PRISIM: Privacy Preserving Synthetic Data Simulator *** Supplementary Materials ***

Dr. Subhrajit Samanta [*1], **Shantanu Chandra**[2], **Dr. Prakash PKS**[3], **Srinivas Chilukuri**[4] and **Srinivas Alva**[5]
ZS Associates

Abstract

The Supplementary Material for the paper titled "PRISIM: Privacy Preserving Synthetic Data Simulator " accepted in NeurIPS-23 Workshop (on Synthetic Data for Empowering ML Research) contains,

- Generative Model Details
- Dataset Details
- Runtime Details
- Additional Results
- Implementation Details
- Privacy vs Utility Regression Analysis

1 Generative Model Details

In this section first we give a brief overview of the generative models followed by their detailed architecture and hyperparameter set-up (whenever applicable).

1.1 CTGAN

Conditional Table GAN or CTGAN [Xu *et al.*, 2019] is a Generative Adversarial Network (GAN) based approach for generating multivariate tabular data. CTGAN conditions the generation of the continuous columns based on the discrete to retain the bi-variate correlations of the original dataset. CTGAN when trained well provides high fidelity synthetic samples, however training can be tricky and slow.

For the experiments in the main paper, we have utilized the following architecture and hyperparameter set-up for CTGAN,

- **Embedding dimensions**: Size of the random sample passed to the Generator. Defaults to 128.
- **Generator dimensions**: Size of the output samples for each one of the Residuals. A Residual Layer will be created for each one of the values provided. Defaults to (256, 256).
- **discriminator dimensions**: Size of the output samples for each one of the Discriminator Layers. A Linear Layer will be created for each one of the values provided. Defaults to (256, 256).
- **Generator learning rate**: Learning rate for the generator. Defaults to 2e-4.
- **Generator decay** :Generator weight decay for the Adam Optimizer. Defaults to 1e-6.
- **Discriminator learning rate**: Learning rate for the discriminator. Defaults to 2e-4.
- **Discriminator decay**: Discriminator weight decay for the Adam Optimizer. Defaults to 1e-6.
- **Batch size**: Number of data samples to process in each step. Defaults to 128.
- **Discriminator steps**:Number of discriminator updates to do for each generator update. Default is 5.
- **Log frequency**: Whether to use log frequency of categorical levels in conditional sampling. Defaults to 'True'.
- **Epochs**: Number of training epochs. Defaults to 300.
- **pac** : Number of samples to group together when applying the discriminator. Defaults to 10.

1.2 TVAE

Tabular VAE or TVAE [Xu *et al.*, 2019] is a variational autoencoder based approach for generating mixed type, multivariate tabular data. TVAE also provides good quality. While faster than CTGAN it still is slower than the statistical Copula based models.

For the experiments in the main paper, we have utilized the following architecture and hyperparameter set-up for TVAE,

- **Embedding dimensions**: Size of the random sample passed to the encoder. Defaults to 128.
- **Compress dimensions**: Size of the encoded output samples for each one of the Residuals. A Residual Layer will be created for each one of the values provided. Defaults to (128,128).
- **Decompress dimensions**: Size of the output samples for each one of the decoding Layers. A Linear Layer will be created for each one of the values provided. Defaults to (128, 128).
- **Batch size**: Number of data samples to process in each step. Defaults to 128.
- **Epochs**: Number of training epochs. Defaults to 300.

[*]Dr. Samanta is the corresponding author for this publication.

- **Learning rate**: Learning rate defaults 2e-4.

1.3 GC

Gaussian Copula or GC is a statistical method of generating multivariate tabular data where the copula takes care of the correlation between the features. This method is fast, quite scalable, provides decent fidelity, and also does not require any hyper-parameter tuning. This should be the preferred approach for a quick analysis or if fidelity of the synthetic samples is not a concern.

2 Dataset Details

In this section we provide more details on the datasets utilized in this article.

