Membership Inference Attack Against Diffusion-Generated Tabular Data: A Black-Box, Single-Table Study

Technical Report, June 2025

Tejaswi Duptala
MAI program
Ostbayerische Technische Hochschule Amberg-Weiden
Amberg, Germany

***Abstract*—Synthetic data is commonly used to maintain people's privacy while releasing data. It creates novel data that are similar to the original data but are not supposed to invade any person's information. Synthetic data, though, can sometimes give people's information in the original training data. This leakage is a privacy risk. We study this risk with tabular data generated by diffusion models. We evaluate membership inference attacks on 30 datasets from the MIDST challenge. We start with comprehensive data cleaning and preprocessing. We then compute 17 features per record based on synthetic record distances. The features help quantify how much each record is near the synthetic distribution. We train a variety of machine learning models to classify whether a record was in the training data. Models we train are Random Forest, XGBoost, LightGBM, and neural networks. The best model we identified is a combination of these using neural networks and LightGBM in ensemble. The best model we identified identifies up to 28% of training data members with minimal false alarms. Our results show that even well-designed synthetic data can spill sensitive information. This underscores the need for strong privacy guarantees and thorough testing of synthetic data.**

***Index Terms*—membership inference, synthetic data, tabular diffusion model, privacy, machine learning, ensemble, MIDST**

I. Introduction

Data is highly significant in numerous fields, such as research, business, and medicine. Data helps organizations make sound decisions and develop beneficial products. However, data typically contains intimate personal information about individuals. If data is used or published without adequate protection, it can harm people's privacy and lead to undesirable consequences.

To address these privacy concerns, researchers have developed synthetic data techniques. Synthetic data is produced artificially to mimic real data. It preserves overall patterns and distributions but does not include exact personal details. Because of this, synthetic data can be shared more openly without exposing individual identities. Synthetic data has become popular as a privacy-enhancing alternative to using real data directly.

Despite these benefits, synthetic data may still leak information about the original training data. One way to test this weakness is through membership inference attacks (MIA). In an MIA, an attacker tries to determine if a particular record was used to train the generative model. If the attack can successfully identify training members, it implies a privacy risk in the synthetic data or the model that produced it [1].

In this project, we study membership inference attacks on synthetic tabular data generated by diffusion models. Diffusion models are a type of generative model that have shown promising results in data synthesis. We use the MIDST challenge data, which provides a realistic and challenging setting for such attacks. Our study focuses on the black-box, single-table scenario where no internal information of the diffusion model is available; only the synthetic data and challenge data are accessible.

Our goal is to develop an effective attack that can predict whether a record in the challenge set was part of the training data. To do this, we extract features that quantify how challenge records relate to the synthetic data distribution. We then train machine learning classifiers on these features to perform membership inference. Finally, we evaluate the attack's success using metrics widely accepted in privacy research, such as TPR@10%FPR.

II. Methods

A. Data and Problem Setting

In this project, 30 synthetic tabular data sets are present. They are named `tabddpm_1` to `tabddpm_30`. They are MIDST challenge data sets. The challenge is on checking privacy threats in synthetic data produced by diffusion models.

These data sets have the following number of major components:

- **Challenge Set:**
  This set contains records. Some of them are from original training data, known as members. Others are from unused training data, known as non-members. The challenge set is a combination of both types.
- **Ground-Truth Labels:**
  In datasets 1–20, a column `is_train` informs us whether every record in the challenge set is a member (1) or not (0). These labels allow us to train and test our membership inference algorithms. For datasets 21–30, the labels are not given to us. They are only used for testing the attack after training.

