# Exploring the Trade-Off between Privacy and Predictive Power in Synthetic Data Generation

Pratik Kamat
Krannert School of Management
Purdue University
West Lafayette, USA
kamat1@purdue.edu

Anusha Reddy
Krannert School of Management
Purdue University
West Lafayette, USA
reddy118@purdue.edu

Naveen Shaji
Krannert School of Management
Purdue University
West Lafayette, USA
shaji@purdue.edu

Prashanth Suresh
Krannert School of Management
Purdue University
West Lafayette, USA
suresh80@purdue.edu

Amisha Turkel
Krannert School of Management
Purdue University
West Lafayette, USA
turkel@purdue.edu

Matthew A. Lanham
Krannert School of Management
Purdue University
West Lafayette, USA
lanhamm@purdue.edu

***Abstract— Abstract: Data scientists often face privacy concerns when using real data. Synthetic data generation offers a solution by creating data that follows the patterns and business constraints of real data while providing differential privacy and satisfactory predictive performance. However, increased differential privacy comes at the cost of model performance. In this study, we explore different synthetic data generators to identify the balance between differential privacy and predictive power. Our target audience is data scientists who want to use synthetic data to achieve business impact while maintaining data security and privacy of sensitive data. By using synthetic data, data scientists can avoid data privacy issues while still obtaining reliable results for their analytics solutions.***

***Keywords—data privacy, synthetic data generation, predictive power vs differential privacy.***

## I. Introduction

As our lives become more online and more comprehensive and high-quality data is generated, machine learning will become a powerful tool for businesses to use to drive strategy and decision-making. However, the collected information can also include personal identifying information, financial and health data, exposing customers to identity theft and fraud. Therefore, it is imperative to protect sensitive information, such as personal and financial information, from unauthorized access, use, and disclosure. Data privacy is essential for firms looking to grow customer trust in their digital platforms. Moreover, governments have imposed strict regulations on data privacy, which aim to safeguard individuals' personal information. Under these regulations, firms handling customer data must act ethically and responsibly. In addition, organizations must be responsible for data breaches and individuals must be protected from being able to access, control, or delete their personal information.

Because of already existing regulations such as HIPAA and GDPR, as well as an increasing number of regulations, organizations are racing to establish their privacy operational policies (Rimol, 2022). However, regulations rarely correlate with statistical measures and often specify abstract requirements (Lutkevich, 2020) (Wolford, 2022). These abstract requirements frequently require human analysis and drive up the operational costs of firms (Krishna, 2022). Any automated operations will involve much tighter constraints than required by regulation, with most small and mid-sized firms choosing simply to opt out of transferring the data at the cost of efficiency. In current practice, firms enforce data privacy by using encryption and by conducting privacy impact assessments often using third-party services. Most firms also employ a policy of data minimization by collecting and storing only the minimum necessary data. In addition to these costly processes, companies incur costs in re-training employees to the evolving privacy requirements.

The current operational practices impose severe constraints on firms in optimizing the cost of maintaining data privacy with the predictive power of data. In recent literature, two main methodologies have been proposed to alleviate these problems, namely Federated Learning and Synthetic Data Generation. Federated Learning is a distributed learning strategy that decouples the training of machine learning models from the need to integrate data centrally. In Federated Learning the data stays at the edge locations and the model is sent to the edge devices for training. The model updates from each node after training are sent back to the server and then aggregated. The aggregated update is then used to update the model before iterating through the entire process. Training is stopped only when the model is stable, i.e., when the magnitude of the aggregated updates falls below a certain threshold. However, Federated Learning models are suboptimal for heterogeneous distributions, costly, and vulnerable to third-party attacks over the internet (Li et al., 2020) (Goetz & Tewari, 2020) (Li, 2019).

Synthetic data generation aims to restructure already existing data into a form that maintains privacy without losing predictive power. In Synthetic data generation, the complexities of the real-world data are generalized using a model. In general, a learning algorithm is used to estimate the parameter of the model using real data. The model is then revered using the Bayes theorem to generate artificial data. Here the data need not be transferred across the network multiple times nor does the model require expensive encryptions. Therefore, for learning across organizations or different business units across the same organization (fewer nodes) synthetic data can be used as a more secure and cheaper alternative to Federated Learning. However, in synthetic data generation, these advantages come at the cost of predictive power. Hence organizations must analyze the tradeoff between predictive power and regulatory risk to optimize their operational efficiency.

