Downstream Task-Oriented Generative Model Selections on Synthetic Data Training for Fraud Detection Models

Yinan Cheng*, Chi-Hua Wang †,
Vamsi K. Potluru ‡, Tucker Balch §, Guang Cheng¶

January 3, 2024

Abstract

Devising procedures for downstream task-oriented generative model selections is an unresolved problem of practical importance. Existing studies focused on the utility of a single family of generative models. They provided limited insights on how synthetic data practitioners select the best family generative models for synthetic training tasks given a specific combination of machine learning model class and performance metric. In this paper, we approach the downstream task-oriented generative model selections problem in the case of training fraud detection models and investigate the best practice given different combinations of model interpretability and model performance constraints. Our investigation supports that, while both Neural Network(NN)-based and Bayesian Network(BN)-based generative models are both good to complete synthetic training task under loose model interpretability constrain, the BN-based generative models is better than NN-based when synthetic training fraud detection model under strict model interpretability constrain. Our results provides practical guidance for machine learning practitioner who is interested in replacing their training dataset from real to synthetic, and shed lights on more general downstream task-oriented generative model selection problems.

Key Words: Generative Model Selections, Synthetic Training Datasets, Fraud Detection, Accuracy-Interpretability Tradeoff, Data-centric Machine Learning, Train on Real Test on Synthetic.

*Dept of Statistics, UC Davis. Email: yncheng@ucdavis.edu
†Department of Statistics and Data Science, Email: chihuawang@ucla.edu
‡J.P. Morgan AI Research
§J.P. Morgan AI Research
¶Department of Statistics and Data Science. Email: guangcheng@ucla.edu

1 Introduction

Synthetic Data is receiving increasing attention from both academics and industry [24]. Such attention is due to synthetic data's potential to accelerate innovation and support decision-making without violating modern privacy regulations (e.g. GDPR [1] or CCPA). The potential for acceleration and support of the modern machine learning lifecycle is lucrative and hence invites both machine learning researchers and practitioners investigation on the benefits and limitations of using synthetic datasets [41, 16]. In particular, **what is the prize and price on *machine learning model training process* if we replace from real training dataset with a synthetic training dataset?**

Indeed, the prize is individual-level privacy protection, and the price is performance degradation. Although the current community does not reach an agreement on how effective the synthetic data approach [40], existing studies [22] observe performance degradations from replacing real with the synthetic training datasets. In practice, there are three family of generative models to synthesize training dataset: Machine Learning-based [17, 10, 15], Neural network-based [32, 44] and Bayesian network-based [46, 19, 3] generative models. Due to the predicament of striking a balance between accuracy and privacy leakage for Machine Learning-based generative models [24], in this paper, we focus on Neural network-based and Bayesian network-based generative models. Given primitive privacy protection by replacing the real training dataset with the synthetic dataset, the following question guides our investigation: **Which family of synthetic data generative models sufferers least performance degradation?** We define the above key research question as the **Generative Model Selection (GMS)** problems. In particular, we investigate the generative model selection problem in the context of the fraud detection model (FDM) training procedure.

The fraud detection model is an integral part of the modern fraud management process [42, 2, 37], where the utility-interpretability trade-off is of particular importance to stop fraudsters to limit fraud impact in financial service operations [31]. The utility is the key performance metric to measure how effective the trained fraud detection model detecting potential fraudulent operations. However, FDM with high utility also incurs a high number of false alerts, leading to potential high human resource cost[31]. In practice, those alerts reported by FDM are reviewed by fraud experts, demanding model interpretability to make decisions on free or suspend the potential fraudulent operation. Consequently, **insights on the impact of synthetic dataset trained FDM, especially around the utility-interpretability trade-off, are desired and of practical importance**. In answering GMS problems in synthetic dataset trained FDM, this paper aims to solve the following two questions:

- **Utility-oriented GMS**: *Given a cost-specific metric*, which family of generative models suffers the least degradation?
- **Interpretability-oriented GMS**: *Given a specific interpretability constraint*, which family of generative models suffers the least degradation?

1.1 Contributions

In this paper, we adopt a downstream task-oriented approach to the evaluation of generative models introduced in section 2.1. Instead of assessing the generative data distribution by looking into the correlation or distance between real and synthetic datasets, we examine the performance of fraud detection models trained on Neural network-based or Bayesian Network-based synthetic training data. Our evaluation is from 3 dimensions: data (Synthetic Data generative models), models (fraud detection models), and metrics (accuracy, AUROC, recall, precision). Our examination focus on two class of generative model selection problems: Utility-oriented generative model selections and Interpretability-oriented generative model selections.

We provide 3 guidance for fraud data scientists and machine learning practitioners interested in the utility-interpretability behind using synthetic training data to train their fraud detection models

- We systematically compare different performance metrics for fraud detection models to give the best advice on different priorities to stop fraudsters.
- We systematically compare different machine learning model candidates for fraud detection models from the layer of interpretability.
- We give guidance to select synthetic data generative models for different model-metric combinations.

Utility-oriented Generative Model Selections The key question of the Utility-oriented generative model selection problem is on which generative model should be used to generate synthetic training data to train fraud detection under *given metric*. We found the best choice of the generative model family is *metric-dependent*. In short, we provide *insights on Utility-oriented GMS.(Section 4.2)*

- Accuracy did not show a preference between neural network-based (NN-based) and Bayesian network-based (BN-based) models.
- AUROC and Recall prefer NN-based generative models.
- F1 score and Precision prefers BN-based generative models.

Interpretability-oriented Generative Model Selections On the other hand, the interpretability-oriented generative model selection problem asks for the best family of generative models to generate training data *given fraud detection model class*. We found that the Bayesian Network-based method is better for an intrinsic interpretable model class, while both generative model families are good for the complex model classes. In short, we provide *insights on Interpretability-oriented GMS. (Section 4.3).*

- Intrinsic interpretable model class and medium interpretable model class prefers BN-based generative models.
- Not-Easy interpretable model class shows no preference between NN-based and BN-based models.

1.2 Paper Organization

This paper is structured as follows. Section 2 gives a comprehensive reviews on related research communities across synthetic data generative models (Section 2.1), fraud detection model interpretability (Section 2.2) and fraud detection metric utility (Section 2.3). Section 3 gives experiment details on how to synthesize training dataset (Section 3.1), choices of fraud detection model class (Section 3.2) and different fraud detection utility metrics (Section 3.3). Section 4 reports result of our evaluation on how generative models has effect on the imbalance of real training dataset (Section 4.1), and results on Utility-oriented GMS (Section 4.2) and Interpretability-oriented GMS (Section 4.3). Section 5 talks about our conclusion, new concerns and future direction.

2 Relate Work

2.1 Synthetic Data Generative Models

Neural Network-based generative models. *Synthetic Data Vault* (SDV) [33] provides three neural network-based generative models to synthesize data from a single table. Generative adversarial networks (GANs) are commonly used tools for fraud detection synthetic data due to their ability to address imbalanced datasets via data augmentation [11, 26, 27]. SDV provides a GAN-based generative model proposed in [43], namely conditional tabular GAN (CTGAN). CopulaGAN, another GAN-based model included in SDV, is a variation of the CTGAN Model which takes advantage of the cumulative distribution function (CDF) based transformation. Another type of neural network-based generative model is based on variational autoencoders (VAEs) [25]. SDV also provides a VAE-based generative model, namely, TVAE [43], to synthesize tabular data. Besides neural network-based generative models, SDV offers GaussianCopula [29] to model the covariances between features in addition to the distributions [28].

Bayesian Network-based generative models. *DataSynthesizer* (DS) [34] has three modes to invoke modules: random mode, independent attribute mode, and correlated attribute mode. The correlated attribute mode uses the GreedyBayes algorithm to construct Bayesian networks to model correlated attributes, which helps to retain the correlation among variables. Another important parameter in DS is epsilon which represents differential privacy to address data protection and privacy issues. When epsilon approaches 0, the presence or absence of a single case in the input will be undetectable in the output.

2.2 Fraud Detection Model Interpretability

(i) Intrinsic Interpretable Model class. Logistic Regression [8] provides predictions based on the estimated probability of an event occurring. A transaction will be predicted as a fraud if its estimated probability of being a fraud passes some threshold. Decision Tree [9] is one of the non-parametric supervised learning models used for classification tasks. It predicts classes of transactions via decision rules for data features. K-nearest Neighbors (KNN) [7, 18] is another commonly used non-parametric supervised learning method. Under

the assumption that similar points can be found near each other, it uses proximity to make predictions about the class of a transaction.

