Annotated Reference

Han Xintian

December 8, 2018

1 Adversarial Examples

1.1 Adversarial Attacks

1.1.1 LBFGS Attack

Intriguing properties of neural networks [11]. Given an image x, L-BFGS tries to find a different image x' that is close to x. They solve the following constrained problem to find x':

$$\begin{aligned} \min \quad & \|x - x'\|_2^2 \\ s.t. \quad & C(x') = l \\ & x' \in [0,1]^n, \end{aligned}$$

where l is the target label. The original problem is hard to solve. They solve the following problem instead.

$$\begin{aligned} \min \quad & c \cdot \|x - x'\|_2^2 + \text{loss}_{F,l}(x') \\ s.t. \quad & x' \in [0,1]^n, \end{aligned}$$

where loss function $\text{loss}_{F,l}$ is a function that maps an image to a positive label l, for example, cross-entropy loss. It aims to perform targeted attack.

1.1.2 Fast Gradient Sign Method (FSGM)

Explaining and Harnessing Adversarial Examples [4]. FSGM is a fast algorithm. It will not produce very close adversarial examples. Given an image x, FSGM sets

$$x' = x - \varepsilon \cdot \text{sign}(\nabla \text{loss}_{F,t}(x)),$$

where ε is chosen to be sufficiently small so as to be undetectable, and t is the target label. An improved version is Iterative Gradient Sign. Begin by setting $x'_0 = x$, we have

$$x'_i = Clip_{x,\varepsilon}\{x'_{i-1} + \alpha \text{sign}(\nabla_{F,t}(x'_{i-1}))\}$$

It is an untargeted attack.

1.1.3 JSMA

The limitations of deep learning in adversarial settings [7]. JSMA is short for Jacobian-based Saliency Map Attack. It is still a gradient based attack. They use the gradient $\nabla Z(x)_l$ to compute a saliency map, which models the impact each pixel has on the resulting classification. They choose selected number of pixels changing which will make the target class more likely and other classes less likely.

1.1.4 Deepfool

Deepfool: a simple and accurate method to fool deep neural networks [5]. They image the neural network are generally linear with a hyperplane separating each class from another. They create adversarial examples for this simplified problem and use these examples to attack neural network. They use the adversarial examples to attack another classifier.

1.1.5 Carlini's l_0, l_2 and l_∞ Attack

Defensive distillation is not robust to adversarial examples [1]. The Carlini's attack mainly solve the problem:

$$
\begin{aligned}
\min \quad & \mathscr{D}(x, x+\delta) \\
s.t. \quad & C(x+\delta) = t \\
& x+\delta \in [0,1]^n
\end{aligned}
$$

Here, $\mathscr{D}$ could be l_0, l_2 or l_∞. They define the objective function f such that $C(x+\delta) = t$ if and only if $f(x+\delta) \leq 0$. When they find f, they tries to minimize

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

LBFGS is a special case of this method.

1.2 Defense Methods

1.2.1 Distillation

Distillation as a defense to adversarial perturbations against deep neural networks [8]. Introduce a method that raises the temperature in training to T in the $\exp(X/T)$ and set $T = 1$ when testing. It is a method to hide the gradient and could be attacked by [1].

1.2.2 Adversarial Training

Towards Deep Learning Models Resistant to Adversarial Attacks [5]. Defense method over all first order attack? Adversarial training over projected gradient method. Adversarial training over projected gradient method will also be robust to other first order attack method. None first order attack are hard to reach by a first order method even restarting randomly.

Other findings: 1. Large capacity network is more robust. The loss value decreases after adversarial training.

1.2.3 Wasserstein Adversarial Training

Certifying Some Distributional Robustness with Principled Adversarial Training [10]. Develops theoretical bounds for perturbation in the Wasserstein neighborhood and find minimax classifier using Augmented Lagrangian.

1.3 Other

1.3.1 Sensitivity and generalization in neural networks [6]

Empirically show that smaller generation gap corresponds to lower sensitivity. Use frobenius norm of the Jacobian and number of transitions (curvature of the functions) to characterize sensitivity. Create close to manifold datasets and off manifolds datasets. Close to manifolds datasets by combination of digits from the same class. Off manifolds by random inputs and combination of digits from different class.

1.3.2 Adversarially Robust Generalization Requires More Data [9]

The paper explains lower bounds on the minimax objective function for adversarial adversarial examples (minimum over all the parameters and maximum over all the perturbations) for Gaussian distribution and Bernoulli distribution data. They find that for Gaussian distributed data, all the learning algorithms have a constant minimax loss lower bound but for the Bernoulli distributed dataset only linear algorithms have a lower bound. The lower bound is suitable for nonlinear algorithms. They believe MNIST is like Bernoulli distribution dataset so it does not have a lower bound when using nonliner classifiers like neural network.

1.3.3 A Spectral View of Adversarially Robust Features [3]

They demonstrate the importance of learning adversarial robust features from spectral graph theory. They build a graph on distances between two images and use the second eigenvector of the graph Laplacian matrix to be a robust feature.

1.3.4 Detecting Adversarial Samples from Artifacts [2]

We investigate model confidence on adversarial samples by looking at Bayesian uncertainty estimates, available in dropout neural networks, and by performing density estimation in the subspace of deep features learned by the model.

2 Explanation

References

[1] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. *arXiv preprint arXiv:1608.04644*, 2016.

[2] Reuben Feinman, Ryan R Curtin, Saurabh Shintre, and Andrew B Gardner. Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*, 2017.

[3] Shivam Garg, Vatsal Sharan, Brian Zhang, and Gregory Valiant. A spectral view of adversarially robust features. In *Advances in Neural Information Processing Systems*, pages 10158–10168, 2018.

[4] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[5] 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.

[6] Roman Novak, Yasaman Bahri, Daniel A Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Sensitivity and generalization in neural networks: an empirical study. *arXiv preprint arXiv:1802.08760*, 2018.

[7] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, pages 372–387. IEEE, 2016.

[8] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597. IEEE, 2016.

[9] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Mądry. Adversarially robust generalization requires more data. *arXiv preprint arXiv:1804.11285*, 2018.

[10] Aman Sinha, Hongseok Namkoong, and John Duchi. Certifying some distributional robustness with principled adversarial training. 2018.

[11] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.