Synthetic Data Generation

ARISTOTLE UNIVERSITY OF THESSALONIKI

Koukolis Evangelos, Grosomanidis Odysseas, Laliotis Georgios

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΑΠΘ Π.Μ.Σ. ΣΤΗΝ Τεχνητή Νοημοσύνη

Introduction

Synthetic data plays a crucial role in the fields of artificial intelligence and machine learning, addressing several challenges associated with real-world data usage. The reasons that can lead to creating synthetic data could be many, for instance the protection of personal data, the non-existence of data due to ethical or legal restraints, because the data collection and labelling is too costly, for the enrichment of the process of learning and testing with more data, and in general to create more data when there is a need.

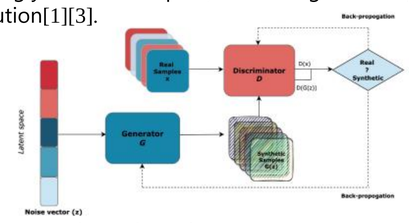
This specific work aims to research and evaluate techniques related to the creation of synthetic data. We try to work with some real case scenarios. We had used 3 datasets, to showcase 3 use cases of synthetic data creation:

- Data synthetization to investigate if a modal trained and tested with fully synthetic data can outperform a model trained and tested on the original data (health insurance lead prediction)
- Data synthesization of the underrepresented class, in a problem with high class imbalance (credit card fraud)
- Data synthesization, creating synthetic time series data, for future value prediction (Electricity Consumption and Production)

Methods

GANs - Main Idea

Generative Adversarial Networks (GANs) consist of two neural networks— a Generator and a Discriminator— that compete in a game-like scenario. The Generator creates data samples aiming to mimic real data, while the Discriminator evaluates them, distinguishing between the generated samples and actual data. Through this adversarial process, the Generator learns to produce increasingly realistic outputs, enhancing its ability to replicate the true data distribution[1][3].



Both networks improve through a min-max game represented by the cost function [1]:

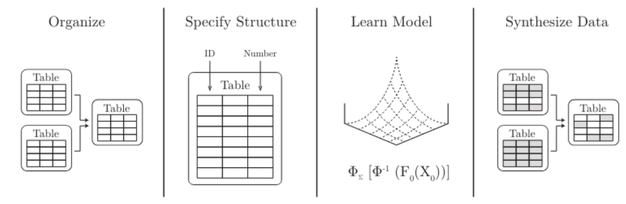
$$min_G max_D = E_{x\sim Pr}[\log D(x)] + E_{z\sim Pz}\left[\log\left(1 - D(G(z))\right)\right]$$

where the Discriminator maximizes the probability of assigning the correct label to both real and generated samples, and the Generator minimizes the probability that its outputs are classified as fake."

Synthesizers - Libraries

Synthesizers: The libraries used SDV and YData synthetic, offer a plethora of synthesizers to choose from:

- YData Synthetic: Offers variations of the GAN methodology, and is appropriate for both tabular and time series data [2][3]
- SDV: Offers various methods based on Gaussian Copula, GAN, or Variational Autoencoders, and is appropriate for tabular and sequential data.[4][5][6]



The user first load the original data, specifies the structure, and metadata (feature names and data types), and then runs the data synthesization library (SDV or Ydata Synthetic) to fit to the existing data, and create new synthetic ones.

