Weight-based Neural Network Interpretability using Activation Tuning and Personalized Products

Hussein Mohsen [1] **Jonathan Warrell** [1] **Martin Renqiang Min** [2] **Sahand Negahban** [3 4] **Mark Gerstein** [1 5 3 4]

Abstract

We introduce approaches to simplifying neural networks and enhancing their interpretability using activation-based neuron tuning and personalized weight matrix products. Inspired by the evolutionary principle of the survival of the fittest, we gradually remove neurons with little to no learning efficacy during training and hypothesize that their absence renders opaque models more interpretable. Experimental results pertaining to cancer and diabetes treatment appear to favor our hypothesis and generate more biomedically salient results. Our approaches also allow for interpretations at the sample level, a feature of particular importance in relation to personalized medicine.

1. Introduction

Wide applicability of neural network models is contingent on our understanding of the underlying dynamics leading to their exemplary performance. In fields like biomedicine, interpretability becomes a necessary bridge to establish trust between AI and medical scientists (Ching et al., 2018). Our goal in this paper is two-fold. First, we aim to understand how neural networks learn. Second, we aim to expand biomedical knowledge by scrutinizing the high number of parameters learned during training — primarily in the form of weight matrices. To these ends, we introduce two complementary approaches: Activation-based Neuron Tuning (ANT) to discard neurons during training, and Personalized Weight Product (PWP) to interpret the resulting network using products of data and weight paths. While each of ANT and PWP can be deployed as a standalone approach that serve different yet related tasks, we connect them through a bio-inspired hypothesis on the learning process of neural networks that renders ANT a favorable precursor to PWP.

[1]Computational Biology & Bioinformatics, Yale University [2]NEC Labs America [3]Computer Science, Yale University [4]Statistics & Data Science, Yale University [5]Molecular Biophysics & Biochemistry, Yale University. Correspondence to: Hussein Mohsen <hussein.mohsen@yale.edu>.

2. Activation-based Neural Tuning

2.1. Hypothesis

Activation-based neural tuning is inspired from biological phenomena where only a subset of entities participating in a process endure or contribute to the final outcome. Whether it is a key cellular pathway whose disruption leads to cancer after inactivating only a few genes, or the brain responding to external stimuli using a small fraction of its neurons, prioritizing biomarkers according to their contribution intensity is a recurring theme in biology. Applicability of this hypothesis on neural network training is centered around two ideas inherent to the training process. The first pertains to the stochasticity of training: networks with different weight initializations yield different learned weights but comparable overall predictive performance, suggesting that neural networks can take multiple "learning routes" to identify patterns in data. The second relates to the comparable predictive performance of networks with different architectures. In supervised learning tasks, network size often reaches a saturation limit where adding neurons does not improve performance.

Our tuning approach trims a network during training to (1) keep only enough neurons to learn target patterns and (2) restrict the "learning route" to untrimmed neurons considered significant by the virtue of receiving concentrated learning flow during training. We hypothesize that discarded neurons could be inducing noise on the learning process. By the end of training, remaining neurons are expected to resemble the "learning bottleneck" of the network, i.e. a small set of neurons that suffice for effective and less noisy learning. This perspective resembles an indirect relation to the "information bottleneck" (Tishby & Zaslavsky, 2015), and from an evolutionary biology angle, it can be seen as a model of Darwin's survival of the fittest. The measure of fitness is based on the level of a neuron's engagement during training measured through an activation function-specific proxy described below.

2.2. Neuron Selection Criteria

For weight updates to effectively navigate the loss function's (L) error surface, gradient magnitudes must take val-

ues higher than 0 or ϵ (i.e. small values pertaining to the saturation problem). To turn off neurons during training, our ANT selection criteria measure the properties of neurons' input distributions, i.e. Z's, to rank them according to the magnitude of weight updates. Neurons with inputs concentrated around activation function-specific favorable intervals are prioritized, while others distant from a concentrated target distribution (Φ_{target}) are permanently turned off. The number of neurons to be removed, n, and number of epochs at which neurons are regularly turned off, k, are both pre-defined parameters that indicate the total number of neurons eliminated from each layer by the end of training, nk.