(ii) Medium Interpretable Model class. Naïve Bayes [45] is a probabilistic classifier based on Bayes' Theorem with the assumption of conditional independence between each pair of data features. The predicted class of a transaction is with the maximum probability. Support Vector Machine (SVM) [13] is designed to find the hyperplane in an n-dimensional space (n is the number of features) that distinctly classifies data points. Random forest [23] is one of the ensemble learning methods. It is composed of abundant decision trees and aggregates all predicted classes of decision trees to identify the most popular result as the prediction.

(iii) Not-Easy Interpretable Model class. The generalized Additive Model (GAM) [21] is a generalized linear model in which the response is linearly dependent on smooth functions. The smooth relationships between the response and each feature can be estimated simultaneously, and the response in test data can be predicted by adding them up. For binary classification, a logit link is applied to data fitting. Xgboost [12] is a scalable, distributed gradient boosting system under the Gradient Boosting framework. It provides parallel tree boosting and improves computational efficiency and model performance. Neural Additive Model (NAM) [4] focuses on a linear combination of deep neural networks that each has a single input feature. The model is fitted via training jointly these neural networks and learning complicated relationships between their inputs and outputs.

2.3 Fraud Detection Metric Utility

Fraud Management Process. Fraud management process [42] has achieved great success in financial service industries due to its capability to catch fraudulent transactions [31]. As the fraud detection models flag some transaction to be suspicious, it may cause interpretability and the fraud agent have difficulty telling whether the flag is a false alert or not. The fraud detection interpretability issues exist in all kinds of financial service applications including credit or debit cards, payments, and loan approval.

One major reason behind this communication bottleneck in the fraud management process is the interpretability of detection. As the fraud data scientist wishes to catch as more suspicious operations as possible, the fraud detection model itself becomes more difficult to interpret by using a more complex model class, e.g. neural network-based model. Consequently, the alerted operations become difficult to interpret, and the fraud experts need to take more time to identify the fraud factor, leading to such a communication bottleneck. Such a late decision allows the fraudsters to maximize their fraud fain, which greatly impairs the effectiveness of the model and also results in enormous economical and opportunity loss. As such, the interpretability in fraud detection model machine learning has become a rising concern.

In order to trade off the model performance and model interpretability, existing works are all focused on a model-centric approach. The model-centric approach takes the mindset of "accuracy first, interpretability later". They first compare the performance of different classes of fraud detection models. Then adopts the model's nature to gain the model interpretability. In addition to the intrinsic interpretable model class, there are various methods proposed to do post-doc methods to gain model-specific interpretability. The main drawback

of the model-centric approach is the lack of a fair way to compare model interpretability in the same framework.

Performance Metrics. We evaluate synthetic data training of FDM with metrics for standard classifier and for imbalanced dataset trained classifier. **(i) Metrics for standard classifier.** Accuracy [35] is one of the commonly used metrics in machine learning tasks. It is useful when all classes are of equal importance, but it can be misleading for an imbalanced dataset. AUROC [20] describes the model's ability to discriminate between positive cases and negative cases. F1 score [38] measures the performance of a model by computing the harmonic mean of the precision and recall of the model. **(ii) Metrics for imbalanced dataset trained classifier.** *Recall* and *precision* [35, 6] are two metrics which are more suitable than accuracy for imbalanced testing datasets. There is an inverse relationship between precision and recall usually. Precision-Recall curve [36] shows the tradeoff between precision and recall for different thresholds. Average precision [39] summarizes such a tradeoff as the weighted mean of precisions achieved at each threshold, where the weight is the increase in recall from the prior threshold.

3 Experiment Setup

3.1 Training Data Synthesis

Dataset-Credit Card Fraud Dataset. We conduct the experiment on the Credit Card Fraud Dataset [14] which is highly imbalanced (0.1727% transactions are frauds). The dataset contains 31 variables. The feature "Amount" is the transaction amount, and the feature "Time" represents the seconds elapsed between each transaction and the first transaction in the dataset. "Class" is the response variable and it takes 1 for fraudulent cases and 0 for other cases. Due to confidentiality, the remaining 28 features are principal components obtained by the means of PCA. In our experiment, we delete the feature "Time" since it has nothing to do with "Class".

Generating Synthetic Training Dataset. SDV is applied with GaussianCopula, CTGAN, CopulaGAN and TVAE, to generate synthetic data. We employ the four generative models on the original dataset respectively and obtain 4 synthetic datasets. DS is utilized with the correlated attribute mode. Synthetic datasets are generated with two generative models. One is with epsilon = 0, and the other is with epsilon = 0.1.

Since the original dataset is highly imbalanced, some synthetic data generative models cannot yield transactions with class 1 (fraudulent cases). It is found that no fraudulent transactions are synthesized when using GaussianCopula and CopulaGAN. Therefore, we leave out the datasets generated by GaussianCopula and CopulaGAN in the following tasks.

3.2 Choice of Fraud Detection Model Class

Synthetic Data Training Procedure. After generating synthetic datasets of the same size as the original dataset, we randomly split the original dataset into training data (70% of the original data) and test data (30% of the original data), and randomly select 70% of each

synthesized dataset as the synthesized training data. Then we fit fraud detection models to the training data and each synthesized training data and predict the results on the test data.

Model class by Interpretability. We fit 9 fraud detection models to analyze utility-interpretability on FDM training tasks. The following list summarizes 9 models class from high to low interpretability:

- **Intrinsic interpretable models**: Logistic Regression (LR), Decision Tree (DT) and K-nearest Neighbors (KNN).
- **Medium interpretable models**: Naïve Bayes (NB), Support Vector Machines (SVM), Random Forest (RF).
- **Not-easy interpretable models**: Generalized Additive Model (GAM), Extreme Gradient Boosting (XGBoost), Neural Additive Model (NAM).

3.3 Choice of Fraud Detection Utility Metrics

To evaluate the performance of synthesized data on fraud detection, we compute accuracy, AUROC, recall, precision, and F1 score, and generate the Precision-Recall curve and average precision (AP) for each machine learning task based on the predicted results.

Basic Performance Metrics for Classifier.

- **Accuracy** is the ratio of the number of correct predictions to the number of all predictions. It describes how the model performs across both classes.
- **AUROC** measures the ability of a classifier to distinguish between the fraudulent class and the nonfraudulent class.
- **F1 score** is the harmonic mean between precision and recall. It provides equal importance to precision and recall.

Performance Metrics for Classifier on Imbalanced Dataset.

- **Precision** is defined as the ratio of true positives to the sum of true positives and false positives. It is the number of correctly predicted fraudulent cases divided by the total number of predicted fraudulent cases for fraud detection.
- **Recall** is defined as the ratio of true positives to the sum of true positives and false negatives. It is the number of correctly predicted fraudulent cases divided by the total number of actual fraudulent cases in the fraud detection scenario.
- **Precision-Recall curve** shows the tradeoff between precision and recall for different thresholds. A high area under the curve represents both high recall and high precision.
- **Average precision** is calculated as the weighted mean of precisions at each threshold, where the weight is the increase in recall from the prior threshold.

Table 1: Percentage of classes

Approach	Class 1 (frauds)	Class 0
Original	0.1727%	99.8273%
CTGAN	59.4919%	40.5081%
TVAE	0.0119%	99.9881%
DS 0	0.1731%	99.8269%
DS 0.1	45.4743%	54.5257%

4 Evaluation

4.1 Comparison of Original to Synthetic Data

(1) Balance. Table 1 shows the degree of imbalance of synthetic training datasets. Datasets generated by CTGAN and DS with epsilon = 0.1 are balanced (59.4919% and 45.4743% transactions are frauds respectively). TVAE synthesizes a more imbalanced dataset (0.0119% transactions are frauds) than the original dataset. When using DS with epsilon = 0, the synthesized dataset contains 0.1731% fraudulent transactions, which is similar to the original dataset.

(2) Correlation. Dataset generated by DS with epsilon = 0 maintains the correlation of each pair of two features. However, datasets synthesized by other generative models do not retain the correlation between features.

4.2 Results on Utility-oriented GMS

In order to compare the performance of the original data and synthesized data in fraud detection tasks, we generate line charts to show the results of each metric, for data generated with different approaches as models' interpretability increases.

(1) Accuracy. Since the original dataset is highly imbalanced, the accuracy on test data is very close to 1 for each fraud detection model. In Figure 1, *training dataset generated by TVAE, DS with epsilon 0 and DS with epsilon 0.1 also yield high accuracy.* The accuracy of data synthesized by DS is even higher than the accuracy of the original data for Naïve Bayes. CTGAN has lower accuracy than other approaches for all fraud detection models.

It is remarkable that **accuracy of Naïve Bayes is highly unstable across different synthetic training datasets**, while other fraud detection models are with stable accuracy except for CTGAN-based training data. Although DS with epsilon 0.1 has the highest accuracy for Naïve Bayes, its accuracy for Decision Tree is the second lowest. Compared with the original data, data synthesized by DS with epsilon 0 has almost the same or higher accuracy.

