Towards trustworthy explanations with gradient-based attribution methods

Ethan Labelson
Partners for the Future Program
Cold Spring Harbor Laboratory
Friends Academy

Rohit Tripathy
Simons Center for Quantitative Biology
Cold Spring Harbor Laboratory

Peter Koo
Simons Center for Quantitative Biology
Cold Spring Harbor Laboratory
koo@cshl.edu

Abstract

The low interpretability of deep neural networks (DNNs) remains a key barrier to their wide-spread adoption in the sciences. Attribution methods offer a promising solution, providing feature importance scores that serve as first-order model explanations for a given input. In practice, gradient-based attribution methods, such as saliency maps, can yield noisy importance scores depending on model architecture and training procedure. Here we explore how various regularization techniques affect model explanations with saliency maps using synthetic regulatory genomic data, which allows us to quantitatively assess the efficacy of attribution maps. Strikingly, we find that generalization performance does not imply better saliency explanations; though unlike before, we do not observe a clear tradeoff. Interestingly, we find that conventional regularization strategies, when tuned appropriately, can yield high generalization and interpretability performance, similar to what can be achieved with more sophisticated techniques, such as *manifold mixup*. Our work challenges the conventional knowledge that model selection should be based on test performance; another criterion is needed to sub-select models ideally suited for downstream post hoc interpretability for scientific discovery.

1 Introduction

Deep neural networks (DNNs) have demonstrated promising predictive performance across many scientific domains, including regulatory genomics, and post hoc model interpretations often reveal that they base their decisions on meaningful patterns, such as sequence motifs [1, 2, 3, 4]. However, it is well known that model explanations with attribution methods – such as saliency maps [5], integrated gradients [6], and DeepLIFT [7] – are fragile [8, 9], making it a challenge to know whether a feature is truly important or just an artifact of the interpretability method. The question remains, how should we choose which DNN to interpret for scientific discovery? The current model selection standard is based on the model that yields the best generalization performance. Should test performance be the basis for model selection when the downstream application is model interpretability?

To yield higher quality post hoc explanations, there have been several approaches that have focused on improving the attribution method itself [10, 11] or imposing direct regularization on the attribution maps during training [12, 13]. Other approaches – such as adversarial training [14], manifold mixup [15], and randomized smoothing [16] – attempt to solve the crux of the issue by encouraging DNNs to learn more robust and smoother functions, which as a consequence, leads to better explanations

NeurIPS 2021 AI for Science Workshop.

with gradient-based attribution methods [17]. However, in many fields like computer vision, it remains a challenge to quantitatively compare the efficacy of local explanations, because there often is no pixel-level ground truth and the relevant features are complex and hierarchical (eg. edges, textures, and shapes). In genomics, it is possible to generate realistic data that are embedded with biologically meaningful patterns, called motifs, while the other positions are completely randomized – the complete ground truth can be established at a "pixel-level". Thus, analysis using regulatory genomic data is well-positioned to explore the aforementioned question in a quantitative manner.

Here, we perform a systematic comparison of different training hyperparameters – such as dropout rate, batch size, learning rate, and early stopping – to explore the interplay between a DNN's prediction accuracy and the efficacy of explanations with attribution methods, specifically saliency maps. Strikingly, we find that as training progresses, a DNN's explanation with gradient-based attribution methods can significantly be affected without noticeably influencing its generalization performance. Although we found that advanced regularization techniques, like manifold mixup, can significantly improve interpretability for DNNs that employ traditional hyperparameter choices in genomics, we also found that conventional regularization techniques, when tuned appropriately, can yield significantly higher quality model explanations while maintaining a high generalization performance. Together, our work suggests that prediction performance alone is not a reliable metric to choose which model should be used for downstream interpretability analysis, and standard regularization techniques can be effective at training models that generalize well and are highly interpretable with attribution methods.

2 Experiment overview

Dataset. To quantify the efficacy of local attribution maps, we analyzed synthetic data that recapitulates a billboard model of gene regulation from Ref. [18]. Briefly, positive sequences were embedded with 3 to 5 "core motifs" randomly selected with replacement from a pool of 5 known motifs. Negative sequences were generated in a similar way with the exception that the pool of motifs also includes a large set of "background motifs." Background sequences can thus contain core motifs; however, it is statistically unlikely for these sequences to resemble a positive class. 20,000 sequences were randomly split into train (0.7), valid (0.1), and test (0.2) sets. In this dataset, we have ground truth of which motifs were embedded in each sequence along with their locations.