2.3. Calculus Interpretation

In calculus terms, we define neuron activity in terms of its gradients' updates during optimization. By virtue of the chain rule used to calculate gradient values during each backward pass, overall gradients are affected by derivatives of neuron activation functions with respect to their inputs (i.e. middle term of eq. (1)).

$$\frac{\partial L}{\partial w_{ij}^{l-1}} = \frac{\partial z_j^l}{\partial w_{ij}^{l-1}} \frac{\partial a_j^l}{\partial z_j^l} \frac{\partial L}{\partial a_j^l}, \tag{1}$$

where L is the loss function, l and $l-1$ are subsequent layers, i is the source neuron index in layer $l-1$, j is the destination neuron index in layer l, a^l is the activation function in l, $a_j^l = a^l(z_j^l)$, $z_j^l = w_j^{l-1} a^{l-1} + b_l$, w_{l-1}^j is the weight vector incoming from $l-1$ to neuron j in l, and b_l is the bias term of layer l. We explain next how we accordingly select neurons to turn off based on input distributions to activation functions. We focus on the cases of ReLU and sigmoid functions and describe a rationale that generalizes to other functions for neuron selection.

2.4. Φ_{target} for Sigmoid and ReLU neurons

Derivative of the sigmoid function $\sigma'(x) \in]0, 0.25]$, with its highest values at $x \in [-3, +3]$ (Figure 1A). To encourage active updates in a layer's neurons, we select the target distribution for sigmoid to be $\Phi_{target}^{Sigmoid} \sim \mathcal{N}(0,\ 1.5)$, a distribution with high peaked-ness centered around $\mu = 0$ and $\in$ [-3, +3] (Figure 1C). Neurons receiving input distributions (Z) close to $\Phi_{target}^{Sigmoid}$ encourage non-zero and relatively large sigmoid gradient values ($\gg \epsilon$) resulting active overall neuron gradient updates during backpropagation. In contrast, input distributions furthest from $\Phi_{target}^{Sigmoid}$ lead to recurring 0 and ϵ-like gradients impeding progress during optimization. ANT uses Kullback–Leibler to measure the difference between the histograms of both distributions: Φ, and Z over validation data points.

A similar rationale is adopted to select Φ_{target} for ReLU, where Φ_{target}^{ReLU} encourages positive, larger derivatives and discourages 0-valued ones. Generally, the gradient of ReLU is either 0 or 1 depending on the input value passed during the forward pass: positive values lead to a derivative of 1, while negative values lead to a derivative of 0 (Figure 1B). We select Φ_{target}^{ReLU} to be an "inverted power law" distribution representing a considerably higher density shifted towards positive values (Fig. 1D). This same goal can drive the selection of target distributions that favor high activity regions of other activation functions.

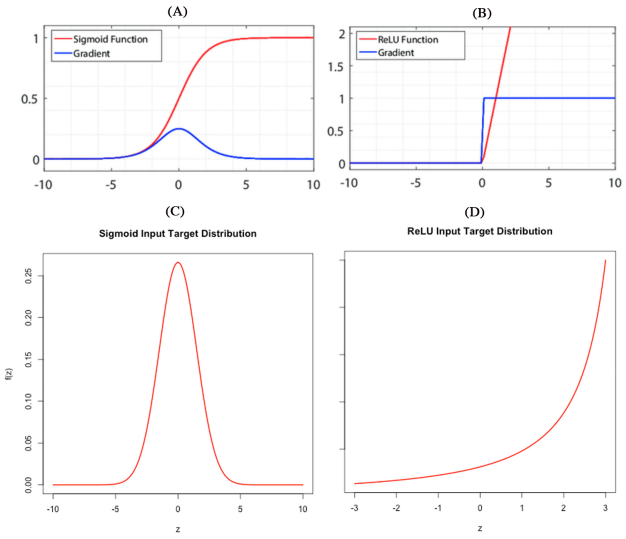

Figure 1. Sigmoid and ReLU specifications with respect to the tuning algorithm. Curve and gradient of (A) Sigmoid and (B) ReLU and target distribution for (C) Sigmoid and (D) ReLU neurons.

2.5. Algorithm

We lay out the steps of ANT in Algorithm 1.

