Adversarial POMCP

Alessandro Rodeghero
Lorenzo Tabarelli

Project

The project plans to emulate the behavior of a black-box, the victime, through the creation of a model, a neural network. The constitution of an NN that emulates the behavior of the oracle allows us to "open" the black box and be able to observe the logic of classification. Knowing the logic of the black box, attacks can be planned via adversarial examples.
An adversarial example, also referred to as an attack, is an input that has been generated to be misclassified by a machine learning model. These inputs are usually a high dimensional input such as images or audio samples.

Attacking Black-Box Models:

When we know nothing about how a model works, we refer to it as a ‘black-box’. To generate an initial dataset, we can use the black-box model giving it random inputs and observe its classification response for those inputs.

The objective of the attack is to move a point over a victim model’s decision boundary. For example, we’ll take a point that is normally classified as class ‘0’ and push it over the victim model’s decision boundary to be classified as class ‘1’. This change to the original point is also called a *perturbation* when using higher dimensional data because we’re making a very small change to the input.

The jacobian-based dataset augmentation technique aims to train another model, called the *substitute model*, to share very similar decision boundaries as the victim model. Once a substitute model is trained to have almost the same decision boundaries as the victim model, an adversarial perturbation that is created to move a point over the substitute model’s decision boundary will likely also cross the victim model’s decision boundary. This can be achieved by exploring the space around the victim model’s decision space and determining how the victim responds.

Training the substitute model

Jacobian-based dataset augmentation works where a random sample of the initial data is taken and used to train a very poor substitute model. The adversarial examples are created from the dataset. These are a step in the direction of the model’s gradient to determine if the black-box model will classify the new data points the same way as the substitute model.

The augmented data is labeled by the black-box model and used to train a better *substitute model* that gets a more precise understanding of where the black-box model’s decision boundary is. After a few iterations of this, the substitute model shares almost the exact same decision boundaries as the black-box model.
Ultimately, with a small sample of data, a few iterations of the data augmentation and labeling, a black-box model can be successfully attacked.

In our case

the black box(oracle) information we possess is limited:

- 81 features for each sample(spatial point)
- two-class classifications: ‘0’ and ‘1’

Substitute model:

The substitute model is constituted as a neural network composed by 3 layers:

- input layers of one neuron per feature, 81 in total
- hidden layers: 2 layers, 16 neurons each
- output layer: 2 neurons, one per class.

Initial dataset

The initial dataset is generated as 380 samples with relative labels (190 samples per class), divided into train-set (80) and test-set (300). These are the points from which we will begin to generate synthetic data through jacobian augmentation.
From this information available, the network must be trained in such a way that it behaves like the oracle.

Each sample is 81 elements leght and is crafted following the *Dirichlet distribution*:

- each element must be $\in [0, 1]$
- the sum of all elements must be 1

These two conditions must be respected also during the data augmentation process.

The label of a sample is defined after 30 predictions of the sample by the black-box: the most frequent classification is stored as the sample label.