- **HR** This open source data[1] contains records for the employees of an imaginary organization. Often used for attrition prediction we picked it from Kaggle for its popularity. The machine learning task we used however in this study was income prediction.
- **MIMIC-III** This Electronics Health Care data is quite popular in the medical AI community and contains Electronics Health Records for patients admitted to ICU. For more details please refer to [Johnson *et al.*, 2016]. We utilized the demography and medical history information (as they are tabular) in this article. BMI is used as the target variable for a regression task to compute MLU.
- **AIRLINE** This tabular data[2] is picked from Kaggle again for its popularity. It contains a survey related to the satisfaction of airline passengers. This is also a mixed type data and we predict the satisfaction level (classification) as the ML task.
- **HEART** This tabular data is also from Kaggle[3] and it contains information about patients with heart diseases. For MLU we predict the death event.
- **ADULT** This dataset[4] is picked from the UCI machine learing repository and also quite popular among researchers. Contains census data. We used this data because we wanted to compare proposed DbP model with state-of-the-art DP-GAN models on their metric values (as reported in the paper [Tantipongpipat *et al.*, 2021]). We compared aggregated JSD value (categorical only) and MLU (on salary prediction) as utility metrics for this dataset against SOTA models as in [Tantipongpipat *et al.*, 2021].

In most cases we utilized the recommended feature selection from Kaggle and did not use all the columns. Please find the summary of the same in table 1.

[1]https://tinyurl.com/bdkpjyvv
[2]https://tinyurl.com/8rc5xw2r
[3]https://tinyurl.com/2p9furuf
[4]https://tinyurl.com/ytvsee64

3 Run-time Analysis

. In this section we provide a detailed run-time analysis. We sort the datasets by their sample size first and track the run-time for all the three generators with prescribed hyperparamter as provided in table 2.

The results are provided in the following table,

		Runtime in Minutes		
Dataset	Size	CTGAN	TVAE	GC
HEART	300	6	1	0.2
HR	1500	14	3	0.5
AIRLINE	26000	76	29	2
ADULT	50000	145	62	5

Table 2: Runtime analysis

We observe that CTGAN takes significantly longer time to train than TVAE. Whereas GC is extremely fast therefore always advisable for a quick analysis. The generation quality and fidelity of course takes a hit with GC and same can be observed from the qualitative analysis. So in terms of quality in general we observed that CTGAN $>$ TVAE $>$ GC whereas in terms of speed GC $>$ TVAE $>$ CTGAN. This is further summarized in the figure 1.

4 Additional Results

we have performed multiple qualitative studies to further demonstrate the utility of DbP private data (the quantitative privacy attack results are already demonstrated in the main paper). We adopt mainly three types of analysis as listed below .

- **Univariate Histogram/KDE Comparison** we pick the features and compare their univariate distribution (KDE for continuous, histogram for categorical) between the real and private data (DbP) for qualitative analysis on the utility. However, in this document we have shared the figure for one randomly picked feature only.
- **Bi-variate Correlation-Heatmap** The lower triangular matrix of the pairwise cross-correlation matrix is presented in the form of a heatmap where higher correlation corresponds to a brighter color. Two heatmaps are presented here, the left one is with the original data and right one with DbP private data (integer encoding is done for the purpose of computing correlations).
- **tSNE Visualization** With tSNE plot we project the multi-dimensional data into a 2-D feature space for visualization. A significant overlap between the real and the private (DbP) data indicates good retention of utility.

These analysis are done for each of the generative model on each of the dataset.

4.1 Results with CTGAN

In the first set of experiments, CTGAN is utilize as the generator. Results are provided for the same.

Dataset	No. of samples	No. of features	Mixed-type?	No. of continuous features	No. of discrete features
HR	∼1.5k	10	Yes	5	5
MIMIC-3	∼2k	14	Yes	8	6
AIRLINE	∼26k	23	Yes	4	19
HEART	∼0.3k	13	Yes	7	6
ADULT	∼50k	15	Yes	5	10

