MATH6380P Project-3: Exploring the Robustness of Neural Network

HUANG Zhichao, WEN Ruixue, LIANG Zhicong
Department of Mathematics, Hong Kong University of Science and Technology

Introduction

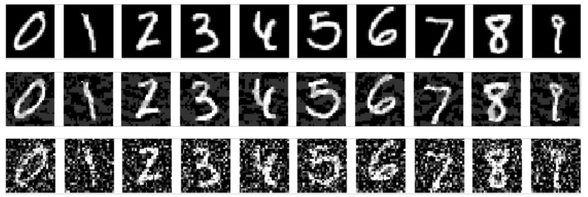
In the project, we engage to explore the robustness of several kinds of neural networks: LeNet trained with standard SGD (Undefended), Parseval LeNet with orthogonal weight matrices (Parseval), LeNet trained with gradient penalty (Gradient Penalty) and LeNet with decomposed convolutional filter (DCFNet). We use MNIST as our datasets and fast sign gradient method (FGSM) and gaussian noise as our attack methods. Our experiment shows that DCFNet give the best performance in defending FGSM attack, while Gradient Penalty outperformances others in defending Gaussian noise.

Methodology

Lipschitz constant can be constrained for each layer to guarantee the Lipchitz constant is smaller than 1 for the whole network. For activation layer like ReLU and linear layer, its Lipchitz constant is easy to constrained. For convolutional layer, Lipschitz constant can be constrained by its kernel explicitly. The l_2 norm of the layer is smaller than $\sqrt{k}\left|\left|W\right|\right|_2$ and the l_∞ norm is equal to $\left|\left|W\right|\right|_\infty$, where k is the kernel size. So we can constrained the norm of each layer directly by its weight. Parseval constrain $W^T W = I$ is one kind of constrain making the singular value of matrix all equal to 1. It can be done by Cayley transforms with update $W \leftarrow (1+\beta)W - \beta W W^T W$

Gradient penalty is most recently used in WGAN. Its aim is the same as Parseval network's: small perturbation in the input space won't significantly change the output. And its idea is intuitive. Without considering the Lipschitz constant, one need only to directly add an additional term in the loss regarding the norm of gradient with respect to the input image. Here, we refer gradient as that of the logit layer preceding the softmax layer instead of the original loss term. And l1-norm is used.

DCFNet computes weights of the neural network by decomposing each convolutional filter into a truncated expansion with pre-fixed bases in the spatial domain, where the expansion coefficients remain learned from data. When filters are approximated by Fourier-Bessel bases, the network is predicted to be more robust than its CNN counterpart due to the ignoring high-frequency information property of the FB bases.



Examples of different attacks. 1) The first row is the original image; 2) the second row is FGSM adversarial example with epsilon=0.2; 3) the third row is gaussian noise example with variance=0.2

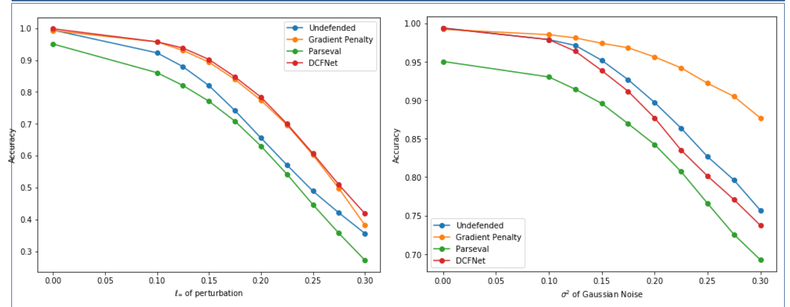
Experimental Setting

As for the training procedure, we set training epoch as 100 and batch size as 32. Fixing these hyperparameters are essential for fair comparison. Because we find that, the larger the epoch is, the more robustness the models are.

Original model trained with large epoch will be even more robustness than refined model trained with small epoch. What's more, the learning rate will halve every 20 epoches.

As for the attack procedure, we use FGSM and gaussian noise with clipping. That is, the pixel of image used to attack would be restricted to the original pixel range.

Result and Conclusion



As shown above, DCFNet is of most robustness against FGSM attack while Gradient Penalty against Gaussian Noise. This result is little bit out of our expectation that DCFNet may be more robust against high-frequency noise with its low-frequency FB bases. Parseval network and Gradient Penalty work in similar way but Gradient Penalty beats Parseval for a huge margin. In the training time, we find the loss for Parseval is very hard to go down, therefore the margin may be small and the input can be easily attacked without perturbating the output much. On the contrary, Gradient Penalty may balance the loss and the norm of gradient, which result in better resistance to noise. Actually, we also find that the network is more resistant to FGSM if it is trained for longer time, which may be caused by growing margin as the training proceeds.

Future

In the future, more attacks should be implemented to verified the robustness of these three networks. What's more, Lipschitz constant and gradient of input should be computed numerically, confirming our expected properties in corresponding networks. In addition, visualization in feature map space and input space are of our interest, too.

Links

https://github.com/HuangZhiChao95/MATH6380P-Project3

References

1. Cisse, Moustapha, et al. "Parseval networks: Improving robustness to adversarial examples." arXiv preprint arXiv:1704.08847 (2017).
2. Gulrajani, Ishaan, et al. "Improved training of wasserstein gans." Advances in Neural Information Processing Systems. 2017.
3. Qiu, Qiang, et al. "DCFNet: Deep Neural Network with Decomposed Convolutional Filters." arXiv preprint arXiv:1802.04145 (2018).