Lecture 25: Robustness (Part 2)

Instructor: Aditya Bhaskara *Scribe: Khawar Murad Ahmed*

CS 5966/6966: Theory of Machine Learning

April 14th, 2022

Abstract

In this lecture, we first went over corruption of data, generation of adversarial examples, adversarial training. Finally, there was a brief description of Differential Privacy.

1 Introduction

As a recap, we defined the following terms:

- *Training Time Corruption*: Adversary corrupts the small fraction of inputs.
- *Test Time Corruption*: Adversary evaluates model on inputs with "imperceptible error" added.

There are a few techniques that could be used to mitigate such a threat. At a high level, one can use entire data but limit the influence of outliers/noisy points. That problem can be tackled by either using median instead of mean or truncating the gradients. There is also the idea of "Inlier Pursuit", the idea that inliers enforce one another. Hypothetically, it is possible that if only the inliers were present, then a "simpler" model is possible. Currently, robust mean estimation is getting studied, which could be used as a subroutine in other algorithms, Gradient Descent for example.

Random Sample Consensus (RANSAC) is one of the examples, but there are other more sophisticated outlier filtering algorithms.

2 Robustness of Trained Models

Deep Learning models that generalize well are surprisingly brittle [Szegedy et al. 2013]. Even though Deep Learning models have a lot of parameters, a slightly perturbed input would be out of the distribution of inputs that the model learned, so generalization doesn't really matter. These inputs are referred to as Adversarial Examples. Roughly, if there was a real input x, the adversarial example would be $x + \delta$, with δ being the perturbation on the real input x. There are currently two popular ways of generating adversarial examples (with the assumption you have access to the model):

- Fast Gradient Sign Method (FGSM): Way to use gradients to carefully choose direction of the gradient.
- Projected Gradient Descent: Iterative procedure to maximally affect loss function.

The reason we need adversarial examples is because Nerual Net doesn't try to ensure that there is a set margin. So this problem could only really be solved with exponential data.

3 Adversarial Training

Recall that traditionally, the goal is for minimizing empirical risk, i.e. minimize $\mathbb{E}(x, y; w)$ where x is the input in a distribution, y is the label, and w are the parameters. For adversarial training, the goal is to minimize the empirical "robust risk", i.e. $min\,\mathbb{E}\,max\ \ l(x+\delta, label(x); w)$ where x is in the input distribution, and $||\delta|| \leq \epsilon$. This is also known as saddle point optimization. This would also create a trade off between robustness and "regular accuracy. For example, model M that uses "regular" ERM might have an accuracy of 80% but only 5% on noisy data, while model M', using minimax training, would have an accuracy of 65%, but accuracy on the perturbed images be 60%.

4 Differential Privacy

A similar approach of adding a little bit of noise to data is also used in order to make certain information unidentifiable, while having it be available for aggregate statistics. While one or two characteristics might not be enough, eventually more and more data becomes identifiable. A clever formulation by Dwork, Nissim, Naor and Smith is the following:Suppose my trusted party ensures that all query that analyst gives, the answer(x) and answer (x') are indistinguishable. So essentially $Answer(x) \approx Answer(x+\delta)$, where δ is some added noise.