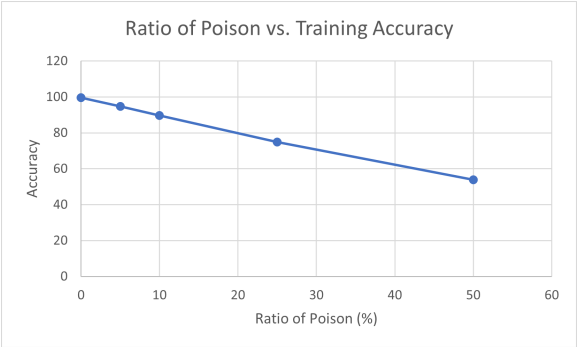
CS 579 Trusworthy Machine Learning - Homework 3

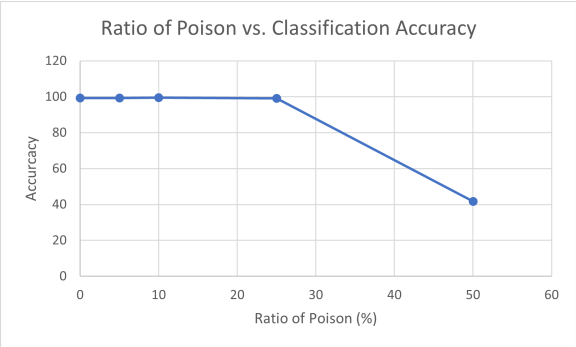
Eugene Yong

May 23, 2023

Task I: Poisoning Attack Against Logistic Regression Models

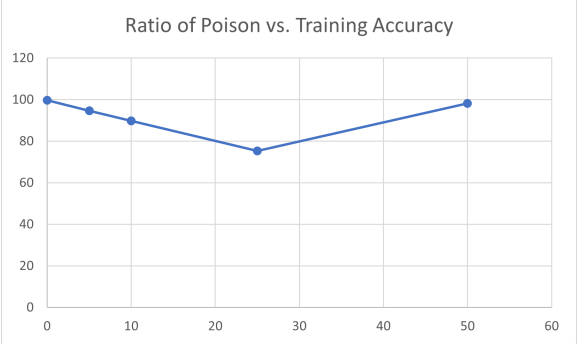


(a) Ratio of Poison vs. Training Accuracy

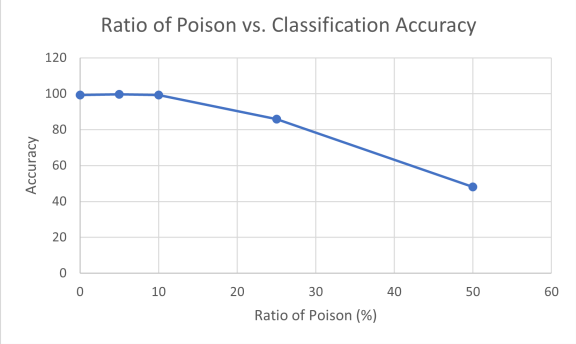


(b) Ratio of Poison vs. Classification Accuracy (Test set)

Figure 1: Results of Random Label Flipping Attack to Logistic Regression Models



(a) Ratio of Poison vs. Training Accuracy



(b) Ratio of Poison vs. Classification Accuracy (Test set)

Figure 2: Results of Random Label Flipping Attack (Only flipping label of Ones) to Logistic Regression Models

Given that my implementation is correct. My guess is that by randomly flipping the label, we can't effectively shift the decision boundary of the model. Maybe the effect of randomly flipping labels from both classes cancel out each other. This can be confirmed by looking at the Ratio of Poison vs. Training Accuracy graph. The training accuracy is always $\sim$ 99% minus the ratio of poison. Which means it is still learning the same decision boundary as if using clean dataset and only missclassify the image that the label was flipped.

To confirm that hypothesis, I create another sets of contaminated training sets by only flipping the labels of the ones. The classification accuracy actually goes down for ratio=25%. One interesting note is that the training accuracy of ratio=50% actually goes back up to $\sim$ 98%, meaning that the model is actually able to fine a decision boundary that fit the contaminated training set.

Task II: Poisoning Attacks on Deep Neural Networks

# of Poisons	1	5	10	25	50	100
# of Successful Attacks	4	4	5	5	5	5

Table 1: Number of Successful Attacks Over 5 Targets

This is expected as in the original paper, they claim that the attack works by only adding one poison instance. However, my attacks fail on one target using only 1 and 5 poisons. One thing stand out to me is that the loss of predicting that target instance using the original ResNet18 model is much lower than predicting the other 4 target instance. My guess is that the original decision boundary fit very well into the space of that particular target instance, making it harder to shift the decision boundary using only a few poisons.