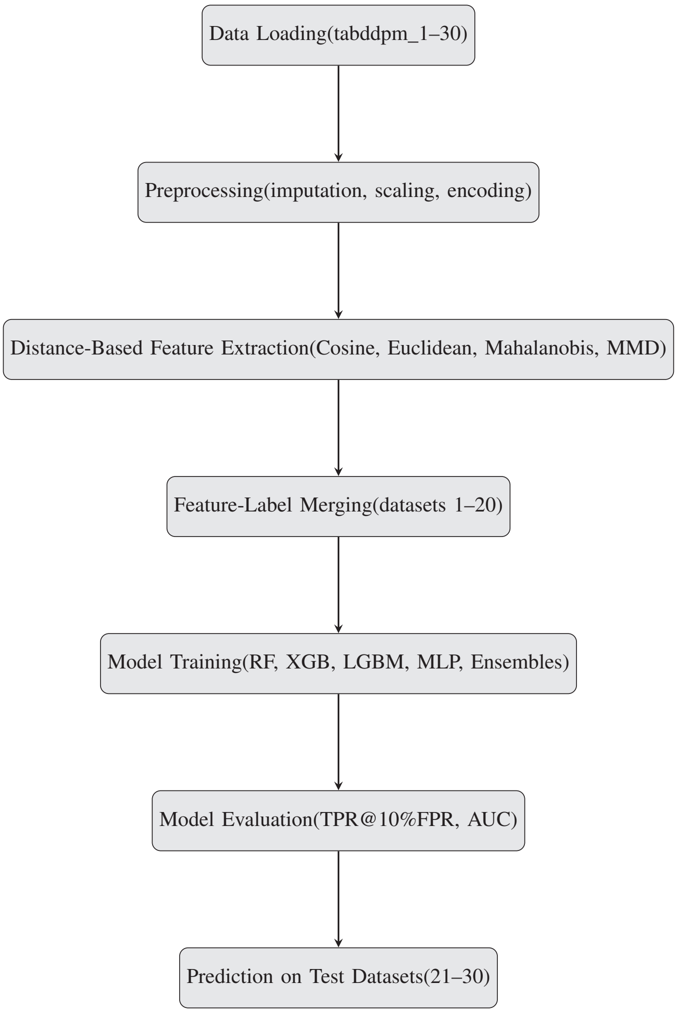
Data Loading(tabddpm_1–30)
Preprocessing(imputation, scaling, encoding)
Distance-Based Feature Extraction(Cosine, Euclidean, Mahalanobis, MMD)
Feature-Label Merging(datasets 1–20)
Model Training(RF, XGB, LGBM, MLP, Ensembles)
Model Evaluation(TPR@10%FPR, AUC)
Prediction on Test Datasets(21–30)

Figure 1. Workflow of the membership inference attack pipeline.

- **Synthetic Dataset:**
  This dataset is created by the diffusion model. It tries to simulate the real training data but does not have any precise original records. The synthetic data is used to help determine membership by comparing challenge records to it.

Datasets 1–20 contain additional training data that use to train and calibrate the techniques. These sets help us learn how to determine membership. Datasets 21–30 are held out as a test set. This keep them only for the final evaluation to see how well the attack works on new data.

The first goal is to construct a membership inference attack. This attack needs to accept a challenge record and make an estimation of whether or not it was employed in training. The challenge and synthetic datasets are employed separately for this. There is no knowledge about the internals of the diffusion model, such as its parameters or training.

This is referred to as a black-box, single-table scenario. “Black-box” indicates that we only get to see the data outputs, not the model workings. “Single-table” indicates that the data is delivered in one table, rather than being divided into several related tables.

B. Data Preprocessing

It is important that data need to be prepared carefully before training and analyzing the models with the data. The same preprocessing is performed for all 30 datasets so that the data gets cleaned and standardized.

1) Missing Values: There may be missing values in particular columns of the data. The missing values can pose an issue to machine learning models if they are not handled properly. For numerical columns such as transaction date, amount, and balance, the model impute missing values using the column mean. This is among the simplest and most effective methods of imputing missing values. For categorical columns such as transaction type, operation, symbol, and bank, we replace missing values with the most frequent category for each column. This maintains the consistency of the data.

2) Categorical Encoding: Machine learning algorithms usually work with numbers, but there are codes or words in categorical columns. In order to be able to use such columns, we convert each categorical feature into a list of binary columns with the help of one-hot encoding. For example, if column `transaction_date` has three categories — A, B, and C — one-hot encoding creates three new columns with information on whether a record belongs to each category or not. Models can understand categorical data in the correct way with this.

3) Numerical Scaling: Numerical features generally have different scales and ranges. For example, amount can be in thousands while `transaction_date` can be a small number. In order to ensure our models learn well, especially neural networks, we apply standard scaling. Standard scaling normalizes each numerical feature to have mean zero and standard deviation one.

Scaling stabilizes and speeds up the training process.

4) Global Fitting of the Preprocessing Pipeline: For preventing data leakage, we fit the preprocessing pipeline globally with all available training data from datasets 1 through 20. This means that calculate the means, modes, and scaling parameters solely on training data, never holdout or test sets. After fitting, apply the same preprocessing to all datasets, including holdout datasets 21 through 30. This ensures an equal and consistent treatment of all the data.

5) Removal of Identifier Columns: Certain columns, such as transaction IDs or account IDs, uniquely identify records but do not help with membership prediction. Including them could be an overfitting or privacy issue. Therefore,removing all ID columns before feature extraction or modeling. The attack is thus founded upon meaningful data patterns rather than on trivial identifiers.

These preprocessing steps yield a clean, numeric, and normalized dataset to move on to the next step: feature engineering.

C. Distance-Based Feature Extraction

To identify whether a challenge record was in the training data or not, there are features that tell us how close or distant it is relative to the synthetic data. For this, a set of 17 features are defined for describing the relation of every challenge record to all the synthetic data. These features are driven by the observation that a training record will somehow be closer to the synthetic data distribution than a non-member. This is motivated by prior work [1], which demonstrated that distance-based statistics can be utilized for membership inference, and this method was adapted for the setting of tabular data.

