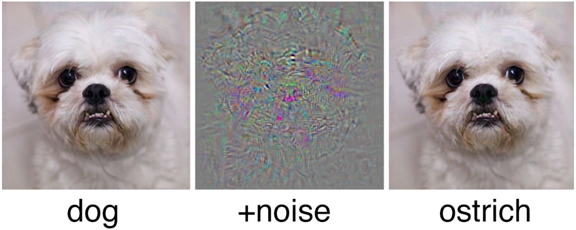
Adversarial Examples

A new evil in town to be aware of...

Sewade Ogun

AMMI, AIMS

Ghana, December 6, 2019



Objectives

1. To show the effect and effectiveness of adversarial examples in machine learning predictions
2. To understand the adversary, and determine how to combat them
3. To enlighten the audience on adversarial security measures.

Outlines

Introduction

- An **adversarial example** is an instance with small, intentional feature perturbations that cause a machine learning model to make a false prediction.[1]
- Adversarial examples are a type of **counterfactual examples** with the aim to deceive the model, not interpret it.

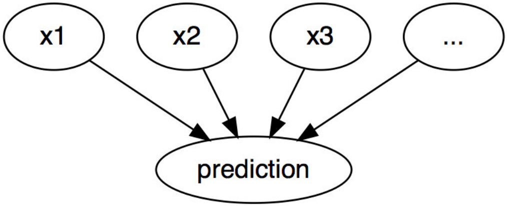


Figure: Causal relationships between inputs of a machine learning model and the predictions.

Properties of Counterfactual Instance

- A counterfactual should be as **similar** as possible to the instance regarding feature values
- Should change as **few** features as possible.
- A counterfactual instance **should** have feature values that are likely.
- It should produce the predefined prediction as closely as possible.

Examples

1. You submit your details for an offer in such a way that the machine classify you as eligible.
2. A spam detector fails to classify an email as spam. The spam mail has been designed to resemble a normal email, but with the intention of cheating the recipient.
3. A machine-learning powered scanner scans suitcases for weapons at the airport. A knife was developed to avoid detection by making the system think it is an umbrella.
4. Self-driving cars can be deceived by images to misclassify stop-signs.

Techniques

1. Minimize the distance between the adversarial example and the instance to be manipulated, while shifting the prediction to the desired (adversarial) outcome.
2. Perturb the example using the gradients of the model, which of course only works with gradient based models such as neural networks,
3. Use the prediction function to train a model to generate new examples, (which makes these methods model-agnostic.)

Our focus will be on how adversarial examples affect image classifiers with deep neural networks.

Gradient based optimization approach

$$\min \, loss(f(x+p), y_{adv}) + c.|p|$$

where x is an image, p is the changes to the pixels to create an adversarial image, y_{adv} is the desired outcome class, and the parameter c is a balancing factor.

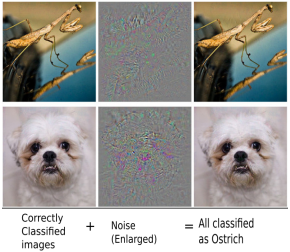


Figure: Examples generated on Alexnet using GB[2]

[2] Szegedy, Christian, et al. "Intriguing properties of neural networks."(2013)

Fast gradient sign method

$$x_{adv} = x + \epsilon Sign(\nabla_x J(\theta, x, y))$$

where x is the gradient of the models loss function with respect to the original input pixel vector x, y is the true label vector for x and θ is the model parameter vector.

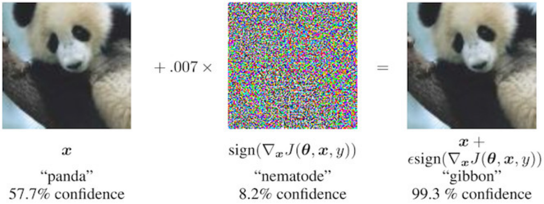


Figure: NN predicts Gibbon for a perturbed panda image[3]

[3] Goodfellow et al. "Explaining and harnessing adversarial examples."(2014)

Changing a single pixel

Uses **differential evolution** to find out which pixel is to be changed and how.

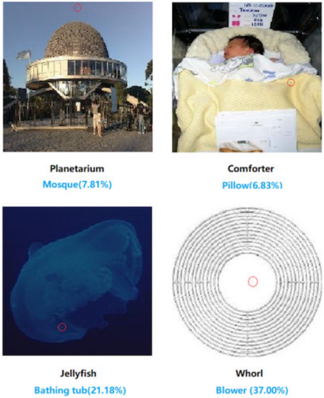


Figure: Changing a single pixel (marked with circles) to deceive a NN to predict the wrong class instead of the original class.[5]