Table 1: Dataset Details

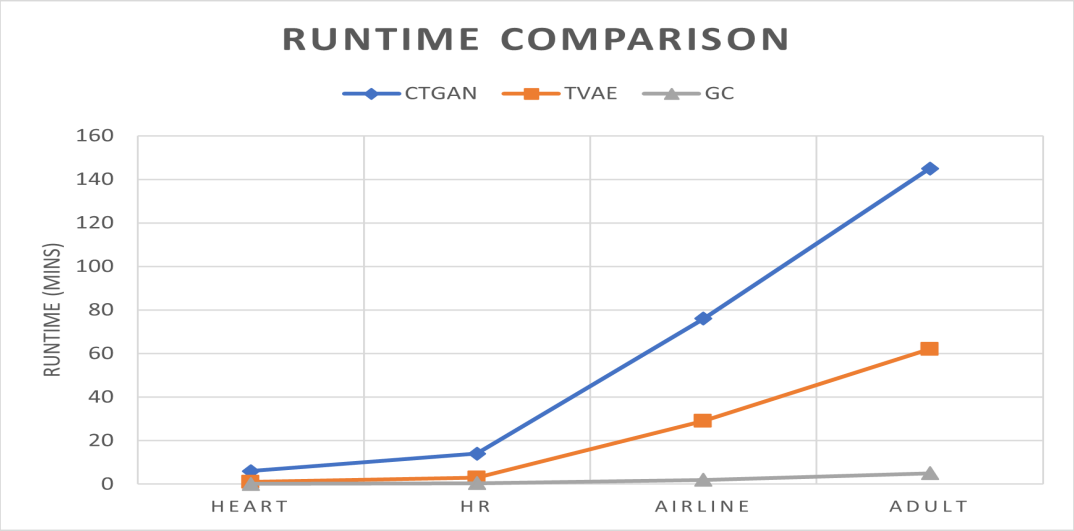

Figure 1: Runtime Analysis for the generative models

Results on HR dataset The qualitative results are presented for HR dataset with the CTGAN + DbP generated private with the following plots in figure 2, 3 and 4.

Results on MIMIC-3 dataset The qualitative results are presented for MIMIC-3 dataset with the CTGAN + DbP generated private with the following plots in figure 5, 6 and 7.

Results on AIRLINE dataset The qualitative results are presented for AIRLINE dataset with the CTGAN + DbP generated private with the following plots in figure 8, 9 and 10.

Results on HEART dataset The qualitative results are presented for HEART dataset with the CTGAN + DbP generated private with the following plots in figure 11, 12 and 13.

Results on ADULT dataset The qualitative results are presented for ADULT dataset with the CTGAN + DbP generated private with the following plots in figure 14, 15 and 16.

4.2 Results with TVAE

In the second set of experiments, TVAE is utilize as the generator. Results are provided for the same.

Results on HR dataset The qualitative results are presented for HR dataset with the TVAE + DbP generated private with the following plots in figure 17, 18 and 19.

Results on MIMIC-3 dataset The qualitative results are presented for MIMIC-3 dataset with the TVAE + DbP generated private with the following plots in figure 20, 21 and 22.

Results on AIRLINE dataset The qualitative results are presented for AIRLINE dataset with the TVAE + DbP generated private with the following plots in figure 23, 24 and 25.

Results on HEART dataset The qualitative results are presented for HEART dataset with the TVAE + DbP generated private with the following plots in figure 26, 27 and 28.

Results on ADULT dataset The qualitative results are presented for ADULT dataset with the TVAE + DbP generated private with the following plots in figure 29, 30 and 31.

4.3 Results with GC

In the third set of experiments, GC is utilize as the generator. Results are provided for the same.

Results on HR dataset The qualitative results are presented for HR dataset with the GC + DbP generated private with the following plots in figure 32, 33 and 34.

Results on MIMIC-3 dataset The qualitative results are presented for MIMIC-3 dataset with the GC + DbP generated private with the following plots in figure 35, 36 and 37.

Results on AIRLINE dataset The qualitative results are presented for AIRLINE dataset with the GC + DbP generated private with the following plots in figure 38, 39 and 40.

Results on HEART dataset The qualitative results are presented for HEART dataset with the GC + DbP generated