Baseline model. We employed a baseline convolutional neural network (CNN) from Ref. [18], which consists of 5 convolutional layers, followed by a fully-connected hidden layer (see Appendix A for additional details). By default, dropout [19] is incorporated after each convolution with a rate of 0.1 and the dense layer with 0.5. We uniformly trained each model by minimizing the binary cross-entropy loss with mini-batch stochastic gradient descent (128 sequences) for 40 epochs with Adam updates using default parameters [20]. Each model was trained 5 times with different random initializations. Variations of the baseline model and training hyperparameters are detailed when outlining specific experiments.

Quantifying interpretability. After training, we computed a saliency map [5] for each positive-label sequence and multiplied it by the inputs, i.e. grad-times-input. We generated the distribution of saliency scores at positions where ground truth motifs were embedded and the distribution of saliency scores at other positions, as described previously [18]. We quantified the separation of these two distributions using the area under the receiver operating characteristic curve, which we call the interpretability AUROC.

3 Manifold mixup improves saliency-based explanations

It is hypothesized that the (un)reliability of gradient-based attribution methods, like saliency maps arises due to a DNN that learns a 'noisy' function, which can still yield good generalization performance but renders their gradients less reliable for model interpretability. Manifold mixup is a regularization strategy that has demonstrated an ability to converge to solutions that exhibit smoother functions [15]; to our knowledge, manifold-mixup has not been previously applied to genomics. Here, we ask the question - can manifold mixup improve local explanations from saliency maps?

Table 1: Performance comparison of manifold mixup versus standard training.

	40 Epochs		Early Stopping	
Model	Classification AUROC	Interpretability AUROC	Classification AUROC	Interpretability AUROC
Standard-1	0.975±0.001	0.703±0.008	0.975±0.002	0.712±0.009
Standard-4	0.971±0.002	0.63±0.016	0.981±0.001	0.748±0.006
Standard-8	0.971±0.000	0.609±0.004	0.983±0.001	0.754±0.004
Manifold-1	0.942±0.007	0.656±0.010	0.953±0.007	0.702±0.008
Manifold-4	0.978±0.002	0.699±0.003	0.982±0.001	0.751±0.009
Manifold-8	0.978±0.004	0.711±0.005	0.984±0.000	0.747±0.021

We trained a baseline CNN model and different scaled versions – where parameters in each convolutional and dense layers were multiplied by a factor of 1, 4 and 8 – with and without manifold mixup (applied to random hidden layers throughout training) and assessed the efficacy of local explanations given by saliency maps (Sec. 2). Interestingly, we found that manifold mixup improves the classification and interpretability AUROCs compared to a baseline CNN for larger model sizes when evaluation is performed at the end of training, i.e. 40 epochs (Table 1). For a smaller model size, the poorer performance of manifold mixup may be due to a regularization that is too strong. However, if we evaluate the model interpretability performance at the checkpoint specified by early stopping (based on maximum classification AUROC on the validation set), which occurs around 5-15 epochs, the CNNs with standard training now yield comparable performance relative to manifold mixup (Table 1). Both training approaches improve their overall interpretability performance while their classification performance is only improved marginally. This suggests that CNN models with standard training and manifold mixup largely learn similar representations early in training. Stochastic gradient descent learns functions of increased complexity [21], so it may be that earlier on, the function being learned by both methods is equally smooth. As training progresses, the stronger regularization provided by manifold mixup maintains a smoother function, while the weaker regularization of a relatively low dropout rate of 0.1 is insufficient, leading to a noisier function that degrades interpretability but largely maintains generalization performance. The slight decrease observed in manifold mixup's interpretability AUROC as training progresses suggests that, while it provides a stronger regularizer than the standard training, it is still not enough to maintain optimal performance here.