In this work, we use multiple benchmark datasets and actual industry datasets to generate synthetic data (Choi, 2018). The industry dataset contains nine features of customer information and an imbalanced binary classification target. (For client confidentiality purposes we shall not reveal a detailed description of the data). We then design and implement various synthetic data generation methods. The generated synthetic data is analyzed for privacy compliance and compared with the real data using a differential privacy ratio (Dwork, 2008). We also train the models using real and synthetic data and compare accuracies to measure the loss in predictive power. The results are then interpolated to establish a tradeoff between privacy and predictive power.

## II. Literature Review

Synthetic data generation is the process of simulating new data sets that are used either in conjunction with real data or as a substitute for it. Synthetic data is generated using mathematical and statistical models, rather than being collected from real-world sources and is intended to resemble real-world data in terms of its distribution, statistical properties, and other characteristics. In the business world, synthetic data has several applications, including data privacy and confidentiality, data augmentation, model validation, and risk assessment. The growing demand for data in the business world, as well as the pressing concern to protect sensitive information, drive the need for synthetic data. As businesses collect and process more and more data, they face growing challenges in protecting sensitive information, sharing data with stakeholders, and managing the risk associated with data breaches and privacy violations.

Despite the numerous benefits of synthetic data, there is a lack of formalized methodologies or processes for generating and evaluating synthetic data. The goal of this review is to gain a better understanding of the processes and methodologies used in existing studies to create synthetic data. We also hope to compare and evaluate several machine learning models and the trade-offs between model accuracy and data privacy to determine the best approach for the specific dataset and business problem. In this research paper, we study the various methodologies for generating synthetic data and assess their efficacy and applicability to our dataset. Our research begins by examining simple data generation methods and gradually moving towards more complex deep-learning-based models.

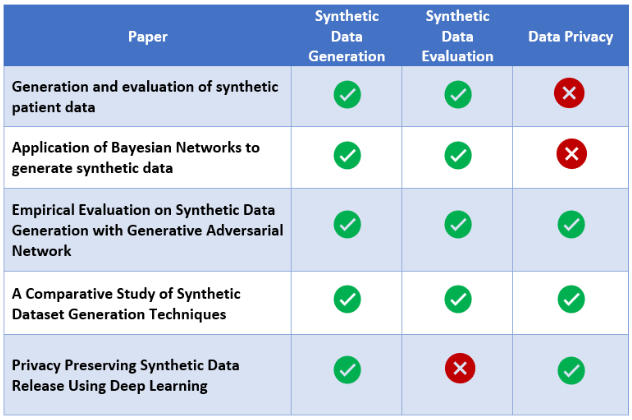
The methods explored include individual marginal sampling, decision trees, linear regression, random forest, Bayesian networks, and deep learning-based General Adversarial Networks (GANs). Goncalves et al., 2020 evaluate various techniques for generating synthetic data and measure the quality of the data generated based on different metrics. The first method generates data through sampling from individual marginals. In this method, only individual distributions are captured, and synthetic data is generated without considering the multi-variate relationships present. While computationally efficient, this method fails to account for the statistical dependencies between variables, making it a less suitable option for our dataset. Goncalves et al., 2020 also discuss another method, Bayesian Networks, which construct direct acyclic graphs based on variable dependencies. The Bayesian network learns the structure of the dataset and then uses this structure to estimate conditional probabilities. D et al., 2021 investigate this idea further by employing a Bayesian network to generate synthetic health data. It elaborates on building multiple healthcare datasets with a mix of numerical and categorical variables and then compares the results to the existing medBGAN method. To accomplish this, a greedy-Bayesian network based on a score-based algorithm is employed. The Bayesian network outperforms the medBGAN model and produces datasets that are like real-world data. Bayesian networks seek to capture the causal relationships between variables, making them a viable option for modeling and predicting on our dataset. It also introduces noise into the conditional probabilities at different nodes, ensuring data privacy.