3. Personalized Weight Product

The idea of leveraging weight matrix products to interpret trained neural networks was first introduced long before deep learning garnered its recent popularity, namely with Garson's algorithm (Garson, 1991). Recent cancer genomics research highlighted the high heterogeneity of cancer subtypes, emphasizing the need for patient- or subgroup-level treatments, a trend that falls under a set of practices that became known as "personalized medicine." (Campbell et al., 2020; Pon & Marra, 2015) Driven by this and other recent trends in biomedicine, we introduce PWP with an ability to estimate the contribution of input features to prediction on batch, subset, or individual sample levels. We

Algorithm 1 Activation-based Neuron Tuning (ANT)

Input: Data D, Network Net, Weight matrices W, Target layers T, tuning step k and number of tuned neurons n
for $epoch = 1$ **to** E epochs **do**
  SGD(Net, D, W, L)
  if epoch % k = 0 **then**
    for layer $l \in T$ **do**
      $S^l = D_{KL}(Z_i^l \| \Phi_{target}^l) \forall$ neuron $i \in l$
      $N_{tuned}^l = N_{tuned}^l \cup \underset{1..n}{\text{argmax}}\, S^l$
      Remove neurons $\in N_{tuned}^l$ from the network
    end for
  end if
end for

also leverage biomedical domain knowledge to incorporate the signs of the weights during matrix multiplication to mimic the important directionality of interactions between genes or clinical phenotypes pertaining to disease. Unlike Garson's algorithm which uses the absolute values of every weight matrix, we use absolute values only in the final step after signed matrices take part in iterative multiplication.

Let W be the set of weight matrices learned during network training. PWP's iterative weight products are calculated as follows:

$$PWP_I = X \,.\, |W_1 \,.\, W_2 \,...\, W_{N_{layers}}|, \qquad (2)$$

where I is the set of inputs and X is the dataset based on which input contributions are to be calculated.

4. Results

We evaluate ANT and PWP on MNIST and two biomedical datasets to predict drug response in acute myeloid leukemia (AML) (Tyner et al., 2018) and hospital readmission of diabetes patients (Strack et al., 2014; Goudjerkan & Jayabalan, 2019). The first set of experiments investigates the possibility of deteriorating performance caused by neuron removal by ANT. The seconds studies the performance of PWP as a standalone approach and combined with ANT. Reported results are aggregated over 10 reproducible runs.

4.1. One-Layer and Two-Layer Tuning

While the predictive performance of a trained network is not the central goal of ANT, this performance must not be sacrificed in exchange of higher levels of interpretability. To this end, we compare ANT-tuned networks with baseline models (i.e. without neuron tuning). Each baseline model constitutes 3 layers with its hyperparameters selected using `hyperopt` (Bergstra et al., 2013). We note that a slightly higher performance has been achieved on the MNIST dataset using CNNs, but we focus on the fully-connected neural nets, the target network type of our current approaches.

Results from all predictive tasks demonstrate that ANT maintains high AUC, accuracy, and precision across all three datasets while turning off up to 70% of the first hidden layer's neurons (Figure 2). Similar results are obtained when tuning two hidden layers while shrinking the model up to 50-80% (Appendix).

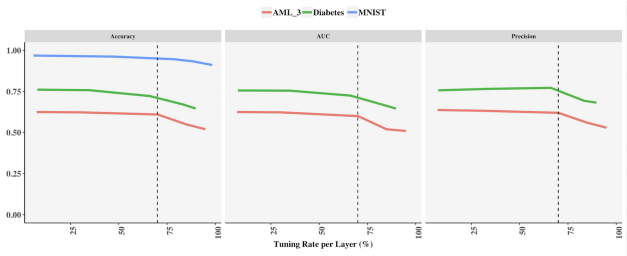

Figure 2. ANT single-layer tuning results. High predictive performance is maintained while triming up to 75% first hidden layer neurons.

4.2. Biomedical Interpretation: Cancer Genomics and Clinical Diabetes

In the first task, we perform biological enrichment analysis on the top 100 genes prioritized by PWP *vs* Garson's algorithm out of >26,000 input features encompassing gene expression and genomic variation profiles. Enrichment results returned by the DAVID databsae (Huang et al., 2007) demonstrate that PWP identifies significantly more biological entities associated with AML than Garson's: "AML" term count, number of associated publications, and statistically significant (Benjamini p-value < 0.05) chart and clustering annotation records. More interestingly, PWP applied on ANT-tuned models (labeled PWP-ANT) achieves better performance than PWP alone. PWP-ANT gene set annotation is also the only one to include AML as a directly reported GAD disease. A similar pattern is observed in the diabetes patient readmission task. PWP-ANT's top 5 features included 3 of the gold standard clinical features curated based on expansive literature review, compared to two features using PWP alone and only one feature by Garson's algorithm (Figure 3(B)). These results highlight the significance of using signed network matrices to capture interactions between features. We also note that PWP variants achieved significantly better results compared to randomly selected genes as another baseline in the AML task.