Hence, *DS with epsilon 0 performs the best in accuracy overall for fraud detection tasks.* Bayesian network-based generative models are selected by accuracy metric due to their better performance in Figure 1, where both Bayesian network-based generative models outstrip neural network-based generative models.

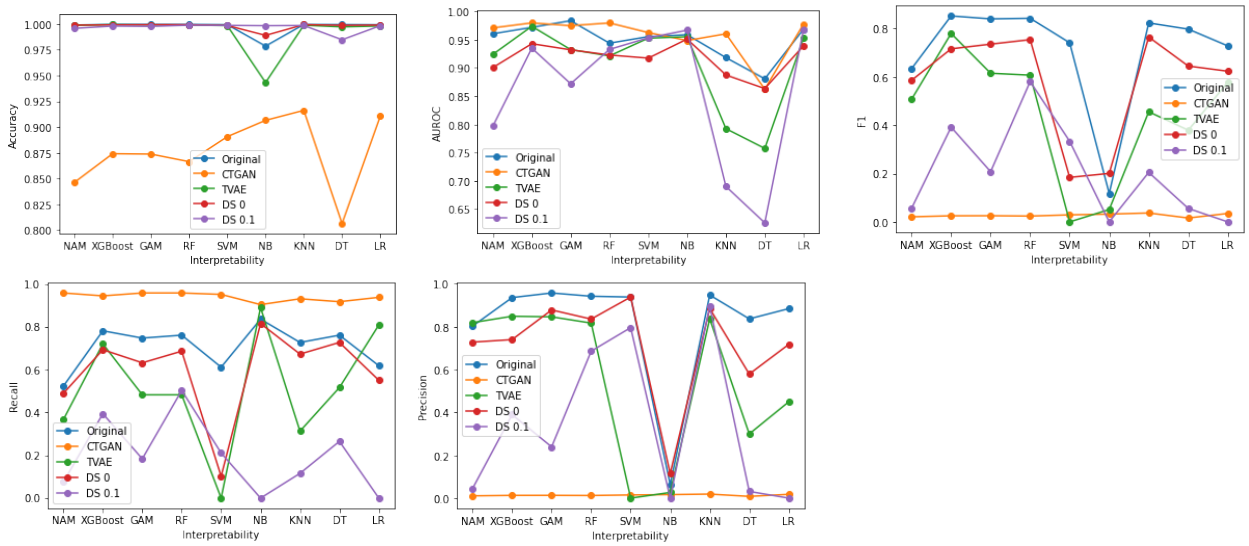

Figure 1: Results to solve Utility-oriented GMS: Utility Metrics for Fraud Detection Classifiers

(2) AUROC. For AUROC, the original data performs the best for GAM and Decision Tree. DS with epsilon 0.1 improves the performance for Naïve Bayes, CTGAN and TVAE improve the results for XGBoost, and only CTGAN increases the value of AUROC for the other 5 fraud detection models. DS with epsilon 0 produces smaller AUROC than the original data for all models. **Naïve Bayes and Logistic Regression show robustness while the result of AUROC varies considerably among data generative approaches for KNN and Decision Tree**. Figure 1 indicates that CTGAN is a suitable data generative technique to synthesize data considering the performance of AUROC, but for Naïve Bayes, DS with epsilon 0.1 is preferred.

In the fraud detection scenario, *Naïve Bayes has better discrimination between fraudulent transactions and nonfraudulent transactions when the training data is generated by DS with epsilon 0.1.* When the training data is synthesized by CTGAN, Logistic Regression, KNN, SVM, Random Forest, XGBoost and NAM perform better at distinguishing between fraudulent cases and nonfraudulent cases. Thus, **AUROC prefers neural-network based generative models, especially CTGAN which achieves comparable performance to the original data**.

(3) F1 score. According to Figure 1, *only DS with epsilon 0 has higher F1 score than the original for Naïve Bayes.* For other fraud detection models, the original data has the best performance, especially for SVM for which there is a noticeable difference in F1 score between the original data and synthesized data. In addition, it is found that *F1 scores of CTGAN are almost 0 for all fraud detection models.* Even though F1 scores are relatively similar for Naïve Bayes, none of the models show robustness for all datasets. Data generated by DS with epsilon 0 yields the highest F1 score, but it seems doubtful that this approach is suitable for Naïve Bayes because all F1 scores, including the F1 score of DS with epsilon 0, are very low for Naïve Bayes.

Combining recall and precision, DS with epsilon 0 improves the performance of Naïve

Bayes for fraud detection. Other fraud detection models show better performance when they are trained by the original dataset. In conclusion, **F1 score selects Bayesian network-based generative models since overall DS with epsilon 0 has the best performance among all generative techniques**.

(4) Recall. *CTGAN has considerably higher recall* than other approaches in Figure 1. Compared with the original dataset, data generated by TVAE also increases recall scores for Naïve Bayes and Logistic Regression while DS with epsilon 0 and DS with epsilon 0.1 perform worse than the original for all fraud detection models. *Only Naïve Bayes is robust for recall when leaving out DS with epsilon 0.1.* Because of the remarkable performance, **CTGAN is the suitable technique to generate synthetic data when we focus on recall scores**.

All machine learning models yield more correctly predicted frauds over total actual frauds in the fraud detection tasks when the training data is synthesized by CTGAN. More fraudulent cases are recognized properly when using CTGAN to generate the training data. Therefore, **neural network-based generative models are preferred by recall** because CTGAN surpasses all of the other methods and even distinctly exceeds the original data.

(5) Precision Compared with the original, TVAE increases the precision score a little for NAM, and DS with epsilon 0 improves the performance in precision for Naïve Bayes in Figure 1. Data synthesized by DS with epsilon 0 produces the same precision score to the original data for SVM. We can see that DS with epsilon 0.1 has worse performance than the original for all fraud detection models, and all precision scores of CTGAN are close to 0. Except for Naïve Bayes, all models show differences in precision scores among approaches. For NAM, TVAE is a suitable method for data generation, and for SVM and Naïve Bayes, DS with epsilon 0 can synthesize data comparable to the original data when focusing on precision. There are no synthetic data generative models with comparable precision to the original dataset for other machine learning models.

For fraud detection, NAM can correctly predict more frauds among all predicted fraudulent cases when the training data is generated by TVAE. Naïve Bayes has more correct predictions for cases predicted to be fraudulent when the training data is synthesized by DS with epsilon 0. Other fraud detection models produce a higher ratio of correct predicted frauds to all fraudulent predictions when it is trained by the original data. Among all data generative tools, DS with epsilon 0 has the best performance overall and CTGAN performs the worst, so **precision prefers Bayesian network-based generative models**.

4.3 Results on Interpretability-oriented GMS

The original dataset yields Precision-Recall curves in the top right corner and the highest average precision for all machine learning models except for Naïve Bayes according to Figure 2. DS with epsilon 0.1 has the Precision-Recall curve in the top right corner and the highest average precision for Naïve Bayes, followed by DS with epsilon 0. For Logistic Regression, TVAE produces the second highest average precision, and the Precision-Recall curve is just lower than the original. CTGAN and TVAE have Precision-Recall curves just lower than the original and the same average precision for XGBoost. For the other 6 fraud detection models, DS with epsilon 0 yields the second highest average precision, and the Precision-Recall curve is just lower than the original.

