Relational Or Single: A Comparative Analysis of Data Synthesis Approaches for Privacy and Utility on a Use Case from Statistical Office

Manel Slokom[1,2(✉)], Shruti Agrawal[1], Nynke C. Krol[1], and Peter-Paul de Wolf[1]

[1] Statistics Netherlands, The Hague, The Netherlands
{s.agrawal,nc.krol,pp.dewolf}@cbs.nl
[2] Centrum Wiskunde & Informatica, Amsterdam, The Netherlands
m.slokom@cwi.nl

Abstract. This paper presents a case study focused on synthesizing relational datasets within Official Statistics for software and technology testing purposes. Specifically, the focus is on generating synthetic data for testing and validating software code. Our study conducts a comprehensive comparative analysis of various synthesis approaches tailored for a multi-table relational database featuring a one-to-one relationship versus a single table. We leverage state-of-the-art single and multi-table synthesis methods to evaluate their potential to maintain the analytical validity of the data, ensure data utility, and mitigate risks associated with disclosure. The evaluation of analytical validity includes assessing how well synthetic data replicates the structure and characteristics of real datasets. First, we compare synthesis methods based on their ability to maintain constraints and conditional dependencies found in real data. Second, we evaluate the utility of synthetic data by training linear regression models on both real and synthetic datasets. Lastly, we measure the privacy risks associated with synthetic data by conducting attribute inference attacks to measure the disclosure risk of sensitive attributes. Our experimental results indicate that the single-table data synthesis method demonstrates superior performance in terms of analytical validity, utility, and privacy preservation compared to the multi-table synthesis method. However, we find promise in the premise of multi-table data synthesis in protecting against attribute disclosure, albeit calling for future exploration to improve the utility of the data.

Keywords: Relational data · data synthesis · inference · constraints · Single vs Multi

The views expressed in this paper are those of the authors and do not necessarily reflect the policy of Statistics Netherlands.

J. Domingo-Ferrer and M. Önen (Eds.): PSD 2024, LNCS 14915, pp. 403–419, 2024.
https://doi.org/10.1007/978-3-031-69651-0_27

1 Introduction

In recent years, the need for synthetic data has gained a lot of attention, particularly in official statistics. Synthetic data offers a viable solution to privacy concerns, allowing organizations to share and utilize data without compromising sensitive information. Various use cases for synthetic data include data release, testing, education, data augmentation, and bias mitigation [7]. For example, synthetic data can be publicly released to enhance transparency, increase collaboration among parties, or for educational purposes. In this paper, we investigate the use of synthetic data for testing technologies/algorithms. We collaborate with the Social Security department at Statistics Netherlands, exploring how synthetic data can validate and test their implementations effectively.

The dataset under investigation comprises two tables linked by a one-to-one relationship. This structure presents unique challenges in generating synthetic data that maintains the integrity of both inter-table and intra-table relationships. We aim to generate synthetic data while preserving these connections, ensuring the synthesized data remains faithful to the real dataset's structure. The generated synthetic data has to adhere to the constraints provided by the software engineers of social security. Several approaches have been proposed in the literature for generating single synthetic data, such as data distortion by probability distribution [14], synthetic data by multiple imputation [19], and synthetic data by Latin Hypercube Sampling [2]. In [5], the authors proposed an empirical evaluation of different machine learning algorithms, e.g., classification and regression trees (CART), bagging, random forests, and Support Vector Machines for generating synthetic data. For multi-table data synthesis, the authors in [18] proposed the Conditional Parameter Aggregation method for synthesizing relational data, emphasizing the need to account for the influence of child tables on parent tables. In [13] the authors present Incremental Relational Generator (IRG), which uses GANs to synthetically generate interrelated tables. In our study, we compare state-of-the-art single and multi-table data synthesis approaches. We use two open public toolkits: SynthPop[1], and the Synthetic Data Vault (SDV)[2]. As we generate synthetic data, it is crucial to evaluate its utility and assess disclosure risks. For utility measure evaluation, we compare the performance of several regression models trained on real and synthetic datasets and tested on real data. Regarding disclosure risk, we focus on measuring the potential of the different synthetic datasets to protect against attribute disclosure, ensuring that sensitive information remains protected.

Our main research question examines *how can we generate (relational vs. single) synthetic data that protects users' private data while maintaining the utility of the data for testing technologies/algorithms purposes?* In essence, we aim to answer the following research questions through this study:

– *SubRQ1*: How can we create relational synthetic data? To what extent can we generate synthetic data by combining data from sources as a single table?

[1] https://synthpop.org.uk/.
[2] https://sdv.dev/.

- *SubRQ2*: Which method achieves the best synthesis quality on the grounds of analytical validity, utility, and privacy risks?
- *SubRQ3*: What are the risks of disclosure from different synthesis approaches?

2 Background and Related Work

In this section, we provide a brief overview of existing techniques for synthetic data generation and measuring the disclosure risk.

