On Population Fidelity as an Estimator for the Utility of Synthetic Training Data

Alexander Florean[1], Jonas Forsman[1], Sebastian Herold[2]

***Abstract*— Synthetic data promises to address several challenges in training machine learning models, such as data scarcity, privacy concerns, and efforts for data collection and annotation. In order to actually benefit from synthetic data, its utility for the intended purpose has to be ensured and, ideally, estimated before it is used to produce possibly poorly performing models. Population fidelity metrics are potential candidates to provide such an estimation. However, evidence of how well they estimate the utility of synthetic data is scarce.**

In this study, we present the results of an experiment in which we investigated whether population fidelity as measured with nine different metrics correlates with the predictive performance of classification models trained on synthetic data.

Cluster Analysis and Cross-Classification show the most consistent results w.r.t. correlation with F1-performance but do not exceed moderate levels. The degree of correlation, and hence the potential suitability for estimating utility, varies considerably across the inspected datasets. Overall, the results suggest that the inspected population fidelity metrics are not a reliable and accurate tool to estimate the utility of synthetic training data for classification tasks. They may be precise enough though to indicate trends for different synthetic datasets based on the same original data.

Further research should shed light on how different data properties affect the ability of population fidelity metrics to estimate utility and make recommendations on how to use these metrics for different scenarios and types of datasets.

I. INTRODUCTION

The utilization of synthetic data in machine learning (ML) model training has gained significant traction due to its potential to address data scarcity, privacy concerns, and the high costs and required time associated with data collection and annotation [1], [2], [3], [4]. Synthetic data generation techniques offer a promising avenue for augmenting training datasets and improving the robustness and generalization capabilities of ML models [5], [6]. While the potential benefits of synthetic data are evident, their actual effectiveness in model training depends on the data's utility, i.e. the degree to which the data are suitable for training models that will show the desired predictive performance and execute the intended task well. Integrating synthetic data of low utility into the training process can lead to poor generalizations, biased models, or ineffective training, all of which might cause poor performance of products and services based on the trained models [2].

Therefore, it appears desirable to be able to reliably estimate the utility of synthetic data before their integration into the training pipeline. Such an estimation could not only make the model training process more efficient and sustainable by avoiding unnecessary training iterations based on poor synthetic data. It could also inform adjusting the data generation process to reach desired or contractually agreed upon levels of utility when synthetic data is shared and help to accurately quantify trade-offs between privacy preservation and utility.

Population fidelity is defined as the degree of accuracy to which synthetic data mimic the original data in terms of statistical properties and underlying characteristics or patterns [7]. As great population fidelity means that a synthetic dataset resembles the original dataset it is constructed from closely, one would expect this measure to be a natural substitute for utility. After all, we would expect similar performance of models trained on different but highly similar data, such as an original dataset and high-fidelity synthetic data generated from it.

However, there is not a single, established way to compute population fidelity, instead, the research community has developed a corpus of different metrics over recent years [8]. They differ in the techniques they utilize to determine the level of similarity between an original dataset and its synthetic counterpart and stretch, for example, from comparing the distributions of values in the datasets with means of statistical testing to applying machine learning to measure how distinguishable the two datasets are. The motivation of this study is to explore whether or not these metrics confirm the assumed association between utility and population fidelity. The goal is to address the following research question: to what degree are different population fidelity metrics capable of estimating how well ML-based classification models trained on synthetic data will perform compared to their counterparts trained on the corresponding real data?

For this purpose, we conducted an experiment in which we trained four different classification models for each of five different original datasets. Additionally, we derived several synthetic datasets from each original dataset and trained classification models for the same tasks on the synthetic datasets. All models were evaluated using their F1-score. For all synthetic datasets, we computed nine different population fidelity metrics, expressing, using different techniques, how closely the datasets mimic the properties of the corresponding original data. Finally, we performed a correlation analysis between those metrics for each dataset and the F1 performance of the corresponding models relative to the models trained on the original data.

[1]CGI Sverige AB, Tynäsgatan 6, 652 24 Karlstad, Sweden alexander.florean@cgi.com, jonas.forsman@cgi.com

[2]Department of Mathematics and Computer Science, Karlstad University, 651 88 Karlstad, Sweden sebastian.herold@kau.se

The remaining article is structured as follows. Section II provides an overview of population fidelity metrics and summarizes existing work on estimating synthetic data utility. In Sec. III, we explain the experiment design in detail. The results of the experiment are presented in Sec. IV and discussed in Sec. V. Finally, Sec. VI completes the article with concluding remarks.

II. BACKGROUND

A. Population Fidelity Metrics

Population fidelity, the degree to which synthetic data resemble the original data, can be measured by different metrics. In this section, we briefly explain the metrics that were compared in this study.

Woo et al. describe the **Cluster Analysis** measure as a population fidelity metric [9]. The fundamental idea of the approach is to fit a clustering model to the dataset that results from merging the original data and the synthetic data and to analyse the distribution of synthetic and original data points in each cluster. If the synthetic data resemble the original data closely, the proportion of original data points in each cluster should be similar and close to the overall proportion. The metric therefore computes a sum of squared error between these proportions per cluster and the overall proportion, weighted by cluster sizes.

The same authors also describe the **Propensity Mean Squared Error (pMSE)** as a population fidelity metric. It is based on the idea of fitting a classification model to the same merged dataset to predict whether a data point is synthetic or original. For the resulting predictions, the propensity score is computed [10]. In the case of a synthetic dataset that is perfectly indistinguishable from the original one, the expected propensity scores would be equal to the proportion of synthetic data points in the merged data as that proportion would correspond to the "probability" of a randomly picked data point being synthetic. The overall metric is therefore defined as the mean squared error between the propensity scores and that proportion. The closer the synthetic data resemble the original, the closer the resulting value is to zero.

Cross-classification takes the generated synthetic data for training several classification models [11]. For each categorical feature in the dataset, a model is trained with that feature as the target and all other features as predictors. The models are then tested on the original data. The average performance of these models is interpreted as a measure of population fidelity.

Likelihood measures fit probabilistic models to the synthetic data that reflect the likelihood that the synthetic data belong to the same distribution as the original data. **Bayesian Networks Log Likelihood (BNLogLikelihood)** fits a Bayesian Network to the original data and generates a likelihood estimate for each synthetic data point [12]. The final score is the average of these estimates. **Gaussian Mixture Log Likelihood (GMLikelihood)** works similarly but fits a Gaussian Mixture Model instead [13].

TABLE I: Overview of the investigated population fidelity metrics.