1) Distance Metrics Computed: For each record in the challenge set, we compute three kinds of distances to all synthetic samples:

- **Cosine Distance:**
  It is the angle between two vectors. It captures similarity in direction, not magnitude differences. The less the cosine distance, the more similar the records are in orientation.
- **Euclidean Distance:**
  It is the straight-line distance between two points in the feature space. It captures absolute differences between records.
- **Mahalanobis Distance:**
  Accounts for correlations among features by considering the covariance structure of the synthetic data. It's a measure of statistical distance, so records that differ in unusual ways will have greater distances.

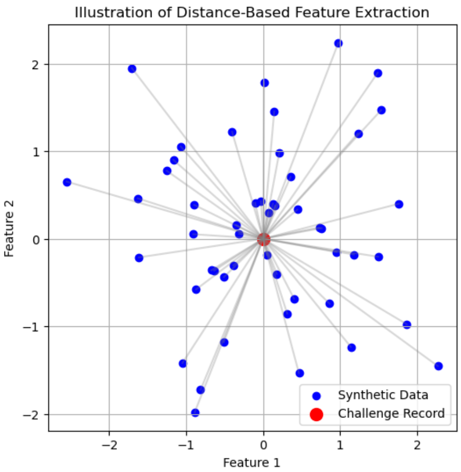

Figure 2. Illustration of distance-based feature extraction. The red point represents a single challenge record. Blue points are synthetic data samples. Grey lines indicate distance calculations used to compute summary statistics for membership inference.

Summary Statistics for Each Distance

For each of these distance measures, we calculate the following five summary statistics over the distances from one challenge record to all synthetic records:

- Minimum distance: The nearest synthetic record to the challenge record.
- Mean distance: The average distance to all synthetic records.
- Standard deviation: A statistic of how spread out the distances are.
- Median distance: The median when ordering distances.
- Top-5 mean: The average of the five smallest distances, capturing local similarity.

This provides 15 features (3 types of distance × 5 statistics).

2) Maximum Mean Discrepancy (MMD) Features: On top of raw distances, we also calculate two additional features based on Maximum Mean Discrepancy (MMD), a statistic of differences between distributions:

- **MMD with Radial Basis Function (RBF) kernel:**
  Captures complex nonlinear relationships by comparing the distribution of the challenge record and the synthetic distribution in a smooth way.
- **MMD with Linear kernel:**
  Calculates the linear difference between the distributions.

Both these MMD values provide complementary information about how well a challenge record fits into the global distribution of synthetic data.

3) Final Feature Vector: Concatenating the distance measures and MMD values, each challenge record is now a 17-dimensional feature vector. The vector captures how "typical" or "atypical" the record is relative to the synthetic data.

These features are inputs to the membership prediction machine learning models.

D. Model Training and Model Selection

Once after deriving the 17 distance-based features for each challenge record, training machine learning models comes next. The models try to classify members (training-used records) and non-members (records not used in training) on these features.

In the project experiment with some common classifiers that are known to work well on tabular data:

- **Random Forest:**
  This is an ensemble of many decision trees. Each one of them is learned on a randomly chosen subset of the data. The final prediction is made by an average of all predictions. Random Forest helps to fight overfitting and provides robust results.
- **XGBoost:**
  XGBoost is an abbreviation for Extreme Gradient Boosting. It builds trees in sequence, with every new tree trying to compensate for the errors made by the earlier ones. This boosting method produces high accuracy and was widely used during data science competitions.
- **LightGBM:**
  LightGBM is a gradient boosting algorithm. It's highly efficient and speedy. It uses an innovative approach to building trees that can handle large datasets faster with good accuracy.
- **Multi-layer Perceptron (MLP):**
  MLP is a neural network. It consists of layers of neurons that are fully connected and learn intricate, nonlinear relationships between the data. In MLP, features are pre-scaled to allow the network to be learned efficiently.
- **Ensemble Models:**
  Soft voting to average over multiple classifiers is employed to boost performance. Each model provide us with predicted probabilities and then average these out to make the final prediction. Ensembles are generally better and more reliable than individual models.

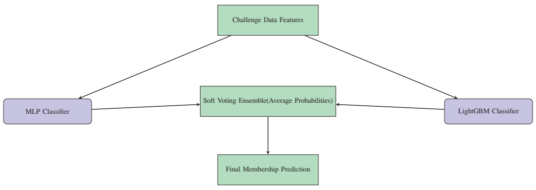
Challenge Data Features
MLP Classifier
Soft Voting Ensemble(Average Probabilities)
LightGBM Classifier
Final Membership Prediction

Figure 3. Ensemble model combining MLP and LightGBM classifiers via soft voting of predicted probabilities.

The training was performed on models using datasets 1 to 20. In the model combine the datasets and split the data into a training dataset and a validation dataset. This splitting is done carefully to keep the ratio of members and non-members the same in both sets.