2.1 Synthetic Data Generation

Synthetic data have been around for quite some time in the world of Statistical Disclosure Control (SDC). However, in recent years a lot of renewed interest in synthetic data has developed. Partly because of new computational possibilities but just as well in view of new regulations like the European General Data Protection Regulation (GDPR, [8]). Synthetic data are available in two flavors: fully synthetic and partially synthetic. In the current paper, we will focus on fully synthetic data: all attributes of all records are synthesized based on the real data [4,6].

Single-table Synthesis. Several approaches are available for generating synthetic single data, including multiple imputations [19], Latin Hypercube Sampling [2], machine learning approaches like classification and regression trees (CART), bagging, random forests, and Support Vector Machines [5]. The authors showed that data synthesis using CART results in synthetic data that provides reliable predictions and low disclosure risks. CART, being a non-parametric method, helps in handling mixed data types and effectively captures complex relationships between attributes [5]. Other approaches involve generative models like General Adversarial Networks, especially CTGAN, and Tabular Variational Autoencoders (TVAE) [25], and TableGAN [17].

Multi-table Synthesis. for relational data, in [18], the authors introduced Conditional Parameter Aggregation (CPA). CPA addresses the challenge of maintaining relationships between tables in a relational database. It operates by iterating through each record in a parent table and performing a conditional lookup to gather data from all child tables that reference it. In [13] the authors present Incremental Relational Generator (IRG), which uses GANs to synthetically generate a table-by-table synthetic relational database. In [10], the authors propose FakeDB, a general framework to generate synthetic data that preserves a wide variety of semantic integrity constraints as well as a broad set of statistical properties, across an entire relational database. In [9], the authors have conducted a comprehensive study for applying GAN to relational data synthesis.

2.2 Disclosure Risk Measures and Threat Model Formulation

Disclosure risk is defined as the risk that a user or an attacker can use the protected data to derive sensitive information on an individual among those in the real data [3]. Different types of disclosure are mentioned in the context of statistical disclosure control [12]: identity disclosure, attribute disclosure, and inferential disclosure. In the context of fully synthetic data, identity disclosure is often considered to be a non-threat. However, depending on the accuracy of the generating process, still a (very) small identification risk could remain. Moreover, attribute disclosure or inference disclosure is very well possible with synthetic data. Recent developments to estimate attribute disclosure in synthetic data include the so-called Correct Attribution Probability (CAP) [11,15,24]. CAP assumes that the attacker knows the values of a set of key attributes for an individual in the original data set, and aims to learn the respective value of a target sensitive attribute. From machine learning, in [22,23], the authors discuss a use case of synthetic data related to releasing trained machine learning models. They investigate privacy risks associated with model inversion attribute inference attacks. In our view, it is still not clear how protective synthetic data are in terms of statistical disclosure. Indeed, in [21] it is stated that *'disclosure risk measures for synthetic data after its generation are still ad-hoc, and a more formal framework is needed for measuring the risk of attribute disclosure'.*

Table 1. The threat model that we address in our paper.

Component	Description
Adversary: Objective	To infer if a target individual has received assistance.
Adversary: Resources	The attacker has a pre-trained classifier or a subset of data to train one. The subset of data can also be the synthetic data.
Vulnerability: Opportunity	Possession of clean-text data and the ability to infer individual's sensitive data
Countermeasure	Make access to real data unreliable

In our work, the measuring and mitigation of privacy risks of synthetic data are founded on the concept of the threat model. A threat model is a theoretical framework that defines what constitutes a privacy violation or breach, such as linking identity to a record, resulting in the leakage of sensitive information. In a widely recognized schema proposed by [20], a threat model comprises two key components: the *adversary* and the *vulnerability*. First, *the adversary's objective* defines what the adversary seeks to accomplish, with potential goals including re-identification attacks, inference attacks, or membership inference attacks. Second, *the adversary's resources* define what the adversary can do, encompassing different levels of knowledge and resources. We focus on black-box scenarios, where the adversary has limited knowledge of the system. Next, *the*

vulnerability-opportunity determines what an adversary is willing to do. Finally, *the vulnerability-countermeasure* component suggests potential solutions to protect against specific attacks. In Table 1, we provide details about the threat model that we aim to address in this paper.

3 The Applied Synthesis Approaches

There are different ways to create fully synthetic data. Based on the number of tables to be synthesized, we have investigated single-table and multi-table synthesis approaches. Our work explores different synthesis methods that accommodate various types of data structures. Since the relationship between the tables in the current scope is one-to-one (1-1), it is possible to merge them and synthesize them as a single table. Alternatively, the two tables can be synthesized independently using a multi-table synthesis method.

Data synthesis is a two-step process [6]. The first step consists of training a model using the real data to learn the joint distributions. The second step involves generating synthetic values for each attribute in turn, using the estimated model for the conditional distribution of that attribute, and using as input the synthetic values already produced for the previous attributes.

