Adverserial Examples

RITESH YADAV(MA17BTECH11009)

June 2019

What are Adverserial Examples?

- Adverserial Examples are inputs to machine learning models that an attacker has intentionally designed to cause the model to make a mistake; they're like optical illusions for machines.
- It is difficult for a human eye to differ between a real example and and an adverserial example.
- Let's say there is a ML system M and input sample C which we call a clean/real example. Let's assume that sample C is correctly classified by the ML system, i.e. M (C) = y_{true} . It's possible to construct an adversarial example A which is perceptually indistinguishable from C but is classified incorrectly, i.e. M(A) != y_{true} .
- They can be pose as a great threat to the future world of AI.

Examples of The Threat in Physical World :

It is already highlighted in the research papers.

Types of Attacks :

- **Black-Box Attack :** where the adversary is a normal user who knows only the output of the model.
- **White-Box Attack :** here the adversary has complete knowledge about the model being attacked like weights, biases, hyper parameters used.
- **Targeted Attack :** where the target of attack is to generate a particular class as an output.
- **Non-Targeted Attack :** where their is no particular target for the attack and output can be in any random class.

Why do they happen?

- In general, a neural network is a computational graph where classification decisions are driven by weights and biases optimized on training data and doesn't explicitly apply logical reasoning for decisions that's why by small change in these weights and bias we can generate a perturbed example that can misguide the model to a mis-classification with a very high accuracy.
- **Hypothesis :** Adverserial Examples come from the model being far too linear and extrapolating in linear fashions when it shouldn't.
- Effect of adverserial example can be seen majorly on linear models , since highly non-linear models are able to restrict these upto huge extent.
  Examples of linear Models : SVMs, K-means algorithm

Categorization of Adverserial Attacks :

Adveserial attacks can be categorized on the basis of three dimensions :

- **Threat Model**
- **Perturbation**
- **Benchmark**

Threat Model :

Threat Model categorizes the adverserial attacks on the basis of different falsifications and threats generated by them and the amount of loss by it. We can further decompose it into four aspects :

- **Adverserial Falsification :**
  - False positive attacks generate a negative sample which is misclassified as a positive one (Type I Error).A correct software being declared as an malware in malware detection model is an example of it.
  - False negative attacks generate a positive sample which is misclassified as a negative one (Type II Error).
- **Adversary's Knowledge :**
  - White-Box Attack
  - Black-Box Attack
- **Adverserial Specificity :**
  - Targeted Attack
  - Non-targeted Attack

Threat Model :

- **Attack Frequency :**
  - One-time attacks take only one time to optimize the adversarial examples.
  - Iterative attacks take multiple times to update the adversarial examples.

Perturbations :

In adverserial learning we always do small amount of perturbations such that they are closed to original ones according to human eyes but are misclassified by the machines withb very high confidence percentage. There are three aspects of Perturbations :

- **Perturbation Scope**
- **Perturbation Limitation**
- **Perturbation Measurement**

Benchmark :

Adversaries show the performance of their adversarial attacks based on different data sets and victim models. This inconsistency brings obstacles to evaluate the adversarial attacks and measure the robustness of DL models. Large and high-quality data sets and complex and high-performance DL models usually make adversaries/defenders hard to attack/defend. The diversity of data sets and victim models also makes us hard to tell whether the existence of adversarial examples is due to data sets or models.Hence, for this we have to sometime set some BENCHMARKS.

Methods For Generation of Adverserial Examples :

FAST GRADIENT SIGN METHOD(FGSM):

- FGSM is to add the noise (not random noise) whose direction is the same as the gradient of the cost function with respect to the data. The noise is scaled by epsilon, which is usually constrained to be a small number via max norm. The magnitude of gradient does not matter in this formula, but the direction (+/-).

Questions :

- How to use adverserial examples to improve ML , even when there is no adversary?
- What to do we mean by dimensionality of Adverserial Examples?
- What is Softmax classifier and regression?
- Low quality image can affect the Adverserial Examples?
- How can we generate Adverserial Examples for Non-Linear Models?
- Will Adverserial Examples also affect a output which is coming with very high accuracy or Has only one output class?
- Why multiple classifiers assign the same class to Adverserial Examples?
- Applications of Adverserial Learning portion from Adverserial : Attacks and Defences for DL.

GENERATIVE ADVERSERIAL EXAMPLES(GANs) :

- Generative Adversarial Network(GAN) which is able to learn from a set of images and create an entirely new **fake** image which isn't in the training set.
- GANs can be used in many ways :
  - to generate new images based on some databases
  - to do 'inpainting' or 'image completion'.It could be that we want to remove parts of the image.
- There are two components of GANs which work against each other :
  - **Generator :** It starts off by creating a very noisy image based upon some random input data. Its job is to try to come up with images that are as real as possible.
  - **Discriminator :** It tries to determine whether a image is real or fake.

Working of GANs :

Let us have an image x which our discriminator D is analyzing.D(x) gives a low value near to 0 if the image looks normal or 'real' and a higher value near to 1 if it thinks the image is fake - this could mean it is very noisy or added with some noise for example. The generator G takes a vector z that has been randomly sampled from a very simple, but well known, distribution e.g. a uniform or normal distribution. The image that is produced by G(z) should help to train the function at D. We alternate showing the discriminator a real image (which will change its parameters to give a low output) and then an image from G (which will change D to give a higher output).At the same time we want G to develop the images which seems to D as real one and it mis-classifies as real one by assigning low output or 0 . Similarly, D will try to improve it's ability to discriminate between the real and fake ones. We want G to minimize the output of D whilst D is trying to maximize the same thing.Hence,they are playing a **min-max** game against each other, which is where we get the term **adversarial training**.

Mathematics Behind GANs :

Let us assume model is following a distribution like uniform distribution :

- Let the known distribution be p_z .We will draw a random vector z from p_z.
- Let us assume the data generated by the distribution normally be p_{data}.
- Our generator will try to learn its own distribution p_g. **Our goal is to show :** $p_g = p_{data}$.

We have two networks to train :

- We want to minimize D(x) if x is drawn from our true distribution p_{data} i.e. maximize D(x) if it's not.
- and maximize D(G(z)) i.e. minimize 1−D(G(z)).

Mathematics Behind GANs :

Formally it can be express as a **min-max game** :

$$\min_G \max_D V(D,G) = \mathbb{E}[logD(x)] + \mathbb{E}[log(1 - D(G(z))] \tag{1}$$

Important Points :

- The current researcher consensus is that adversarial examples aren't a product of overfitting, but rather of high-dimensional input and the internal linearity of modern models.
- Fast gradient method no longer work when the image is rotated or viewing angle changes

Study of Following Research Papers :

- **Adversarial Examples: Attacks and Defenses for Deep Learning**
- **Practical Black-Box Attacks against Machine Learning**
- **ADVERSARIAL EXAMPLES IN THE PHYSICAL WORLD**
- **Mathematical Analysis of Adversarial Attacks**
- **EXPLAINING AND HARNESSING ADVERSARIAL EXAMPLES**