Adversarial Robustness

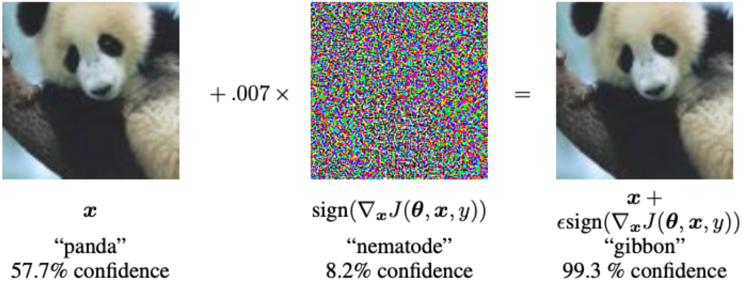
Alicia Tsai

aliciatsai@berkeley.edu

September 2019

Adversarial Examples

- An adversarial example $x^* = x + \delta$ is constructed from a benign sample x by adding a perturbation vector δ under an allowable perturbation region Δ.



Adversarial Examples

- The adversary wants to find such a perturbation that maximizes the loss function.
- The perturbation can be found by solving an optimization problem

$$\max_{\delta} \quad L(h_\theta(x+\delta), y) \tag{1}$$

$$\text{s.t.} \quad \delta \in \Delta \tag{2}$$

where $x^* = x + \delta$ is an adversarial example, L is the loss function, h_θ is the hypothesis function, and y is the label.

Worst-case Loss

- We refer this to the worst-case loss; hence, the expected worst-case loss for the entire data set D is

$$\frac{1}{|D|} \sum_{(x,y)\in D} \max_{\delta\in\Delta} L(h_\theta(x+\delta), y) \tag{3}$$

where $|D|$ is the total number of the data point.

Robust Classifier

- We want to train a classifier that is robust under the aforementioned worst-case scenario.
- The training task can be formulated as the following *min-max* optimization problem.

$$\min_{\theta} \frac{1}{|D|} \sum_{(x,y)\in D} \max_{\delta\in\Delta} L(h_{\theta}(x+\delta), y) \tag{4}$$

Attack and Defense

- Given the *min-max* framework,
  - any attack method can be viewed as approximately solving the inner maximization problem and
  - any defense is approximately solving the outer minimization problem.

Training a Robust Classifier

- Given the above formulation, we can train a robust classifier by
  1. Solve the inner maximization problem for each pair of (x, y) in the training set D

$$\delta^*_{(x,y)} = \arg\max_{\delta \in \Delta} L(h_\theta(x + \delta), y) \tag{5}$$

  2. Update model parameters θ by gradient descent

$$\theta := \theta - \frac{\alpha}{|D|} \sum_{(x,y) \in D} \nabla_\theta L\big(h_\theta(x + \delta^*_{(x,y)}), y\big) \tag{6}$$

  where α is the step size.

Training a Robust Classifier

- We typically cannot solve the inner maximization since it's usually non-convex.
- The community has found that if the inner maximization problem is solved "well enough", then this strategy can perform well[1].
- If we cannot solve it exactly, then we can either
  - lower bound it
  - upper bound it

[1] Aleksander Madry et al. *Towards Deep Learning Models Resistant to Adversarial Attacks*. 2017. eprint: arXiv:1706.06083.

Lower Bounding The Inner Maximization

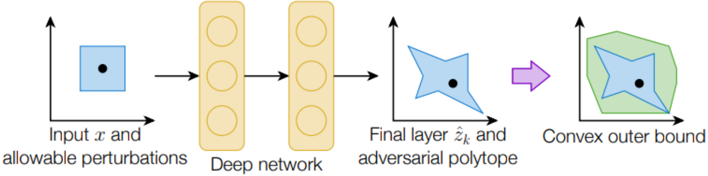
Adversarial Attack

- Any feasible δ gives us a lower bound on the inner objective value. This is equivalent to constructing an adversarial example.
- One simple way to find a feasible δ is by performing (projected) gradient ascent on δ to maximize the inner objective function.

Upper Bounding The Inner Maximization

Convex Relaxation

- For a typical multi-layer neural network, the inner maximization problem is non-convex.
- We can construct a *convex outer bound* on this non-convex adversarial polytope[2].



[2] J. Zico Kolter and Eric Wong. "Provable defenses against adversarial examples via the convex outer adversarial polytope". In: *CoRR* abs/1711.00851 (2017). arXiv: 1711.00851. URL: http://arxiv.org/abs/1711.00851.

Upper Bounding The Inner Maximization

Interval Bound Propagation (IBP)

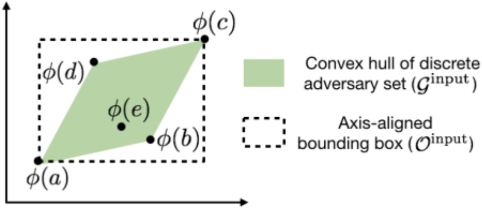
- Solving the convex relaxation is rather complicated. An easy and extremely efficient way to obtain an upper bound is via bound propagation.
- Given a perturbation region $\|\delta\|_\infty \leq \epsilon$, we know that $l_0 \leq x_0 \leq u_0$ for a given input x_0, and hence we can propagate the bound through the network

$$l_i \leq \Phi(W_i x_{i-1} + b_i) \leq u_i \tag{7}$$

- This is an even relaxed version of the convex relaxation.

Adversarial Word Substitution

- Interval bound propagation is used to train a robust model against word substitution attack[3].



[3] Robin Jia et al. *Certified Robustness to Adversarial Word Substitutions*. 2019. eprint: arXiv:1909.00986.

Certification for Robustness

- The upper bound can be used to determine whether or not an adversarial example exists within a certain perturbation region.
- One way to determine this is by considering the targeted attack of a given input against every possible class.
- This means that no point within the perturbation region exists that will change the class prediction.

$$h_\theta(x+\delta)_{y'} - h_\theta(x+\delta)_y < 0, \ \forall y' \neq y \tag{8}$$

Certification for Robustness

- This provides a guarantees on the adversarial robustness.
- If we cannot make the true class activation lower than any other classes even in the convex outer polytope (or any relaxed version), then we know that **no** norm-bounded adversarial perturbation of the input exists that could mis-classify it.