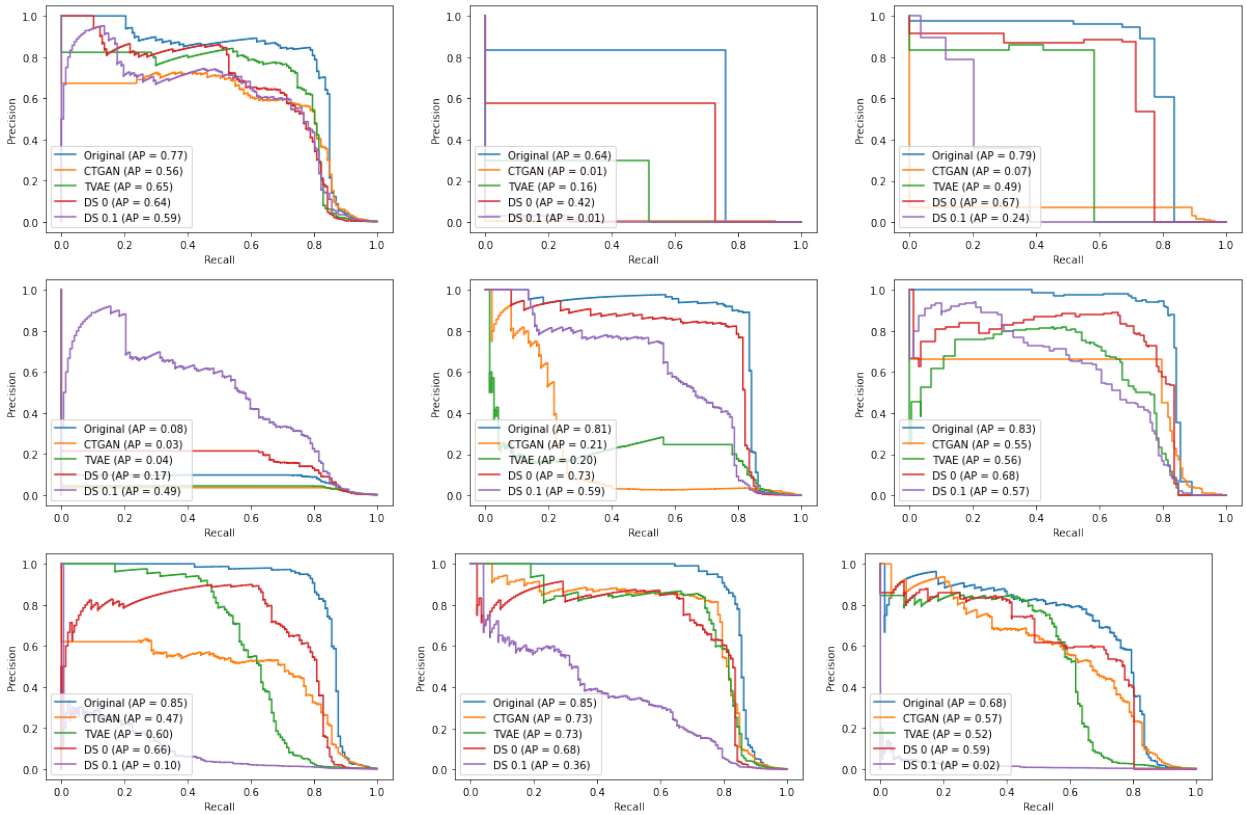

Figure 2: Results to solve Interpretability-oriented GMS: Precision-Recall curve and Average Precision

(i) Intrinsic Interpretable Model class. In the first row of Figure 2, there is no noticeable difference between neural network-based models and Bayesian network-based models in Precision-Recall curves for Logistic Regression. So, Logistic Regression indicates that neural network-based and Bayesian network-based generative models have comparable performance. Decision Tree and KNN suggest Bayesian network-based generative models as better tools for synthetic data generation.

(ii) Medium Interpretable Model class. In the second row of Figure 2, **all medium interpretable models select Bayesian network-based generative models to synthesize data**. Bayesian network-based generative models even have a better performance than the original for Naïve Bayes. Although Bayesian network-based generative methods perform better for all medium interpretable machine learning models, the difference is that Naïve Bayes prefers DS with epsilon 0.1 while SVM and Random Forest prefer DS with epsilon 0.

(iii) Not-Easy Interpretable Model class. The third row of Figure 2 presents the parallel performance of Bayesian network-based and neural network-based generative models for not-easy interpretable fraud detection models. For GAM, TVAE and DS with epsilon 0 show comparable performance. For XGBoost and NAM, CTGAN, TVAE and DS with epsilon 0 have similar Precision-Recall curves. Curves for DS with epsilon 0.1 are located in the bottom left corner for all not-easy interpretable models.

4.4 Results on Synthetic Augmented Training

By *synthetic augmented training*, we mean the training Dataset for ML classifier is a mixture of source real dataset and certain percentage of synthetic data. In particular, we consider five differen degree of such real-synthetic mixture: `syn0.1, syn0.2, syn0.3, syn0.4, syn0.5`. For example, `syn0.2` means the training dataset is the mixture of 100% source real dataset and 20% of synthetic dataset from the synthesizer.

The full empirical results on Synthetic Augmented Training from metric-oriented and synthesizer-oriented are given at section A. We summarize key insights below. (1) `CTGAN`-augmented training datasets improves the AUROC and recall of synthetic trained ML classifier. However, `CTGAN`-augmented training datasets damages accuracy, F1 score, precision and precision-recall curve across all synthetic trained ML classifier. (2) `TVAE`-augmented training datasets in general do not improve or damage the utility of synthetic trained ML classifier across all 6 performance performance metrics and all 9 ML Classifier. (3) `PrivBayes`-augmented training datasets improves the accuracy of synthetic trained ML classifier. However, `PrivBayes`-augmented training datasets damages AUROC, F1 score, recall, precision across all synthetic trained ML classifier.

5 Conclusion

In this paper, we provide a practical evaluation of generative model selections for synthetic training of fraud detection models. Our evaluation framework covers data, models, and metrics and provides results to answer utility-oriented generative model selections and interpretability-oriented generative model selections.

One promising future work direction is to develop *generative model auditing process* for generative models and their synthetic datasets. Such model auditing process has been explored in the domain-agnostic and model-agnostic way [5], but more task-oriented studies on model generative model auditing process are in demand. Indeed, such a generative model auditing process will generate values for data scientists and machine learning practitioners in integrating synthetic datasets into their daily workflow and leading a more trustworthy machine learning lifecycle in the near future.

References

[1] 2018 reform of eu data protection rules.

[2] Aisha Abdallah, Mohd Aizaini Maarof, and Anazida Zainal. Fraud detection system: A survey. *Journal of Network and Computer Applications*, 68:90–113, 2016.

[3] John M Abowd and Lars Vilhuber. How protective are synthetic data? In *International Conference on Privacy in Statistical Databases*, pages 239–246. Springer, 2008.

[4] Rishabh Agarwal, Nicholas Frosst, Xuezhou Zhang, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. *arXiv preprint arXiv:2004.13912*, 2020.

[5] Ahmed Alaa, Boris Van Breugel, Evgeny S Saveliev, and Mihaela van der Schaar. How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models. In *International Conference on Machine Learning*, pages 290–306. PMLR, 2022.

[6] Kent Allen, Madeline M Berry, Fred U Luehrs Jr, and James W Perry. Machine literature searching viii. operational criteria for designing information retrieval systems. *American Documentation (pre-1986)*, 6(2):93, 1955.

[7] Naomi S Altman. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185, 1992.

[8] Joseph Berkson. Application of the logistic function to bio-assay. *Journal of the American statistical association*, 39(227):357–365, 1944.

[9] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. *Classification and regression trees*. Routledge, 2017.

[10] Gregory Caiola and Jerome P Reiter. Random forests for generating partially synthetic, categorical data. *Trans. Data Priv.*, 3(1):27–42, 2010.

[11] Charitos Charitou, Simo Dragicevic, and Artur d'Avila Garcez. Synthetic data generation for fraud detection using gans. *arXiv preprint arXiv:2109.12546*, 2021.

[12] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[13] Corinna Cortes and Vladimir Naumovich Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 2004.

[14] Andrea Dal Pozzolo, Olivier Caelen, Reid A Johnson, and Gianluca Bontempi. Calibrating probability with undersampling for unbalanced classification. In *2015 IEEE symposium series on computational intelligence*, pages 159–166. IEEE, 2015.

[15] Jörg Drechsler. Using support vector machines for generating synthetic datasets. In *International Conference on Privacy in Statistical Databases*, pages 148–161. Springer, 2010.

[16] Khaled El Emam, Lucy Mosquera, and Richard Hoptroff. *Practical synthetic data generation: balancing privacy and the broad availability of data*. O'Reilly Media, 2020.

[17] Josh Eno and Craig W Thompson. Generating synthetic data to match data mining patterns. *IEEE Internet Computing*, 12(3):78–82, 2008.

[18] Evelyn Fix and Joseph Lawson Hodges. Discriminatory analysis. nonparametric discrimination: Consistency properties. *International Statistical Review/Revue Internationale de Statistique*, 57(3):238–247, 1989.

[19] Grigoriy Gogoshin, Sergio Branciamore, and Andrei S Rodin. Synthetic data generation with probabilistic bayesian networks. *Mathematical biosciences and engineering: MBE*, 18(6):8603, 2021.

[20] James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

[21] Trevor Hastie and Robert Tibshirani. Generalized additive models: some applications. *Journal of the American Statistical Association*, 82(398):371–386, 1987.

[22] Markus Hittmeir, Andreas Ekelhart, and Rudolf Mayer. On the utility of synthetic data: An empirical evaluation on machine learning tasks. In *Proceedings of the 14th International Conference on Availability, Reliability and Security*, pages 1–6, 2019.

[23] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.