Though the Bayesian Networks technique appears promising, it is critical to be mindful when leveraging Bayesian networks that the network may not always capture more complex relationships among variables and may converge at the local optima. Deep learning models provide excellent solutions to address the limitations of Bayesian networks. One such model- General Adversarial Network (GAN) is discussed in detail (Lu et al., 2019). GAN engages a deep neural network classification model to learn the hidden structures in data. The neural network includes two types of networks: one for the generator and one for the discriminator. The former continues to synthesize data from the learned data distribution, whereas the latter distinguishes between data from the original dataset and synthetic data from the generator. The network achieves differential privacy through Laplace and PrivBayes mechanisms. However, Neural Networks are historically known to be typically effective for large training data and features. For a dataset with a limited number of rows, the model tends to introduce heavy bias and overfit to the training data. Furthermore, if the dataset is small, the network may not be exposed to enough variation in the data and thus fail to learn robust representations.

Differential privacy is one of the most important applications in the context of synthetic data as it helps ensure that synthetic data cannot be used to re-identify individual records in the original dataset. Differential privacy ensures that the synthetic data can be used for research, analysis, or other purposes without jeopardizing the privacy of the individuals whose data is being used by adding random noise to the data. The implementation of differential privacy in synthetic data generation is a complex process and requires careful consideration of the trade-off between privacy protection and the accuracy of the synthetic data. To ensure that the synthetic data retains the properties of the original data while still being privacy-preserving, it is necessary to use techniques that are specifically designed for differential privacy, such as DP-SYN as described by Abay et al., 2018. The authors present a deep learning-based auto-encoder technique called DP-SYN that is designed to preserve privacy. Several experiments are conducted on benchmark datasets to compare the performance of DP-SYN with other existing techniques and conclude that DP-SYN outperforms them. The results of various evaluation experiments (Dandekar et al., 2018) on decision trees, linear regression, and random forest models for synthetic data generation reveal that the decision tree is the most effective synthesizer. Since decision trees are known as unstable predictors therefore, we need more research to confirm their effectiveness for our dataset.

To determine the optimal solution for our dataset, we must compare and assess the various methods using evaluation metrics. Goncalves et al., 2020 introduce various means for making comparisons between real and synthetic data. The first method is to compare individual variables graphically in order to understand univariate differences. Following this, the correlation table, two variable Kolmogorov, and clustering algorithms are used to ensure that multivariate relationships in the synthetic dataset are maintained. These tests can be used for each of the methods discussed above, and the degree to which each one resembles the real dataset can be determined. The best-suited model can then be used to generate synthetic data from our data.

TABLE 1. SUMMARY OF LITERATURE REVIEW AND STUDY COMPARISON



| Paper | Synthetic Data Generation | Synthetic Data Evaluation | Data Privacy |
|---|---|---|---|
| Generation and evaluation of synthetic patient data | ✓ | ✓ | ✗ |
| Application of Bayesian Networks to generate synthetic data | ✓ | ✓ | ✗ |
| Empirical Evaluation on Synthetic Data Generation with Generative Adversarial Network | ✓ | ✓ | ✓ |
| A Comparative Study of Synthetic Dataset Generation Techniques | ✓ | ✓ | ✓ |
| Privacy Preserving Synthetic Data Release Using Deep Learning | ✓ | ✗ | ✓ |

## III. Data

In this study, we use the sales dataset provided by the client which is a timeshare-based vacation and travel planning company. The sales data consists of transactions/purchases done by their customers, with the data collected at regular intervals. The data provided is a snapshot of the transactions at a point in time. Each row of the dataset represents a single sale, and the columns represent features of members like the tenure of membership with the company, history of purchases made, and additionally, features of the sale like the type of product purchased, location of the resort, etc. The primary key of this table is a randomly generated ID. There are about 50,000 rows of data available, and the variables present are a combination of categorical and continuous values.

The second table provided contains the Id and a flag whether the customer made the sale or not. This table is related to the sales data through the ID column which plays the role of the foreign key.

The two datasets combined will enable us to capture the univariate distributions of individual columns and understand multivariate relationships between columns for customers who made a purchase and for those who did not. Table 1 provides a brief description of the different tables used in this study.

TABLE 2. DATA DESCRIPTION

| Table No. | Table Name | Description |
|---|---|---|
| 1 | Sales Data | ~50,000 rows of Purchase history for members |
| 2 | Outcome Data | Details of whether the customer made the purchase or not |

## IV. Methodology

In this study, the aim is to establish the trade-off between the loss of predictive power and the level of privacy maintained in synthetic data generation. For conducting the experiments, we use the class-imbalanced industry dataset mentioned in the data section. The real data is sampled using a random stratified sampling to generate test and train splits with 80% and 20% data. Then train data is fit into a logistic regression model to learn the binary classification problem. And the test data used to evaluate model accuracy, area under the ROC curve and to generate a lift chart. These curves and metrics will serve as the benchmarks to measure the loss in predictive power when using synthetically generated data.