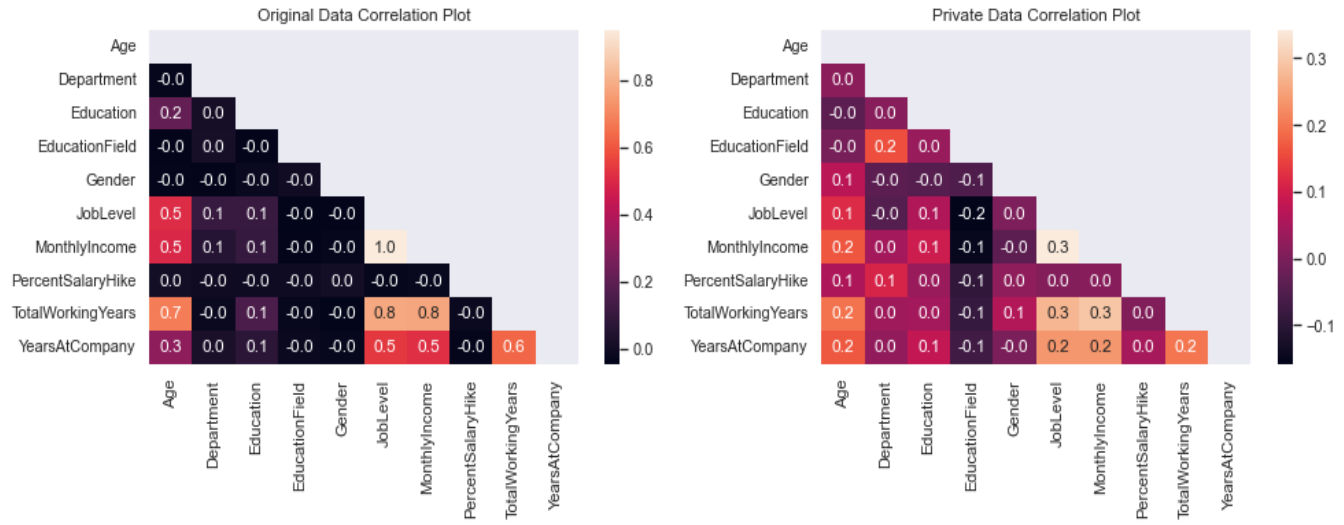

Figure 2: Correlation heatmap comparison for HR dataset with CTGAN + DbP

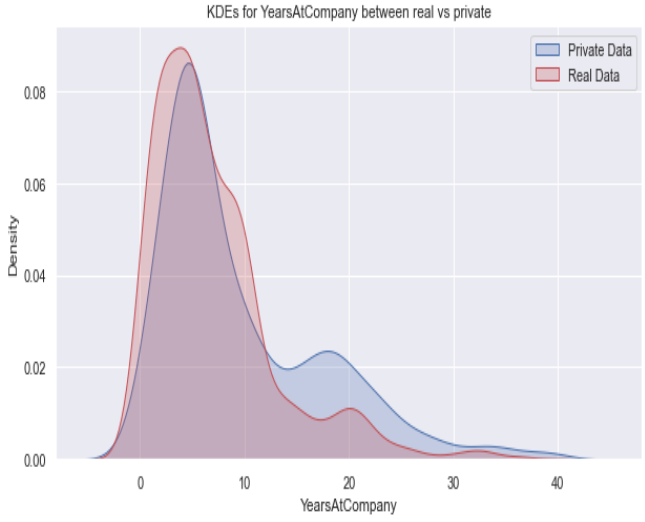

Figure 3: Univariate comparison for a randomly picked feature for HR dataset with CTGAN + DbP