The fine-tuning of the models is performed by trying different hyperparameters and training configurations. This fine-tuning allows the models to learn the best way to differentiate between members and non-members.

1) Evaluation Metrics: The project main evaluation metric is **TPR@10%FPR**:

- **True Positive Rate (TPR):** The percentage of actual members that are appropriately detected by the model.
- **False Positive Rate (FPR):** The percentage of non-members that are incorrectly identified as members.
- **TPR@10%FPR:** The TPR when FPR is 10%. That is when it look at the true member detection rate with only permitting 10% false alarms. This is a critical measure since it accounts for privacy risk and the attacker's tolerance for error.

This measure is employed since it most closely aligns with the privacy challenges in the MIDST challenge. Greater TPR@10%FPR suggests an improved membership inference attack.

The model also compute **AUC** (Area Under the ROC Curve) as a second measure. AUC provides an estimate of the model's total ability to separate members from non-members at all thresholds of classification.

2) Random Guessing Baseline: To make sure our models are accomplishing something useful, in the project random guessing baseline is also tried. This baseline makes random guesses at membership and should score an TPR@10%FPR of about 0.10, i.e., it is just guessing without any information.

The heavily trained models ought to perform considerably better than this baseline in order to demonstrate that they actually learn privacy-sensitive data.

Summary of Model Exploration

In the project have trained a number of classifiers including Random Forest, XGBoost, LightGBM, and Multi-layer Perceptron. And explored the ensembles of the above-mentioned models to leverage their complementary strengths. Models were sorted based on TPR@10%FPR and AUC on validation sets (datasets 1–20). The best performer among the models was an MLP ensemble with LightGBM, which optimized the TPR@10%FPR.

E. Evaluation Process

During the assessment of our membership inference attack performance, we design an apt evaluation process based on development and end testing stages.

1) Validation Stage: While the project create models and tune parameters, utilize datasets 1 to 20 that have ground-truth membership tags.The project divide the datasets into a training set and validation set through stratified sampling in order to promote fairness and trustworthiness in our assessment.

a) Stratified Splitting: The data is split such that both the train and validation sets have roughly equal numbers of members (`is_train = 1`) and non-members (`is_train = 0`). This avoids class bias due to class imbalance and is an actual test of model performance.

b) Purpose: Validation set is employed to make an estimate of model parameters and select the best model without consideration of the test data. It acts as a placeholder in order to come up with an estimate of how the model will perform on new, unseen data.

2) Final Test Phase: Now that the model have the top-performing model based on validation,it apply this model to the datasets 21 to 30, which constitute a holdout test set.

a) Labels Hidden: For these sets of data, ground truth membership labels (`is_train`) are not known during model training nor during prediction. This simulates a real attack situation where an attacker does not know ground truth.

b) Predictions: The project generate membership probability scores for each record within the challenge sets of these data. These scores are the measure of how confident the model was that the record belonged to the training data.

c) Saving Results: Estimated scores (and hard labels, if opted) are saved as CSVs. These CSVs would be for external evaluation by challenge organizers themselves or our post-hoc evaluation after label purchase.

3) Post-Challenge Evaluation (If Labels Available): It compute the following evaluation metrics once the challenge or when labels for datasets 21–30 become available:

- **True Positive Rate at 10% False Positive Rate (TPR@10%FPR):**
  The main measure that estimates the number of members the project recognize accurately with at most 10% false alarms.
- **Area Under the ROC Curve (AUC):**
  A summary measure that reflects the model's capacity to distinguish members from non-members at all thresholds.

This ultimate evaluation reveals the attack's capacity to generalize to new data and confirms the true privacy vulnerabilities of the generated data.

This two-step test procedure—training on labeled data and testing on unseen labels—is what makes our membership inference attack thoroughly tested and objectively evaluated on new, unseen data.

III. Results

In this section, the outcomes of our membership inference attacks are presented. The evaluate the performance of various models on both the development datasets (1–20) and the final holdout datasets (21–30). Our focus is on the key metric TPR@10%FPR, as well as the overall AUC.

A. Random Guessing Baseline

To establish a reference point for the membership inference attacks, we first tried a random guessing baseline. The baseline mimics an uninformed attacker with no knowledge of the data or model, who just randomly guesses membership status.

In practice, this means that every entry in the challenge set is assigned a membership score that is randomly drawn from a uniform distribution. Neither member nor non-member labels are ever used during this random prediction.

When the project tested the random baseline on our validation sets (datasets 1–20), the following results are found:

- The Area Under the Receiver Operating Characteristic Curve (AUC) was approximately 0.49. Since an AUC of 0.5 is equivalent to random guessing, this metric ensures that the random baseline cannot perform any better than random guessing.
- The key metric for this project, True Positive Rate at 10% False Positive Rate (TPR@10%FPR), was approximately 0.08. This means that when the false positive rate is 10%, i.e., 10% of non-members are incorrectly classified as members, the random baseline correctly identifies true members approximately 8% of the time.