Before generating synthetic data, it's imperative to mark the difference in using just the train data for generating synthetic data against using the entire dataset. Using the entire dataset also allows synthetic data generators to map the complexities in test data. This, in-turn exposes the comparisons between synthetic data generators to concerns of overfitting. Hence, we feed only train data to each individual synthetic data generators studied. The data sets so produced are compared with the real data to compute the differential privacy ratio and understand the level of privacy that was maintained. We also add noise to the generated datasets using random samples from a Laplace distribution of different scales to enhance privacy. The different scales of noise added allows us observe the trade-off between privacy and predictive power on a granular scale.

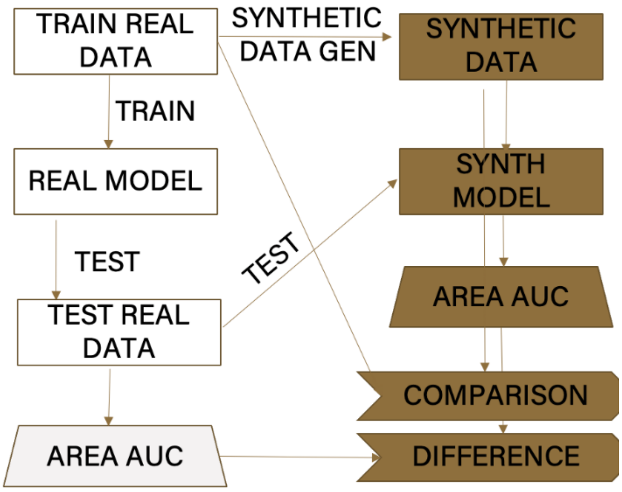


FIGURE 2. EXPERIMENT SETUP

From the synthetic datasets generated, similar to real data, logistic regression models are fit to learn the binary classification problem. Then these models are evaluated using the real test data to obtain model accuracy, area under the ROC curve and to generate a lift chart. The synthetic data model's area under ROC curve is then subtracted from the real model's area under ROC curve to measure the drop in predictive power. Finally, the results for drop in predictive power are then interpolated with the differential privacy ratio values against the scale of noise axis to generate the drop in predictive power versus differential privacy graphs for each individual synthetic data generator.

## V. MODELS

Our selection for the models to generate synthetic data was driven by its anticipated performance on our dataset as well to cover diverse methodologies. Thus, in this work we use Data Synthesizer, Gaussian Copula and CTGAN models to generate synthetic data and evaluate its effectiveness.

**DataSynthesizer:** We chose Data Synthesizer as it heavily relies on Bayesian methods which are known to perform well for noisy and heterogenous data. Data Synthesizer constructs a greedy Bayesian network based on information gain. The network construction is also fixed to the same number of children for each parent ( defined by the user). The conditional probabilities of the connected nodes are the estimated using the real data. To generate synthetic data, the joint distribution defined by the network and the learned parameters is sampled. Engineered features in data sets introduces specific constraints that the data need to follow. However, since the network created is not a fully connected network, it is impossible to define and maintain these constrains while generating synthetic data using DataSynthesizer.

**Gaussian Copula:** We chose Gaussian Copula as it a heavier model that maps on to every pair of joint distribution between feature as opposed to the just the one defined by the network in the case of Data Synthesizer. The model also allows you to predefine gaussian distribution for individual variables. Gaussian Copula allows for data constraints to be maintained while generating synthetic data. However, the cost of development of the model scales along the feature axis as there is a considerable manual effort in analyzing and choosing the right pre- determined distribution for each feature.

**CTGAN:** The above two models studied heavily relies on classical statics and machine learning approaches. CTGAN however is a deep learning based adversarial network. We chose CTGAN to cover diverse methods of synthetic data generation and study its impact. The general idea behind adversarial networks is that it comprises of two different neural network modules which have adversarial objectives. The first module is a generator model, which is trained with the objective of generating artificial data that closely resembles the original data. The second module is discriminator model which is trained with the objective of differentiating between real and synthetic data. In our particular use case, we stop the training when the discriminator model is no longer able to discriminate between the real and synthetics data. CT Gan allows for data constraints to be maintained while generating synthetic data. However, as it is a deep learning model, its computationally costly and requires a GPU-based machine to train on. The training times are approximately 30 minutes for 200 epochs of training a 50000-row dataset divided into batches of 32 (Tesla T4 Nvidia GPU).

