To Prune or not to Prune: A Chaos-Causality Approach to Principled Pruning of Dense Neural Networks

Rajan Sahu$^{1[0009-0003-6201-5507]}$, Shivam Chadha$^{2[0009-0001-3921-7106]}$, Archana Mathur$^{3[2222--3333-4444-5555]}$, Nithin Nagaraj^{4}, and Snehanshu Saha$^{5,6[0000-0002-8458-604X]}$

[1] Dept. of CSIS, Birla Institute of Technology and Sciences Pilani, Pilani, India,
`f20190572@pilani.bits-pilani.ac.in`
[2] Dept of Mathematics, BITS Pilani, Goa Campus, Goa, India
`f20190704@goa.bits-pilani.ac.in`
[3] Dept. of ISE, Nitte Meenakshi Institute of Technology, India
`archana.mathur@nmit.ac.in`
[4] Consciousness Studies Programme, National Institute of Advanced Studies, Bangalore, India, `nithin@nias.res.in`
[5] Dept. of CSIS and APPCAIR, BITS Pilani, Goa, India
[6] HappyMonk AI, `snehanshus@goa.bits-pilani.ac.in`

Abstract. Reducing the size of a neural network (pruning) by removing weights without impacting its performance is an important problem for resource-constrained devices. In the past, pruning was typically accomplished by ranking or penalizing weights based on criteria like magnitude and removing low-ranked weights before retraining the remaining ones. Pruning strategies may also involve removing neurons from the network to achieve the desired reduction in network size. We formulate pruning as an optimization problem to minimize misclassifications by selecting specific weights. To accomplish this, we have introduced the concept of chaos in learning (LEs) via weight updates and exploiting causality to identify the causal weights responsible for misclassification. Such a pruned network maintains the original performance and retains feature explainability

Keywords: Granger Causality · Chaos · Lyapunov Exponent · Weight pruning.

Appendix

1 Plots on WeightWatchers run on different datasets

It is also crucial to examine the impact of our pruning technique on the training process of the model and determine if the relevance of feature importance is maintained and if the pruned network is properly trained or not. To accomplish this goal, we utilized two diagnostic tools, *namely WeightWatcher (WW) and*

SHAP. The alpha lies between 2.0 and 6.0 on every layer, however, layer 3 of the dense network and the (baseline) magnitude pruning method is not trained well (alpha < 2.0). The ESD plots of the three types of training (dense, LEGCNet-PT, LEGCNet-FT) manifest a heavy-tailed distribution of eigenvalues on each layer indicating the layers are well-trained (figure 3). A careful observation of figure 3 reveals the following: ESD plot of a layer, where the orange spike on the far right is the tell-tale clue; it's called a Correlation Trap, A Correlation Trap refers to a situation where the empirical spectral distributions (ESDs) of the actual (green) and random (red) distributions appear remarkably similar, except a small correlation shelf located just to the right of 0. In the random ESD (red), the largest eigenvalue (orange) is noticeably positioned further to the right and is separated from the majority of the ESD's bulk. This phenomenon indicates the presence of strong correlations in the layer, which can potentially affect the overall behavior and performance of the network. Layers have an overlap of random and original ones when they have not been trained properly because they look almost random, with only a little bit of information present. And the information the layer learned may even be spurious. This is the case of a well-trained layer.

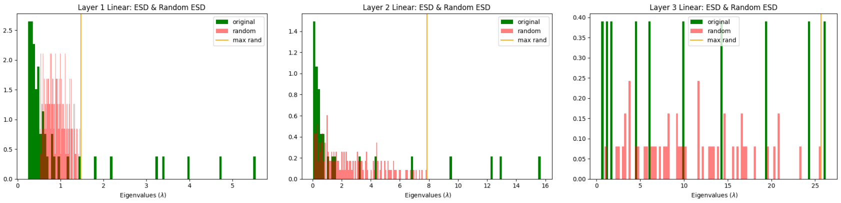

Fig. 1. WW plots for Dense network

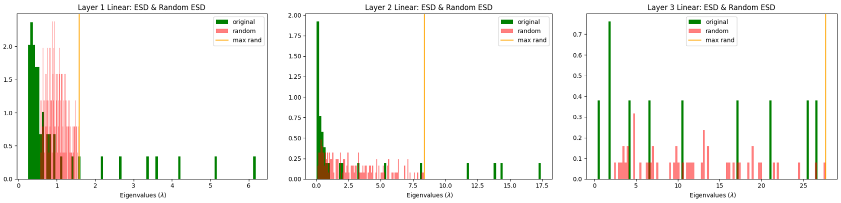

Fig. 2. WW plots for LEGCNet-FT network

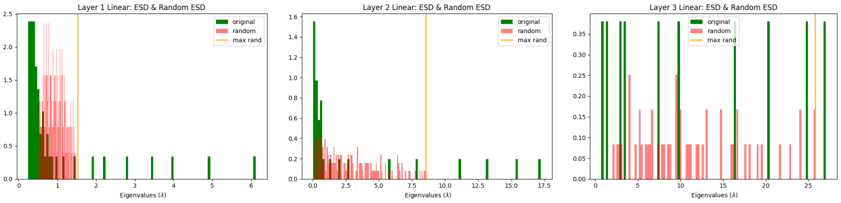

Fig. 3. WW plots for layer-wise Dense and LEGCNet-FT (figures 1 and 2) and LEGCNet-PT (figures 3) networks on MNIST data. Plots reveal the correct training of the proposed architectures.