3.1 Single Data Synthesis Approach

In this section, we describe the single table synthesis approaches we use in our experiments. We select two state-of-the-art approaches: (1) CART is a fully conditional specification (FCS) method, (2) TVAE is a generative model.

Synthetic Data Generation Using CART. CART takes as a parameter the matrix of predictors to model the data and sample the synthesized records. The first column to be synthesized is sampled from the distribution in the real data. The sequence of the synthesis of columns and the predictors of each column are important hyperparameters. After fitting the decision tree to a specific set of inputs, a synthesized value is generated by randomly selecting an item from the leaf node where the input parameters fall. This approach maintains reasonable analytical validity while ensuring that exact replicas of real data are not produced. However, it's crucial to tune the hyperparameters of the tree to prevent overfitting, which helps mitigate privacy risks. In our experiments, we use the Synthpop package in R that offers a sequential synthesis approach [16]. Synthpop provides sampling methods based on linear models or decision tree-based models. We use CART in our experiments as it has shown to perform the best in the literature [5].

Synthetic Data Generation Using TVAE. This is a neural network based on encoder-decoder architecture adapted for tabular data. This synthesizer uses the variational-autoencoder architecture to learn a model from real data and create synthetic data [25]. The encoder is a neural network that outputs the

parameters of the normal distribution of the latent space parameters. The latent space parameters from the normal distribution are further fed into the decoder. Thus, TVAE completely hides the input parameters from being passed into the synthetic data. We note that even though TVAE underperforms CTGAN, it is chosen for this study since it takes time of the order of 1/10 that by CTGAN [25]. TVAE is implemented in the SDV Python package with the default network architecture.[3]

3.2 Relational Data Synthesis Approach

In a relational database, tables are often interconnected, with one table referencing records in another. To effectively synthesize relational datasets while preserving their complex dependencies, we use the Hierarchical Modeling Algorithm (HMA).[4] HMA recursively models the relationships across all tables in a dataset, ensuring that the generative process respects the hierarchical and relational structure inherent in the data. This approach involves training individual models for each table, conditioned on the context provided by their related tables. By capturing how fields in different tables interrelate, HMA constructs a comprehensive representation of the entire dataset. During the data generation phase, HMA sequentially generates synthetic data for each table, maintaining the learned dependencies and ensuring consistency across the dataset. This method allows for the creation of realistic and coherent synthetic data, suitable for various downstream applications such as testing and analysis, while protecting the privacy of the real data.

4 Experimental Setup

In this section, we describe our data, as well as the privacy, and utility measures.

4.1 Data Set

Our data consists of two tables. The first table, *GBA*, contains records for 27 million unique individuals in the Netherlands. Each record corresponds to a unique individual and includes basic information such as an ID (*RINPERSOON*), country of birth (*GBAGEBOORTELAND*), year of birth (*GBAGEBOORTEJAAR*), gender (*GBAGESLACHT*), and the year of birth of the person's parents (*GBAGEBOORTEJAARMOEDER* and *GBAGEBOORTEJAARVADER*). Note that there are missing values for some records, specifically 38% missing values for the mother's birth year and 41% for the father's birth year.

The second table, *Bij*, contains records for 0.47 million unique individuals who have received some form of social benefit. Each record provides information about whether the individual received one of three kinds of benefits (*bijstand*,

[3] https://docs.sdv.dev/sdv/single-table-data/modeling/synthesizers/tvaesynthesizer.
[4] https://docs.sdv.dev/sdv/multi-table-data/modeling/synthesizers/hmasynthesizer.

ioaw, *ioaz*), an ID (*RINPERSOON*), dates when the benefits first started (*aanvangbijstand*, *aanvangioaw*, *aanvangioaz*), and the start and end dates for the benefits in the corresponding year (*Aanvbijstandpersoon*, *Eindbijstandpersoon*).

For the purpose of single-table synthesis, these two datasets were inner joined on the ID. Due to the complexity and time needed to run the experiments, especially for GAN, we randomly selected 50K individuals. The 50K records are used to generate both single and multi-table synthetic data. The relationships between the tables in our dataset are depicted in Fig. 4 (Appendix Sect. A). For multi-table synthesis, the tables were joined later on for the purposes of analytical, utility, and privacy assessments.

4.2 Measuring Utility

As discussed earlier, the main purpose of using synthetic data is to test technologies/algorithms. The first step in this evaluation is to validate the utility of synthetic data compared to real data. In our case study, this involves maintaining the same data structure, uni- and bi-variate distributions, and respecting the constraints (cf. Sect. 5). In this section, we examine the effectiveness of a linear regression model in predicting the duration (measured in days) individuals receive benefits. We adopt the $\mathcal{TSTR}$ (train on synthetic and test on real) and $\mathcal{TRTR}$ (train on real and test on real) strategies. For a fair comparison, linear regression models are trained on real and synthetic datasets, respectively, and then tested on exclusive real test data randomly sampled from the entire population, totaling approximately 2000 records.