## VI. RESULTS

To analyse the data generated by each of the models, we evaluate the data through 3 dimensions namely Information captured, Model Performance and Privacy preservation while also checking if constraints the data are maintained. The following methods are employed to compare the similarity of synthetic data to real data in terms of the information captured: K-Means clustering to understand the overall structure of data, Univariate distributions to understand individual distributions, Correlation matrix to check if relationship between variables are preserved. To understand the model performance of model developed using synthetic data, we use ROC score as the data is highly imbalanced and additionally gain chart as well. The privacy of the dataset generated is evaluated using a Euclidean Distance based Nearest neighbor algorithm which provides the % of records which are different from that of the real data. These methods are implemented to all the 3 models that were developed to generate synthetic data.

The following are the results obtained for **Data Synthesizer**:

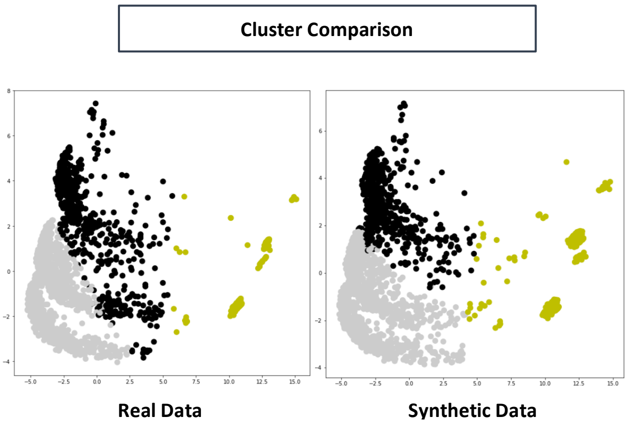


FIGURE 3: CLUSTER COMPARISON

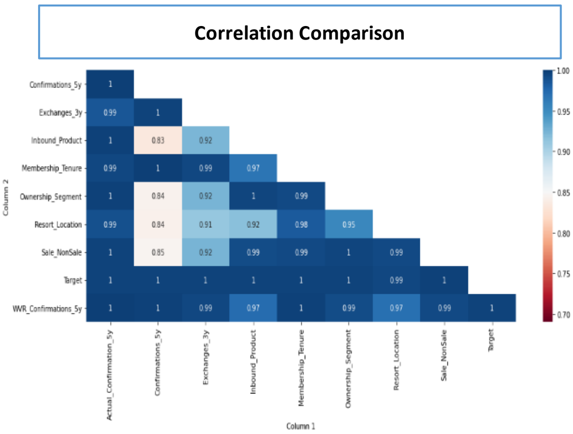


FIGURE 4: CORRELATION COMPARISON

97% of similarity is observed both in the univariate distributions and correlations. Clusters formed are very similar to that of real data.

We obtained a ROC score of 0.729 for the real data while 0.725 for synthetic data and obtained 81% privacy score for the synthetic data. The gain charts were also similar for both real and synthetic data.

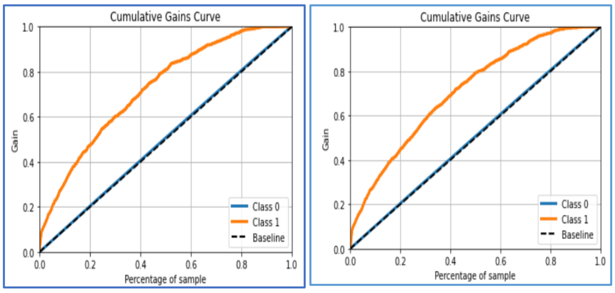


FIGURE 5: GAIN CURVE COMPARISON

These results suggest that Data Synthesizer is able to capture the information well and provide a good model performance while preserving privacy reasonably. However, the constraints in the real data are not being followed in the synthetic data.