2 Remaining WW plots for random and magnitude pruning

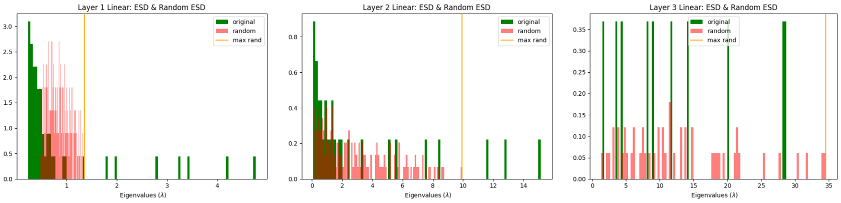

Fig. 4. WW plots for Random-Pruned network, layers 1,2 and 3

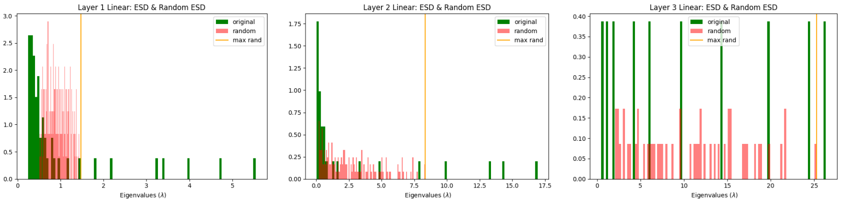

Fig. 5. WW plots for Magnitude-Pruned network, layers 1,2 and 3

3 Shap values and feature importance computed on Vowel, Banknote, and Titanic datasets for the 2 models - Random pruning Network, Magnitude-based pruning network

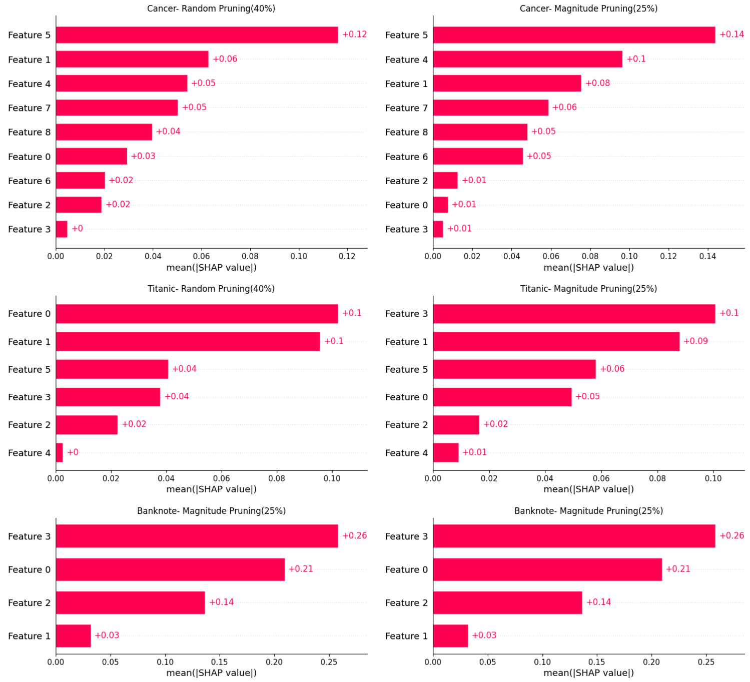

Fig. 6. Shap values and feature importance computed on Vowel, Banknote, and Titanic datasets for the 2 models - Random pruning Network, Magnitude based pruning network; the feature importance for the dense network is different from the original dense network.

4 Shap values and feature importance computed on Cancer, Banknote and Titanic datasets for all the three models - Dense Network, LEGCNet-FT and LEGCNet-PT

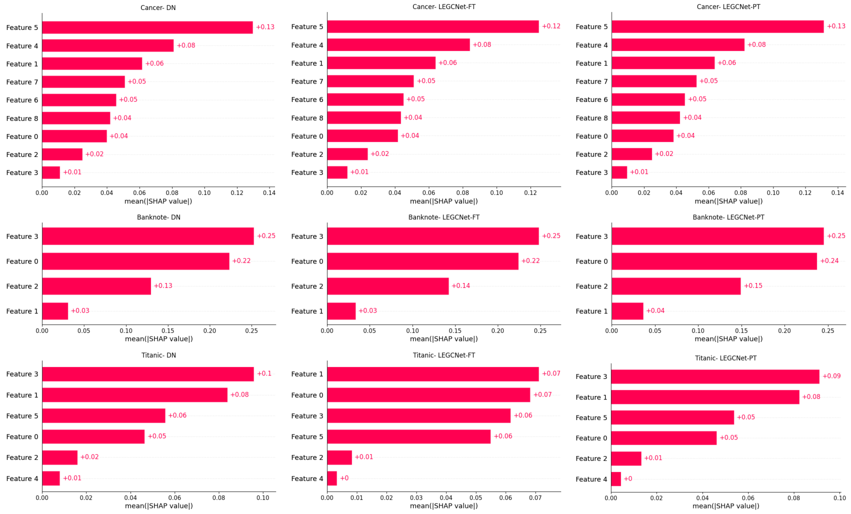

Fig. 7. Shap values and feature importance computed on Cancer, Banknote and Titanic datasets for all the three models - Dense Network, LEGCNet-FT and LEGCNet-PT; the feature importance for the dense network is same as LEGCNet-FT and LEGCNet-PT

References