PROVABLY ROBUST NEURAL NETWORKS' PROPERTIES AS FUNCTIONS OF WIDTH AND DEPTH

BENJAMIN ASCH AND PATRICK BALES

Introduction and Background

As deep learning becomes increasingly prevalent in technological sectors that are responsible for people's lives, security in deep learning becomes a forefont issue. Much of the research surrounding security has focused on the creation of adversarial examples[1] and the development of more robust networks[2][3]. (Papernot, McDaniel, et al., 2015) developed defensive distillation with the intent of creating more robust networks by reducing the number of blind-spots introduced by the non-linearity of neural networks. Initial testing was promising, but (Carlini and Wagner, 2017) managed to demonstrate that defensive distillation fails to remove the existence of adversarial examples.

(Carlini and Wagner, 2017) achieve this by introducing attack methods based on the l_0, l_2, and l_∞ distance metrics. Where they write the general problem of finding adversarial examples as,

$$\begin{aligned} &\min \mathcal{D}(x, x+\delta) + c \cdot f(x+\delta) \\ &\text{s.t. } x+\delta \in [0,1]^n \end{aligned} \tag{1}$$

, where $\mathcal{D}(\cdot)$ is our chosen distance function, $C(\cdot)$ is our classification, and $f(\cdot)$ is chosen such that $C(x+\delta) = t \iff f(x+\delta) \leq 0$. Largely in response to this finding, (Wong and Kolter, 2018) created a method of designing provably robust neural networks. They achieve this through a convex relaxation of the adversarial polytope, which is defined as,

$$\mathcal{Z}_\epsilon(x) = \{f_\theta(x+\Delta) : ||\Delta||_\infty \leq \epsilon\} \tag{2}$$

(Wong and Kolter, 2018) then pose their problem as an optimization problem:

$$\begin{aligned} &\min \vec{c}^T \hat{z}_k \\ &\text{s.t. } \hat{z}_k \in \mathcal{Z}_\epsilon(x) \end{aligned} \tag{3}$$

Here, $\vec{c} = \vec{e}_{y^*} - \vec{e}_{y^{targ}}$, where $\vec{(e)}_{y^*}$ is the unit vector with 1 in the index of the true label of x and 0 elsewhere, and $\vec{e}_{y^{targ}}$ is the unit vector with 1 in the index of the adversarial target label of x and 0 elsewhere. This problem is relaxed by forming a convex outer bound on $\mathcal{Z}_\epsilon$:

$$z_i \geq 0, z_i \geq \hat{z}_i, -u_i \hat{z}_i + (u_i - l_i) z_i \leq -u_i l_i, \quad \forall i \in [1, n] \tag{4}$$

They[2] then show that the dual to this problem is equivalent to a backward pass of the primal, and that minimizing over the dual loss is equivalent to minimizing over the worst-case loss of the primal:

$$\begin{aligned} &\max L(f_\theta(x+\Delta), y) \leq L(-\mathcal{J}_\epsilon(\vec{x}, g_\theta(\vec{e}_{y^*}\vec{1}^T - I)), \vec{y}) \\ &\text{s.t. } ||\Delta||_\infty \leq \epsilon \end{aligned} \tag{5}$$

Here, L is the loss function and J is the dual of the network. Ensuring that this loss is greater than zero provides us with a robustness certificate for the network.

Previous work, however, fails to analyze the effect of dimensionality on this lower bound and the robustness of the network using this training method.

Methodology

In this project, we wished to test and analyze the influence a Deep Neural Network's width and depth have on the network's accuracy and robustness to adversarial norm-based perturbations. To achieve this, we modified our existing three-layer network model to instead have a variable number of layers with variable width, where these dimensions were determined by some user input or defaulted to the initial model's dimensions (a width of 256 nodes in hidden layers and a depth of 3 layers). Additionally, we modified the dual bounds and dual forward functions to be able to create a convex relaxation of the adversarial polytope for d-layered networks as well as to be able to calculate a generalized dual function for when calculating loss. This dual forward function formula was obtained by using induction to generalize the three-layer dual network formula to a depth of n and is as follows:

$$d^*(\vec{x}, \vec{c}) = \max_{\vec{\nu}}(-\vec{\nu}_1^T \vec{x} - \epsilon||\vec{\nu}_1||_1 - \sum_{i=1}^{n-1} \vec{\nu}_{i+1}^T \vec{b}_i + \sum_{i=2}^{n-1} \sum_{j \in S_i} l_{ij} ReLU(\vec{\nu}_{ij})) \tag{6}$$

