A Appendix

A.1 Hyperparameter spaces for trained CTGANs

The tested hyperparameters were:

- Batch Size (100 to 5000);
- Epochs (50 to 1000);
- Generator embedding layer dimension (8 to 256 neurons);
- Number of layers and neurons per layer in the generator (1 to 3 layers, 128 to 512 neurons per layer);
- Number of layers and neurons per layer in the critic (1 to 2 layers, 64 to 256 neurons per layer);
- Learning rates of the generator and critic.

Default values were used for omitted hyperparameters available in CTGAN's [38] implementation [2].

Additionally, undersampling the dataset was included as hyperparameter, where the target prevalence was increased to either 5%, 10%, or 20%. Not performing undersampling was also one possible value for this hyperparameter.

A.2 Hyperparameter spaces for trained LightGBM models

The tested hyperparameters for trained LightGBM models were:

- Number of estimators (20 to 10000);
- Maximum tree depth (3 to 30 splits);
- Learning rate (0.02 to 0.1);
- Maximum tree leaves (10 to 100);
- Boosting algorithm (GBDT, GOSS);
- Minimum instances in leaf (5 to 200);
- Maximum number of buckets for numerical features (100 to 500);
- Exclusive feature bundling (True or False).

Default values were used for omitted hyperparameters available in LGBM's [44] official implementation [3].

A.3 Results of Generative Models

In this section, we present the evaluation results of the 70 trained generative models. Out of these 70 models, 20 ($\approx$28%) were not able to produce a candidate sample that followed the observed distribution of month and prevalence in the original datasets. These were excluded from the analysis, as they were incapable of learning the distribution of the data over time to an acceptable extent. We present a table with the best performing generative models, when testing with the generated train and test sets.

The first three columns of metrics represent the obtained predictive performance (TPR with thresholding at 5% FPR) with the possible combinations of datasets. Here the column **Train & Test** represents training and testing on the generated dataset; the column **Train Set** represents training on the generated train set and testing on the original test set. This corresponds to the evaluation methodology presented in tabular GAN literature [38, 43]. Lastly, the column **Test Set** represents training on the original training set and testing on the generated test set. The name of the column, therefore, represents which subset of the data is generated in this evaluation step. The selection criterion for the generative model was the performance when training and testing on generated data (in

[2] `https://github.com/sdv-dev/CTGAN/tree/v0.4.3`

[3] `https://lightgbm.readthedocs.io/en/v3.2.1/Parameters.html`

Table 2: Results of the evaluation on trained generative models (Top 5 Models).

ID	**Train**: gen. **Test**: gen. ↓	**Train**: gen. **Test**: orig.	**Train**: orig. **Test**: gen.	**KS Metric**	**Correlation Diff.**
1 (Anonymized)	56.0%	62.7%	37.2%	0.125	0.021
1	54.8%	63.1%	44.2%	0.074	0.018
2	51.2%	63.6%	41.3%	0.077	0.025
3	50.6%	65.3%	39.6%	0.078	0.017
4	49.4%	53.3%	32.0%	0.071	0.027
5	48.4%	62.5%	40.7%	0.086	0.024
Mean (Std.)	26.5% (16.3%)	30.9% (23.3%)	16.7% (13.7%)	0.127 (0.061)	0.031 (0.012)

a descending manner). This was done since, in practice. both training and testing will be performed with generated data.

No model was able to achieve performance similar to training and testing on the original data, which was of 75.4% TPR. Observing the table results, we notice a larger degradation in performance when using the generated test set only. The selected model, in fact, obtained the best performance with the generated test set, while other models produced slightly better results with the generated training sets. In this regard, a part of the models was not capable of converging, with performances close to a random estimator in the ROC space (TPR matching FPR). Regarding the statistical similarity metrics, we observe that these values are not correlated with the ML performance of the datasets; however, the best generated datasets in terms of ML performance tend to also have better statistical similarity metrics.