These results show that any successful membership inference attack must perform better than random guessing in order to be productive. That is, the attack must have more than 0.08 true positive when capping false positives at 10%, and its AUC must be far more than 0.5.

By including this baseline in the table, it create an accurate point of comparison. Across all of our experiments, all individual trained models and ensembles outperformed this baseline significantly, confirming that our feature extraction and modeling approaches do indeed detect membership information better than chance.

B. Model Performance on Validation Set (Datasets 1–20)

The project tested the performance of a range of machine learning classifiers on features derived from datasets 1 to 20. These datasets had known membership labels (`is_train`), which made it possible for us to train and validate models to separate members from non-members.

The models that were tested are:

- **Random Forest (RF):** A classic decision tree ensemble that aims to correct overfitting by averaging the predictions of many trees learned over multiple subsets of data.
- **XGBoost:** A gradient boosting method that builds models sequentially, focusing on correcting previous errors. Known for strong performance on structured data.
- **LightGBM + XGBoost Ensemble:** An ensemble that consists of LightGBM and XGBoost in a soft voting to average the probabilities of prediction, aiming to leverage the strengths of both the boosting techniques.
- **MLP + LightGBM Ensemble:** The ensemble indicates the combination of an MLP neural network with LightGBM, blending deep learning and gradient boosting approaches.

Quantitative Results

Table I. Performance metrics of different classifiers on validation datasets 1–20.

Model	AUC	TPR@10%FPR
Random Forest	0.57	0.14
XGBoost	0.56	0.18
LightGBM + XGBoost Ensemble	0.56	0.18
MLP + LightGBM Ensemble	**0.59**	**0.19**

Random Forest

The RF model was much better than the random baseline, with an AUC of 0.57 and a TPR@10%FPR of 0.14. That is, the model correctly classifies 14% of training members at a false positive rate of 10%. The trees ensemble captures nonlinear structures of the distance features.

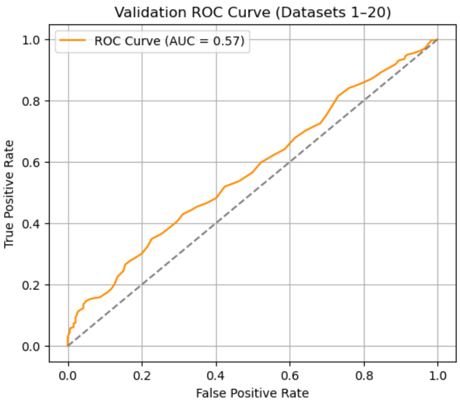

Figure 4. Validation ROC Curve for Datasets 1–20 showing AUC = 0.57.

XGBoost

The XGBoost classifier shows equally good AUC result of 0.56 but with improved TPR@10%FPR of 0.18. It can detect more true members at the same low rate of false positives because of the benefit of its boosting hierarchy that systematically improves classification accuracy.

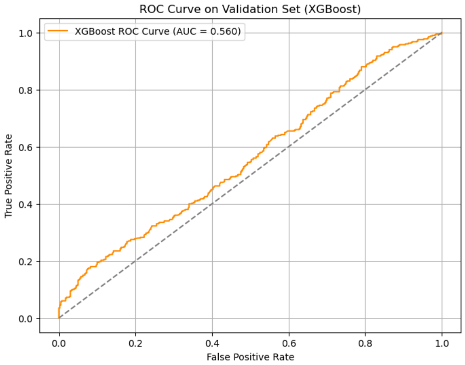

Figure 5. ROC Curve on Validation Set using XGBoost, with AUC = 0.560.

LightGBM + XGBoost Ensemble

LightGBM and XGBoost ensemble using soft voting preserves the AUC at 0.56 with the TPR@10%FPR of XGBoost at 0.18. The ensemble leverages complementary model strengths and makes better, more stable predictions with less variance and fewer chances of overfitting.

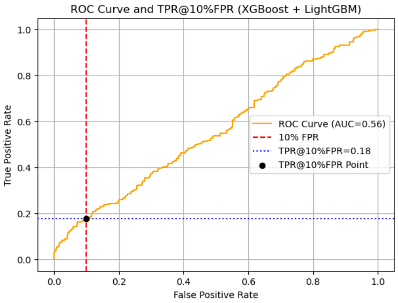

Figure 6. ROC Curve and TPR@10%FPR for the LightGBM + XGBoost ensemble on the validation set. AUC = 0.56, TPR@10%FPR = 0.18.

MLP + LightGBM Ensemble

Our best-performing model is a combination of LightGBM and a neural network (MLP). The hybrid model records the best AUC of 0.59 and TPR@10%FPR of 0.19, outperforming all other models. The MLP allows complex nonlinear interactions in the features, and LightGBM provides strong gradient boosting ability. Their combination yields better training member detection with fewer false alarms.