This formulation is supported by the findings calculating $\mathcal{J}_\epsilon$ in E. Wang and Z. Kolter's paper on using a convex expansion on the adversarial polytope to create provably robust networks, as discussed in the introduction and background section [2]. Thus, by modifying the dual bounds, dual forwards, and the network's general and forward function construction, we were able to generalize the network to work for user-inputted dimensions. Furthermore, in order to examine the efficacy of using an n-depth and m-width dual network to train a provably robust network, we trained a network using a robust training function calculating iteration loss using the loss incurred by this dual network, which was then able to be directly compared to the loss incurred by the forward primal network (and subsequently the networks' corresponding accuracies and index classifying values for the MNIST data set) to compare the two networks' results. Our findings from these modifications and subsequent tests and analyses are discussed below.

Results

Depth analysis: We first analyzed the influence of network depth on the network's accuracy classifying images from the MNIST data set adversarially perturbed by an FGSM algorithm. An example of an unperturbed image from the MNIST data set is shown in figure 1, and an example of the adversarial perturbations incurred by this image by such a method is shown in figure 2. In our analysis, we compared the accuracy of our provably robust dual network with the accuracy of the forward primal network, as well as related these to the primal accuracy a baseline network of width 256 nodes and depth 3 layers. Due to the long runtime of training a robust network – on a network of width 256 and depth 6, one training

epoch with 15000 iterations took 3 hours to compute using a Mac M1 Processor – only four network depths – 3, 4, 5, and 6 – were able to be analyzed, each of which only being trained over one epoch due to this computing time. All networks analyzed were also of width 256 to maintain consistency, and the results of these tests can be viewed in figure 3. A similar process was repeated using network widths 150, 256, 512 with a constant depth of 3, the results of which are discussed below and can be viewed in figure 4. Ultimately, we found that there exists a sharp decrease in accuracy of the robust dual network model directly correlated with increases in network depth, whereas the forward primal network experiences only a very slight, relatively unnoticeable decrease in accuracy as network depth increases (though this decrease in accuracy would get far more noticeable at much larger depths). Additionally, while it may simply be the result of trial randomness, it appears that the final measurement taken at a depth of 6 indicates that the robust network's accuracy was beginning to level out as depth increased further, though this certainly would require far more data to be gathered before coming to such a conclusion. Regardless, network depth had a major impact on the accuracy differences between the two network and loss calculation approaches, with the greatest difference occurring at the largest depth measured – 82% accuracy for the primal network versus only 39% accuracy for the dual network at 6 layers.

It is important to note, however, that both the forward primal and dual networks were trained over only one epoch of 15000 iterations. While this certainly reduces the confidence and accuracy of our networks and thus findings, training the dual network over more epochs would require a computer with more processing power than is available to us at this time. While the primal could be run with more epochs with no major difference in runtime up to a certain point, for consistency we decided to use a constant epoch and iteration number for every experiment. Thus, future experiments could train the networks over a greater number of epochs or even examine the differences in accuracy resulting from training both networks with a different epoch number. Nevertheless, the accuracy trends depicted by the dual and primal network classifiers do follow what may be expected of them, as – especially for multilevel perceptrons and other feed-forward deep neural networks – increasing the number of layers has a tendency to cause the model to over-fit data, resulting in adversarial perturbations having a much larger effect on the network's accuracy. This trend is certainly noticeable in the general decrease in accuracy rates as depth increases for both the robust loss and primal loss models. Additionally, the drastic decrease in the robust dual network's accuracy as depth increases is potentially explainable by the conservative nature of the dual classifier, which assumes an upper bound on the classifier's potential loss and forms a lower bound of the primal network. Such a conservative approach to classification means that there are far more non-adversarial examples flagged as perturbed and thus discarded by the dual network than in the case of the forward primal network; while such an approach is theoretically guaranteed to ensure robustness, the network's accuracy is certainly decreased due to the consistent flagging and thus discarding and misclassification of non-adversarial inputs – a potential tradeoff for the model. A confirmation of this theory would take more analysis on more network depths over a variety of epoch numbers and training datasets to confirm, however. It could also potentially be beneficial and insightful to look at data for networks of different widths and depths as opposed to modifying each factor one at a time.