The following are the results obtained for **Gaussian Copula**:

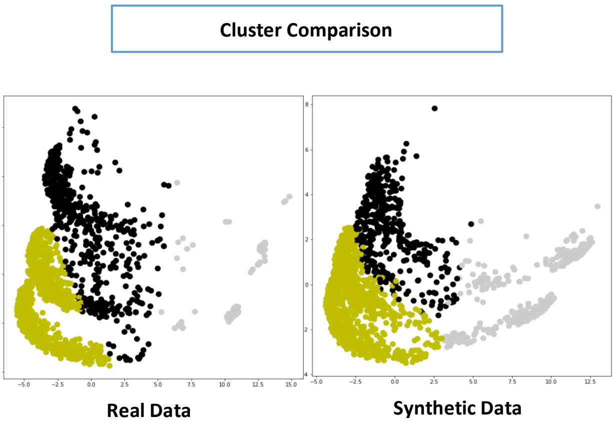


FIGURE 6: CLUSTER COMPARISON

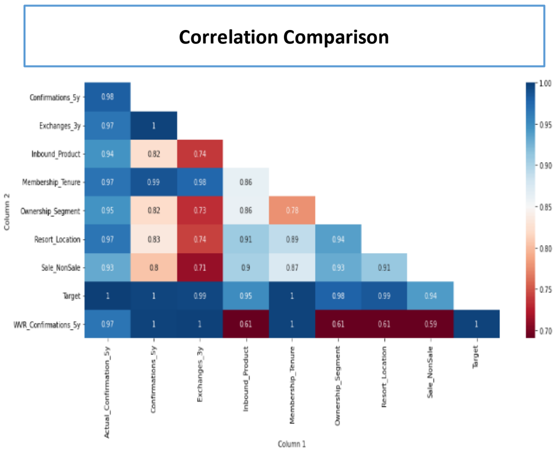


FIGURE 7: CORRELATION COMPARISON

90% of the univariate distributions are similar between real and synthetic data while 87% of the correlations are maintained in the synthetic data. Clusters formed are very similar to that of real data.

We obtained a ROC score of 0.729 for the real data while 0.716 for synthetic data and obtained 94%

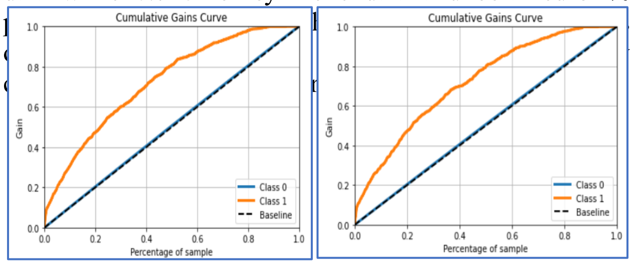


FIGURE 8: GAIN CURVE COMPARISON

These results suggest that compared to Data Synthesizer, this model is able to preserve more privacy at the cost of small drop in model performance and the information captured. This model, however, gives the user the flexibility to define constraints between variables and generate data based on these constraints.

The following are the results obtained for **CTGAN**:

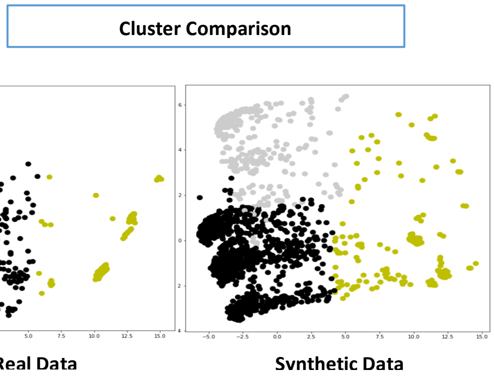


FIGURE 9: CLUSTER COMPARISON

The clusters formed are highly dissimilar while we obtained a similarity of 92% for univariate distributions and 88% for correlations.

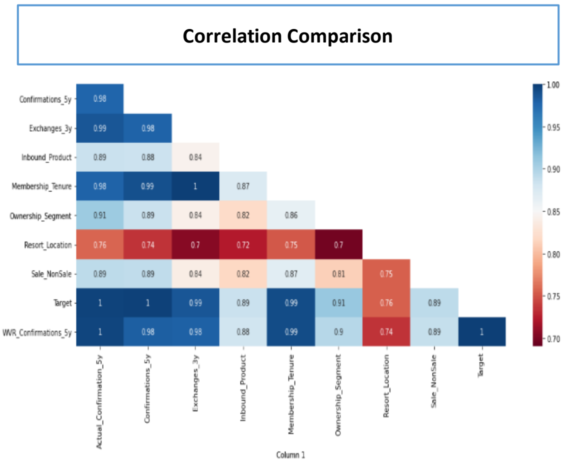


FIGURE 10: CORRELATION SIMILARITY

We obtained a ROC score of 0.729 for the real data while 0.65 for synthetic data and obtained 78% privacy score for the synthetic data. The gain charts obtained suggests that there is a 10% drop in opportunity coverage when compared to real data.