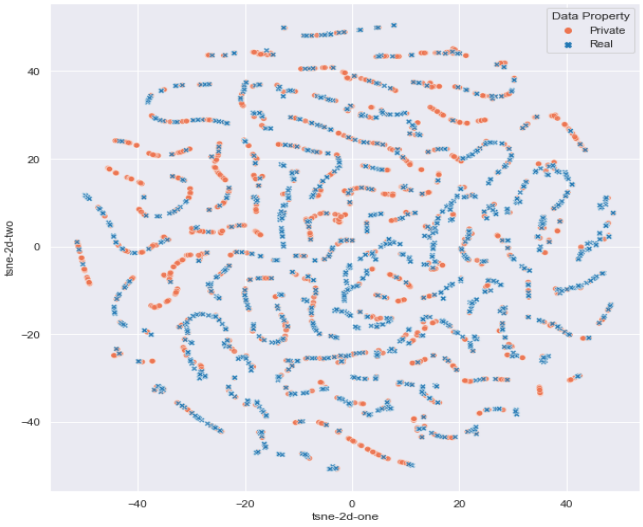

Figure 4: tSNE comparison for HR dataset with CTGAN +DbP

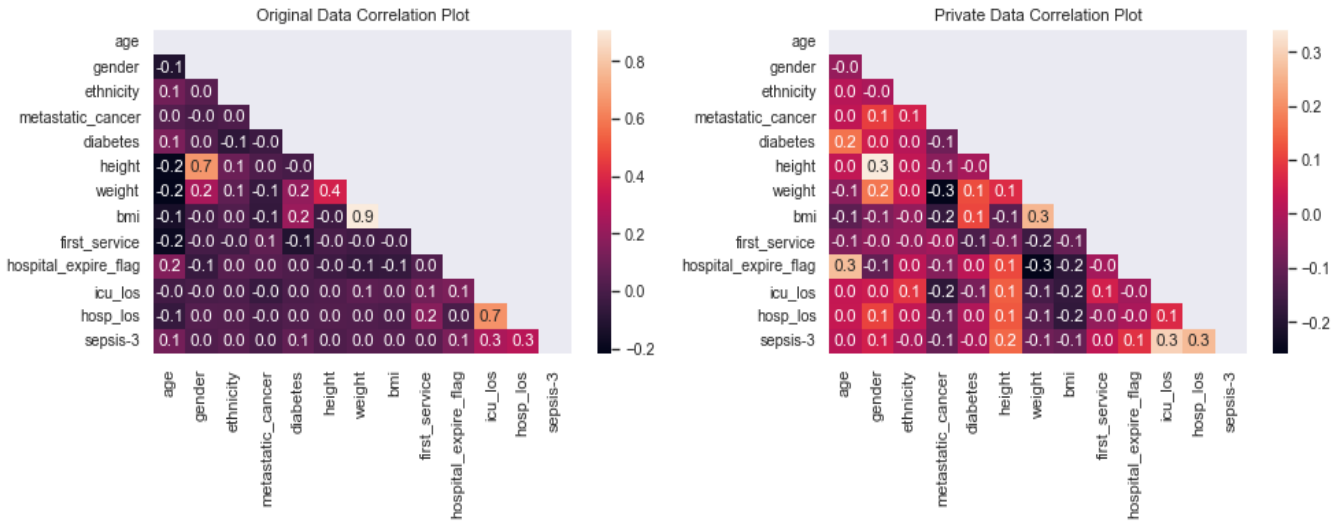

Figure 5: Correlation heatmap comparison for MIMIC-3 dataset with CTGAN + DbP

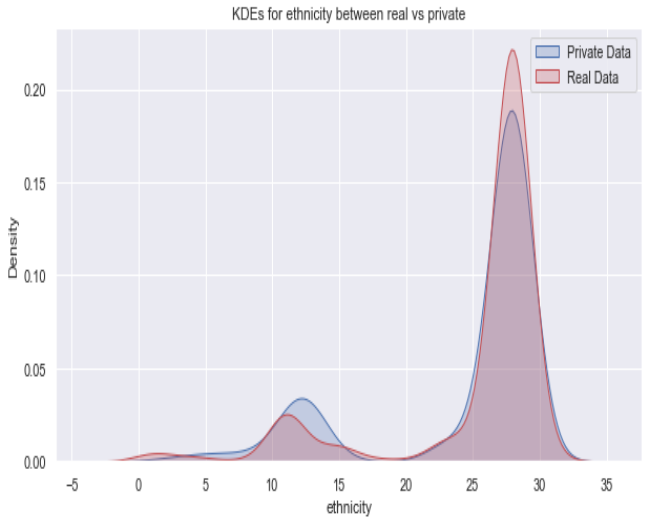

Figure 6: Univariate comparison for a randomly picked feature for MIMIC-3 dataset with CTGAN + DbP

