Introduction

In the assignment number two, I learned how to implement Membership Attack. I implemented it in two ways. First, for the ICP, I implement Membership Attack on 2 different datasets using the same model structure. After that, for the assignment, I implemented an attack using a different model but on the same dataset. For my work, I used techniques described in Cyphercat GitHub, which I found using one of the references of the PowerPoint slides. In this paper I will go over the methods I used as well as what was done and what can be improved.

Objectives

The main objective of this assignment was to learn how to implement a membership attack when we either know the structure of the model(which was implemented in ICP) or the dataset that was used to train the model. The secondary goal was to learn what is precision, recall, and accuracy and how to calculate them and use them to see how effective the attack model is. And the third objective was to learn how to defend against the membership inference attack and what types of defense mechanisms can be used.

Approaches/Methods

To accomplish my goals, I defined three models. One of the models is the target model that I am trying to attack. The second model is the Shadow Model, that is different from the target model in its structure, but it tries to mimic it in classifying the images in CIFAR10 dataset. And the third model is Binary Classification model built with neural networks. All of the models are using Adam optimizers since for this task with trial and error method I found out that it is

working better comparing to SGD. I approach the problem the following way. I first train all the model and then perform an attack while evaluating the accuracy of the models, and precision as well as recall of the attack. I used 100 epochs for each model, 128 was my batch size for each model, the learning rate was 0.001. Before building the models, I loaded data and created transforms for training and test portions of the data. Then I divided the data in 4 parts to create in and out loaders for target and shadow models. I also used cuda in order to utilize gpu to reduce the amount of waiting for training and evaluation of the models.

Workflow

1. Download Datasets
2. Create Transforms
3. Split the datasets described above
4. Activate cuda
5. Define and train the models described above
6. Evaluate models

Datasets

I used CIFAR10 dataset which has images of 10 different classes. This is a labeled dataset.

Parameters

- Learning Rate = 0.001
- Batch size = 128

- Number of epochs = 100
- GPU from google Collaboratory
- CIFAR10 dataset
- Various parameters of the model networks

Evaluation and Discussion

The attack precision in both cases of attacks (with unknown dataset or model) was 0.63%. The attack accuracy was higher when the dataset was unknown, but the model is known and is 70.79 vs 70.52 when the model is unknown. None of my trials reached 85% accuracy of attack. I think that is due to a very simplistic model that was used for binary classification with neural networks. There is more job to be done with this model to increase its accuracy.

Conclusion

In this paper I discussed how I implemented the membership inference attack. For the conclusion part, I would like to talk about possible defenses that were used. The first one I will discuss is breaking linearity of the target model. I used Rectified Linear Unit twice for the target model in order to break linearity, this is probably the reason the attack precision is so low. Those were two types of defenses I used in order to decrease the attack accuracy of the attacking model. In future, I would like to use simpler neural network to increase attack accuracy and then very complex networks to make attack accuracy even lower than they are now.