Outcome Attribute and the Other Attributes. With the dataset available in official statistics, it is of interest to determine which factors influence the duration of time a person receives social benefits. This outcome attribute $Days.benefits$ (number of days people are on benefits) is derived from available dates in the dataset. Figure 1 illustrates the distribution of the outcome attribute in both real and synthetic datasets. The distribution generated by the CART method shows the closest resemblance to the real data. The Multi method follows, while the distribution produced by the TVAE method appears noticeably distinct compared to the real data distribution.

To predict the number of days on benefits, we utilize attributes such as year of birth (*GBAGEBOORTEJAAR*), gender (*GBAGESLACHT*), and the presence of other benefits (*ioaw*). These attributes are used to assess model performance across all datasets. It is important to note that our selection of real data included only individuals receiving benefits at a specific date. Some attributes contained significant missing values or were not relevant for modeling our outcome.

We opt for a simple linear regression model to illustrate the disparities between real and synthetic data. This model is transparent and widely used in statistical research. We depict our formula in Eq. 1:

$$\text{Days.Benefits}_i = \alpha + \beta_1 \text{GBAGEBOORTEJAAR}_i + \beta_2 \text{GBAGESLACHT}_i + \beta_3 \text{ioaw}_i \quad (1)$$

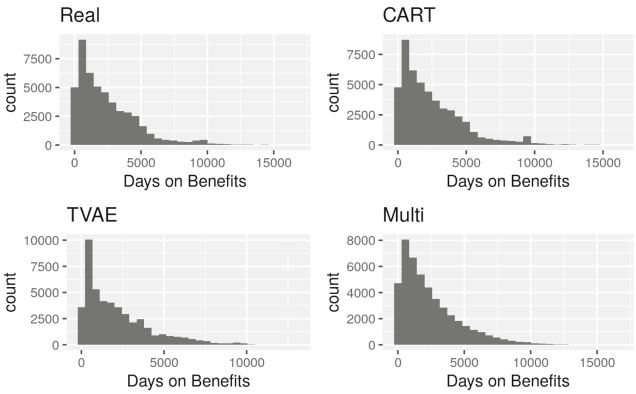

Fig. 1. The distribution of the number of days people are on benefits for the real data (top left) and synthetic data generated using the CART, TVAE, and Multi-table.

We use the Root Mean Squared Error (RMSE) as a performance indicator for the regression model. Lower values indicate a better fit. More results are present in the Appendix (Sect. A.2)

4.3 Measuring Attribute Inference Attack

Following our threat model (cf. Sect. 2.2), we evaluate attribute inference attacks using three machine learning algorithms. In this section, we describe the subset of data available to the attacker, the inference attack models, and the metrics used to measure the success of an attack.

Subset of Data. In our experiments, we assume the attacker has access to a subset of data or a pre-trained model. This subset, possibly obtained through scraping or as an internal actor. The subset of data includes information on gender, birth date, parents' birth dates, and country of origin. Additionally, the attacker has a dataset of 10K target individuals for whom they aim to infer whether they received assistance (Bijstand). The binary outcome attribute "bijstand" (1 for received assistance, 0 otherwise) is notably unbalanced. The attacker uses this subset of data to train machine learning classifiers, which are then applied to the target data for inference. We also explore different sizes of attacker training data to determine the minimum number of records needed for a successful attack.

Machine Learning Models. *Naive Bayes (NB)* is a simple yet powerful probabilistic machine learning model based on Bayes' theorem. Second, *Decision Tree (DT)* is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. Third, *XGBoost (GBC)* is a powerful and efficient implementation of the gradient boosting framework [1].

To evaluate the success of our attribute inference attacks, we use: The *F1-score macro-average* measures a test's accuracy by considering both precision

and recall. It is the harmonic mean of precision and recall, with an F1 score ranging from 0 (worst) to 1 (perfect precision and recall). The macro-average approach calculates the F1-score independently for each class and then averages them, treating all classes equally. The *MCC* takes into account true and false positives and negatives, providing a balanced measure even if the classes are of very different sizes. It returns a value between −1 and +1, where 1 indicates perfect prediction, 0 indicates random prediction, and −1 indicates total disagreement between prediction and observation. We repeat the attribute inference attack experiments ten times and report the average and standard deviation.

5 Analytical Validity

Analytical validity of synthetic data is crucial in the evaluation process, ensuring its suitability for software testing. This involves maintaining identical data structures and types across all three synthetic datasets compared to the real data. Additionally, we verify that our synthetic datasets adhere to the rules and constraints provided by the social security team for our testing purposes. There are 6 conditional constraints as listed in Table 2, which specify the expected behavior of numerical date columns when the corresponding binary column is either 1 or 0. In the real data, these constraints are observed for records where the binary column is either False or True. The CART synthetic data fully satisfies these constraints. However, the TVAE and Multi synthesis models fail to learn these constraints. For instance, only 4.1% of records in the real data where bijstand = False meet constraint 1, whereas only 4.0% of records in the TVAE and Multi synthetic data adhere to this constraint, compared to 100% compliance in the real data and CART synthesized data. Similar discrepancies are observed for constraints 2 through 6 as well (further details are in Sect. A.1).