[24] James Jordon, Lukasz Szpruch, Florimond Houssiau, Mirko Bottarelli, Giovanni Cherubin, Carsten Maple, Samuel N Cohen, and Adrian Weller. Synthetic data–what, why and how? *arXiv preprint arXiv:2205.03257*, 2022.

[25] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[26] A Langevin, T Cody, S Adams, and P Beling. Synthetic data augmentation of imbalanced datasets with generative adversarial networks under varying distributional assumptions: A case study in credit card fraud detection. *Journal of the Operational Research Society*, pages 1–28, 2021.

[27] Alex Langevin, Tyler Cody, Stephen Adams, and Peter Beling. Generative adversarial networks for data augmentation and transfer in credit card fraud detection. *Journal of the Operational Research Society*, 73(1):153–180, 2022.

[28] Majlinda Llugiqi and Rudolf Mayer. An empirical analysis of synthetic-data-based anomaly detection. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pages 306–327. Springer, 2022.

[29] Guido Masarotto and Cristiano Varin. Gaussian copula marginal regression. *Electronic Journal of Statistics*, 6:1517–1549, 2012.

[30] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, 2019.

[31] Anna Nesvijevskaia, Sophie Ouillade, Pauline Guilmin, and Jean-Daniel Zucker. The accuracy versus interpretability trade-off in fraud detection model. *Data & Policy*, 3, 2021.

[32] Noseong Park, Mahmoud Mohammadi, Kshitij Gorde, Sushil Jajodia, Hongkyu Park, and Youngmin Kim. Data synthesis based on generative adversarial networks. *arXiv preprint arXiv:1806.03384*, 2018.

[33] Neha Patki, Roy Wedge, and Kalyan Veeramachaneni. The synthetic data vault. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 399–410. IEEE, 2016.

[34] Haoyue Ping, Julia Stoyanovich, and Bill Howe. Datasynthesizer: Privacy-preserving synthetic datasets. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, pages 1–5, 2017.

[35] David MW Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*, 2020.

[36] Vijay Raghavan, Peter Bollmann, and Gwang S Jung. A critical investigation of recall and precision as measures of retrieval system performance. *ACM Transactions on Information Systems (TOIS)*, 7(3):205–229, 1989.

[37] Nick F Ryman-Tubb, Paul Krause, and Wolfgang Garn. How artificial intelligence and machine learning research impacts payment card fraud detection: A survey and industry benchmark. *Engineering Applications of Artificial Intelligence*, 76:130–157, 2018.

[38] Yutaka Sasaki et al. The truth of the f-measure. *Teach tutor mater*, 1(5):1–5, 2007.

[39] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge, 2008.

[40] Theresa Stadler, Bristena Oprisanu, and Carmela Troncoso. Synthetic data–anonymisation groundhog day. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1451–1468, 2022.

[41] Giorgio Visani, Giacomo Graffi, Mattia Alfero, Enrico Bagli, Davide Capuzzo, and Federico Chesani. Enabling synthetic data adoption in regulated domains. *arXiv preprint arXiv:2204.06297*, 2022.

[42] Wesley Kenneth Wilhelm. The fraud management lifecycle theory: A holistic approach to fraud management. *Journal of economic crime management*, 2(2):1–38, 2004.

[43] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional gan. *Advances in Neural Information Processing Systems*, 32, 2019.

[44] Lei Xu and Kalyan Veeramachaneni. Synthesizing tabular data using generative adversarial networks. *arXiv preprint arXiv:1811.11264*, 2018.

[45] Harry Zhang. The optimality of naive bayes. *Aa*, 1(2):3, 2004.

[46] Jun Zhang, Graham Cormode, Cecilia M Procopiuc, Divesh Srivastava, and Xiaokui Xiao. Privbayes: Private data release via bayesian networks. *ACM Transactions on Database Systems (TODS)*, 42(4):1–41, 2017.

A Experiment Results on Synthetic Data Augmented Training

This section provides additional empirical results for Synthetic Augmented Training discussed at Section 4.4.

Metric-Oriented results.

- Figure 3 gives Synthetic Augmented Training result under Accuracy performance metric.
- Figure 5 gives Synthetic Augmented Training result under AUROC performance metric.
- Figure 6 gives Synthetic Augmented Training result under F1 performance metric.
- Figure 4 gives Synthetic Augmented Training result under Recall performance metric.
- Figure 7 gives Synthetic Augmented Training result under Precision performance metric.

Synthesizer-Oriented results.

- Figure 8 gives Synthetic Augmented Training result in Precision-Recall curve for CTGAN synthesizer.
- Figure 9 gives Synthetic Augmented Training result in Precision-Recall curve for TVAE synthesizer.
- Figure 10 gives Synthetic Augmented Training result in Precision-Recall curve for DS0 synthesizer.
- Figure 11 gives Synthetic Augmented Training result in Precision-Recall curve for DS1 synthesizer.

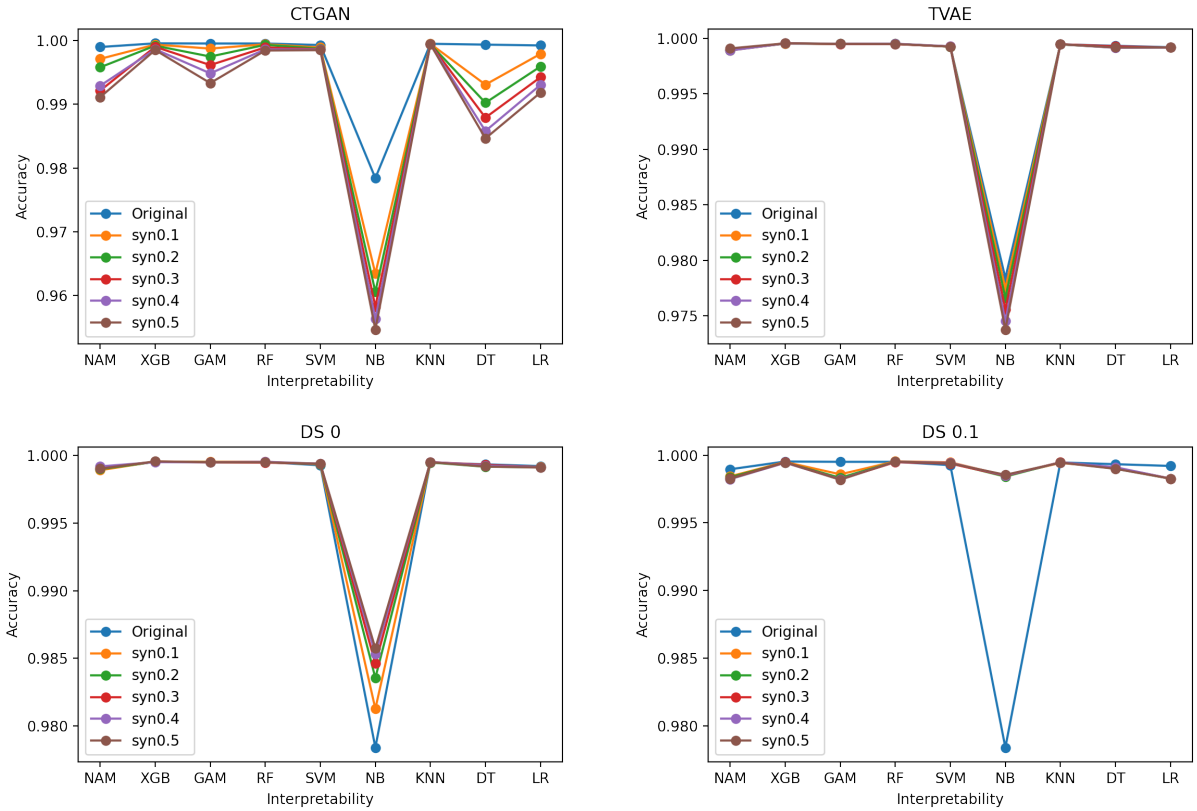

Figure 3: Accuracy. CTGAN-augmented training dataset damages synthetic trained classifier utility. PrivBayes-augmented training dataset improves synthetic trained classifier utility.