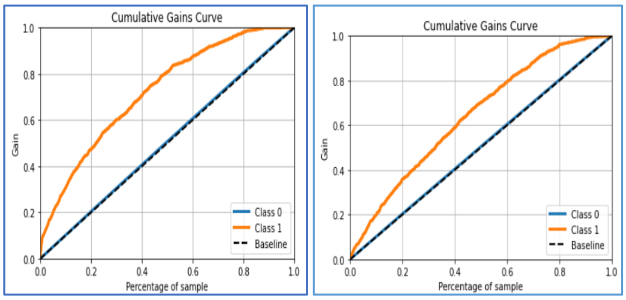


FIGURE 11: CLUSTER COMPARISON

Overall, these results suggest that this is the worst performing model compared to the other two models in model performance and privacy.

One noticeable trend that was observed was that with increasing privacy, there was a drop in the model performance. To study this further, privacy in the form of laplace noise was added to each of the 3 models to study its impact on the model performance.

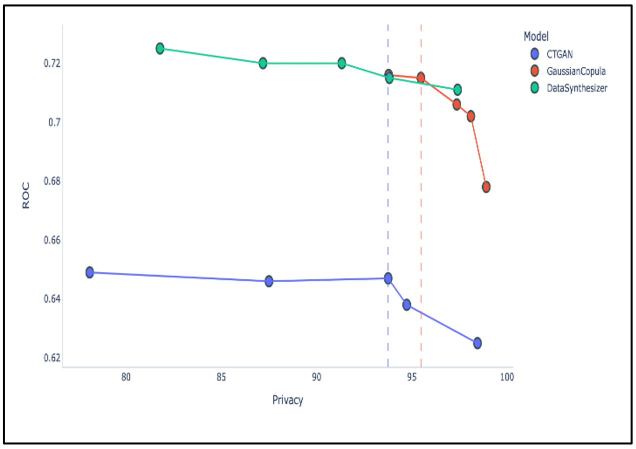


FIGURE 12: GAIN CURVE COMPARISON

The baseline models indicate the models that were developed without the addition of noise. From the graph, it is evident that as the privacy increases, model performance drops slightly and after 95% privacy, we can see that the drop is steep. Businesses can utilize this information to understand which model would suit their use case based on their objectives.

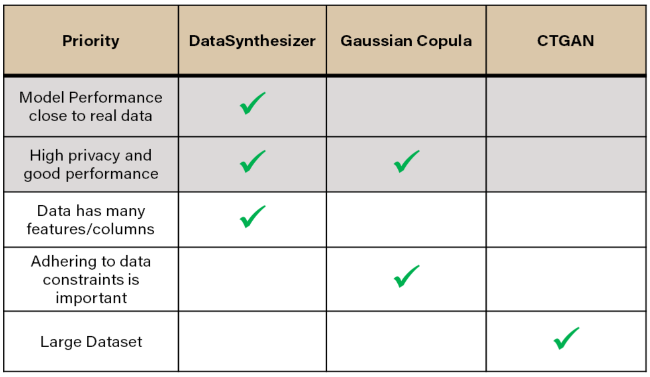
## VII. CONCLUSION

Our analysis of the three synthetic data generation models - DataSynthesizer, Gaussian Copula, and CTGAN - has revealed that the optimal model selection depends on the specific use case. For example, if the priority is to have model performance close to real data, DataSynthesizer may be the best option. However, if high privacy and good performance are important, both DataSynthesizer and Gaussian Copula could be considered. If the data has many features/columns, then DataSynthesizer may be preferred. On the other hand, if adhering to data constraints is crucial, then Gaussian Copula might be a better choice. For larger datasets, CTGAN should be a better option since neural network based model works better on large datasets.

| Model | Information Captured | Model ROC | Data Privacy |
|---|---|---|---|
| DataSynthesizer | Good | Good | Average |
| Gaussian Copula | Average | Good | Good |
| CTGAN | Bad | Bad | Average |