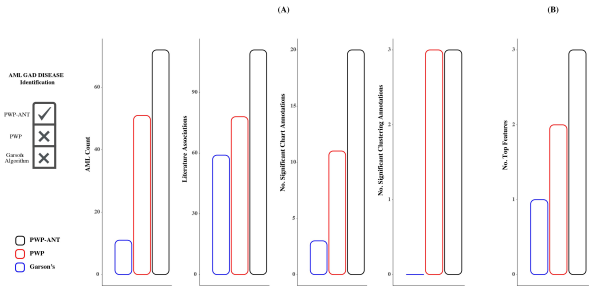

Figure 3. AML and Diabetes Results. (A) PWP-ANT's top gene list uncovers more biomedical annotations pertaining to AML than that of PWP alone or Garson's algorithm. (B) PWP-ANT prioritizes more clinically important features than both approaches.

4.3. Personalized Interpretations

A severe limitation of the Garson's algorithm's weight matrix product approach is its estimation of a singular value for each feature's contribution to the output. The data-driven nature of PWP allows it to identify prioritized features on a sample- or subset-levels of interest. To examine the potential of PWP-ANT as an attribution method, we run PWP-ANT on MNIST with three input datasets: (i) all images of all digits, (ii) all images of each digit separately, and (iii) only two images of the same digit. Prioritized pixels varied depending on the subset being considered. On dataset (i), PWP-ANT highlights pixels pertaining to specific features of multiple digits included the set. Interestingly, these pixels are located in discriminative locations that allow for the distinction between similar-looking digits such as the edges in the center of 3 and 8 or 0 and 9 (white rectangles of Figure 4(A)). On subset (ii), prioritized pixels become more specific to the target digit. Each row in Figure 4(B) highlights the same pixels prioritized to cover discriminative features of the target digit (0, 1 or 7 shown as examples). Selected pixels might also demonstrate locations where the digit of interest uniquely has no pixels. For instance, being the only digit without a single pixel in the center, these pixels were highlighted for digit 0 (center top image of 4(B), orange rectangle). Prioritized pixels become even more specific for subset (iii) as shown in 4(C). When only two images of the same digit are provided to the method, PWP-ANT uncovers the specific edge pixels of these particular images. We note that no retraining of any baseline or ANT-tuned network was required in these or other experiments, and the specificity of prioritized features is based solely on data provided to PWP as described in Eq. 2 with minimal computational overhead.

5. Future Directions

We introduce efficient approaches to simplifying neural networks and enhancing our understanding of learned parameter values. Driven by biomedical domain knowledge,

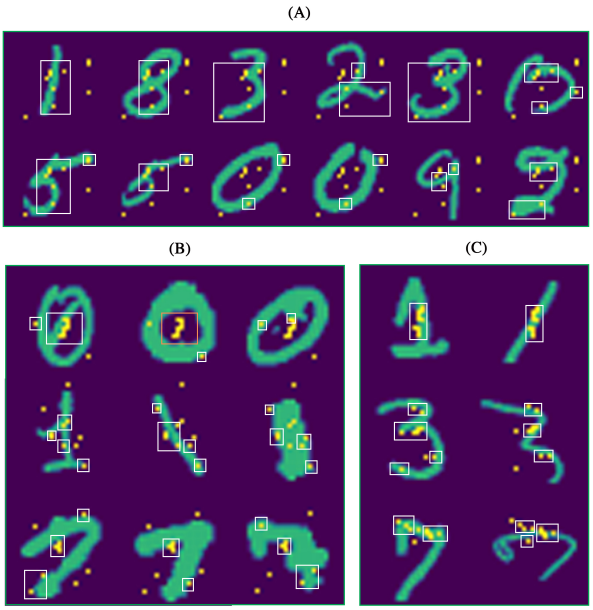

Figure 4. Representative MNIST results. Highlighted pixels prioritized by PWP-ANT capture the important discriminative features used to distinguish digits in the input in each of three scenarios: (A) all data including all digits, (B) all data of for a single digit, and (C) two data points of the same digit.

our results highlight the importance of learned weight signs and the efficacy of adopting a parsimonious perspective in training yielding smaller networks. While we demonstrate the improvement our method introduces to its closest counterpart (i.e. weight-based Garson's algorithm), experiments can be expanded in relation to related work by: (i) comparing ANT's tuning to other methods including the lottery ticket theory (Frankle & Carbin, 2018) and the work in (Morcos et al., 2018), (ii) extending PWP to detect feature interactions in line with weight-based Garson's algorithm-inspired work in (Tsang et al., 2018), or (iii) elaborating on the attributive side of PWP in comparison with other attribution methods (Springenberg et al., 2015; Shrikumar et al., 2017; Weinberger et al., 2020) that have made significant recent advances with potential opportunities for additional improvement.

