TML Assignment 1

GitHub Repository

1. Task Description

This project addresses the **Membership Inference Attack (MIA)** challenge. The goal is to predict whether a given data point was part of a machine learning model's training set. The task is binary classification where only the target model and a public dataset are provided.

---

2. Proposed Solution

We designed a solution based on the shadow model paradigm coupled with score-based inference techniques. The approach involves training multiple shadow models to simulate the behavior of the target model, followed by extracting informative features to train an attack model.

Shadow Model Architecture

We utilized **ResNet-18** as the base architecture for all shadow models. These models were trained using the provided public dataset (`pub.pt`) with a 70/30 train-validation split. Training was conducted over a maximum of **200 epochs** with **early stopping** to prevent overfitting.

Feature Extraction

We extracted the following features from model softmax outputs:

- Maximum Confidence
- Entropy
- Margin (Top-1 - Top-2 probability)

These were computed for both member and non-member samples during shadow model evaluation.

Attack Model: Gradient Boosting with LiRA

For the attack model, we utilized a **Gradient Boosting Classifier** (GBM) trained on the extracted shadow features. We employed the **Likelihood Ratio Attack (LiRA)** method for final inference, estimating

confidence distributions of member vs. non-member samples and computing scores based on their ratio.

Final Configuration

In the final system, we adopted a shadow modeling framework comprising **seven independently trained ResNet-18 classifiers**. Each shadow model was optimized using **early stopping** over a maximum of **200 epochs** to enhance generalization and prevent overfitting.

For the membership inference attack (MIA), we employed a **score-based LiRA (Likelihood Ratio Attack)** strategy, leveraging the model confidence distributions learned from shadow data. To perform the final classification, we utilized a **Gradient Boosting Classifier (GBM)** as the attack model, trained on shadow model outputs including **confidence, entropy, and margin statistics**.

This configuration yielded the most competitive performance across both local and server evaluations, achieving a **TPR@FPR=0.05 of 0.06166666666666667** on the official leaderboard.

---

3. Experiments and Results

We experimented with several configurations to improve MIA performance:

Method	Local AUC	Local `TPR@0.05`	Server `TPR@0.05`	Server AUC
7 Shadow Models (Baseline MLP)	0.5871	0.1265	0.043	0.5017
Gradient Boosting + Confidence Features	0.6539	0.1604	0.0573	0.5141
GB + Raw Logits + Confidence Features	0.6614	0.1671	0.0546	0.5223
GB + 30 Epochs + Weight Decay	0.6497	0.1560	0.0603	0.5078
Voting Classifier	0.6525	0.1656	0.044	0.5104
LightGBM	0.8886	0.5375	0.0566	0.5063
Gradient Boosting + LiRA + 7 Shadow + 200 Epochs	0.5984	0.1145	**0.0617**	**0.5121**

---

4. Lessons Learned

- Confidence-based LiRA methods consistently outperform threshold-based classifiers.
- Increasing shadow models improves robustness but may introduce noise beyond a point.
- Early stopping and feature engineering significantly impact attack effectiveness.

5. Files and Their Descriptions

- `main.py` : Trains shadow models, extracts features, and trains the attack classifier.
- `submit.py` : Loads shadow stats and performs LiRA-based attack on the private dataset.
- `attacks/attack_model.py` : Defines the attack model (Gradient Boosting).
- `attacks/feature_extractor.py` : Extracts softmax features (confidence, entropy, margin).
- `data/dataset.py` : Dataset wrappers and shadow data split logic.
- `models/shadow_model.py` : Shadow model training logic (ResNet-18).
- `config.py` : Configuration values and constants.
- `test.csv` : Final submission file with membership scores.
- `shadow_model.pt` : Saved PyTorch model from shadow training.
- `attack_model.pkl` : Trained Gradient Boosting attack model.

---

6. References

- **Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr.** *Membership Inference Attacks From First Principles.* arXiv preprint arXiv:2112.03570, 2021.
- **Gauri Pradhan, Joonas Jälkö, Marlon Tobaben, and Antti Honkela.** *Hyperparameters in Score-Based Membership Inference Attacks.* arXiv preprint arXiv:2502.06374, 2025.
- **Gongxi Zhu, Donghao Li, Hanlin Gu, Yuan Yao, Lixin Fan, and Yuxing Han.** *FedMIA: An Effective Membership Inference Attack Exploiting "All for One" Principle in Federated Learning.* arXiv preprint arXiv:2402.06289v2, 2024.