Metric	Range	Value of Maximal Fidelity
BNLogLikelihood	$(-\infty, 1]$	1
Cluster Analysis	$[0, \infty)$	0
ContinuousKLD	$[0, 1]$	1
Cross Classification	$[0, 1]$	1
Chi-Statistic Test	$[0, 1]$	1
DiscreteKLD	$[0, 1]$	1
GMLogLikelihood	$(-\infty, 1]$	1
KSComplement	$[0, 1]$	1
pMSE	$[0, 0.25]$[1]	0

The **Kullback-Leibler divergence (KLD)** also known as relative entropy or information divergence is a measure of statistical distance [14], [15]. It quantifies the difference between two probability distributions, offering a way to measure the information loss when using one distribution to approximate another. For the experiment, two different variants were considered. **DiscreteKLD** considers only categorical while **ContinuousKLD** analyses numerical features.

The **Kolmogorov-Smirnov Complement (KSComplement)** is a measure from the SDMetrics library used to quantify the quality of synthetic data by comparing the cumulative distribution functions (CDFs) of the original and synthetic datasets [9]. It is based on the Kolmogorov-Smirnov (KS) Statistic Test, a non-parametric statistical test that evaluates the maximum distance between the CDFs of two datasets. It tests the null hypothesis that the two datasets are drawn from the same distribution, where a value of zero indicates high similarity in distributions. The KSComplement adapts this approach by providing the complement to the traditional KS statistic, focusing on the similarity between distributions, meaning the value of one indicates similarity rather than zero.

The **Chi-Statistic Test (CSTest)** measure is based on the statistical test of the same name to assess the similarity between two distributions of data [16]. It is implemented in the SDMetrics library as a population fidelity measure and calculates the statistical significance of differences between observed frequencies of values in the synthetic data and the expected frequencies as present in the original data. This measure only considers categorical features.

Tab. I lists the introduced population fidelity metrics, their ranges, and values indicating maximal fidelity.

B. Related Work Investigating the Association between Population Fidelity and Utility

The literature addressing the question of to which degree different population fidelity metrics are able to estimate utility is scarce. Dankar et al., although not directly touching upon the issue, describe a similar study that investigates the utility of different synthetic data generators [8]. To that end, they inspected different data generation methods and

[1]More general, the range is $[0, \max(c^2, (1-c)^2)]$, c being the ratio of synthetic data in the merged dataset. In the experiments, c is equal to 0.5.

evaluated the performance of classification models trained on the generated synthetic data. Four different fidelity metrics, including pMSE as only population fidelity metrics, were computed for all synthetic datasets. While the main results and discussion focus on the performance of the synthetic data generators, a side result shows that there was only a low level of agreement between the fidelity metrics on best-performing data generators, and largely weak correlations between the metrics. The authors conclude on that front that no single metric might be sufficient to evaluate the utility of synthetic data.

Goncalves et al. present a study on generating synthetic patient data and evaluating utility and privacy risks [11]. They assess different synthetic data generation methods, including probabilistic models, classification-based imputation models, and generative adversarial neural networks. The study uses various metrics to evaluate data utility and privacy risks. While the article reflects on the utility of synthetic data using different population fidelity metrics, the focus is not on investigating the relationship between utility and population fidelity, the terms are used rather synonymously. Therefore, no performance metrics were analysed and no correlation analyses or analyses of the agreement between population fidelity metrics were performed.

Dankar and Ibrahim investigate the various usage configurations for generating synthetic data and their effects on its utility and resulting models [17], including the effect of data preprocessing and whether tuning should be applied to synthetic data for classification models. They also address the question of whether pMSE can predict the accuracy of the resulting classification models. Similar to the experiment we present in this article, they generated synthetic datasets based on several original datasets and analyse fidelity and performance. In contrast to our work, in which we ignore the technique used for data generation, Dankar and Ibrahim analyse the results w.r.t. to the generation techniques applied, and focus on accuracy as a performance measure only.

The results suggest that neither preprocessing data prior to generating synthetic data nor tuning on synthetic data yielded any significant benefit. The authors therefore argue that there is a benefit in sharing tuning settings of the original data along with synthetic data. However, this is based on the ideal setting where the user knows beforehand of the type of analysis that will be performed on the data or that the user of synthetic data will have access to the original data, which is rarely the case, in particular when synthetic data is used to protect sensitive data [18].

As for the ability of pMSE to predict accuracy, the results show only a weak correlation with the resulting performance, which the authors measure as an absolute difference in accuracy with models trained on original data.

Our work aims to extend these insights in three ways. Firstly, we believe that other performance metrics than accuracy might be more accurate to relate population fidelity with, as many classification problems are inherently imbalanced. Accuracy is, in those cases, not an appropriate performance measure. Secondly, the practical question motivating our research is whether or not synthetic data can replace the original data for model training purposes, i.e., which level of performance we get *relative* to using the original data. Looking at the absolute performance difference like Dankar and Ibrahim can be misleading: An absolute loss of 10% in accuracy weighs heavier if the accuracy of the model trained on original data was 40% than when it was 95%. Thirdly, we extend the set of investigated population fidelity metrics to get a more comprehensive picture of the relationship between population fidelity and classification performance.

III. EXPERIMENT DESIGN

As introduced in Sec. I, the motivating research question for this study is to which degree population fidelity metrics can estimate the utility of synthetic tabular data for classification tasks. The process of the experiment to address this research question is illustrated in Fig. 1.

The starting point for the experiment is a set of original datasets. Each of them is prepared and cleaned in step 1. The resulting cleaned datasets serve as inputs for two subsequent activities. In step 2, they are used for training baseline classification models using several supervised learning algorithms for classification problems. These models are then evaluated using several performance metrics, including the F1-score. Step 3 consists of generating several synthetic datasets using generative adversarial networks (GANs) [19], [20]. These datasets then serve as training data for new classification models in step 4, using the same classification algorithms as in step 2. The resulting models are evaluated in terms of a relative F1-score. This measure takes into account how well the corresponding baseline model from step 2 performed such that the value reflects how a model trained on synthetic data compared to the same model trained on original data. In step 5, nine different population fidelity metrics are computed for each synthetic dataset. After having obtained the necessary values in steps 4 and 5, step 6 finally consists of performing several descriptive statistics and a correlation analysis between the model performance as measured as relative F1-scores and the considered population fidelity metrics.

The following subsections explain the individual steps in more detail.

A. Step 1: Prepare Data

The first step involved selecting appropriate datasets and preparing them for the subsequent steps. The selection of datasets had to meet several criteria:

- The datasets should contain tabular data of independent data points (excluding, e.g., time series).
- The datasets should vary in number and types of predictors.
- The datasets should be of a manageable size as the available computing resources were limited.
- The datasets should be freely accessible to allow the research community to replicate the study.