Width analysis: FGSM-perturbed data was also classified for various networks with widths 150, 256, 512, and 750 and depths of 3 (see figure 4). We found minimal variation in classification accuracy among these networks with only a slight, and potentially statistically insignificant, increase for wider networks. This finding intuitively makes sense in connection with past work on wide-shallow networks [4]. (Simon et al., 2021) found that they were able to approximate any positive semidefinite dot-product kernel with the NTK [5] of a one-layer fully connected network. Despite the fact that their experiments with kernel regression assume infinitely-wide networks, they found that their approximations regarding the behavior of these networks remained informative down to width 50 and almost indistinguishable differences at width 500 [6]. This, in combination with our data, seems to suggest that minimal benefits are received from increasing network width beyond 500. Nevertheless, our experimentation and subsequent analysis of the relationship between dual and primal network robustness and accuracy versus hidden layer width suffered from similar issues as in our depth analysis, and thus similar future courses of action to those spoken about above – more epochs, repeated experiments, more width measurements – can be taken for strengthening this measured relationship.

Ultimately, it was found that robustness stays relatively consistent – and may even increase up to some point – with increases in network hidden layer width, whereas robustness and network accuracy has a tendency to decrease as network depth increases. Due to the dual network's upper bounding of expected loss, while the model is provably robust due to its forumlation, its overall robustness and accuracy drastically decreases as network depth increases, a reason for which we believe being one or both of (1) more depth resulting in model overfitting – which more strongly affects the dual network than the primal since the dual is formulated from an upper bound of potential network loss so will have much larger swings in accuracy as the model becomes more overfitted – and (2) the dual network flagging more non-perturbed inputs as adversarially perturbed since the larger depth results in a larger potential loss and thus a lower primal lower bound, which further results in much lower dual network acceptance. Interestingly the width of the network's hidden layers didn't appear to have too large an impact on the accuracies and robustness of either network, though this requires further analysis and experimentation to be sure of. Consequently, we conclude that it is likely best if dual network formulations are simplified to smaller depths when trying to create high-accuracy and provably robust networks, and that while greater network hidden layer widths may influence accuracy slightly, it is likely best to use a larger width up to some point – though this boundary must be determined through more experimentation and data analysis and by training the networks with more than one epoch.

Despite our findings, there is still much work to be done to receive theoretical confirmation of our empirical results. (Simon et al., 2021) work seems to suggest that single-layer FCN's achieve maximum expressivity on the set $\mathbb{S}^\infty \times \mathbb{S}^\infty$ for all positive semidefinite dot-product kernels and that subsequent non-linear layers decrease expressivity. Their approaches to this kind of work could prove fruitful in subsequent research in an attempt to develop a theoretical understanding of why these deeper networks achieved such a large robust loss. We also fail to understand the level of improvement that we will see in these networks a priori. Along a similar vein as the aformentioned research, NTKs may prove to be a powerful tool in determining the robustness of a network ahead of time. (Tsilivis and Kempe, 2023) showed

that an NTK's eigenvalues were indicative of the robustness of a feature, where they defined a feature as γ-robust:

$$\mathbb{E}_{x,y\sim\mathcal{D}}[\inf_{\delta\in\mathcal{B}} \mathbf{1}_{\text{argmax}_{i\in[k]}\phi_i(x+\delta)=y}] = \gamma \tag{7}$$

Their work showed that features corresponding to higher eigenvalues were learned first and demonstrated a decomposition of their output function, f, using an eigendecomposition of the NTK. They claim, however, that the network uses both robust and non-robust features to learn and confer it an advantage. They were unable to produce non-trivial results on multi-class outputs with a smaller, decomposed approximation. That being said, they provide no theoretical understanding of this. Using the approaches to NTK analysis in [6] to answer unanswered questions in [7] is an area ripe for future work.