Figure 7: tSNE comparison for MIMIC-3 dataset with CTGAN +DbP

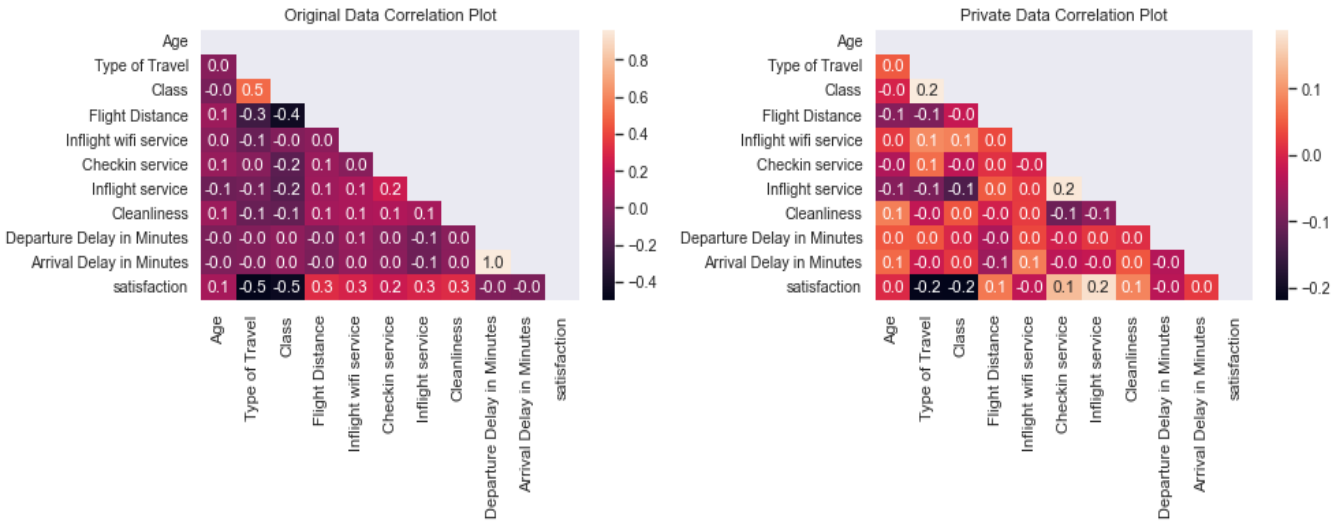

Figure 8: Correlation heatmap comparison for AIRLINE dataset with CTGAN + DbP. Note: For AIRLINE dataset, we have shown the heatmap for a subset of the features as the whole image could not fit in the diagram. Doing so would make the diagram illegible.

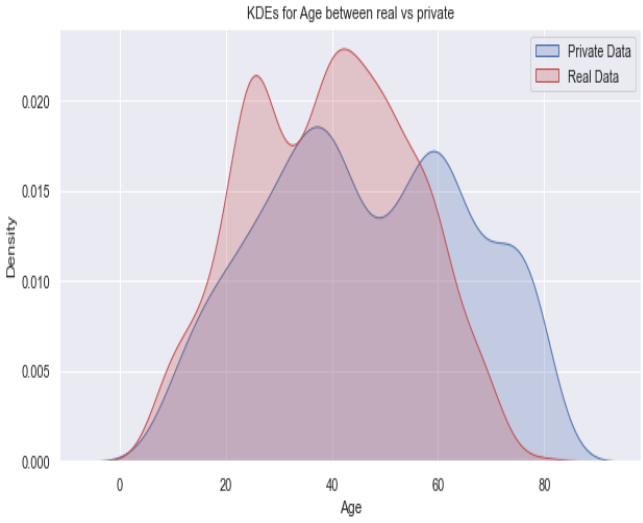

Figure 9: Univariate comparison for a randomly picked feature for AIRLINE dataset with CTGAN + DbP