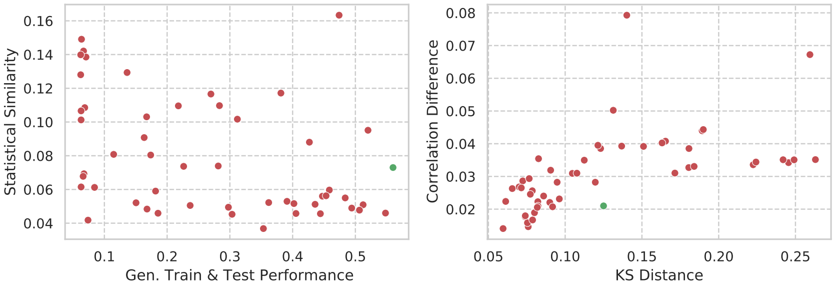

Figure 3: Generative models metrics. The left plot represents performance (with generated train and test sets) versus statistical similarity. The right plot represents the two metrics of statistical similarity. The selected generative model is represented in green.

In these plots, the main conclusion that we can obtain is that there is no clear correlation between ML performance and statistical similarity. The better performing models, however, have better than average results in the statistical similarity metrics.

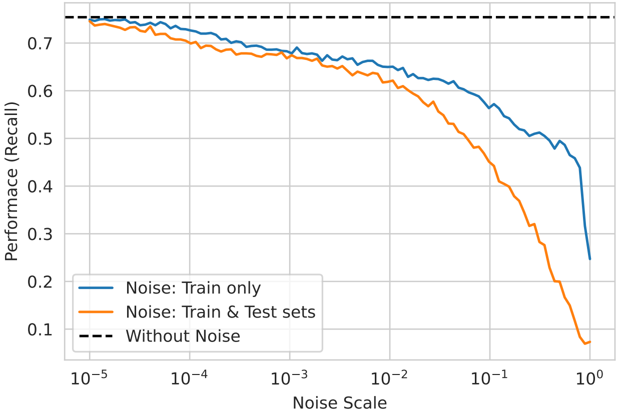

Figure 4: Results of Laplacian noise mechanism. For each point of noise, 100 trials of TPE optimized LightGBM were run, and TPR at 5% FPR in the original test set was registered. The dashed line represents the TPR of the model on data without noise injection. In the blue line, only the training set had noise. In the full orange line, both training set and test set were noisy. The larger the noise scale — i.e., the lower the privacy budget — the larger the decrease in data utility. The orange dashed line corresponds to performance on the original data without added noise.

A.4 Laplacian Noise Mechanism Results

The original data had already been anonymized, and most features were aggregations that do not constitute a significant privacy threat to specific applicants. Still, it was considered important to improve the privacy of the proposed resource. Thus, the process for generating private data was the following: first, prior to training the GAN, we pre-selected a subset of the best and most intuitive features to also improve convergence of the GAN. Second, we added Laplacian noise [41] to the original data, with varying privacy budgets (inverse of the noise scale), and added noise to its categorical features as well. Third, continuous columns that contained personal information were categorized (customer age and income). The GAN was then trained on noisy data, with blocked access to information on real applicants. Finally, a filter was applied after generation to guarantee that no generated instance matched an original data point, and some key categorical features went through label encoding. Figure 4 displays the trade-off between privacy budget (inverse of the noise scale in the xx axis) and model performance on the original test set (yy axis). Given this, we chose the noise scale (10^{-4}) that yielded a TPR of 70% in the noisy test set — which would be later used in the GAN training — corresponding to a decrease in model utility of about 5p.p.. Steeper decreases in utility would compromise the usefulness of the proposed resource. We do not make the GAN model publicly available, which further increases the difficulty of conducting an attack on applicant privacy.

A.5 Correlation of Features

In this section, we compare the correlation between the target variable (fraud label) and each of the features that are available in the dataset. This correlation is calculated for both the original data and the generated base dataset. These values are available in Table 3. We observe that the signs are preserved for almost all the features (except for `feat_22`), *i.e.*, only one feature inverts their natural tendency for fraud. We also observe that correlations are generally slightly weaker for most features, when observing the generated data. This fact might explain the lower ML performance that we observe when training and testing with generated data, as lower correlation values might translate into higher classification difficulty.

Table 3: Spearman correlation between target variable and numerical features in the dataset.