Table 2. Constraint check on the real vs. CART, TVAE, and Multi-Table Synthesized Data (%).

Constraint	Real	CART	TVAE	Multi	Priori
1. No Date for Bijstand when Bijstand = 0	100	100	4.0	4.0	4.1
2. No Date for Ioaw when Ioaw = 0	100	100	96.1	96.2	96.1
3. No Date for Ioaz when Ioaz = 0	100	100	99.5	99.5	99.5
4. Date for Bijstand when Bijstand = 1	100	100	96.1	95.7	95.9
5. Date for Ioaw when Ioaw = 1	100	100	3.3	4.1	3.8
6. Date for Ioaz when Ioaz = 1	100	100	0.0	0.8	0.4

6 Utility Measures

In this section, the results from our analysis on the utility of the synthetic datasets will be presented. In Table 3, the coefficients (betas) and their standard errors from the linear regression model trained on real data and synthetic data. Large differences can be observed, with the linear model trained on synthetic data. Notably, the linear model trained on synthetic data by the CART approach closely resembles the coefficients observed in the model trained on real data, indicating better alignment in predictive performance compared to the other synthetic data generation methods.

Table 3. Coefficients from a linear model and their respective standard errors.

	Real		CART		TVAE		Multi	
	Coef	*Std Err*	Coef	*Std Err*	Coef	*Std Err*	Coef	*Std Err*
(Intercept)	98286.67	*1256.77*	97910.80	*1311.37*	62993.60	*1195.37*	2908.35	*1351.86*
GBAGEBOORTEJAAR	−48.71	*0.64*	−48.45	*0.67*	−31.44	*0.61*	−0.29	*0.69*
GBAGESLACHT	332.89	*19.66*	153.49	*20.32*	1328.67	*24.50*	209.12	*20.94*
ioaw	−1332.74	*231.36*	−1.36	*238.11*	−596.09	*103.39*	3.89	*54.22*

In Table 4, the Root Mean Squared Error (RMSE) for models trained on real data and synthetic data are presented. Again, we see that the RMSE for the model trained on CART data is most comparable to the RMSE for the model trained on real data, suggesting that the CART synthetic data approach provides predictions closest to those derived from real data. Next, the Multi approach performs second best, followed by TVAE.

Table 4. RMSE for models trained on real data and synthetic datasets.

	Real	CART	TVAE	Multi
RMSE	2129.42	2133.43	2291.83	2249.53

In Fig. 2, we show the predicted values of the number of days a person is on benefits against the true number of days they are on benefits. In a perfect prediction, all values would fall on a diagonal line. The predictive value of our model trained on real data can be improved, but the CART model resembles its results. On the other hand, the predicted values from the model trained on synthetic data generated by the TVAE method show a different pattern. For the Multi-method, the pattern in predicted values versus true values diverges even further from the pattern when real data is used.

7 Attribute Disclosure Risk

In this section, we provide our results of the attribute inference attack. In Table 5, we provide our results on attribute inference attack. The results show that mod-

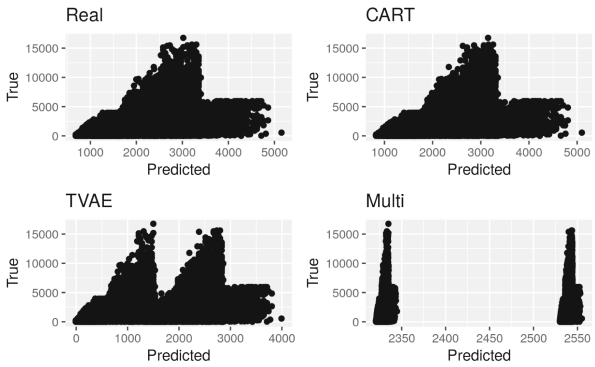

Fig. 2. Utility measure: True values of the outcome attribute in our test set, the number of days people are on benefits, and the predicted values from linear models trained on real data (top left), and on synthetic data generated by CART, TVAE, and Multi.

els trained on TVAE data achieved the highest scores across all metrics, outperforming the random classifier. This indicates a higher risk of sensitive information leakage when using TVAE-synthesized data. In contrast, models trained on CART and Multi data sets have lower scores than the random classifier, suggesting that these two approaches are more effective at protecting sensitive information, thereby providing better privacy. The performance of models trained on CART and Multi data sets demonstrates their potential for reducing data leakage and improving data privacy.

Figure 3 provides a comparison of the F1-scores macro-average for different machine learning classifiers (Random, NB, DT, and GBC) across different fractions of attacker data. We show the performance on different attacker data sizes. These figures compare the performance of attribute inference attack models trained on synthetic data (Multi, TVAE, CART) to that of a model trained on real data.