4 Better classification performance does not imply better model interpretability

To explore how imposing a stronger regularization penalty could improve both generalization and interpretability performance, we performed a systematic exploration of the baseline CNN with 3 scale factors (i.e. 1, 4, and 8) and using different combinations of conventional regularization tech-

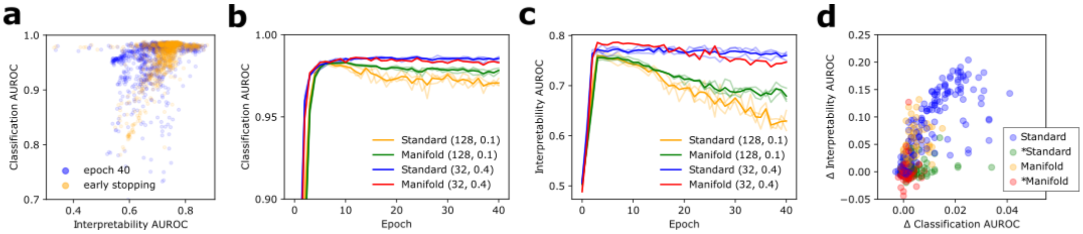

Figure 1: Performance evaluation. (**a**) Scatter plot of classification AUROC versus interpretability AUROC for different for different models trained with standard training and manifold mixup evaluated at 40 epochs (blue) and early stopping (orange). ((**b**)) classification AUROC and ((**c**)) interpretability AUROC throughout training for different models; metrics were evaluated on the test set. ((**d**)) Change in classification AUROC versus the change in interpretability AUROC evaluated at early stopping and 40 epochs for standard training and manifold mixup.

Table 2: Performance of high learning rate models evaluated at early stopping and at 40 Epochs while varying batch size (bs), dropout rate (do), learning rate (lr), and batch normalization (bn).

					Early Stopping		40 Epochs	
					Classification	Interpretability	Classification	Interpretability
Model	bs	do	lr	bn	AUROC	AUROC	AUROC	AUROC
Standard-1	32	0.4	0.01	True	0.978±0.002	0.805±0.027	0.973±0.004	0.804±0.029
Manifold-1	32	0.4	0.01	True	0.981±0.002	0.833±0.027	0.979±0.002	0.831±0.028
Standard-1	64	0.4	0.01	True	0.978±0.003	0.821±0.024	0.968±0.015	0.823±0.024
Manifold-1	64	0.4	0.01	True	0.981±0.002	0.811±0.036	0.979±0.004	0.808±0.036
Standard-1	128	0.4	0.01	True	0.978±0.003	0.821±0.023	0.972±0.007	0.816±0.027
Manifold-1	128	0.4	0.01	True	0.981±0.001	0.799±0.029	0.979±0.003	0.806±0.023
Standard-1	32	0.1	0.01	True	0.981±0.001	0.718±0.007	0.980±0.001	0.709±0.015
Standard-1	32	0.4	0.01	False	0.907±0.018	0.686±0.024	0.568±0.151	0.526±0.058
Standard-4	32	0.4	0.01	True	0.983±0.002	0.719±0.017	0.977±0.005	0.703±0.020
Standard-8	32	0.4	0.01	True	0.982±0.001	0.731±0.017	0.972±0.007	0.711±0.008
Manifold-1	32	0.1	0.01	True	0.983±0.001	0.715±0.029	0.983±0.002	0.684±0.022
Manifold-1	32	0.4	0.01	False	0.902±0.019	0.684±0.016	0.630±0.185	0.547±0.067
Manifold-4	32	0.4	0.01	True	0.987±0.001	0.668±0.025	0.987±0.001	0.677±0.017
Manifold-8	32	0.4	0.01	True	0.986±0.001	0.709±0.018	0.985±0.001	0.670±0.034