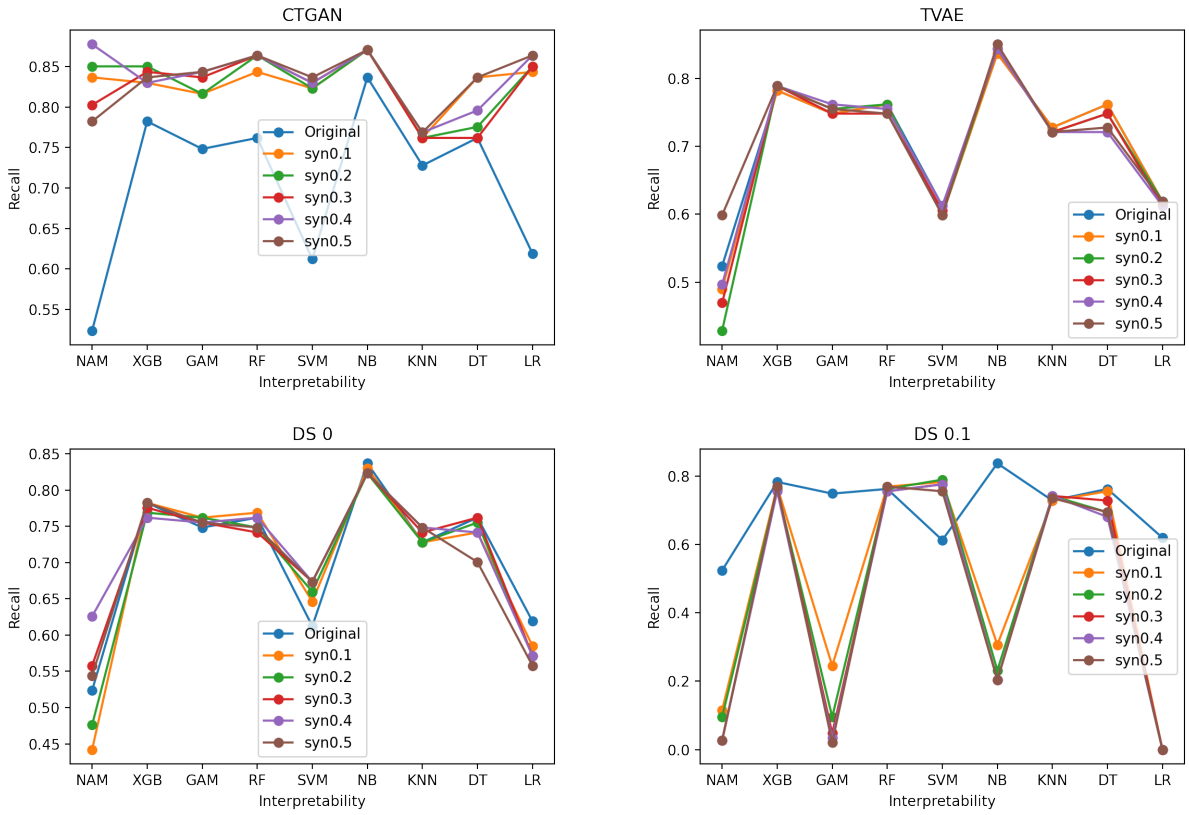

Figure 4: Recall. CTGAN-augmented training dataset improves synthetic trained classifier utility. PrivBayes-augmented training dataset damages synthetic trained classifier utility.

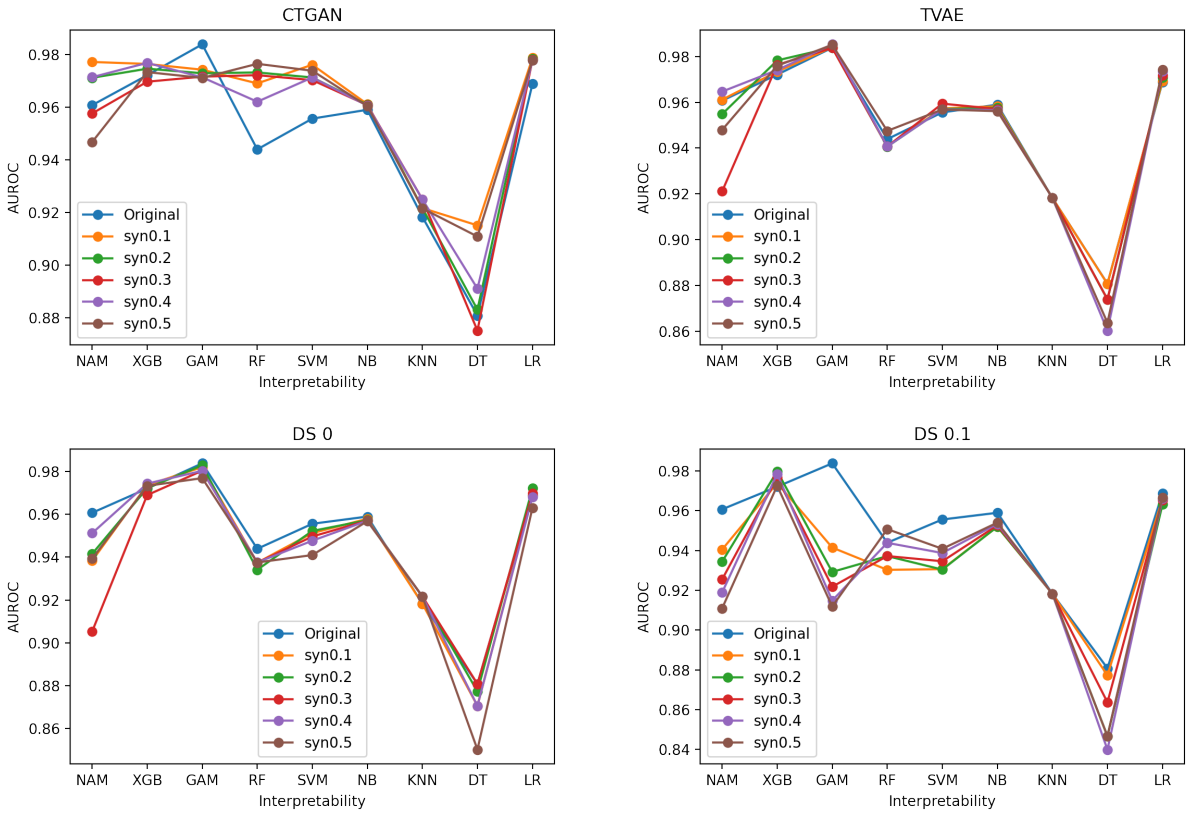

Figure 5: AUROC. CTGAN-augmented training dataset improves synthetic trained classifier utility. PrivBayes-augmented training dataset damages synthetic trained classifier utility.

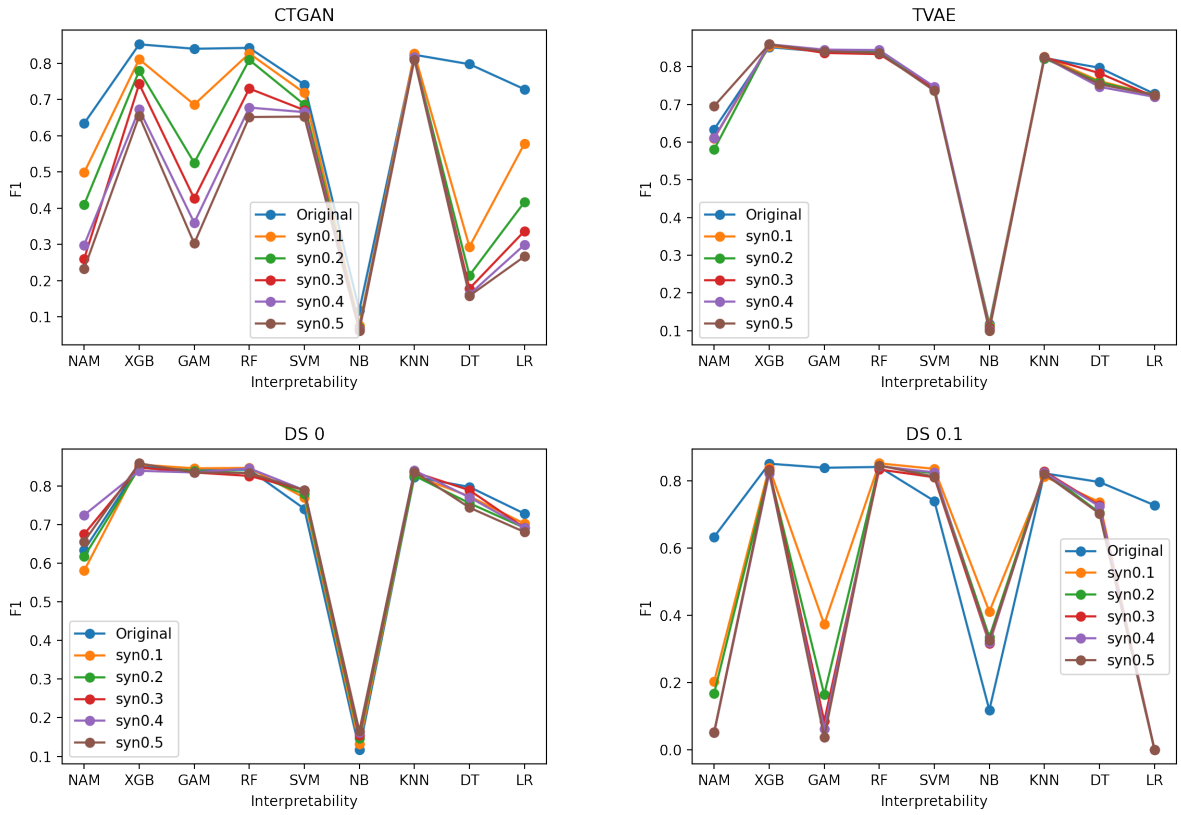

Figure 6: F1. Both CTGAN-augmented and PrivBayes-augmented training dataset damages synthetic trained classifier utility.