Feature	**Original Data**	**Generated Data**
`feat_1`	0.090	0.050
`feat_2`	-0.028	-0.037
`feat_3`	-0.032	-0.046
`feat_4`	0.030	0.049
`feat_5`	0.042	0.058
`feat_6`	0.005	-0.014
`feat_7`	-0.022	-0.018
`feat_8`	0.013	0.006
`feat_9`	-0.001	-0.016
`feat_10`	-0.011	-0.011
`feat_11`	-0.013	-0.014
`feat_12`	-0.034	-0.032
`feat_13`	-0.030	-0.046
`feat_14`	0.055	0.060
`feat_15`	0.036	0.028
`feat_16`	-0.030	-0.035
`feat_17`	-0.010	-0.013
`feat_18`	-0.032	-0.011
`feat_19`	-0.039	-0.035
`feat_20`	0.049	0.057
`feat_21`	0.012	0.017
`feat_22`	-0.003	0.002
`feat_23`	-0.036	-0.050
`feat_24`	0.089	0.037
`feat_25`	0.013	0.013
Mean Absolute Difference	-	0.011

A.6 Distributions of Customer Age Protected Attribute

Customer Age

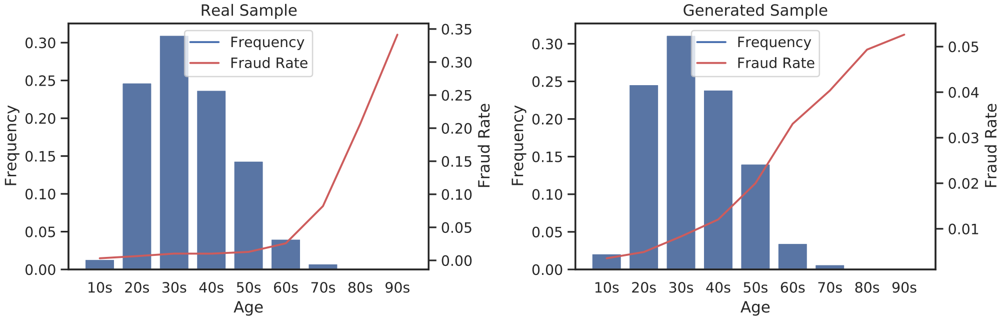

Figure 5: Distribution of age and prevalence of fraud by age in real (left) and generated (right) datasets. Ages truncated to 80, due to the lower frequencies and higher noise in higher values.

A.7 Overlap in Most Important Features

In this section, we replicate the study presented in Section 3.2, for the method of selecting relevant features for the generative model, with different models, namely Random Forests, Decision Trees and Logistic Regressions. For each algorithm, we trained 100 models with hyperparameters selected by a random search algorithm. From these, we selected the 5 best performing models for each algorithm and joined the 30 most important features for each algorithm. In the case of Logistic Regressions, feature values were standardized with the distributions in train. Results are condensed in Table 4.

Table 4: Feature overlap with different models. Compared with LightGBM results (LGBM) and selected features (final). Recall results presented for best model.

Model Group	**N. Overlap Feats (LGBM)**	**Jaccard Sim. (LGBM)**	**N. Overlap Feats (Final)**	**Jaccard Sim (Final)**	**Recall @ 5% FPR (best)**
Random Forest	27	0.443	21	0.477	69.1%
Decision Tree	27	0.450	24	0.480	60.6%
Log. Regression	24	0.407	15	0.333	62.9%

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] The main contribution is a suite of datasets for the evaluation of ML methods. This is described in Section 3.
   (b) Did you describe the limitations of your work? [Yes] This is discussed in Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] This is discussed in Section 5.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments (*e.g.*, for benchmarks)...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (*e.g.*, data splits, hyperparameters, how they were chosen)? [Yes] This information is included both in the description of the dataset, Section 3.1 and Appendix.
   (c) Did you report error bars (*e.g.*, with respect to the random seed after running experiments multiple times)? [No] We do not experiment by varying random seeds in the process. We apply hyperparameter optimization algorithms, and show the results in Section 4 and Appendix.
   (d) Did you include the total amount of compute and the type of resources used (*e.g.*, type of GPUs, internal cluster, or cloud provider)? [Yes] This information is included in Section 3.2 for the generative models. The calculation regarding the training of models in the datasets was omitted, as it was negligible when compared to the computation time of the generative models.
4. If you are using existing assets (*e.g.*, code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] Discussed in Section 3.1.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] This is discussed throughout the paper, and one of the main reasons to do feature engineering and use a generative model.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]