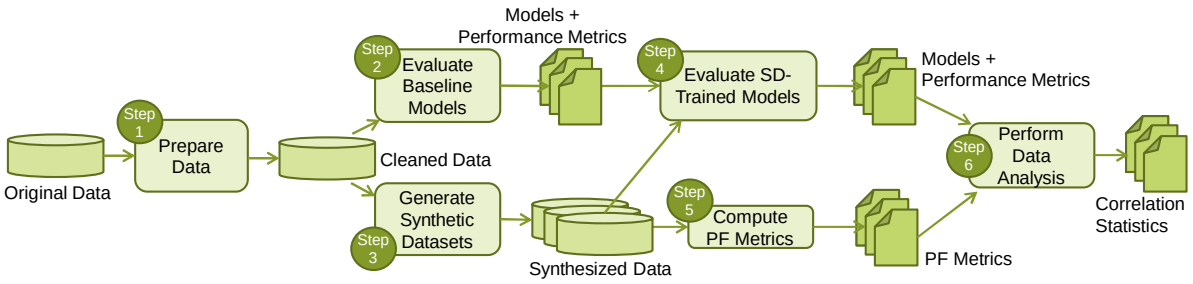

Fig. 1: Overview of the experiment.

TABLE II: The datasets selected for the study.

Dataset name	**# samples (original size)**	**# predictors (num./cat./ord.)**	**# labels**
D_1: Adult	2261 (45222)	15 (6/8/1)	2
D_2: Bank	2260 (45211)	16 (7/7/2)	2
D_3: Diabetes	768	8 (8/0/0)	2
D_4: MNIST	3500 (70000)	784 (784/0/0)	10
D_5: Titanic	891	7 (4/3/0)	2

Tab. II lists the selected datasets. The Adult dataset contains census data and has been used to showcase classifications for predicting high income based on personal data [21]. The Bank dataset relates to direct marketing campaigns by a Portuguese banking institute [22]. It has been used for training models classifying whether or not a client would subscribe to a financial product. The Diabetes dataset centres around predicting early diabetes in female patients based on diagnostic measurements [23]. The MNIST dataset is a collection of images of handwritten digits in CSV format [24]. Lastly, the Titanic dataset contains passenger data of the famous vessel and its ill-fated voyage [25]. The data is often used for educational purposes to illustrate ML-based classification, mostly in predicting the chance of survival of passengers.

We included the MNIST dataset although it strictly speaking does not contain tabular data but image data represented in tabular format. Firstly, the number of features (representing individual pixels in a $28x28$ picture) differs significantly from the other datasets. Secondly, the data can be easily visualized and provide a first intuitive grasp of the derived synthetic versions' utility (or the lack of it).

Due to resource limitations that were observed during trial runs of the overall experiment, we downsampled the Adult, Bank, and MNIST datasets to 5% of their original size, using stratified sampling to keep imbalances in the data. The remaining data preparations were largely about imputation, i.e. dealing with missing values in the datasets. Depending on semantic meaning and type of features, we applied techniques that seemed adequate after discussions among the authors. Please refer to the replication package for more details on this process.

B. Step 2: Evaluate Baseline Models

In order to produce and evaluate baseline models to compare the classification models trained with synthetic data, we created classifiers applying four different classic machine learning algorithms, Logistic Regression, K-Nearest Neighbors, Random Forest, and Support Vector Machines. This resulted in four baseline models B_i^a for each dataset D_i, a indicating the algorithm used for training.

The data was split into 80% training and 20% test data. For hyperparameter tuning, we applied a 10-fold cross-validation and we utilized the tree-structured Parzen estimator algorithm from the Optuna library [26]. Although we focused on the F1-score in the later analysis, we recorded several additional performance metrics (measured on the test set), such as accuracy, precision, recall, Matthews correlation coefficient, and Cohen's kappa score. This way, we (and other interested researchers) can easily rerun the experiment investigating the association between population fidelity and these performance measures as well.

C. Step 3: Generate Synthetic Datasets

For generating the synthetic data required for the experiment, we used *conditional tabular GANs (CTGAN)* [27], a variation of generative adversarial networks (GANs) [20], for four of the datasets. GANs for synthetic data generation are (pairs of) neural networks trained on original data that, after training, are able to produce synthetic data statistically similar to the original data. By changing the number of training epochs, the fidelity of the resulting data can be influenced: too few epochs during training will lead to data that resembles the original data less accurately. The possibility to easily manipulate the fidelity (and, hence, likely utility) in creating synthetic datasets made GAN architectures well suited for our experiment. As CTGANs were shown to outperform other GANs for tabular data, we selected these for generating synthetic data. In early test runs of the experiment, the CTGAN model showed poor performance for the MNIST dataset. Following Xu, we decided to use the TVAE model for the MNIST dataset instead, significantly improving the performance [27].

For each original dataset, we then created five different generators, each trained for a different number of epochs (10, 100, 500, 1000, 1500). Each generator was then run 10 times to generate synthetic datasets of the same size as the original dataset. In total, this resulted in 50 synthetic datasets

per original dataset each of which can be described as $S^e_{i,j}$, for $j = 1, \ldots, 10$, a dataset based on the original dataset D_i, created by the generator trained for e-many epochs.

For details on settings used for training the synthetic data generators, please refer to the replication package referred to in Sec. III-G.

D. Step 4: Evaluate SD-Trained Models

This step closely followed the process outlined in Step 2, with two key distinctions. Firstly, while training was performed on the synthetic datasets, testing and evaluation were done on the original data, not a held-out part of the synthetic dataset. Secondly, tuning comes in two flavours. Each of the algorithms considered in step 2, was run twice to create two different variants of setting the hyperparameters:

- Variant A, reusing the hyperparameters of the corresponding baseline model from step 2.
- Variant B, based on newly tuned hyperparameters, determined using the same technique as in step 2.

This choice was made to complement the results of Dankar and Ibrahim who, in their experiments, did not see significant differences between these two variants [17].

Overall, this resulted in eight classification models $M^{e,a,v}_{i,j}$ for each synthetic dataset $S^e_{i,j}$, a and v referring to the algorithm used for learning and v to the tuning variant, respectively.

As for the evaluation of performance, we compute in this step a relative F1-score that enables us to easily compare the performance of models trained on synthetic data with the corresponding baseline model. We define the relative F1-score of a model trained on synthetic data as

$$\text{rel\_f1}(M^{e,a,v}_{i,j}) := \frac{\text{f1}(M^{e,a,v}_{i,j})}{\text{f1}(B^a_i)}$$

The relative variants of other recorded performance measures (see Sec. III-B) can be defined analogously.

E. Step 5: Compute Population Fidelity Metrics

In this step, we computed the population fidelity metrics explained in Sec. II-A for all the datasets $S^e_{i,j}$. The metrics Cluster Analysis, Cross Classification, and pMSE were implemented from scratch based on their definitions in the literature. The implementations (provided by the first author) were rigorously tested and reviewed by the co-authors to identify bugs and establish a high level of certainty of the implementations' correctness. The remaining metrics were computed using the SDMetrics library to compute [28].

As for the Cluster Analysis implementation, we used two different clustering algorithms. Sklearn's K-Means implementation is utilized for datasets with exclusively numerical features. Other datasets are handled by using the KPrototypes algorithm as implemented in the kmodes library, a versatile clustering algorithm capable of handling mixed datasets [29].