We observe that across all conditions, models trained on TVAE synthetic data have the highest F1 scores surpassing those of models trained on real data. This confirms that TVAE does not help to protect against attribute disclosure. However, looking at the performance of the models trained on CART and Multi, we see that the F1 scores are around 0.5 and below. This demonstrates that Multi and CART offer higher protection against attribute disclosure compared to that of TVAE.

Table 5. Results of the attribute inference attack are measured in terms of F1 macro and MCC. We compare the performance of a random classifier to DT, NB, and XGBoost. We compare the performance of models trained on different training data: Real, TVAE, CART, and Multi. Gray-highlighted scores indicate classifier performance lower than real data. ± denotes the standard deviation over ten runs of the experiments. Note that the test set is the same real target individuals.

Classifiers	Random		DT		NB		XGBoost	
Data Sets	*F1*	*MCC*	*F1*	*MCC*	*F1*	*MCC*	*F1*	*MCC*
Real	0.4901 ± 0.000	0.00 ± 0.000	0.5967 ± 0.0087	0.1943 ± 0.0174	0.5888 ± 0.0024	0.2821 ± 0.0041	0.5341 ± 0.0185	0.1252 ± 0.0402
TVAE	0.4901 ± 0.000	0.00 ± 0.000	**0.6168 ± 0.0009**	**0.2440 ± 0.0026**	**0.5963 ± 0.0018**	**0.2614 ± 0.0015**	**0.6169 ± 0.0089**	**0.2384 ± 0.0174**
CART	0.4901 ± 0.000	0.00 ± 0.000	0.4969 ± 0.0015	-0.0060 ± 0.0030	0.4901 ± 0.000	0.00 ± 0.000	0.4901 ± 0.000	-0.0012 ± 0.0011
Multi	0.4901 ± 0.000	0.00 ± 0.000	0.4948 ± 0.0029	-0.0097 ± 0.0058	0.4901 ± 0.000	0.00 ± 0.000	0.4900 ± 0.000	-0.0045 ± 0.0005

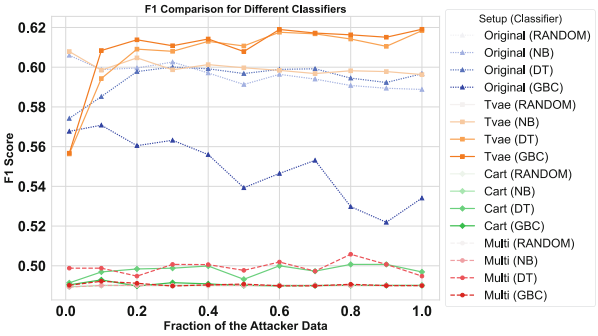

Fig. 3. Attribute inference attack measured using F1 Scores Macro on DT, NB, XGBoost (GBC) for the different synthetic data Multi, TVAE, CART.

8 Conclusion and Future Work

In this paper, we conducted a comparative analysis of different synthesis approaches, focusing on the specific challenge posed by a relational multi-table structure with a one-to-one relationship for testing technologies/algorithms. By juxtaposing single table synthesis against multi table synthesis, we aimed to discern the strengths and limitations of each method. Our approach involved merging the two tables into a single entity for single-table synthesis, facilitating a direct comparison with the multi-table synthesis technique.

Through extensive experimentation, we evaluated the efficacy of various synthesis methods in respecting constraints/rules, ensuring analytical validity, maintaining the utility of the data, and mitigating risks associated with attribute disclosure. Our findings revealed that among the single synthesis approaches, CART emerged as the most effective solution for generating synthetic data within our

particular use case. CART demonstrated superior performance in preserving the integrity of the synthesized data while meeting the constraints imposed by the analytical framework. On the other hand, the multi-table synthesis method demonstrated promise in capturing the intricate inter- and intra-relationships inherent in the data structure. While it proved effective in protecting against attribute disclosure, comparable to the performance of CART, its utility effectiveness faced limitations. This suggests that while the multi-table approach holds potential, further refinement and optimization are necessary to fully exploit the relational structure embedded within the data. Future research should focus on improving the utility of the multi-table synthesis method to ensure its practical applicability across diverse analytical frameworks and use cases.

Acknowledgments. We would like to thank the SOZ team at Statistics Netherlands for providing us with the data and knowledge of the rules. This work was partly supported by the AI, Media, and Democracy Lab, NWA.1332.20.009.