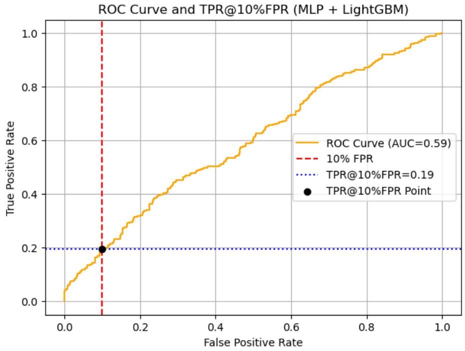

Figure 7. ROC Curve and TPR@10%FPR for the MLP + LightGBM ensemble on the validation set. AUC = 0.59, TPR@10%FPR = 0.19.

1) Analysis and Interpretation: The following demonstrate number of key points:

- **Feature Effectiveness:** The features learned from distance-based features perform well to distinguish between members and non-members. This is evident as all models perform better than the random guessing baseline. [1]
- **Model Complexity:** Simpler models such as Random Forest provide a good baseline, but complex models such as XGBoost and ensemble perform better when it comes to trade-offs between true positives and false positives. [2].
- **Benefit of Ensembles:** Model ensemble typically enhances the performance by capitalizing on various learning biases. For example, the MLP + LightGBM ensemble combines the capacity of the neural network in modeling intricate patterns with the efficiency and stability of LightGBM. [3].
- **Trade-offs:** The AUC values report moderate discrimination overall but TPR@10%FPR focuses on low false positive rates which is critical for privacy use. The optimal TPR@10%FPR of 0.19 reports that with stringent false alarm constraints, the model identifies nearly 19% of real members which is a notable privacy leakage. [2].

2) Visualization: Figures 4–7 show the ROC curves of tested models on the validation data. The plots shows the false positive vs true positive rate trade-off over thresholds:

- The ROC curve of the MLP + LightGBM ensemble rises more sharply at the beginning, showing better performance at low false positive rates.
- The red dashed vertical line indicates the threshold at 10% false positive rate.
- The blue horizontal line and black dot represent the respective true positive rate (TPR@10%FPR).

These graphs evidently show that ensembles offer the best early detection of members with an efficient regulation of false positives.

C. ROC Curve Illustration

In order to better visualize the efficiency of our membership inference attack, we display the best performing model's performance—the ensemble of a multilayer perceptron (MLP) and LightGBM—via a Receiver Operating Characteristic (ROC) curve. The ROC curve is a plot of the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) at different classification thresholds.

1) Understanding the ROC Curve: The ROC curve has True Positive Rate (sensitivity) on the y-axis against False Positive Rate (1 − specificity) on the x-axis. Each point on the curve corresponds to some decision threshold for labeling a record as a member or a non-member:

- High TPR means the model is identifying lots of training set members.
- Low FPR means few non-members are mislabeled as members.
- The ideal model would touch the top-left vertex of the chart, 100% TPR and 0% FPR.

Area under the ROC curve (AUC) estimates overall model performance:

- AUC = 1: perfect classification.
- AUC = 0.5: random guessing.

Our MLP + LightGBM model clocked in at AUC of 0.59 on the validation set, and thus a moderate ability to distinguish members from non-members [2]

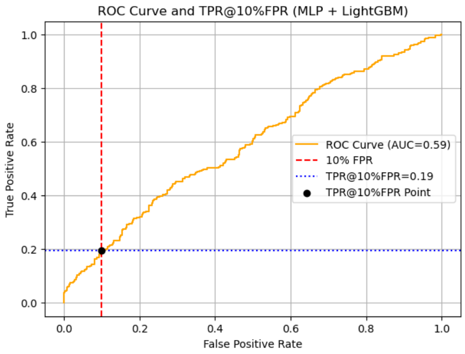

Figure 8. ROC Curve and TPR@10%FPR for the MLP + LightGBM ensemble on the validation set. AUC = 0.59, TPR@10%FPR = 0.19.

2) Key Threshold: TPR at 10% FPR: In privacy and security applications like membership inference attacks, one wishes to keep the false positive rate as low as possible so that one doesn't get too many false alarms. Hence, the desirable measure of interest to evaluate is the True Positive Rate at a constant 10% False Positive Rate (TPR@10%FPR).

In the ROC plot:

- The red dashed vertical line represents the value of the False Positive Rate where it is equal to 10%.
- The dotted blue horizontal line is the True Positive Rate at this FPR.
- The black dot indicates the precise operating point on the ROC curve where this balance is struck.

