training and evaluating PyTorch models, and tools for visualizing the results of the GWAS. Since this new tool makes it easier to implement and compare approaches, I wanted to use it to explore the tradeoffs between different methods for applying neural networks to a GWAS.

Experimental Setup

While most of the models described in recent publications do not have published code associated with them or are missing important pieces such as steps for data processing, I found one place where I could reuse existing code. In creating tutorials for the Selene library, the researchers developed one model, known as Deeper DeepSEA, as a proof of concept for their new tool. Deeper DeepSEA is a variant of the DeepSEA model, described in a 2015 publication to study the regulatory effects of noncoding DNA. The new model is trained on the same dataset as was used in the original DeepSEA publication and is fully open source.

In this project, I plan to begin with the implementation of Deeper DeepSEA and modify it to match the original implementation of DeepSEA, as is described in the original 2015 publication. I will compare the performance of these two models and will also explore other options for convolutional neural network architectures. Finally, I will implement the classifier from a 2020 publication describing the DeepLIFT method, which uses a multilayer perceptron to make predictions.

Dataset

The dataset used for this analysis comes from *The Encyclopedia of DNA Elements* (ENCODE) and is a subset of the data used to train the 2015 DeepSEA model [6]. The dataset contains information about the presence of individual SNPs (genetic variants) and information about transcription factor bindings, which regulate gene expression [6]. In the training of the following models, I will be using the presence of 4000 individual SNPs as the input features and will use values describing transcription factor binding for 1000 genes as the targets.

Training Setup

The models will be implemented in PyTorch and trained using the Selene framework. The training will use PyTorch's stochastic gradient descent optimizer and will be configured to use weight decay and momentum. In training the different methods, it is expected that the convolutional approaches will need to train for substantially longer than the multilayer perceptron. To compare the performance between models, I will be calculating the precision and the area under the receiver operating characteristics curve (AUC) using the test dataset. To observe the training process itself, I will use training loss and validation loss to watch for overfitting. I will also track the precision and AUC during training to see how it improves over time.

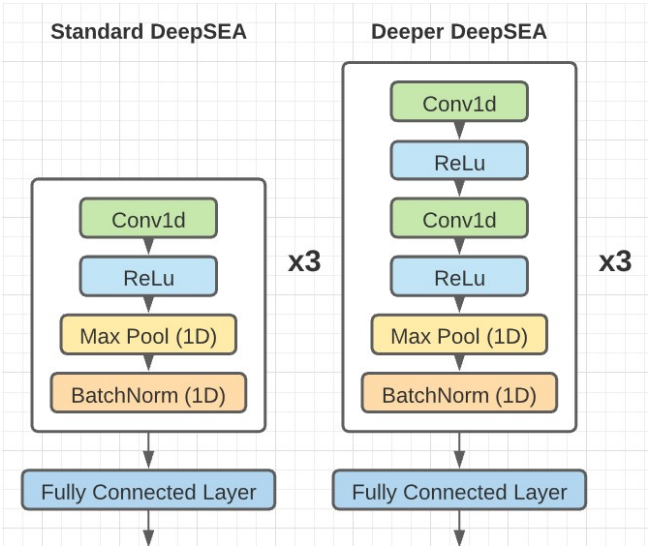
Models

Standard DeepSEA

The developers of DeepSEA used a deep convolutional neural network, alternating convolution and pooling layers to extract features. The convolutional layers are implemented using a one-dimensional convolution operation (with step size 1) that is then transformed by a ReLU function. The three convolution layers have 320, 480 and 960 kernels, respectively [3].

Deeper DeepSEA

This model is very similar to standard DeepSEA, the main difference being that it has an additional convolutional and ReLU layer between pooling layers. This results in a total of six convolutional layers (three more than standard DeepSEA). The differences between the two models are highlighted in the diagram below.



DeepLIFT

DeepLIFT was first described in 2020 and was originally used to identify diabetes risk factors [1]. In this project, I will be applying the classifier from DeepLIFT to the context of gene expression to see how its performance differs. Since the goal of learning gene expression is generally regarded to be more complex, it will be interesting to see how DeepLIFT performs on this application.

