CSC 8228: Privacy-Aware Computing
Project 2 - Report: Membership Inference Attack

Name: **Srinivas Narne**
Panther-Id: **002705961**

Abstract

This project explores the implementation of a Membership Inference Attack (MIA) on a machine learning model trained using the CIFAR10 dataset. The primary goal is to ascertain whether specific data points were included in the training set of the pre-trained model. The significance of this investigation lies in highlighting potential privacy risks associated with machine learning models, especially in handling sensitive data. Through the utilization of various datasets derived from CIFAR10, we designed and trained an attack model capable of identifying the inclusion of individual data points in the training set. The results offer valuable insights into the susceptibilities of machine learning models to privacy breaches, emphasizing the necessity for more robust data protection measures in the model training process.

Introduction

The initiation of a Membership Inference Attack (MIA) involves crucial steps that revolve around submitting data to the target model and obtaining predictions. This initial phase establishes the groundwork for assessing the model's susceptibility to privacy breaches. As data is forwarded to the model for prediction, various input types—ranging from images to textual or numerical information—are processed. Utilizing its learned parameters, the target model generates predictions based on the provided data. The resulting predictive output becomes pivotal in subsequent stages of the MIA, guiding the assessment of the model's accuracy, the storage of result values, and the overall determination of the attack's success. The success of an MIA depends on the intricate interplay between input data, model predictions, and the subsequent analysis of these predictions concerning membership inference.

Approaches Used

I experimented with various machine learning models, including Random Forest and XGBClassifier, recognizing their respective strengths in handling diverse types of information, including numerical and categorical features. Random Forest is particularly versatile, while XGBClassifier, a variant of XGBoost, demonstrates expertise in discerning intricate relationships within the data. In addition to these models, I also employed linear regression and AdaBoost in my analysis. Each model brings its unique characteristics and capabilities to the task at hand.

Methodology

1.Data Preparation:

The project employs multiple datasets derived from the CIFAR10 dataset:

- **Train_data_partial.npy** and **Train_labels_partial.npy**: Represent partial training data and labels.
- **Test_data_partial.npy** and **Test_labels_partial.npy**: Represent partial test data and labels not utilized in training the target model.
- **Test_member_data.npy** and **Test_Non-member_data.npy**: Used for evaluating the attack model's performance.
- **Evaluation_data.npy**: Used for the final evaluation of the attack model.

2.Target Model:

The target model, denoted as **Trained_target_model.h5**, is structured as follows:

- **Input Layer:** Accepts 32x32x3 image data.
- **Convolutional and MaxPooling Layers:** Extract features from images through convolutional operations followed by downsampling.
- **Flattening:** Converts 2D feature maps into a 1D vector.
- **Dense Layers:** Fully connected layers for classification, including the output layer for the CIFAR10 classes.

3.Attack Model Development:

The attack model is designed as a binary classifier, with the primary objective of distinguishing between data used in the training of the target model (members) and data that was not used (non-members).

Linear Regression Approach:

Training Phase:

In the training phase of linear regression, the model is fitted to a dataset containing both member and non-member data points. Linear regression aims to establish a linear relationship between the input features and the target variable by minimizing the sum of squared differences between predicted and actual values.

Prediction Phase:

During the prediction phase, new data points are provided as input to the trained linear regression model, and predictions are calculated based on the established linear relationship.

AdaBoost Approach:

Training Phase:
AdaBoost, in its training phase, works by sequentially fitting weak learners to the dataset. Each weak learner focuses on the data points that were misclassified by its predecessor, and their predictions are combined to create a strong, ensemble model.
Prediction Phase:
In the prediction phase, new data points are presented to the trained AdaBoost model. The model aggregates the predictions of the weak learners to make a final prediction for each data point, incorporating the strengths of multiple weak learners.

XGBoost Approach:

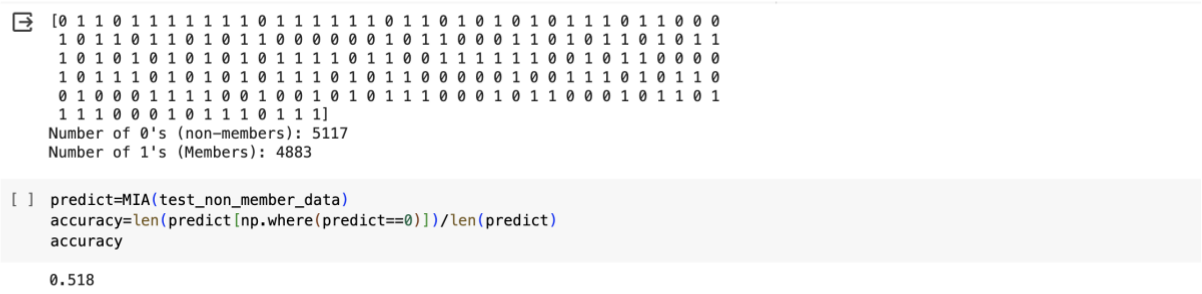
Training Phase:
In the training phase, akin to RandomForest, the XGBoost model undergoes training on a dataset comprising both member and non-member data points. XGBoost, a gradient boosting algorithm, constructs a sequence of decision trees for making predictions. The training process involves iteratively adding trees to rectify errors made by the existing ones.
Prediction Phase:
During the prediction phase, newly introduced data points are input into the trained XGBoost model, and predictions are generated.

4. Evaluation:

The attack model's performance was evaluated using:
Primary Evaluation: Utilizing *Evaluation_data.npy* to assess how well the model could generalize its inference to unseen data.

Results

The XGBoost approach demonstrates a marginally higher accuracy with a dataset consisting of 5,000 member and 5,000 non-member data points, resulting in predictions of 4,883 for members and 5,117 for non-members. The attack model effectively distinguished between member and non-member data points, achieving an accuracy of 51.8%. The incorporation of extra validation datasets played a crucial role in evaluating the model's ability to generalize.



```
[0 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 0 1 0 1 0 1 0 1 1 1 0 1 1 0 0 0
 1 0 1 1 0 1 1 0 1 0 1 1 0 0 0 0 0 0 1 0 1 1 0 0 0 1 1 0 1 0 1 1 0 1 0 1 1
 1 0 1 0 1 0 1 0 1 0 1 0 1 1 1 1 0 1 1 0 0 1 1 1 1 1 1 0 0 1 0 1 1 0 0 0 0
 1 0 1 1 1 0 1 0 1 0 1 0 1 1 1 0 1 0 1 1 0 0 0 0 0 1 0 0 1 1 1 0 1 0 1 1 0
 0 1 0 0 0 1 1 1 1 0 0 1 0 0 1 0 1 0 1 1 1 0 0 0 1 0 1 1 0 0 0 1 0 1 1 0 1
 1 1 1 0 0 0 1 0 1 1 1 0 1 1 1]
Number of 0's (non-members): 5117
Number of 1's (Members): 4883
```

```
[ ] predict=MIA(test_non_member_data)
    accuracy=len(predict[np.where(predict==0)])/len(predict)
    accuracy
```

```
0.518
```

Conclusion

This project showcases the viability of Membership Inference Attacks in uncovering privacy vulnerabilities within machine learning models. It underscores the significance of incorporating resilient privacy-preserving techniques in both model training and data handling processes.