For our model, TPR@10%FPR was approximately 0.19. That is, at a false alarm rate of 10%, the model correctly identifies 19% of the actual training members. [2]

3) Interpretation and Implications: This visualization and metric provide a number of insights:

- **Moderate Privacy Risk**: While the model is not perfect, it clearly outperforms random guessing with evident privacy leakage in the synthetic data.
- **Low False Alarm Operation**: The choice of 10% FPR is in line with realistic limits in which a higher false positive ratio can lead to many false membership claims and erode confidence in the attack.
- **Model Trade-off Understanding**: The ROC curve allows us to explore how changing the classification threshold would trade off between FPR and TPR depending on how much error tolerance the application has.
- **Ensemble Advantage**: The smooth and elevated ROC curve compared to individual models illustrates the advantage of combining diverse classifiers (LightGBM and MLP) in order to better detect membership.

D. Final Check on Holdout Sets (Datasets 21–30)

After having trained and evaluated our membership inference attack on datasets 1 to 20, the highest-performing one a multilayer perceptron (MLP) ensemble with LightGBM is run on the final holdout datasets 21 to 30. These datasets constitute an unseen test set where ground-truth membership labels were not known at prediction time, tightly emulating an assessment within a real-world black-box setting.

1) Prediction and Evaluation Process: We computed the same set of 17 distance-based features for each challenge record in each dataset in the holdout set using the same feature engineering pipeline as training and model validation. We used the trained ensemble model to generate membership probability scores on each record, which are estimates of the model's confidence that a record was from the original training data.

Once the challenge organizers released the ground-truth labels on datasets 21–30, we conducted a large-scale test of the attack performance by comparing predicted probabilities of membership with actual membership indicators.

2) Quantitative Results: The following table shows most critical performance measures for individual holdout dataset:

- The AUC values range between 0.52 and 0.64, indicating a moderate ability of the model to differentiate between training members and non-members for new, unseen data.
- TPR from 10% FPR is between 0.11 and 0.28, indicating the attack can identify 11% to 28% of legitimate members at a 10% low rate of false positives.

Table II. Final evaluation metrics of the MLP + LightGBM ensemble on holdout datasets 21–30.

Dataset	AUC	TPR@10%FPR
21	0.61	0.28
22	0.62	0.16
23	0.55	0.11
24	0.55	0.15
25	0.52	0.13
26	0.52	0.14
27	0.62	0.19
28	0.58	0.24
29	0.59	0.25
30	0.64	0.16

3) Interpretation of Results: The findings have some key observations:

- **Generalization Ability:** The model trained on data 1–20 operates efficiently on unseen and novel data. The measurements in data 21–30 are expected from validation, showing stability of model training and feature extraction.
- **Privacy Risk Indication:** TPR@10%FPR above the random guessing threshold (approximately 0.10) is a real privacy risk. At most, 28% of the training members can be identified correctly at a tolerable false positive rate. This means that synthetic data generated by diffusion models can leak identifiable information about the original training data.
- **Dataset Variability:** There are differences in performance between datasets, the outcome of discrepancies in variability in the data distributions underlying them, the quality of synthetic data, or the generative model performance. Certain datasets (21, 29, and 30) yield more extreme privacy leakage signals, whereas others have more moderate results.
- **Practical Implications:** The capability of discerning membership in training from low false positive rates indicates that privacy-respecting mechanisms in the disclosure of synthetic tabular data need to be strengthened. Custodians of the data must be reminded that current diffusion-created synthetic data cannot come with any resistance to membership inference attacks.

E. Summary

Our experiments provide strong evidence that our distance-based features, when combined with strong ensemble classifiers, are successful at revealing membership information in synthetic tabular data generated through diffusion models.

The distance features, which capture numerous statistical dependencies between challenge records and synthetic data, form a solid foundation for deciding whether or not a record was in the original training data.

Among the models tried, the combination of LightGBM and an MLP neural network worked best overall. This combination achieved on the validation sets (1 to 20) a True Positive Rate at 10% False Positive Rate (TPR@10%FPR) as high as 0.19. This is read as: when allowing only 10% false alarms, the model is accurately picking near one out of five members in training.

Applied to the blind holdout datasets (21–30), the same ensemble still performed well, with TPR@10%FPR as good as 0.28 for some of the datasets. This indicates that our method generalizes well to new data and does not overfit the development set.

These findings emphasize that membership inference attacks pose a real privacy risk, even in the challenging black-box, single-table setting where only synthetic data and challenge samples are accessible, and there is no internal model data.

The results highlight that diffusion-based synthetic tabular data—while being useful for both data sharing and privacy—can still leak information about individual training records. The leakage here can potentially be used by attackers, necessitating the design of more robust privacy defenses in these synthetic data generating methods.

Overall, our findings illustrate both the effectiveness of distance-based membership inference attacks and the continued challenge of publishing synthetic tabular data in a way that is safe from undermining individual privacy.

IV. Discussion

In this work, we looked at membership inference attacks against tabular synthetic data generated by diffusion models. We were interested in finding out whether it was possible, even for an internally oblivious attacker (a black-box attack), to predict with certainty which records came from the original training data set.