For the classification as part of the Cross classification metric, we use multi-layer perceptron (MLP) classifiers, either as implementation for binary or multi-class classification problems. To decide which one to use, the implementation counts the number of different values for the feature of interest and selects the classifier accordingly.

The first step in the implementation of the pMSE measure is to merge the corresponding original dataset with the synthetic one and augment the data points with a binary target feature indicating their origin, original or synthetic. The data is standardized and used for training a logistic regression classifier as proposed in the literature [9]. Upon training, the classifier predicts the likelihood of the test data points being synthetic.

F. Step 6: Perform Data Analysis

In the last step of the experiment, we eventually analyse the collected metrics to answer the motivating research question of how well population fidelity metrics estimate the performance of classification models trained on synthetic data. This question is therefore translated into a set of hypotheses that can be tested using statistical tests for correlation analysis.

First, we test for the most generic hypothesis:

> **Null hypothesis** H^A_0(*pf*): there exists no monotonic relationship between the population fidelity measure *pf* of a synthetic dataset and the relative F1-scores of models trained on that dataset.
> **Alternative hypothesis** H^A_1(*pf*): a monotonic relationship exists between the population fidelity measure measure *pf* and the relative F1-score.

After that, we refine this hypothesis to investigate possible correlations with specific classification models:

> **Null hypothesis** H^B_0(*pf,a,t*): there exists no monotonic relationship between the population fidelity measure *pf* of a synthetic dataset and the relative F1-scores of models trained on that dataset using algorithm a and tuning variant t.
> **Alternative hypothesis** H^B_1(*pf,a,t*): a monotonic relationship exists between the population fidelity measure measure *pf* and the relative F1-score of models using algorithm a and tuning variant t.

Finally, we investigate to which degree a such correlation can be found for the individual original datasets:

> **Null hypothesis** H^C_0(*pf,i*): there exists no monotonic relationship between the population fidelity measure *pf* of a synthetic dataset based on D_i and the relative F1-scores of models trained on that dataset.
> **Alternative hypothesis** H^C_1(*pf,i*): a monotonic relationship exists between the population fidelity measure measure *pf* of a synthetic dataset based on D_i and the relative F1-score of models trained on that dataset.

All hypotheses were tested using Spearman's rank correlation coefficient at a significance level of $\alpha = 0.01$.

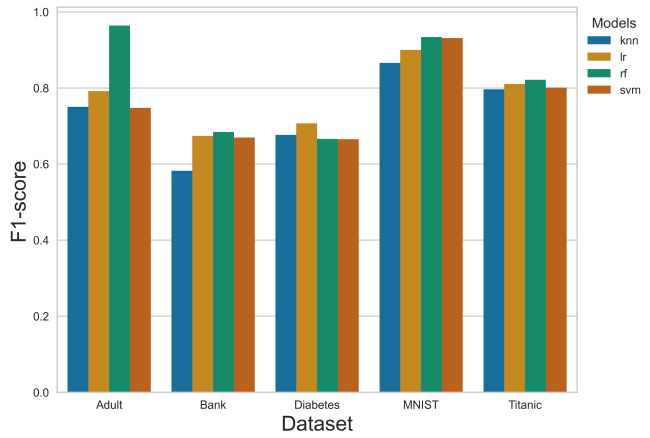

Fig. 2: Baseline model F1-performances (trained on original data).

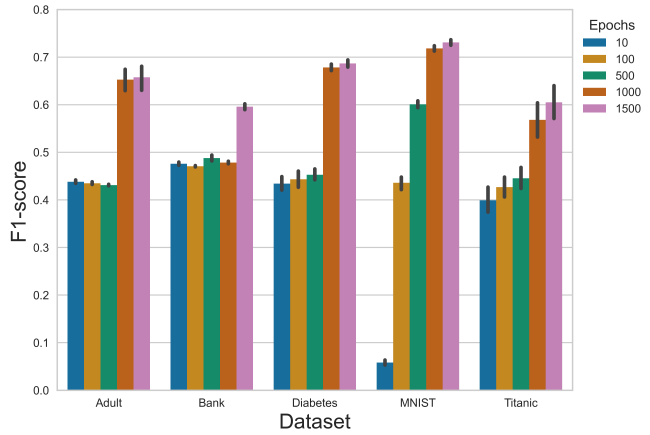

Fig. 3: Mean F1-performance for each dataset and number of generator training epochs.

G. Implementation and Replication Package

The experiment was implemented in Python. The code and documentation are available in the replication package available at `https://github.com/alexanderflorean/SCAI2024_Estimating_Synthetic_Data`

IV. RESULTS

Through the process in step 2, we obtained classification models for each of the datasets. Fig. 2 shows their average F1 performance by training algorithm. As described in Sec. III-C, we generated synthetic datasets with varying training epochs for the underlying generator. This served the purpose of obtaining datasets of varying utility as measured as the F1 performance of the models being trained on the datasets (see Sec. III-D). Fig. 3 illustrates the resulting averaging F1 performances over the number of epochs for each of the datasets. While there is a general trend of the F1 performance increasing with the number of epochs, the shape of the increase varies across the datasets. The greatest variance in F1 can be observed for the MNIST dataset (ranging from less than 0.1 to over 0.7), while for the other cases, the values range between 0.4 and 0.75. Fig. 4 shows the distribution of population fidelity vs. relative F1-score for three selected population fidelity measures. We limited the illustration to three measures due to space limitations. The scatter plots for the remaining population fidelity metrics are available in the full documentation of the experiment (see Sec. III-G).

TABLE III: Results of testing $H_0^A(pf)$: Is there a monotonic relationship between population fidelity and relative F1-score?

Measure	p-value	Correlation / CI (99%)
BNLogLikelihood	0.0000	0.1761 [0.1031, 0.2471]
Cluster Measure	0.0000	-0.5370 [-0.5767, -0.4947]
ContinuousKLD	0.0000	0.2596 [0.2051, 0.3125]
CrCl	0.0000	0.4619 [0.4154, 0.506]
CSTest	0.0000	0.4300 [0.3674, 0.4887]
DiscreteKLD	0.0000	0.3414 [0.2741, 0.4055]
GMLogLikelihood	0.0188	0.0526 [-0.005, 0.1098]
KSComplement	0.0000	0.4425 [0.395, 0.4876]
pMSE	0.0000	-0.4589 [-0.5032, -0.4122]

Each data point in each scatter plot represents a single model $M_{i,j}^{e,a,v}$ trained in step 4 (see Sec. III-D) and its relative F1-score and population fidelity metric. The general distribution of data points hints at a potential negative correlation for cluster measure and pMSE (which both decrease in value with increasing fidelity) and a positive correlation with cross-classification (CrCl). This appears even more pronounced for individual datasets in some cases, like for the MNIST dataset measures with Cluster Measure and CrCl, for which the plots suggest a stronger correlation than for the overall depicted dataset. However, there is also significant spread of values for all measures.