Appendix

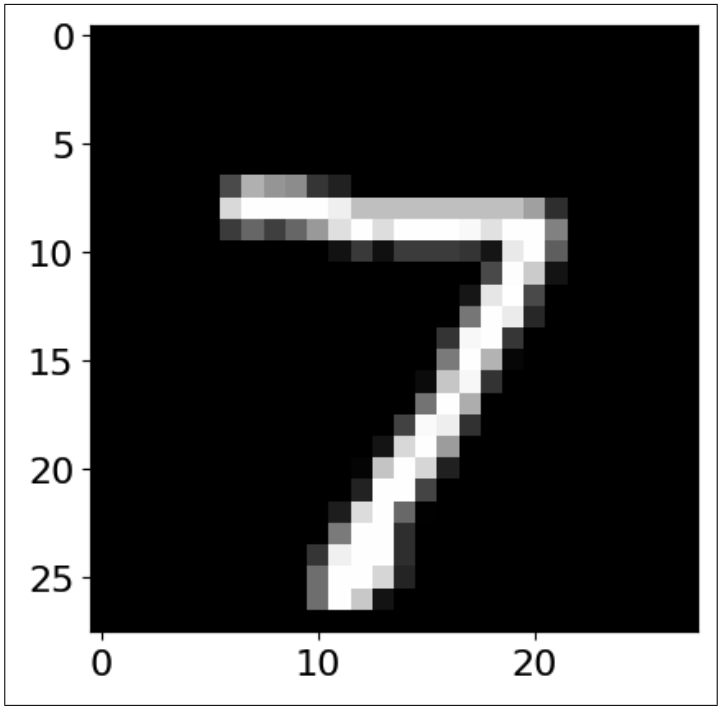

FIGURE 1. Original, Unperturbed Image

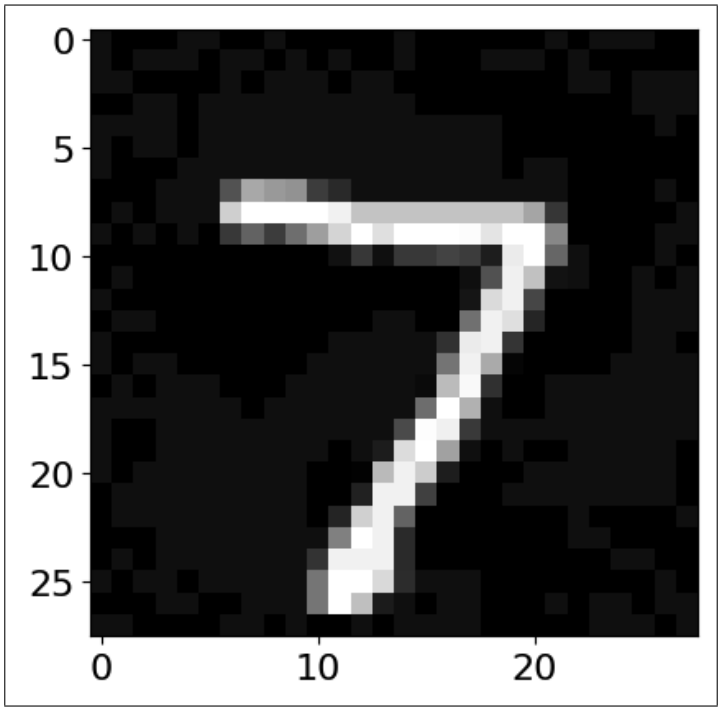

FIGURE 2. FGSM-Perturbed Image

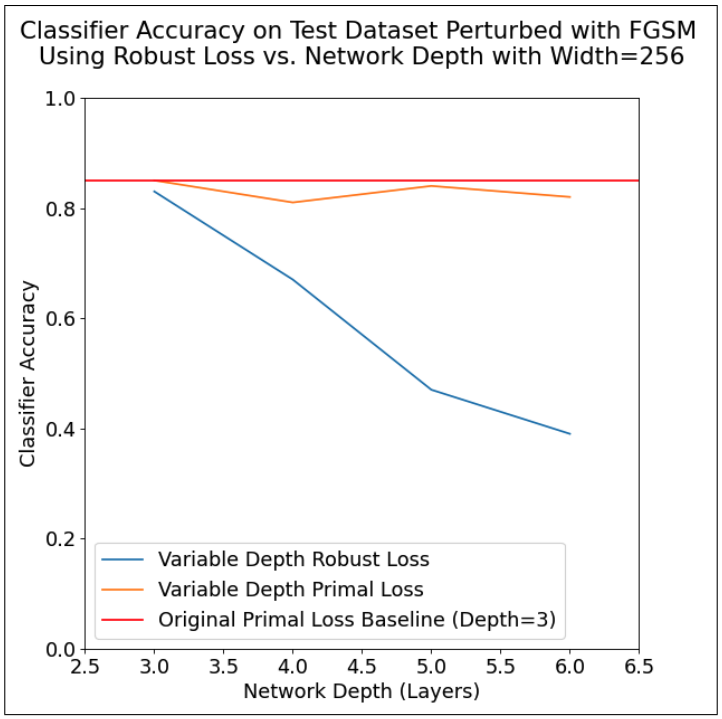

FIGURE 3. FGSM-Perturbed Image Classification Acccuracy vs. Network Depth

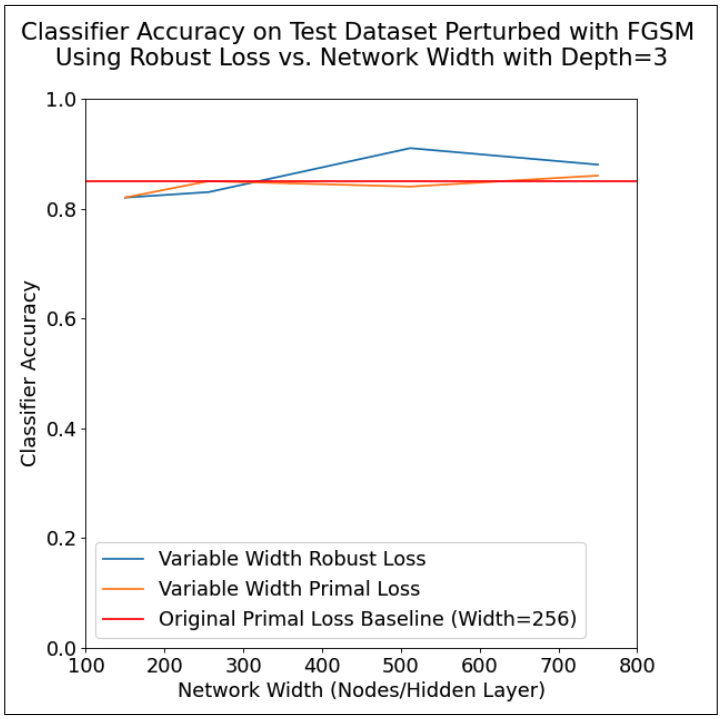

FIGURE 4. FGSM-Perturbed Image Classification Acccuracy vs. Network Width

References

[1] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," *2017 ieee symposium on security and privacy (sp)*, pp. 39–57, 2017.

[2] E. Wong and Z. Kolter, "Provable defenses against adversarial examples via the convex outer adversarial polytope," *International conference on machine learning*, pp. 5286–5295, 2018.

[3] N. Papernot, P. D. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," *CoRR*, vol. abs/1511.04508, 2015. arXiv: 1511.04508. [Online]. Available: http://arxiv.org/abs/1511.04508.

[4] J. B. Simon, S. Anand, and M. R. DeWeese, "Reverse engineering the neural tangent kernel," *CoRR*, vol. abs/2106.03186, 2021. arXiv: 2106.03186. [Online]. Available: https://arxiv.org/abs/2106.03186.

[5] A. Jacot, F. Gabriel, and C. Hongler, "Neural tangent kernel: Convergence and generalization in neural networks," *CoRR*, vol. abs/1806.07572, 2018. arXiv: 1806.07572. [Online]. Available: http://arxiv.org/abs/1806.07572.

[6] J. B. Simon, M. Dickens, and M. R. DeWeese, "Neural tangent kernel eigenvalues accurately predict generalization," *CoRR*, vol. abs/2110.03922, 2021. arXiv: 2110.03922. [Online]. Available: https://arxiv.org/abs/2110.03922.

[7] N. Tsilivis and J. Kempe, "What can the neural tangent kernel tell us about adversarial robustness?," 2023. arXiv: 2210.05577 [cs.LG].