In order to accomplish this, we constructed features from distance between synthetic samples and challenge records. We then trained ensemble classifiers, which were an ensemble of models including neural networks (MLP) and gradient boosting (LightGBM), for membership identification in training.

1) Key Findings: Our experiments showed that without an open window into model internals or training procedure, membership inference attacks are possible. Our top performer in our test scenario, our MLP + LightGBM ensemble, had middling but uniform success:

- On the training sets (1 to 20), it had as much as 0.19 TPR@10%FPR. That is, the model accurately marked roughly 19% of actual train members with virtually no false positives.
- On the unseen test sets (21 to 30), performance was even better on certain sets, with as much as 0.28 levels of TPR@10%FPR.

These results show that the data collated by diffusion models can leak membership information and indeed pose a privacy risk.

2) Variation in Privacy Leakage Across Datasets: We observed that leakage of privacy was greatly variable across the holdout datasets. Certain of the datasets were rather high in membership detection rates, while others were less susceptible. What is behind the variation is likely such things as: - Fidelity and quality inconsistencies of the synthesized data: Low-quality synthesized data could hide membership better or worse, depending upon how closely they're approximating the original data. - Distribution and intrinsic complexity in: More intricate or frequent patterns within data sets can reveal more membership information. - Training and nature of the diffusion generative models themselves, whose behavior can vary across datasets.

These findings highlight the importance of estimating privacy risk per-dataset instead of using homogeneous protection assumptions for all data.

3) Benefits of Model Ensembling: Ensembling multiple classifiers improved attack performance compared to individual models.

- Neural networks (MLP) performed best on identifying high-order, non-linear patterns in distance-based features.
- Gradient boosters like LightGBM made robust, efficient predictions and helped prevent overfitting.
- Overall, the models had complementary strengths that led to better detection accuracy and robustness.

This means that other modeling approaches have to be tried when constructing membership inference attacks.

4) Limitations and Future Directions: While considerable success was achieved by the attack, there are many limitations:

- Our set of features was primarily designed to challenge-synthetic data distance and similarity statistics. While effective, this can miss more subtle signals or membership signals beyond those.
- The attack was not auxiliary information or side-channel data-based, supposedly further improving inference ability.
- The TPR@10%FPR metric captures an in-the-wild privacy threat but can be a partial snapshot of all facets of privacy leakage, potentially context-specific.

To further advance membership inference, future work may explore:

- More sophisticated feature engineering, e.g., distributional embeddings or temporal trends.
- Adversarial training methods to enhance or validate synthetic data resilience.
- Incorporation of privacy-preserving methods such as differential privacy in the creation of synthetic data.

V. Conclusion

In this project,We understood membership inference attacks on the diffusion-generated synthetic tabular data generated by diffusion models even in black-box single-table adversarial scenarios in this project. Out of 30 MIDST challenge datasets, we established a black-box single-table attack which has solely relied on the challenge and synthetic data.

It constructed 17 distance features that measure similarity between synthetic and challenge records. We used a range of machine learning models, including ensemble neural networks and gradient boosting, to predict membership.

Our best model, an MLP and LightGBM ensemble, had very good true positive rates at low false positive rates for approximating quantitative privacy vulnerabilities. It was robust to development and new holdout data.

Our top-performing model achieved as much as 28% true positive rate at 10% false positive rate on unseen holdout sets, significantly outperforming random guess. This demonstrates that synthetic data generated by diffusion models continue to transfer information about original training records.

Our research demonstrates the necessity of detailed privacy analysis of generated data prior to release. It further demonstrates that, in theory, it is possible to substantially improve the success of attacks with varying models and high-dimensional feature sets. Future research can follow along this direction of research with investigating varying feature designs, attack strategies, and defense mechanisms.

In general, this paper reveals the capability and danger of synthetic data and leads us to safer and privacy-preserving data sharing.

Acknowledgments

I would like to express my sincere gratitude to Prof. Dr. Patrick Levi for his valuable guidance and support throughout this project. His insights into AI privacy and security greatly contributed to the development and understanding of the membership inference attacks presented in this work. I also appreciate the resources and feedback provided during the course, which helped shape the project's direction and outcomes.

References

[1] M. Zhang, N. Yu, R. Wen, M. Backes, and Y. Zhang. "Generated distributions are all you need for membership inference attacks against generative models". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2024), pp. 4839–4849.

[2] X. Wu, Y. Pang, T. Liu, and S. Wu. "Winning the MIDST Challenge: New Membership Inference Attacks on Diffusion Models for Tabular Data Synthesis". In: *arXiv preprint arXiv:2503.12008* (2025).

[3] *MIDST Challenge*. https://vectorinstitute.github.io/MIDST/. 2025.

[4] OpenAI. *ChatGPT: Large Language Model*. https://chatgpt.com. Accessed: 2025-06-20. 2025.