A Appendix

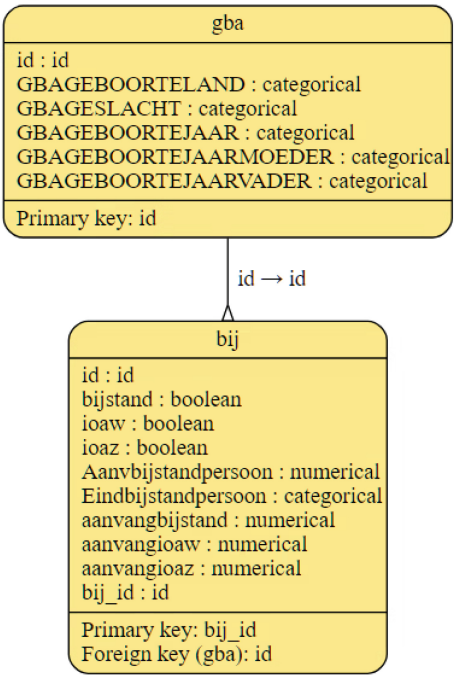

Fig. 4. The metadata of our relational data. We have two tables *gba* and *bij* that are connected through $id \rightarrow id$. The primary id of table *gba* is a foreign key in table *bij*.

A.1 Analytical Validity

Table 6 shows the counts or percentages of categories captured in the categorical or boolean columns synthetic datasets compared to the real and.

Table 6. Measurement of representation of underrepresented categories captured in the real, the CART, TVAE and multi-table synthetic datasets.

Column	Real	CART	TVAE	Multi
Country (unique codes)	218	205	61	200
bijstand = 0	4.1%	4.0%	3.4%	4.1 %
ioaw = 1	3.8%	3.8%	3.9%	4.0%
ioaz = 1	0.4%	0.4%	0.006%	0.5%

Bivariate Distribution. The distribution between column pairs in the real data is compared with those in the synthetic datasets. CART based synthesis outperforms TVAE and Multi model for all but the case of benefit (True or False) vs birth year, as shown in Fig. 5 and Fig. 6.

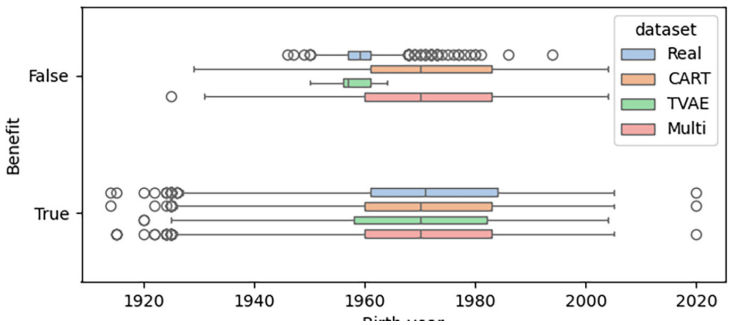

Fig. 5. Distribution of birth year in different datasets who received or did not receive the benefit. People from only a specific range of birth years did not receive the benefit in the real data. This characteristic has been very well captured by the TVAE synthetic data.

A.2 Utility

Residuals. To check the assumptions of our models, we look at the residuals (errors) of the models trained on real and synthetic data. Residuals are visualized in Fig. 7. Although some skewness is present in all plots, the model trained on synthetic data generated by the Multi-method does seem to violate the assumptions of a linear model most.

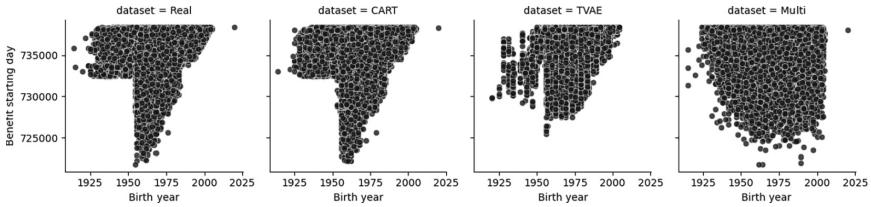

Fig. 6. Starting date of benefit vs birth year. This characteristic has been very well captured in the CART synthetic data.

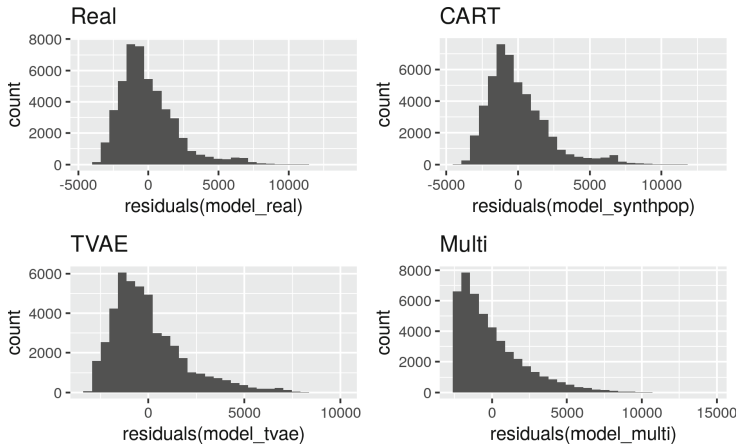

Fig. 7. Residuals of the linear models trained on real data and synthetic data generated using CART, TVAE, and Multi methods.