These visual impressions (and the ones for the missing population fidelity measures) are confirmed by the statistical tests (see Tab. III). The test results for $H_0^A(pf)$ are statistically significant for all population fidelity metrics but GMLogLikelihood. We therefore reject all $H_0^A(pf)$ and assume $H_1^A(pf)$ for all metrics but GMLogLikelihood.

The correlation values and confidence intervals indicate that the strength of the correlation varies across the measures. BNLogLikelihood and ContinuousKLD show only weak correlations while Cluster measure, CrCl, CSTest, DiscreteKLD, KSComplement, and pMSE indicate moderate correlations.

Tab. IV summarises the test results for the hypotheses H_0^B, a refined analysis looking at the correlations on a per-algorithm basis. For five out of the nine population fidelity metrics (Cluster Measure, Cross Classification, CSTest, KSComplement, pMSE), we can consistently reject the corresponding null hypothesis and assume the alternative hypothesis is true across all learning algorithms and tuning alternatives[2]. ContinuousKLD and DiscreteKLD do not exhibit a significant correlation for one to two cases while, on the other hand, BNLogLikelihood and GMLogLikelihood show no correlation except for random forest models and k-NN (BNLogLikelyhood only).

[2]In the table, the prefix o_ indicates reusing hyperparameters from tuning the corresponding baseline model (see Sec. III-D)

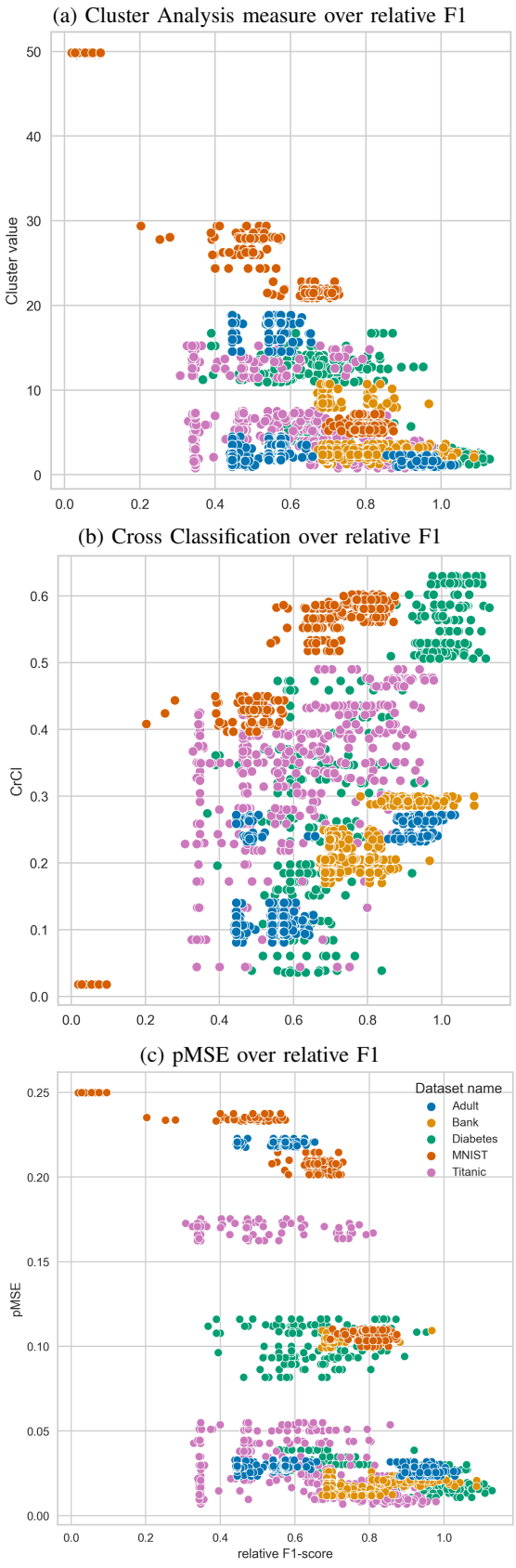

Fig. 4: Scatter plots for three selected population fidelity metrics over the relative F1-score.

TABLE IV: Results of testing $H_0^B(pf,a,t)$: Is there a monotonic relationship between population fidelity and relative F1-score for individual learning algorithms?

Algo-rithm	p-value	Correlation / 99%-CI	p-value	Correlation / 99%-CI
	BNLogLikelihood		**Cluster M. (avg. -0.54, sd. 0.11)**	
knn	0.0038	0.2409 [0.0333, 0.4286]	0.0000	-0.6631 [-0.7453, -0.5611]
o_knn	0.2429	0.0988 [-0.1128, 0.3019]	0.0000	-0.6362 [-0.7238, -0.5283]
lr	0.9152	-0.0090 [-0.218, 0.2007]	0.0000	-0.6030 [-0.6971, -0.4884]
o_lr	0.3555	0.0783 [-0.1332, 0.283]	0.0000	-0.5916 [-0.688, -0.4748]
rf	0.0000	0.6166 [0.4676, 0.7315]	0.0000	-0.4236 [-0.5484, -0.2805]
o_rf	0.0000	0.5875 [0.4313, 0.7096]	0.0000	-0.3202 [-0.4588, -0.1665]
svm	0.6160	-0.0426 [-0.2497, 0.1683]	0.0000	-0.5870 [-0.6842, -0.4693]
o_svm	0.3928	-0.0724 [-0.2776, 0.139]	0.0000	-0.5012 [-0.6137, -0.3688]
	ContinuousKLD		**Cross Class. (avg. 0.47, sd. 0.08)**	
knn	0.0000	0.3567 [0.2062, 0.4907]	0.0000	0.4720 [0.3353, 0.5893]
o_knn	0.0000	0.2666 [0.1088, 0.4112]	0.0000	0.3394 [0.1873, 0.4756]
lr	0.0000	0.3814 [0.2334, 0.5122]	0.0000	0.4590 [0.3204, 0.5783]
o_lr	0.0000	0.3520 [0.201, 0.4866]	0.0000	0.4878 [0.3534, 0.6025]
rf	0.0014	0.2065 [0.0456, 0.357]	0.0000	0.6086 [0.4951, 0.7017]
o_rf	0.8830	0.0096 [-0.153, 0.1718]	0.0000	0.5559 [0.4325, 0.6589]
svm	0.0000	0.3040 [0.1489, 0.4445]	0.0000	0.3969 [0.2506, 0.5255]
o_svm	0.0000	0.2835 [0.1269, 0.4263]	0.0000	0.4392 [0.298, 0.5616]
	CSTest (avg. 0.46, sd. 0.06)		**DiscreteKLD**	
knn	0.0000	0.5350 [0.3667, 0.6693]	0.0000	0.3631 [0.1664, 0.532]
o_knn	0.0000	0.4691 [0.288, 0.6177]	0.0009	0.2747 [0.0693, 0.4577]
lr	0.0000	0.4462 [0.2613, 0.5995]	0.0021	0.2551 [0.0484, 0.4409]
o_lr	0.0000	0.4514 [0.2673, 0.6036]	0.0010	0.2715 [0.066, 0.455]
rf	0.0000	0.5413 [0.3744, 0.6742]	0.0000	0.7172 [0.5977, 0.8056]
o_rf	0.0000	0.4700 [0.2892, 0.6185]	0.0000	0.6438 [0.5021, 0.7518]
svm	0.0000	0.3742 [0.1789, 0.5412]	0.0469	0.1672 [-0.0437, 0.3638]
o_svm	0.0000	0.3620 [0.1652, 0.5311]	0.0747	0.1502 [-0.061, 0.3486]
	GMLogLikelihood		**KSComplement (avg. 0.45, sd. 0.12)**	
knn	0.3619	0.0597 [-0.1038, 0.22]	0.0000	0.4937 [0.3601, 0.6074]
o_knn	0.1603	0.0917 [-0.0718, 0.2504]	0.0000	0.3687 [0.2193, 0.5011]
lr	0.1111	-0.1040 [-0.262, 0.0595]	0.0000	0.4525 [0.313, 0.5728]
o_lr	0.2580	-0.0740 [-0.2336, 0.0896]	0.0000	0.4344 [0.2926, 0.5575]
rf	0.0000	0.3364 [0.184, 0.473]	0.0000	0.7296 [0.6434, 0.7975]
o_rf	0.0000	0.3261 [0.1728, 0.464]	0.0000	0.5366 [0.4099, 0.643]
svm	0.2548	-0.0745 [-0.2341, 0.0891]	0.0000	0.3413 [0.1894, 0.4773]
o_svm	0.1138	-0.1032 [-0.2613, 0.0602]	0.0000	0.2815 [0.1248, 0.4246]
	pMSE (avg. -0.50, sd. 0.07)			
knn	0.0000	-0.5805 [-0.6789, -0.4616]		
o_knn	0.0000	-0.5677 [-0.6685, -0.4465]		
lr	0.0000	-0.4941 [-0.6077, -0.3606]		
o_lr	0.0000	-0.4900 [-0.6043, -0.3558]		
rf	0.0000	-0.5604 [-0.6625, -0.4378]		
o_rf	0.0000	-0.4530 [-0.5732, -0.3136]		
svm	0.0000	-0.4686 [-0.5864, -0.3313]		
o_svm	0.0000	-0.3651 [-0.4980, -0.2154]		