niques, i.e. batch size and dropout rates after each convolutional layer, while keeping a dropout rate of 0.5 for the final hidden dense layer (Appendix B). Strikingly, when we plot the classification AUROC versus the interpretability AUROC at early stopping, we find that models with the best classification performance do not necessarily yield the best interpretability performance. In addition, the interpretability performance is more variable when evaluating the models at the end of training (i.e. 40 epochs) without early stopping (Fig. 1a). This suggests that we should not base model selection solely on generalization performance when the downstream application is model interpretability. Importantly, while conventional wisdom suggests that there is a trade-off between these two properties, we find that models that yield high generalization performance can also yield the best interpretability performance. However, it remains unclear whether there exists a measurable quantity that correlates with interpretability performance without requiring any a prior knowledge of ground truth. Interestingly, we found that a combination of smaller batch sizes (≤ 64) and higher dropout (0.4) consistently provide better results across the 3 baseline CNNs for standard training and manifold mixup (Appendix B); typical batch sizes in genomics are relatively big (around 100) and dropout rates after convolutional layers are relatively small (0-0.2). When regularization is stronger, the generalization and interpretability performance plateaus without a significant decrease as training progresses (Fig. 1b & c), an observation that was only observed before for generalization performance [22]. Surprisingly, even a small decrease in classification AUROC as training progresses leads to a large decrease in interpretability AUROC (Fig. 1d). Thus, one way to assess which model to use for downstream model interpretability may be to assess the change in generalization performance beyond early stopping; a plateau should be observed with no significant drop-off.

5 A new training regime for interpretable models for genomics

By sampling different learning rates, we found that training with a learning rate of 0.001 and no dropout for 2 epochs followed by a higher learning rate of 0.01 and dropout of 0.4 for the remaining epochs leads to a significant improvement in the interpretability performance (Table 2), beyond what was achievable in our large scale hyperparameter search (Appendix B). We hypothesize that the stochasticity from a high learning rate enables the navigation towards a region in parameter space that was otherwise inaccessible with lower learning rates (given our limited initialization strategy), within the flat basin [23]. While previous methods have claimed observing better generalization performance within such regions [24], we only observe improved interpretability, and an ablation study shows that each hyperparameter choice is indeed synergistic. However, the change in classification performance is no longer indicative of better interpretability as was observed in Fig. 1d; nevertheless, this approach can still be informative to filter models from contention for downstream interpretability.

6 Conclusions

We performed a systematic evaluation of the generalization and interpretability performance for different model architectures and training procedures. We found models that yield the best generalization performance do not necessarily lead to the best interpretability performance – though we did not observe a strict trade-off. We could not find a universal metric for model selection that consistently leads to best model interpretability, Interestingly, we observed that standard regularization techniques are often good enough. Even still, it would be interesting to explore other regularization strategies, including adversarial training and Jacobian regularization [25], and to see how well all of these results generalize to *in vivo* data, despite the fact that evaluation remains tricky due to a lack of ground truth.

7 Code

Code for this paper is open source and can be found here:
`https://github.com/thelabelmaker/Regularization-Genomics-NeurIPS`

References

[1] 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.

[2] Alexandra Maslova, Ricardo N Ramirez, Ke Ma, Hugo Schmutz, Chendi Wang, Curtis Fox, Bernard Ng, Christophe Benoist, Sara Mostafavi, et al. Deep learning of immune cell differentiation. *Proceedings of the National Academy of Sciences*, 117(41):25655–25666, 2020.

[3] Žiga Avsec, Melanie Weilert, Avanti Shrikumar, Sabrina Krueger, Amr Alexandari, Khyati Dalal, Robin Fropf, Charles McAnany, Julien Gagneur, Anshul Kundaje, et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nature Genetics*, 53(3):354–366, 2021.

[4] Peter K Koo and Matt Ploenzke. Deep learning for inferring transcription factor binding sites. *Current Opinion in Systems Biology*, 2020.

[5] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv:1312.6034*, 2013.

[6] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017.

[7] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, pages 3145–3153. PMLR, 2017.

[8] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3681–3688, 2019.

[9] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *arXiv preprint arXiv:1810.03292*, 2018.

[10] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[11] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *arXiv:1705.07874*, 2017.

[12] Gabriel Erion, Joseph D Janizek, Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Learning explainable models using attribution priors. *arXiv e-prints*, pages arXiv–1906, 2019.

[13] Alex Tseng, Avanti Shrikumar, and Anshul Kundaje. Fourier-transform-based attribution priors improve the interpretability and stability of deep learning models for genomics. *Advances in Neural Information Processing Systems*, 33, 2020.