References

1. Chen, T., Guestrin, C.: Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794 (2016)
2. Dandekar, R.A., Cohen, M., Kirkendall, N.: Sensitive micro data protection using Latin hypercube sampling technique. In: Domingo-Ferrer, J. (ed.) Inference Control in Statistical Databases. LNCS, vol. 2316, pp. 117–125. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-47804-3_9
3. Domingo-Ferrer, J., Torra, V.: Disclosure risk assessment in statistical data protection. J. Comput. Appl. Math. **164–165**, 285–293 (2004). proceedings of the 10th International Congress on Computational and Applied Mathematics
4. Drechsler, J., Bender, S., Rässler, S.: Comparing fully and partially synthetic datasets for statistical disclosure control in the German IAB establishment panel. Trans. Data Priv. **1**(3), 105–130 (2008)
5. Drechsler, J., Reiter, J.P.: An empirical evaluation of easily implemented, nonparametric methods for generating synthetic datasets. Comput. Stat. Data Anal. **55**(12), 3232–3243 (2011)

6. Drechsler, J.: Synthetic Datasets for Statistical Disclosure Control: Theory and Implementation, vol. 201. Springer, New York (2011)
7. Nations Economic Commission for Europe, U., et al.: Synthetic data for official statistics: a starter guide (2023)
8. European Parliament and Council of the European Union: Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 april 2016 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation). OJ 2016 L 119, pp. 1–88 (2016)
9. Fan, J., Chen, J., Liu, T., Shen, Y., Li, G., Du, X.: Relational data synthesis using generative adversarial networks: a design space exploration. Proc. VLDB Endow. **13**(12), 1962–1975 (2020)
10. Gao, C., Jajodia, S., Pugliese, A., Subrahmanian, V.: FakeDB: generating fake synthetic databases. IEEE Trans. Dependable Secure Comput. 1–12 (2024)
11. Hittmeir, M., Mayer, R., Ekelhart, A.: A baseline for attribute disclosure risk in synthetic data. In: Proceedings of the 10th ACM Conference on Data and Application Security and Privacy, pp. 133–143 (2020)
12. Hundepool, A., et al.: Statistical Disclosure Control. Wiley, Hoboken (2012)
13. Li, J., Tay, Y.: IRG: generating synthetic relational databases using GANs. arXiv preprint arXiv:2312.15187 (2023)
14. Liew, C.K., Choi, U.J., Liew, C.J.: A data distortion by probability distribution. ACM Trans. Database Syst. **10**(3), 395–411 (1985)
15. Elliot, M.: Final report on the disclosure risk associated with synthetic data produced by the SYLLS team (2014). http://hummedia.manchester.ac.uk/institutes/cmist/archive-publications/reports/. Accessed 13 Oct 2023
16. Nowok, B., Raab, G.M., Dibben, C.: synthpop: bespoke creation of synthetic data in R. J. Stat. Softw. **74**(11), 1–26 (2016)
17. Park, N., Mohammadi, M., Gorde, K., Jajodia, S., Park, H., Kim, Y.: Data synthesis based on Generative Adversarial Networks. In: Proceedings of the 44th International Conference on Very Large Data Bases (VLDB Endowment), vol. 11, no. 10, pp. 1071–1083 (2018)
18. Patki, N., Wedge, R., Veeramachaneni, K.: The synthetic data vault. In: IEEE International Conference on Data Science and Advanced Analytics, pp. 399–410 (2016)
19. Rubin, D.B.: Discussion statistical disclosure limitation. J. Official Stat. **9**(2), 461–468 (1993)
20. Salter, C., Saydjari, O.S., Schneier, B., Wallner, J.: Toward a secure system engineering methodology. In: Proceedings of the Workshop on New Security Paradigms, pp. 2–10. NSPW (1998)
21. Schlomo, N.: How to measure disclosure risk in microdata? Surv. Stat. **86**, 13–21 (2022)
22. Slokom, M., de Wolf, P.P., Larson, M.: When machine learning models leak: an exploration of synthetic training data. In: Domingo-Ferrer, J., Laurent, M. (eds.) Proceedings of the International Conference on Privacy in Statistical Databases: Corrected and updated version on arXiv at:https://arxiv.org/abs/2310.08775 (2022)
23. Slokom, M., de Wolf, P.P., Larson, M.: Exploring privacy-preserving techniques on synthetic data as a defense against model inversion attacks. In: Athanasopoulos, E., Mennink, B. (eds.) ISC 2023. LNCS, vol. 14411, pp. 3–23. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-49187-0_1

24. Taub, J., Elliot, M., Pampaka, M., Smith, D.: Differential correct attribution probability for synthetic data: an exploration. In: Domingo-Ferrer, J., Montes, F. (eds.) PSD 2018. LNCS, vol. 11126, pp. 122–137. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99771-1_9
25. Xu, L., Skoularidou, M., Cuesta-Infante, A., Veeramachaneni, K.: Modeling tabular data using conditional GAN. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alche Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 32, pp. 7335–7345 (2019)