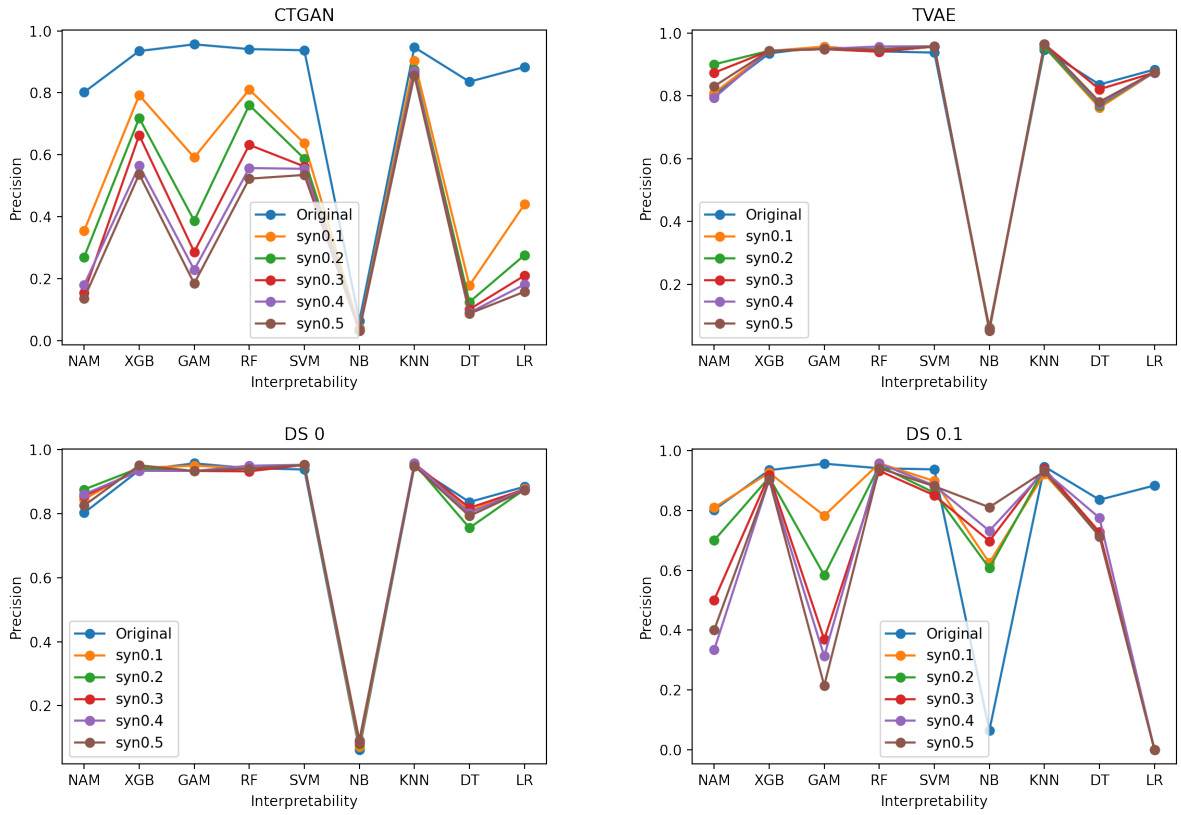

Figure 7: Precision. Both CTGAN-augmented and PrivBayes-augmented training dataset damages synthetic trained classifier utility.

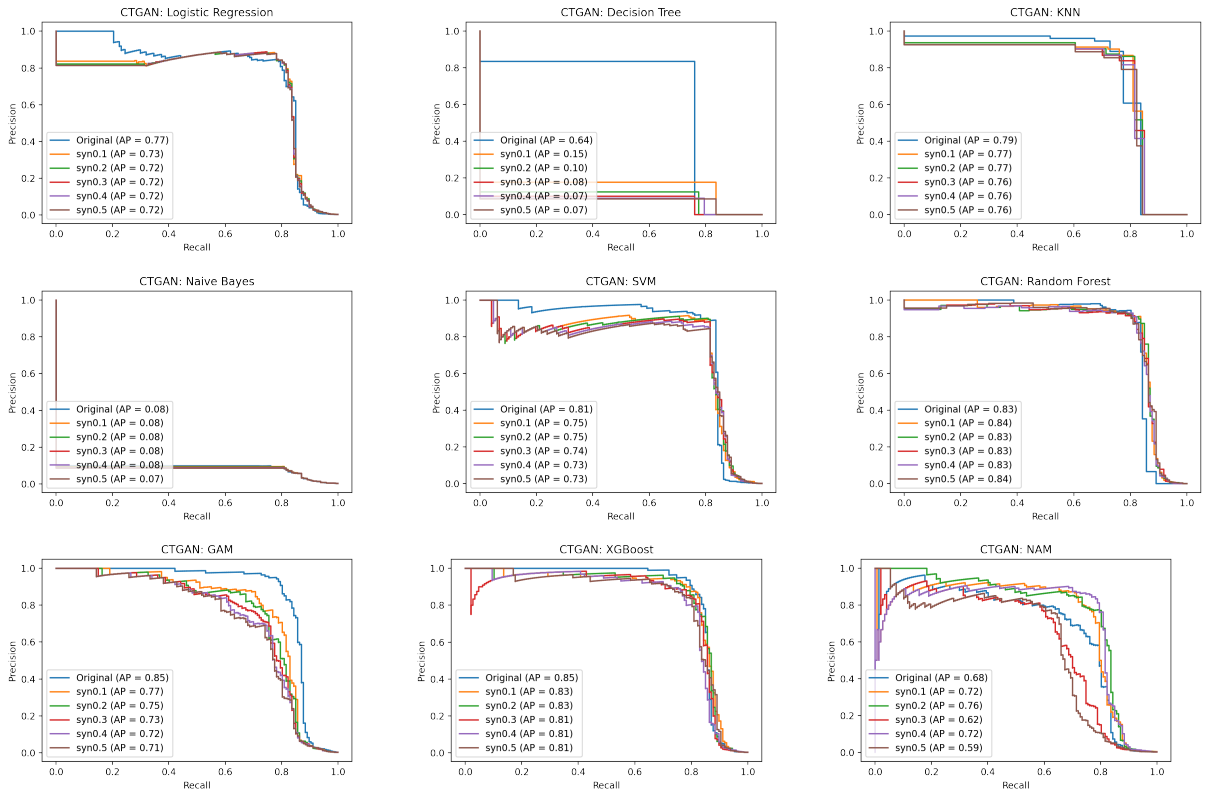

Figure 8: Precision-Recall curve for CTGAN. CTGAN-augmented training datasets in general damages the Precision-Recall curve across all ML classifier.