References

Bergstra, J., Yamins, D., and Cox, D. D. Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. *Proceedings of the 12th Python in Science Conference (SCIPY)*, pp. 13–20, 2013.

Campbell, P. J., Getz, G., Korbel, J. O., Stuart, J. M., Jennings, J. L., Stein, L. D., Perry, M. D., of Whole Genomes Consortium, T. I. P.-C. A., et al. Pan-cancer

analysis of whole genomes. *Nature*, 578(7793):82–93, Feb 2020. ISSN 1476-4687. doi: 10.1038/s41586-020-1969-6. URL https://doi.org/10.1038/s41586-020-1969-6.

Ching, T., Himmelstein, D. S., Beaulieu-Jones, B. K., Kalinin, A. A., et al. Opportunities and obstacles for deep learning in biology and medicine. *Journal of The Royal Society Interface*, 15(141):20170387, 2018.

Frankle, J. and Carbin, M. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

Garson, G. Interpreting neural network connection weights. *AI Expert*, 6:46 – 51, 1991.

Goudjerkan, T. and Jayabalan, M. Predicting 30-day hospital readmission for diabetes patients using multilayer perceptron. *International Journal of Advanced Computer Science and Applications*, 10(2), 2019. doi: 10.14569/IJACSA.2019.0100236. URL http://dx.doi.org/10.14569/IJACSA.2019.0100236.

Huang, D. W. D., Sherman, B., et al. David bioinformatics resources: expanded annotation database and novel algorithms to better extract biology from large gene lists. *Nucleic acids research*, 35:W169–75, 08 2007. doi: 10.1093/nar/gkm415.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the International Conference on Machine Learning (ICML)*, 2000.

Morcos, A. S., Barrett, D. G. T., Rabinowitz, N. C., and Botvinick, M. On the importance of single directions for generalization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

Pon, J. R. and Marra, M. A. Driver and passenger mutations in cancer. *Annual Review of Pathology: Mechanisms of Disease*, 10(1):25–50, 2015.

Shrikumar, A., Greenside, P., and Kundaje, A. Learning important features through propagating activation differences. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. Detecting statistical interactions from neural network weights. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

Strack, B., DeShazo, J., Gennings, C., Olmo, J., Ventura, S., Krzysztof, C., and Clore, J. Impact of hba1c measurement on hospital readmission rates: Analysis of 70,000 clinical database patient records. *BioMed Research International*, 2014, 2014.

Tishby, N. and Zaslavsky, N. Deep learning and the information bottleneck principle. *CoRR*, abs/1503.02406, 2015. URL http://arxiv.org/abs/1503.02406.

Tsang, M., Cheng, D., and Liu, Y. Detecting statistical interactions from neural network weights. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

Tyner, J. W., Tognon, C. E., Bottomly, D., Wilmot, B., Kurtz, S. E., Savage, S. L., Long, N., Schultz, A. R., Traer, E., et al. Functional genomic landscape of acute myeloid leukaemia. *Nature*, 562(7728):526–531, 2018.

Weinberger, E., Janizek, J., , and Lee, S.-I. Learning deep attribution priors based on prior knowledge. In *Proceedings of Neural Information Processing Systems (NeurIPS)*, 2020.

6. Appendix

6.1. ANT Two-Layer Tuning Results

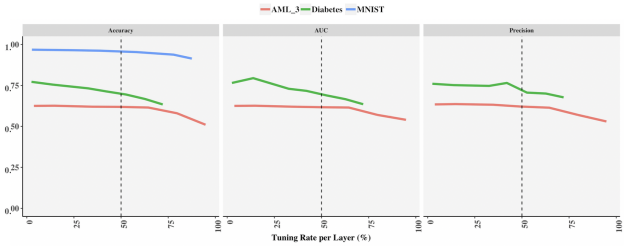

Figure 5. Two-layer tuning results. High predictive performance is maintained up to 50-80% tuning of the two hidden layers on all datasets, higher rate for selected ones.

6.2. Subfigure Credit

Subfigures (A) and (B) of Figure 1 on the curve and derivative values of Sigmoid and ReLU functions are adopted from part of Figure 3-5 in "Ranking to Learn and Learning to Rank: On the Role of Ranking in Pattern Recognition Applications" by Giorgio Roffo, *arXiv:1706.05933*.