Figure 10: tSNE comparison for AIRLINE dataset with CTGAN +DbP

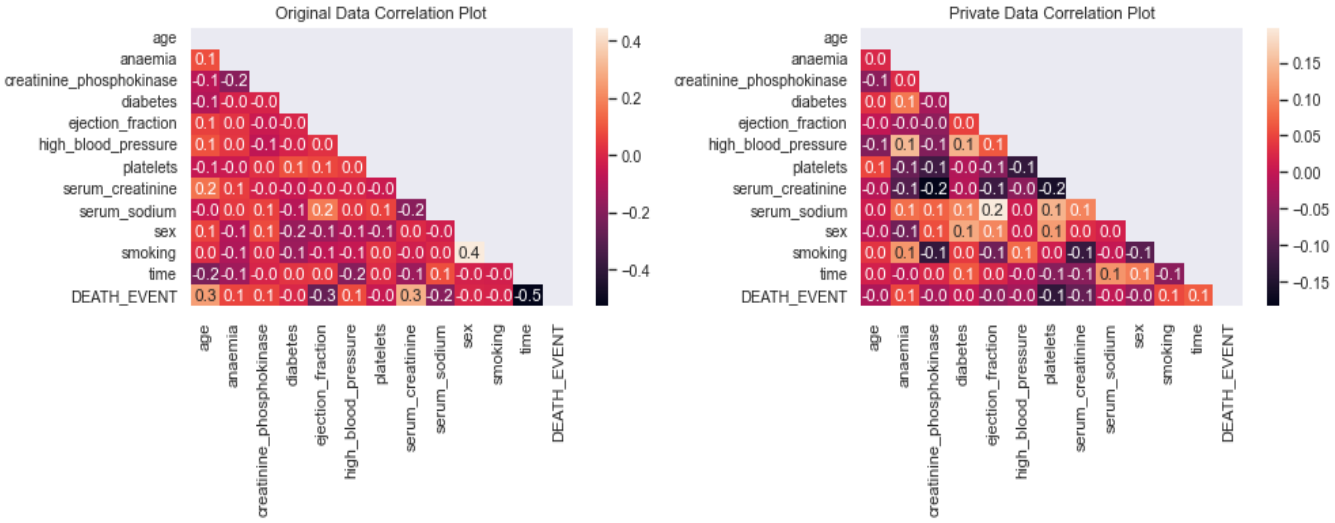

Figure 11: Correlation heatmap comparison for HEART dataset with CTGAN + DbP

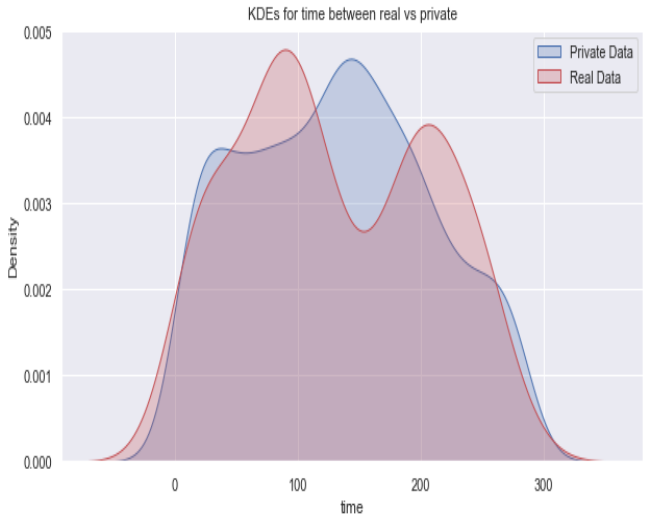

Figure 12: Univariate comparison for a randomly picked feature for HEART dataset with CTGAN + DbP