FIGURE 13: SYNTHETIC DATA GENERATOR SUMMARY

Additionally, our analysis showed that to achieve the best prediction power, using DataSynthesizer without adding noise is recommended. However, when noise is added to increase privacy, both DataSynthesizer and Gaussian Copula showed similar predictive power. Furthermore, our results indicated that there is a significant drop in performance when attempting to increase privacy beyond a certain point for all three models.



| Priority | DataSynthesizer | Gaussian Copula | CTGAN |
|---|---|---|---|
| Model Performance close to real data | ✓ | | |
| High privacy and good performance | ✓ | ✓ | |
| Data has many features/columns | ✓ | | |
| Adhering to data constraints is important | | ✓ | |
| Large Dataset | | | ✓ |

FIGURE 14: SYNTHETIC DATA GENERATOR SUMMARY

In conclusion, when selecting a synthetic data generation model, it is important to consider the specific use case and prioritize the relevant factors such as model performance, data privacy, and adherence to data constraints.

## References

[1] Gartner Identifies Top 5 Trends in Privacy Through 2024 . Retrieved from https://www.gartner.com/en/newsroom/press-releases/2022-05-31-gartner-identifies-top-five-trends-in-privacy-through-2024

[2] HIPAA (n.d.). Retrieved from https://www.techtarget.com/searchhealthit/definition/HIPAA

[3] What is GDPR, the EU's new data protection law? Retrieved from https://gdpr.eu/what-is-gdpr/

[4] Data Privacy is expensive Retrieved from https://venturebeat.com/security/data-privacy-is-expensive-how-to-manage-costs/

[5] T. Li, A. K. Sahu, A. Talwalkar and V. Smith, "Federated Learning: Challenges, Methods, and Future Directions," in *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50-60, May 2020, doi: 10.1109/MSP.2020.2975749.

[6] Goetz, J., & Tewari, A. (2020). Federated Learning via Synthetic Data. *ArXiv*. https://doi.org/10.48550/arXiv.2008.04489

[7] Federated Learning : Challenges, Methods and Future Directions. Retrieved from https://blog.ml.cmu.edu/2019/11/12/federated-learning-challenges-methods-and-future-directions/

[8] Miri Choi 2017 Medical Cost Personal Datasets Retrieved from https://www.kaggle.com/datasets/mirichoi0218/insurance

[9] Dwork, C. (2008). Differential Privacy: A Survey of Results. In: Agrawal, M., Du, D., Duan, Z., Li, A. (eds) Theory and Applications of Models of Computation. TAMC 2008. Lecture Notes in Computer Science, vol 4978. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-79228-4_1

[10] Goncalves, A., Ray, P., Soper, B. et al. Generation and evaluation of synthetic patient data. BMC Med Res Methodol 20, 108 (2020). https://doi.org/10.1186/s12874-020-00977-1

[11] Kaur D, Sobiesk M, Patil S, Liu J, Bhagat P, Gupta A, Markuzon N. Application of Bayesian networks to generate synthetic health data. J Am Med Inform Assoc. 2021 Mar 18;28(4):801-811. doi: 10.1093/jamia/ocaa303. PMID: 33367620; PMCID: PMC7973486.

[12] Pei-Hsuan Lu, Pang-Chieh Wang, and Chia-Mu Yu. 2019. Empirical Evaluation on Synthetic Data Generation with Generative Adversarial Network. In Proceedings of the 9th International Conference on Web Intelligence, Mining and Semantics (WIMS2019). Association for Computing Machinery, New York, NY, USA, Article 16, 1–6. https://doi.org/10.1145/3326467.3326474

[13] Abay, N.C., Zhou, Y., Kantarcioglu, M., Thuraisingham, B., Sweeney, L. (2019). Privacy Preserving Synthetic Data Release Using Deep Learning. In: Berlingerio, M., Bonchi, F., Gärtner, T., Hurley, N., Ifrim, G. (eds) Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2018. Lecture Notes in Computer Science(), vol 11051. Springer, Cham. https://doi.org/10.1007/978-3-030-10925-7_31

[14] Dandekar, A., Zen, R.A.M., Bressan, S. (2018). A Comparative Study of Synthetic Dataset Generation Techniques. In: Hartmann, S., Ma, H., Hameurlain, A., Pernul, G., Wagner, R. (eds) Database and Expert Systems Applications. DEXA 2018. Lecture Notes in Computer Science(), vol 11030. Springer, Cham. https://doi.org/10.1007/978-3-319-98812-2_35