[14] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[15] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447. PMLR, 2019.

[16] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR, 2019.

[17] Christian Etmann, Sebastian Lunz, Peter Maass, and Carola-Bibiane Schönlieb. On the connection between adversarial robustness and saliency map interpretability. *arXiv preprint arXiv:1905.04172*, 2019.

[18] Peter K Koo and Matt Ploenzke. Improving representations of genomic sequence motifs in convolutional networks with exponential activations. *Nature Machine Intelligence*, 3(3):258–266, 2021.

[19] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.

[21] Dimitris Kalimeris, Gal Kaplun, Preetum Nakkiran, Benjamin Edelman, Tristan Yang, Boaz Barak, and Haofeng Zhang. Sgd on neural networks learns functions of increasing complexity. *Advances in Neural Information Processing Systems*, 32:3496–3506, 2019.

[22] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[23] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *arXiv preprint arXiv:1712.09913*, 2017.

[24] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

[25] Daniel Jakubovitz and Raja Giryes. Improving dnn robustness to adversarial attacks using jacobian regularization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 514–529, 2018.

A Model architecture

Our CNN is based on Ref. [18]. Briefly, it is given according to:

1. input (200 nucleotide sequence)
2. convolution (24×f filters, size 19, ReLU)
   dropout (rate = r)
3. convolution (32×f filters, size 7, ReLU)
   dropout (rate = r)
   max-pooling (size 4)
4. convolution (48×f filters, size 5, ReLU)
   max-pooling (size 4)
   dropout (rate = r)
5. convolution (64×f filters, size 5, ReLU)
   max-pooling (size 4)
   dropout (rate = r)
6. fully-connected layer (96×f units, ReLU)
   dropout (rate = 0.5)
7. fully-connected output layer (1 units, sigmoid)

where f is a factor to change the width of the network, here f = 1, 4, and 8, and r is the dropout rate that was set to 0.1 by default, unless specified otherwise.

B Hyperparameter search

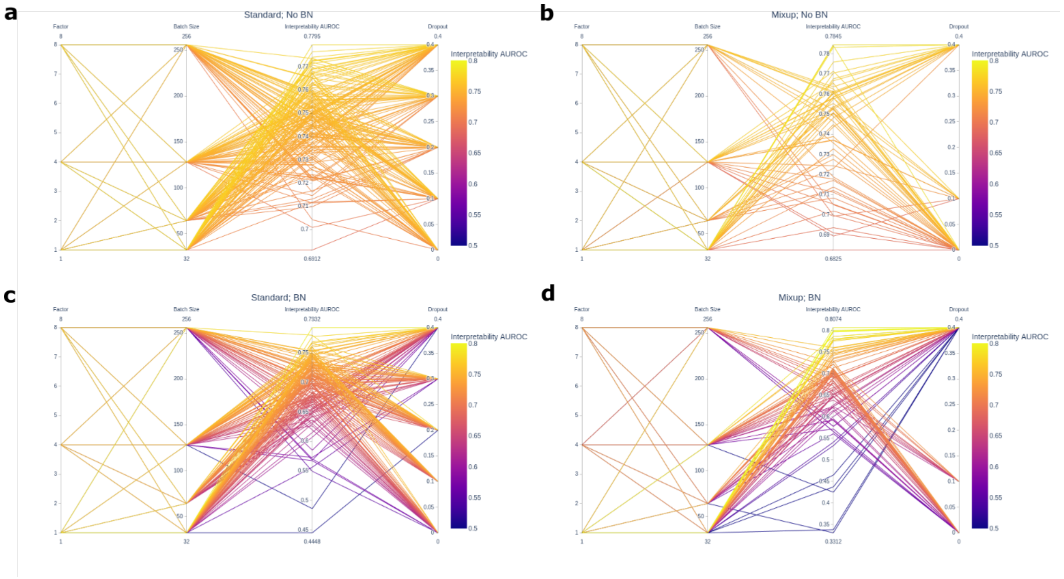

Figure 2: Hyperparameter search for the baseline CNN with (**a**) standard training, (**b**) manifold mixup, and (**c**) standard training and (**d**) manifold mixup where the models include batch normalization prior to activations in each hidden layer.