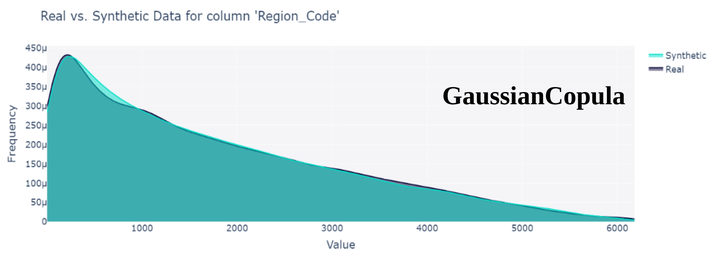
Results

Health Insurance Lead Prediction

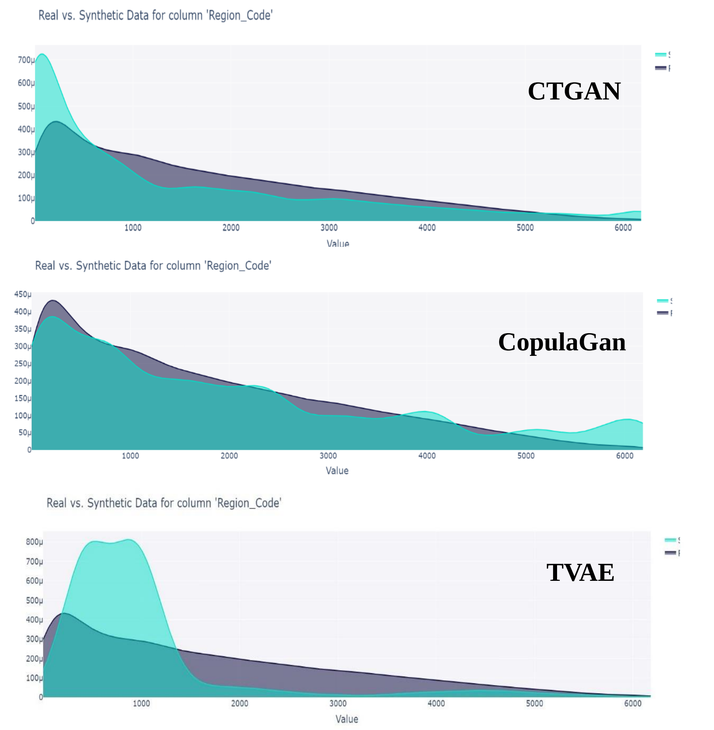
Training + Test data produced by Synthesizer (or Original Data)	Method	Acc	Precision (Macro)	Precision (Weighted)	Recall (Macro)	Recall (Weighted)	F1 (Macro)	F1 (weighted)
Original Data	RF	0.76	0.38	0.58	0.50	0.76	0.43	0.65
GaussianCopula	RF	0.75	0.37	0.56	0.50	0.75	0.43	0.64
CTGAN	RF	0.78	0.39	0.60	0.50	0.78	0.44	0.68
CopulaGan	RF	0.82	0.41	0.68	0.50	0.82	0.45	0.74
TVAE	RF	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Original Data	kNN	0.71	0.50	0.64	0.50	0.71	0.48	0.66
GaussianCopula	kNN	0.70	0.49	0.62	0.50	0.70	0.47	0.65
CTGAN	kNN	0.74	0.53	0.67	0.51	0.74	0.49	0.69
CopulaGan	kNN	0.80	0.50	0.71	0.50	0.80	0.48	0.74
TVAE	kNN	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Table 1: comparison of classification reports for different synthetic train data on the same test data.

Using **GaussianCopula** synthesizer is ideal, when the data follow a Gaussian distribution



Comparison of the same 'Region_Code' feature data distribution for different synthetic data generators:



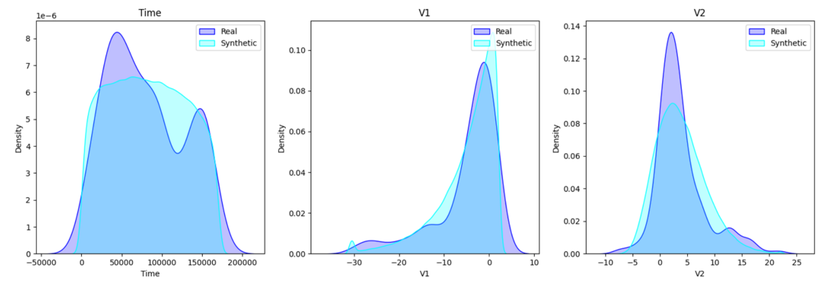
Credit Card Fraud

Training + Test data produced by Synthesizer (or Original Data)	Method	Acc	Precision for class 0 (Weighted)	Precision for class 1 (Weighted)	Recall for class 0 (Weighted)	Recall for class 1 (Weighted)	Precision (Weighted)	Recall (weighted)	F1 (weighted)
Original Data	RF	0.9994	1.00	0.94	1.00	0.76	1.00	1.00	1.00
GaussianCopula	RF	0.9995	1.00	0.95	1.00	0.77	1.00	1.00	1.00
CTGAN	RF	0.9995	1.00	0.90	1.00	0.84	1.00	1.00	1.00
CopulaGan	RF	0.9995	1.00	0.93	1.00	0.77	1.00	1.00	1.00
TVAE	RF	0.9992	1.00	0.74	1.00	0.84	1.00	1.00	1.00
Original Data	kNN	0.9983	1.00	0.86	1.00	0.04	1.00	1.00	1.00
GaussianCopula	kNN	0.9215	1.00	0.01	0.92	0.48	1.00	0.92	0.96
CTGAN	kNN	0.9387	1.00	0.01	0.94	0.34	1.00	0.94	0.97
CopulaGan	kNN	0.9655	1.00	0.02	0.97	0.42	1.00	0.97	0.98
TVAE	kNN	0.9769	1.00	0.04	0.98	0.56	1.00	0.98	0.99
Original Data	LR	0.9991	1.00	0.88	1.00	0.56	1.00	1.00	1.00
GaussianCopula	LR	0.9906	1.00	0.09	0.99	0.52	1.00	0.99	0.99
CTGAN	LR	0.9993	1.00	0.84	1.00	0.78	1.00	1.00	1.00
CopulaGan	LR	0.9992	1.00	0.83	1.00	0.71	1.00	1.00	1.00
TVAE	LR	0.9886	1.00	0.11	0.99	0.83	1.00	0.99	0.99

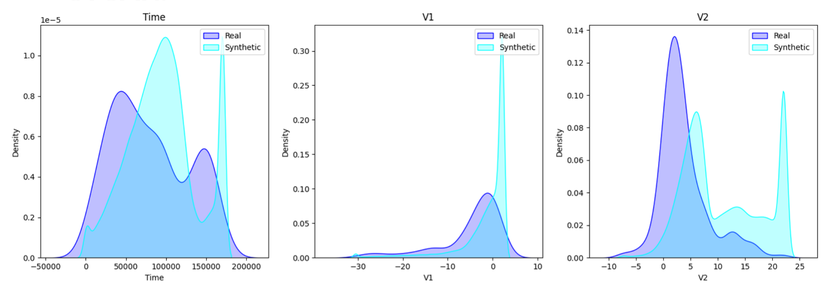
Table 2: comparison of classification reports for different synthetic train data on the same test data.

Distributions of Real and Synthetic Data, for specific features:

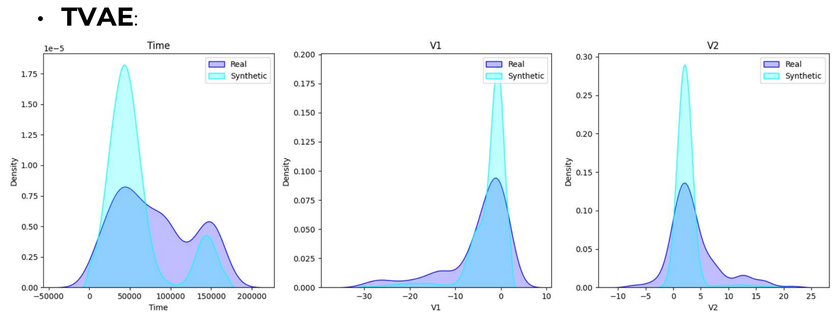
- **GaussianCopula:**



- **CTGAN:**

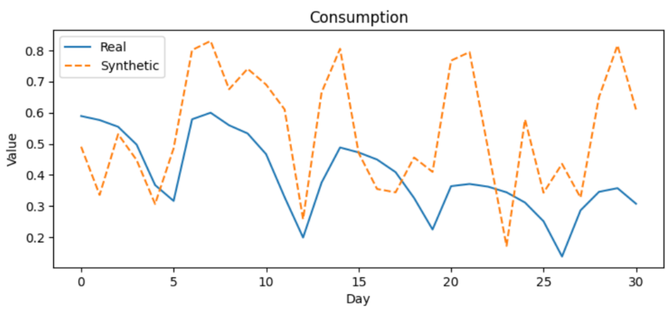


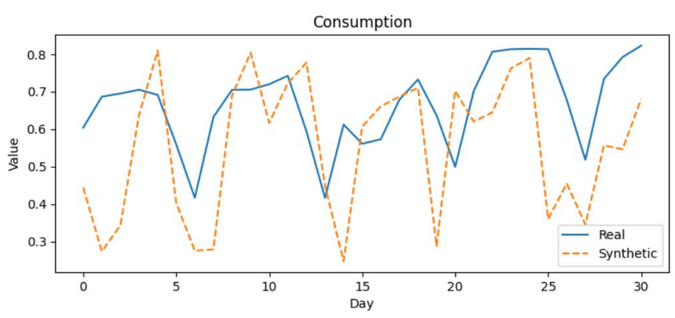
- **TVAE**:

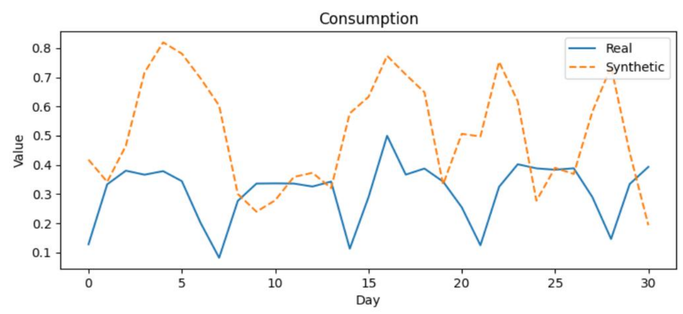


Hourly Electricity Consumption and Production

Comparison of some value windows for real and synthetic data, where the values refer to normalized daily electricity consumption.







Future predictions of electricity consumption based on synthetic historical data

This synthetic data lacks utility.

Evaluation

Fidelity

How similar is this synthetic data as compared to the original training sets

Utility

How useful is this synthetic data for our downstream machine learning applications

Privacy

Has any sensitive data been inadvertently synthesized by our model

- **Fidelity:** We evaluate how similar the synthetic and the original data are, by comparing the distributions (tabular data), and by comparing the time series trends (time-series data).
- **Utility:** We gauge the usefulness of our synthetic data, by training a model using synthetic data, and comparing the performance metrics to a model trained with the original data.
- **Privacy:** We have not enlisted the help of metrics to assess if sensitive data has been encoded in the synthesized data. The privacy concern is not applicable to the credit card fraud dataset, as this was the reason behind having PCA-produced features.

References

[1] Goodfellow, Ian J., Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville and Yoshua Bengio. "Generative Adversarial Nets." Neural Information Processing Systems (2014).

[2] Yoon, Jinsung, Daniel Jarrett and Mihaela van der Schaar. "Time-series Generative Adversarial Networks." Neural Information Processing Systems (2019).

[3] Ghosheh, Ghadeer & Jin, Li & Zhu, Tingting. (2023). A Survey of Generative Adversarial Networks for Synthesizing Structured Electronic Health Records. ACM Computing Surveys.

[4] Mirza, Mehdi and Simon Osindero. "Conditional Generative Adversarial Nets." ArXiv abs/1411.1784 (2014): n. pag.

[5] Xu, Lei, Maria Skoularidou, Alfredo Cuesta-Infante and Kalyan Veeramachaneni. "Modeling Tabular data using Conditional GAN." Neural Information Processing Systems (2019).

[6] Kingma, Diederik P. and Max Welling. "Auto-Encoding Variational Bayes." CoRR abs/1312.6114 (2013): n. pag