In the original 2020 publication describing DeepLIFT, the researchers began by using the training dataset to train a feedforward neural network to predict the probability of a trait (Y) given a set of SNPs (X) [1]. The researchers emphasized that the model may vary based on the context in which this approach is used and should be selected based on performance in the specific context. The model should be optimized using Mean Squared Error or Cross Entropy to calculate loss [1]. Since the original publication did not provide specifics for how the multilayer perceptron should be implemented, I tried several approaches and compared their performance in the table below. In each approach, I made a modification to either the number of hidden layers, the number of nodes per layer, the activation function, or the loss function.

Hidden Layers	Nodes / Hidden Layer	Activation Function	Loss Function	Test Precision	Test AUC
2	6000, 4000	Sigmoid	MSE	0.021	0.502
2	16000, 1000	ReLU	BCE	0.026	0.555
2	16000, 1000	ReLU	MSE	0.026	0.561
2	6000, 4000	ReLU	MSE	0.025	0.546
2	16000, 1000	Sigmoid	MSE	0.025	0.548
3	16000, 8000, 4000	Sigmoid	MSE	0.021	0.500
3	16000, 8000, 4000	ReLU	MSE	0.019	0.466
2	16000, 4000	ReLU	MSE	0.021	0.502
2	8000, 1000	ReLU	MSE	0.024	0.553

After collecting the data in the table above, I chose the best performing variant to use in comparisons with the CNN models. This was the third attempt, which had two hidden layers with 1600 and 1000 nodes respectively.

Results and Discussion

After training and testing each of the models, it appears that Deeper DeepSEA outperforms the two other models across all metrics, and the multilayer perceptron performs significantly worse than the two other models on this application. Performance metrics for each are listed in the chart below.

Model	Validation Loss	Test Precision	Test AUC
Deeper DeepSEA	0.046	0.513	0.973
DeepSEA	0.052	0.449	0.954
Multilayer Perceptron	0.098	0.221	0.542

Multilayer Perceptron Discussion

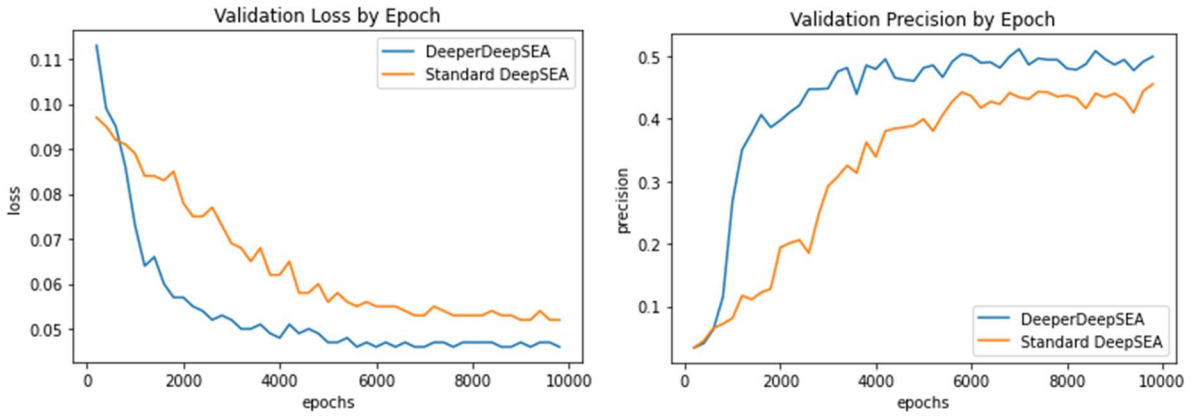
The multilayer perceptron had a significantly lower performance on this dataset than the other models. However, it is possible that in simpler contexts, like the diabetes risk context it was originally described for, it would perform better. In the DeepLIFT publication, their implementation of the MLP performed with >64% recall on the several diabetes-focused datasets from the UK Biobank. However, because the original developers of DeepLIFT did not describe the architecture they used in detail, it is difficult to know whether the poor performance here is due to the architecture itself or the complicated dataset.

In determining the architecture of the MLP, I tried a number of different configurations by adjusting the number of hidden layers, the size of the hidden layers, and the activation functions used. I was surprised to see a nonlinear activation function such as sigmoid did not lead to better performance than a ReLU activation function. However, I believe that the large

number of different architectures which were tried give an indication that a significantly better-performing MLP architecture may not exist in this context.