The five consistent metrics all exhibit moderate correlation with the relative F1-score on average with slightly higher values for the Cluster Measure. Cross classification, CSTest, and pMSE, however, show less variance in the correlation across the different algorithms.

Re-tuning the hyperparameter of models seems favourable over re-using them in some cases for the consistent population fidelity measures, e.g. KSComplement for random forests. In many cases though, the distinction does not influence the resulting correlation significantly.

Tab. V illustrates the results related to H_0^C. As the Diabetes and the MNIST dataset do not contain categorical/cardinal features, the hypothesis could not be tested for metrics BNLogLikelihood, CSTest, and DiscreteKLD. These metrics take only categorical/cardinal features into account and can hence not be applied to these two datasets. Cluster measure and Cross-Classification are the only two population

TABLE V: Results of testing $H_0^C(pf, i)$: Is there a monotonic relationship between population fidelity and relative F1-score for individual datasets?

	BNLogLikelihood		**Cluster M. (avg. -0.54, sd. 0.26)**	
Dataset	p-value	Correlation / 99%-CI	p-value	Correlation / 99%-CI
Adult	0.0000	0.4413 [0.3316, 0.5393]	0.0000	-0.3848 [-0.4892, -0.2696]
Bank	0.0000	0.3961 [0.2818, 0.4992]	0.0002	-0.1827 [-0.3042, -0.0555]
Diabetes	n/a	n/a	0.0000	-0.7347 [-0.7887, -0.6693]
MNIST	n/a	n/a	0.0000	-0.9121 [-0.9314, -0.8876]
Titanic	0.0000	0.3432 [0.2245, 0.4518]	0.0000	-0.4765 [-0.5701, -0.3706]
	ContinuousKLD		**Cross Class. (avg. 0.62, sd. 0.17)**	
Adult	0.0000	0.2851 [0.1625, 0.399]	0.0000	0.4615 [0.3539, 0.557]
Bank	0.1435	-0.0733 [-0.2, 0.0558]	0.0000	0.5498 [0.4533, 0.6336]
Diabetes	0.0000	0.7293 [0.6629, 0.7843]	0.0000	0.7750 [0.7179, 0.8217]
MNIST	0.0000	0.8972 [0.8688, 0.9197]	0.0000	0.8574 [0.8191, 0.8881]
Titanic	0.0000	0.3787 [0.263, 0.4838]	0.0000	0.4475 [0.3384, 0.5448]
	CSTest		**DiscreteKLD**	
Adult	0.0000	0.3315 [0.212, 0.4413]	0.0546	0.0962 [-0.0328, 0.222]
Bank	0.0002	-0.1833 [-0.3047, -0.0561]	0.0002	-0.1849 [-0.3062, -0.0577]
Diabetes	n/a	n/a	n/a	n/a
MNIST	n/a	n/a	n/a	n/a
Titanic	0.0000	0.2178 [0.0918, 0.3369]	0.0000	0.4326 [0.3219, 0.5316]
	GMLogLikelihood		**KSComplement**	
Adult	0.0578	-0.0949 [-0.2208, 0.034]	0.0000	0.4333 [0.3227, 0.5322]
Bank	0.1411	-0.0737 [-0.2004, 0.0554]	0.0980	0.0828 [-0.0462, 0.2092]
Diabetes	0.0000	0.3552 [0.2375, 0.4626]	0.0000	0.6148 [0.5279, 0.6889]
MNIST	0.0000	-0.3946 [-0.4979, -0.2803]	0.0000	0.9253 [0.9044, 0.9419]
Titanic	0.0000	0.2151 [0.089, 0.3345]	0.0000	0.4727 [0.3664, 0.5668]
	pMSE			
Adult	0.0000	-0.2457 [-0.3628, -0.1209]		
Bank	0.2574	0.0568 [-0.0723, 0.184]		
Diabetes	0.0000	-0.7424 [-0.7951, -0.6787]		
MNIST	0.0000	-0.9197 [-0.9374, -0.8972]		
Titanic	0.0000	-0.4683 [-0.563, -0.3615]		

fidelity metrics for which the corresponding null hypothesis H_0^C could be consistently rejected for all datasets and the alternative hypothesis could be assumed. The correlation with the relative F1-score varies a lot across the datasets, even for these two metrics showing standard deviations of 0.26 (Cluster Measure) and 0.17 (Cross Classification), respectively. Overall, the strongest correlation can be observed for the MNIST dataset with only GMLoglikelihood (and the inapplicable metrics) being not strongly correlated with relative F1-performance. The correlation appears to be weakest for the Bank dataset across all metrics, which, for some metrics, is even the single dataset for which a correlation is not statistically significant (ContinuousKLD, KSComplement, pMSE).