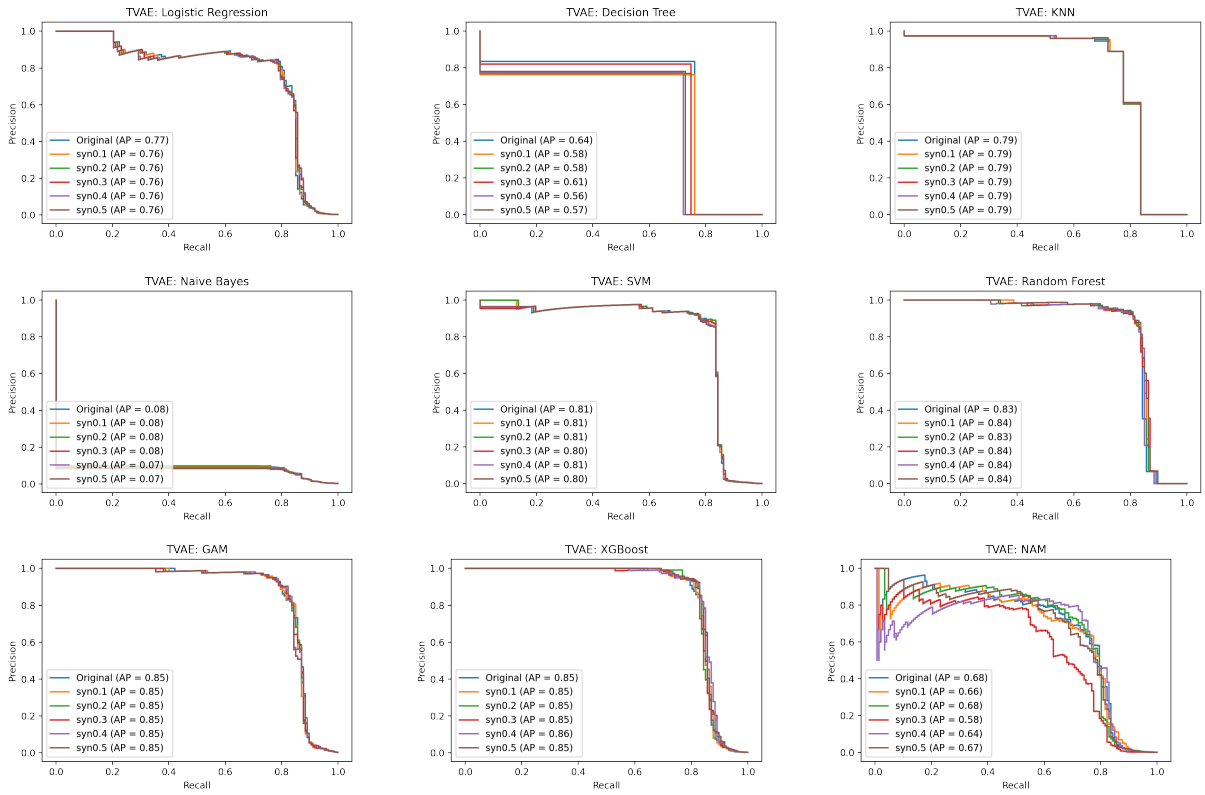

Figure 9: Precision-Recall curve for TVAE. TVAE-augmented training datasets in general do not improve or damage the Precision-Recall curve across all ML classifier.

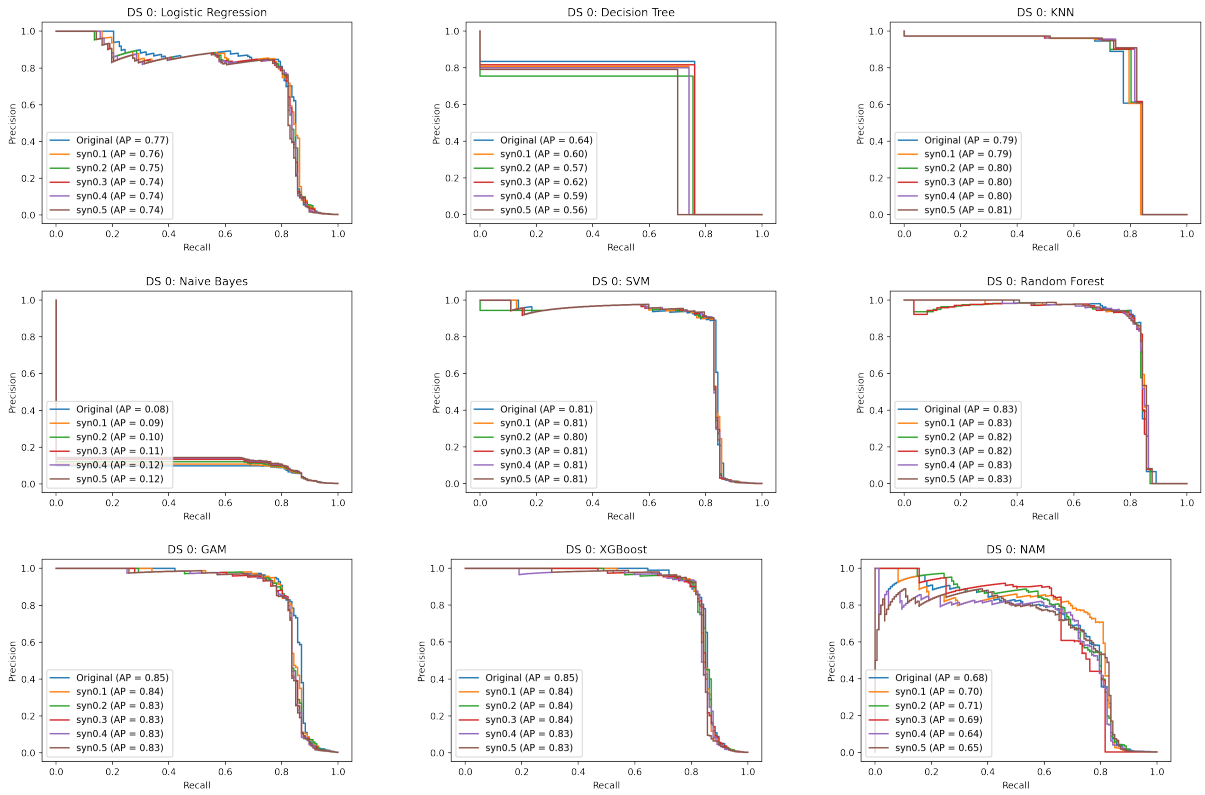

Figure 10: Precision-Recall curve for DS 0. DS 0-augmented training datasets in general do not improve or damage the Precision-Recall curve across all ML classifier.

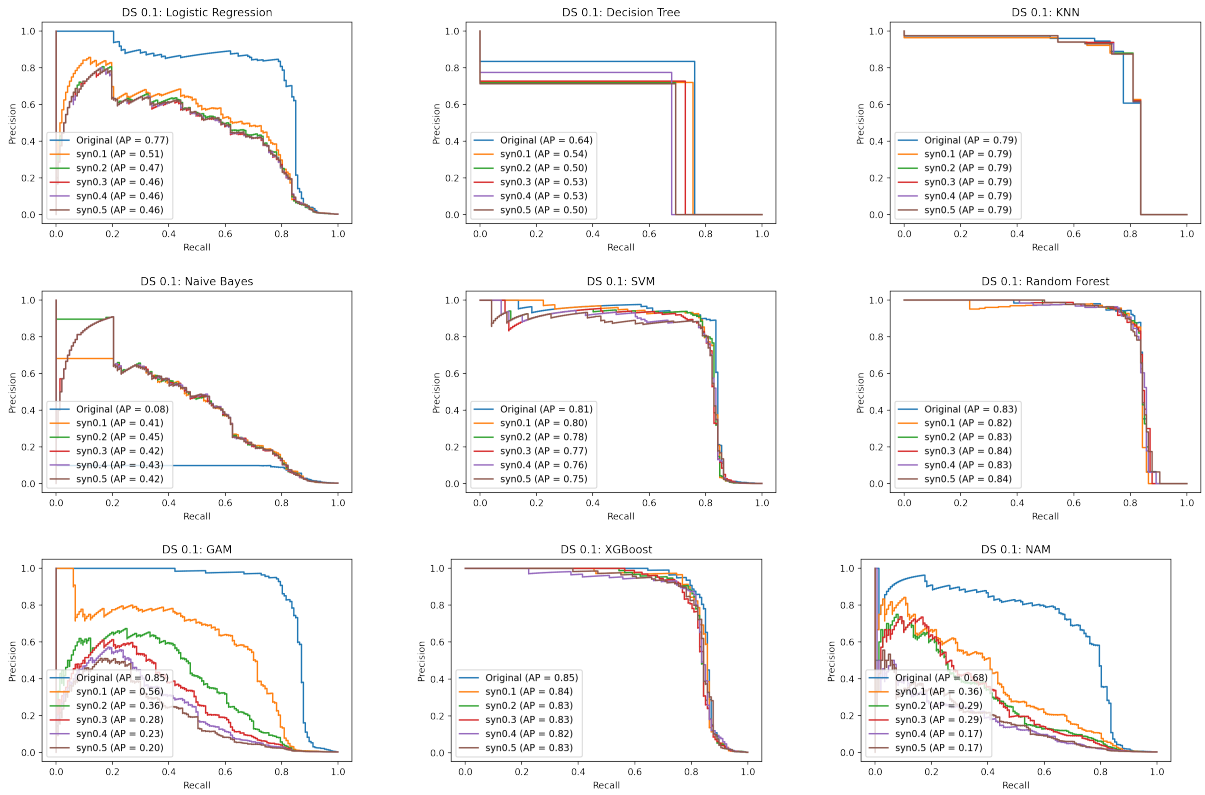

Figure 11: Precision-Recall curve for DS 0.1. DS 0.1-augmented training datasets in general damages the Precision-Recall curve across all ML classifier.