DeepSEA Variant Comparisons

DeepSEA and Deeper DeepSEA both performed well, with similar results to what was described in the original publications. In the original DeepSEA publication, the model performed with an AUC of 0.958, whereas my implementation had an almost identical AUC of 0.954 [3]. For Deeper DeepSEA, my training and metrics show a slightly better performance than the original. In the 2020 publication, Deeper DeepSEA was able to "achieve an average AUC of 0.934" on the full dataset, which is a superset of the dataset I used. In my own training, the model achieved an AUC of 0.973. In training, Deeper DeepSEA initially fits the data much faster than the standard DeepSEA model and levels out faster, as is shown in the graphs below.



In my analysis, I also spent time experimenting with other variants of Deeper DeepSEA. None of these variants outperformed the Deeper DeepSEA model, but they do give some insight into the tradeoffs of different architectures.

Change in Architecture	**Precision**	**Loss**
None (the original Deeper DeepSEA architecture)	0.51	0.046
Adding another set of two convolutions	0.31	0.05
Removing one convolution	0.49	0.02
Lowering the convolutional kernel size to four and the pooling kernel size to 2	0.45	0.02
Adding a third convolution to each set of convolutions	0.45	0.52

Conclusion and Future Work

Amid the many advances in the field of genome wide association studies, it is important to pause and reflect upon the available tools. In the case of studying transcription factor binding and gene expression, it appears that Deeper DeepSEA is the best performing tool which was

assessed. However, it will be important to continue to study the performance of these methods in other contexts, such as the classic application of genome wide association studies where the model's target is the presence of disease. In doing so, we may find that different architectures perform best in different contexts. It will also be important to include other prominent approaches such as the Mental-disorder Genome Score (iMEGES) and DeepWAS in future comparisons.

References

[1] D. Sharma, A. Durand, M.-A. Legault, L.-P. L. Perreault, A. Lemaon, M.-P. Dub, and J. Pineau, "Deep interpretability for GWAS," *arXiv*, Jul. 2020. arxiv.org/pdf/2007.01516v1.pdf

[2] K. M. Chen, E. M. Cofer, J. Zhou, and O. G. Troyanskaya, "Selene: a PyTorch-based deep learning library for sequence data," *Nature Methods*, vol. 16, no. 4, pp. 315–318, 2019. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7148117/

[3] J. Zhou and O. G. Troyanskaya, "Predicting effects of noncoding variants with deep learning–based sequence model," *Nature Methods*, vol. 12, no. 10, pp. 931–934, Aug. 2015. https://www-nature-com.ezproxy.neu.edu/articles/nmeth.3547.pdf

[4] P. Sebastiani, N. Solovieff, S. W. Hartley, J. N. Milton, A. Riva, D. A. Dworkis, E. Melista, E. S. Klings, M. E. Garrett, M. J. Telen, A. Ashley-Koch, C. T. Baldwin, and M. H. Steinberg, "Genetic modifiers of the severity of sickle cell anemia identified through a genome-wide association study," *American journal of hematology*, Jan-2010. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2903007/. [Accessed: 02-Nov-2020].

[5] H. L. Nicholls, C. R. John, D. S. Watson, P. B. Munroe, M. R. Barnes, and C. P. Cabrera, "Reaching the End-Game for GWAS: Machine Learning Approaches for the Prioritization of Complex Disease Loci," *Frontiers in Genetics*, vol. 11, Apr. 2020. https://www.frontiersin.org/articles/10.3389/fgene.2020.00350/full

[6] FunctionLab, "Getting Started with Selene," *GitHub*. [Online]. Available: https://github.com/FunctionLab/selene/blob/master/tutorials/getting_started_with_selene/getting_started_with_selene.ipynb. [Accessed: 12-Dec-2020].

[7] J. Collin, R. Queen, D. Zerti, B. Dorgau, M. Georgiou, I. Djidrovski, R. Hussain, J. M. Coxhead, A. Joseph, P. Rooney, S. Lisgo, F. Figueiredo, L. Armstrong, and M. Lako, "Co-expression of SARS-CoV-2 entry genes in the superficial adult human conjunctival, limbal and corneal epithelium suggests an additional route of entry via the ocular surface," *The Ocular Surface*, Jun. 2020. https://www.sciencedirect.com/science/article/pii/S1542012420300975