V. DISCUSSION

In the following, we discuss the results and their implications on the suitability of population fidelity for estimating utility, provide recommendations for practitioners, and elaborate on the limitations and validity of the study.

A. Suitability of Population Fidelity to Estimate Utility

As outlined in Sec I, the motivating research question of the presented study is to which degree population fidelity can estimate the performance of classification models trained on synthetic data. The experiment explained in the previous sections therefore measures population fidelity with several metrics and checks for correlations with the relative F1-score.

Most population metrics exhibit moderate correlations with relative F1 performance. Only BNLogLikelihood and and ContinuousKLD show weak correlations and GMLog-likelihood fails to show statistically relevant correlations. It must be stated though that even moderate levels of correlation are insufficient for estimating utility. As can be seen in Fig. 4, the data points scatter considerably. Models with a relative F1-score of around 0.8 have a corresponding Cluster Analysis value between 1 and 18, a Cross-Classification score between 0.13 and 0.61, and a pMSE score between 0.01 and 0.17. For reliable utility estimations, this variance is too large.

Cluster Analysis, Cross-Classification, CSTest, KSComplement, and pMSE appear relatively robust against the choice of learning algorithm used for the classifier as the evidence provided for H_0^B shows. CSTest and pMSE exhibit a little less variance in the correlation than Cluster Analysis, Cross-Classification, and KSComplement.

The variance is even lower if we only consider the results for models that were fine-tuned newly in step 5, i.e., when new hyperparameters were computed (see Sec. III-D). In general, in contrast to previous results from similar studies, the results indicate that, in most cases, computing new hyperparameters leads to better (or at least equally good) results in terms of correlation. This is a positive result as the motivation for having population fidelity as a utility estimator is to avoid training a model on real data in the first place and tuning would need to happen based on synthetic training data anyway.

More influential to the degree of correlation than the learning algorithm used seems to be the datasets themselves. The dataset-specific results (H_0^C) show much more variance. For the Diabetes and MNIST datasets, five out of six metrics that can deal with datasets of only numerical features have a strong correlation with the relative F1-performance and score highest for MNIST and second-highest for Diabetes. For the Bank dataset, only for BNLogLikelihood and Cross Classification a moderate correlation was observed while ContinuousKLD and GMLogLikelihood indicated weak inverse correlations. Overall, there is some disagreement between the metrics in terms of correlation which makes us assume that certain dataset properties allow different metrics to estimate utility less or more accurately. As the Diabetes and MNIST datasets show high correlations, a first point might be to clarify the influence categorical and ordinal features have on population fidelity metrics. However, other aspects, like the distribution of features or, the relevance of a feature for the resulting classification need to be investigated further as such a detailed investigation was beyond the scope of this study.

A fact that complicates the effective use of population fidelity metrics to estimate utility is the number of parameters that can be changed. Many of the metrics make use of machine learning themselves, such as pMSE makes use of classification to compute propensity scores, or Cluster Analysis computes clusters in the data merged from original and synthetic data. Therefore, the specific implementation of a population fidelity metric can be influenced by the choice

of learning algorithm, model parameters, hyper-parameters, and settings for training and evaluation. Although literature sometimes recommends certain settings, parameter values, or algorithms, these recommendations are far from complete and do not seem to be evaluated empirically. For pMSE, for example, the literature suggests fitting a logistic regression model to the data while other classification models are, of course, possible (as long as they express class membership as a value that can be interpreted as probability). However, specific model parameters (like, the degree of the polynomial), hyperparameters (e.g., regularization strength), or training and evaluation settings are all parameters that can influence how well the resulting metric estimates the utility of the dataset at hand. A sloppily trained, underfitting model could falsely indicate high-fidelity, synthetic data while, in reality, it does simply not represent a good effort to tell synthetic from original data.

This performance competes with the level of computational complexity that is affordable and reasonable to compute population fidelity. Considering, for example, all the relevant settings, parameters, and hyperparameters in Cross-Classification, ideally as average over models trained with different algorithms, might be too expensive. In the end, one goal of being able to estimate the utility data of synthetic data is to avoid potentially expensive iterations of training the intended model. This reduction in cost and effort should not be made null and void by overly expensive estimators.

B. Recommendations for Practitioners

For estimating the utility of synthetic data based on their population fidelity, the results lead to the recommendation to take fidelity scores with a pinch of salt. They are too imprecise to infer a certain classification performance of potential classification models trained on the evaluated synthetic data.

This does not mean that population fidelity metrics are useless in this context. Cluster Measure and Cross-Classification, and to a somewhat lesser extent, CSTest, KSComplement, and pMSE, can certainly be used to point out qualitative utility differences between synthetic datasets based on the same original dataset. In particular, if classification models trained on synthetic data already exist and new synthetic data needs to be generated (for example, due to unsatisfactory model performance), a comparison of population fidelity scores might be informative to steer the generation efforts.

However, practitioners should be aware of the lack of quantitative information that those metrics currently provide. An increase/decrease in population fidelity measured with any metric cannot, with current techniques, be translated into a proportional change in utility. In addition, population fidelity scores are not comparable across different datasets. Generic scores and any assurances or agreements on synthetic data quality based on them, for example, offered by parties providing synthetic data generation services, should be scrutinized carefully.

In order to provide more accurate utility estimations, research needs to investigate the influence of dataset characteristics on population fidelity measures and develop recommendations on how to measure utility in different scenarios.

C. Limitations & Validity

We limited the experiment in several aspects. Firstly, we only consider structured data consisting of independent data points, i.e., tabular data without any associations between data points as they would exist, e.g., in time series. The rationale for this was to keep the scope of the study at a manageable scope while covering a practically relevant type of data. Other types of data would require different ways of expressing classification performance and fidelity as well as other techniques to generate synthetic data, e.g. time series [30]. We will address other types of data in the near future.

Secondly, the experiment focuses on classification tasks only, again, to keep the study scope manageable. Future work will include regression as intended task, unsupervised learning, and forecasting for time series data.

Thirdly, we limited the data analysis to a correlation analysis towards the relative F1-score, ignoring other performance measures or alternatives to represent the performance difference between original and synthetic training data. Other measures will be considered in our future work.

The restriction to five datasets as subjects of the study may pose a threat to the external validity of the results. We believe, however, that the discovered challenges with using population fidelity metrics as estimators for utility are not limited to the sample but that they, in fact, may be generalized to a large number of other datasets. The datasets were picked based on availability, popularity in the machine learning community, and technical criteria (see Sec. III-A) without any knowledge about their suitability for synthetic data generation or their fidelity. We therefore consider any selection bias towards the desired results highly unlikely.

The lack of recommended values for the parameters of the considered population fidelity metrics was already explained in Sec. V-A. This naturally forms a threat to construct validity, together with the limited empirical evidence on the "right" way to parameterize the metrics. We followed advice from the literature as far as possible in using and implementing them. Design/parameterization choices are documented in the replication package for other researchers to review and repeat the experiment with different settings. The same holds for the choice of the relative F1-score as the classification performance measure, which can easily be replaced by others.

VI. CONCLUSION

Synthetic (training) data have to meet several quality attributes. In many scenarios, they should protect sensitive information contained in original data. If used as training data in machine learning, synthetic data must show high utility, i.e. lead to models of high predictive performance. In this article, we addressed the question to which degree population fidelity metrics can be applied to estimate this utility for classification models trained on synthetic data.

The first contribution of the article is the results of an experiment that analyses the correlation between nine different population fidelity metrics and the F1 performance of classification models trained on synthetic data based on five different datasets. As the results suggest, those metrics are too weakly correlated to serve as estimators in general but some of them can be used to indicate trends in utility among different synthetic datasets based on the original data. The second contribution is an experimental framework that enables other researchers to easily investigate similar correlations with more datasets, other population fidelity metrics, and different performance measures.

The results so far show that the degree of correlation depends a lot on the datasets and their characteristics. Future work needs to investigate which, and how, data characteristics influence the ability of population fidelity metrics to estimate utility. Only then, reliable estimations of synthetic data utility based on population fidelity will be possible.

REFERENCES

[1] C. Arnold and M. Neunhoeffer, "Really Useful Synthetic Data – A Framework to Evaluate the Quality of Differentially Private Synthetic Data," Oct. 2021. [Online]. Available: http://arxiv.org/abs/2004.07740

[2] J. Jordon, L. Szpruch, F. Houssiau, M. Bottarelli, G. Cherubin, C. Maple, S. N. Cohen, and A. Weller, "Synthetic data – what, why and how?" 2022.

[3] J. Taub, M. Elliot, and J. W. Sakshaug, "The Impact of Synthetic Data Generation on Data Utility with Application to the 1991 UK Samples of Anonymised Records," *Transactions on Data Privacy*, vol. 13, no. 1, pp. 1–23, Apr. 2020, publisher: Instituto de Estudios Documentales sobre Ciencia y Tecnología (IEDCYT), Ciencia y Tecnología.

[4] Lei Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling Tabular data using Conditional GAN," Oct. 2019, arXiv:1907.00503 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1907.00503

[5] "Unity's Danny Lange explains why synthetic data is better than the real thing at Transform 2021," https://venturebeat.com/ai/unitys-danny-lange-explains-why-synthetic-data-is-better-than-the-real-thing-at-transform-2021-2/, Jul. 2021.

[6] S. I. Nikolenko, *Synthetic data for deep learning*. Springer, 2021.

[7] J. Snoke, G. M. Raab, B. Nowok, C. Dibben, and A. Slavkovic, "General and Specific Utility Measures for Synthetic Data," *Journal of the Royal Statistical Society Series A: Statistics in Society*, vol. 181, no. 3, pp. 663–688, 03 2018. [Online]. Available: https://doi.org/10.1111/rssa.12358

[8] F. K. Dankar, M. K. Ibrahim, and L. Ismail, "A multi-dimensional evaluation of synthetic data generators," *IEEE Access*, vol. 10, pp. 11 147–11 158, 2022.

[9] M.-J. Woo, J. Reiter, A. Oganian, and A. Karr, "Global measures of data utility for microdata masked for disclosure limitation," *Journal of Privacy and Confidentiality*, vol. 1, 04 2009.

[10] P. R. Rosenbaum and D. B. Rubin, "The central role of the propensity score in observational studies for causal effects," *Biometrika*, vol. 70, no. 1, pp. 41–55, 04 1983. [Online]. Available: https://doi.org/10.1093/biomet/70.1.41

[11] A. Goncalves, P. Ray, B. Soper, J. Stevens, L. Coyle, and A. P. Sales, "Generation and evaluation of synthetic patient data," *BMC medical research methodology*, vol. 20, no. 1, pp. 1–40, 2020.

[12] "BNLogLikelihood," https://docs.sdv.dev/sdmetrics/metrics/metrics-in-beta/data-likelihood/bnloglikelihood, [Online; Accessed 28th Feb 2024.].

[13] "GMLikelihood," https://docs.sdv.dev/sdmetrics/metrics/metrics-in-beta/data-likelihood/bnloglikelihood, [Online; Accessed 28th Feb 2024.].

[14] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951, iSBN: 0003-4851 Publisher: JSTOR.

[15] T. Van Erven and P. Harremos, "Rényi divergence and Kullback-Leibler divergence," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 3797–3820, 2014, iSBN: 0018-9448 Publisher: IEEE.

[16] G. K. Kanji, *100 Statistical Tests*, 3rd ed. Sage Publications, 2006.

[17] F. K. Dankar and M. Ibrahim, "Fake it till you make it: Guidelines for effective synthetic data generation," *Applied Sciences*, vol. 11, no. 5, 2021. [Online]. Available: https://www.mdpi.com/2076-3417/11/5/2158

[18] S. James, C. Harbron, J. Branson, and M. Sundler, "Synthetic data use: exploring use cases to optimise data utility," *Discover Artificial Intelligence*, vol. 1, no. 1, p. 15, Dec. 2021. [Online]. Available: https://doi.org/10.1007/s44163-021-00016-y

[19] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks," Jun. 2014. [Online]. Available: http://arxiv.org/abs/1406.2661

[20] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, Oct. 2020. [Online]. Available: https://dl.acm.org/doi/10.1145/3422622

[21] "Adult dataset," *UCI Machine Learning Repository*, 1996. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/adult

[22] "Banknote authentication Data Set," *UCI Machine Learning Repository*, 2013. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/banknote+authentication

[23] "Diabetes Dataset." [Online]. Available: https://www.kaggle.com/datasets/mathchi/diabetes-data-set

[24] "MNIST in CSV." [Online]. Available: https://www.kaggle.com/datasets/oddrationale/mnist-in-csv

[25] "Titanic - Machine Learning from Disaster." [Online]. Available: https://kaggle.com/competitions/titanic

[26] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," Jul. 2019. [Online]. Available: http://arxiv.org/abs/1907.10902

[27] L. Xu, "Synthesizing tabular data using conditional GAN," Thesis, Massachusetts Institute of Technology, 2020. [Online]. Available: https://dspace.mit.edu/handle/1721.1/128349

[28] "Synthetic data metrics," Jan. 2023. [Online]. Available: https://docs.sdv.dev/sdmetrics/

[29] N. J. de Vos, "kmodes categorical clustering library," 2015. [Online]. Available: https://github.com/nicodv/kmodes

[30] Z. Lin, A. Jain, C. Wang, G. Fanti, and V. Sekar, "Using GANs for sharing networked time series data: Challenges, initial promise, and open questions," in *Proceedings of the ACM Internet Measurement Conference*, ser. IMC '20. New York, NY, USA: ACM, 2020, p. 464–483. [Online]. Available: https://doi.org/10.1145/3419394.3423643