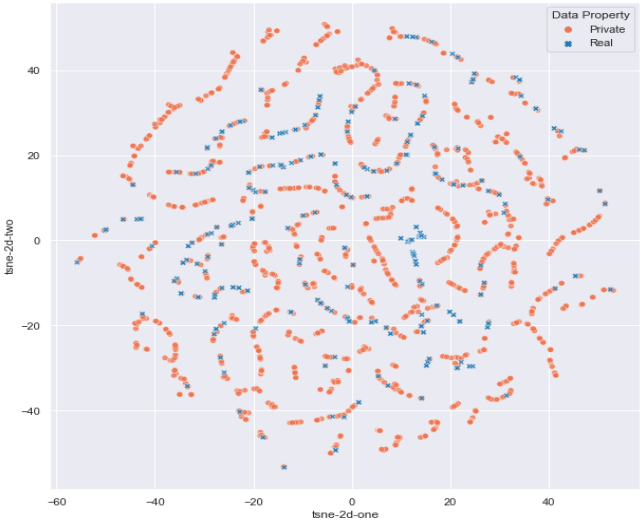

Figure 13: tSNE comparison for HEART dataset with CTGAN +DbP

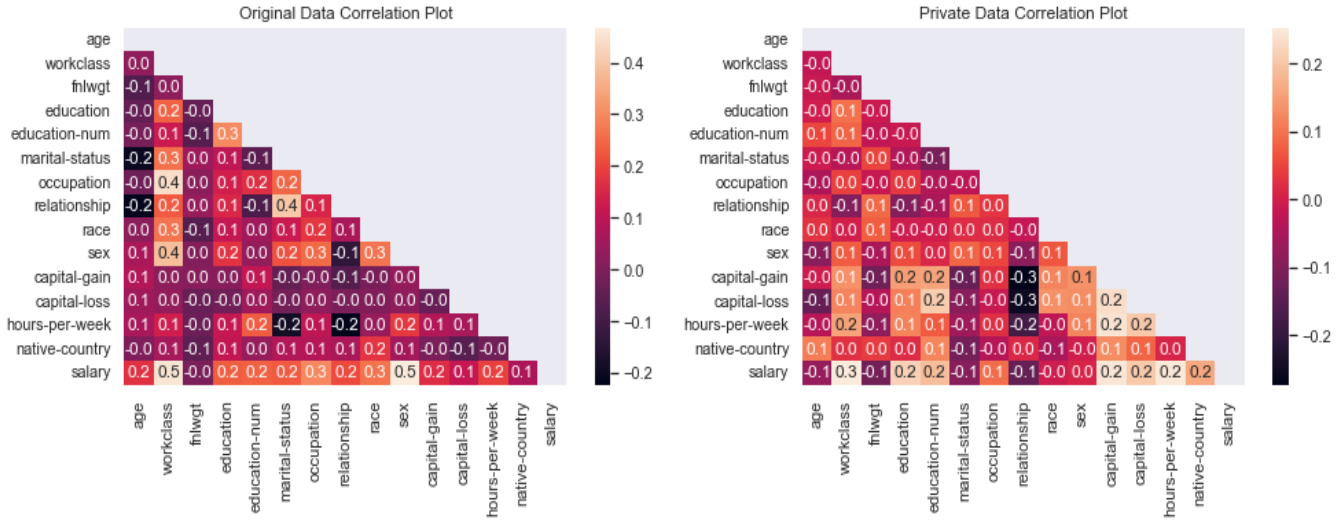

Figure 14: Correlation heatmap comparison for ADULT dataset with CTGAN + DbP

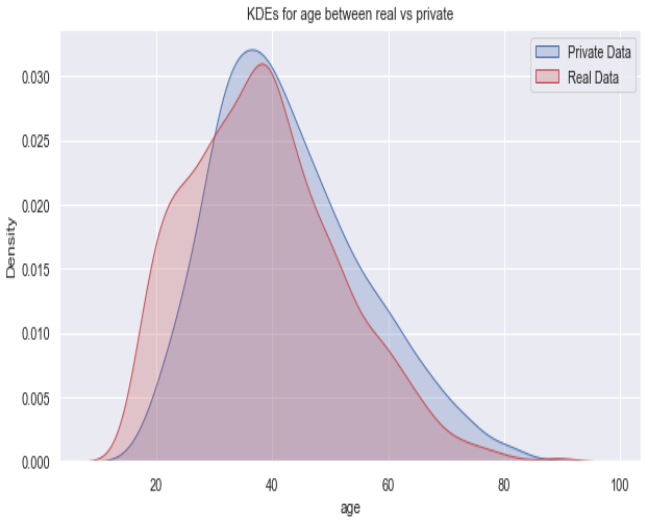

Figure 15: Univariate comparison for a randomly picked feature for ADULT dataset with CTGAN + DbP

Figure 16: tSNE comparison for ADULT dataset with CTGAN +DbP

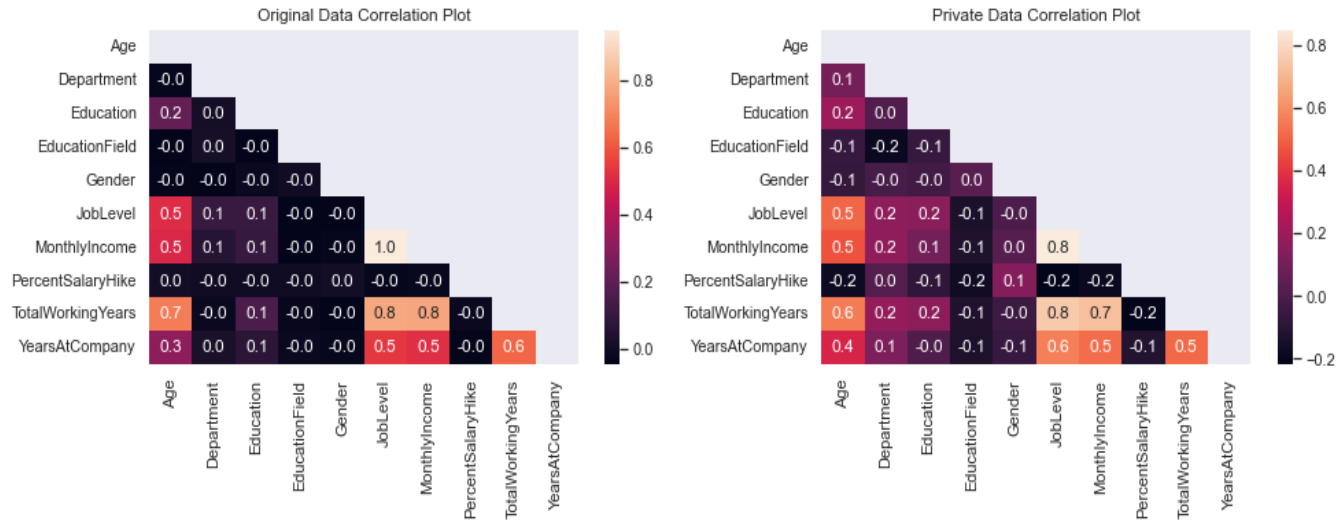

Figure 17: Correlation heatmap comparison for HR dataset with TVAE + DbP

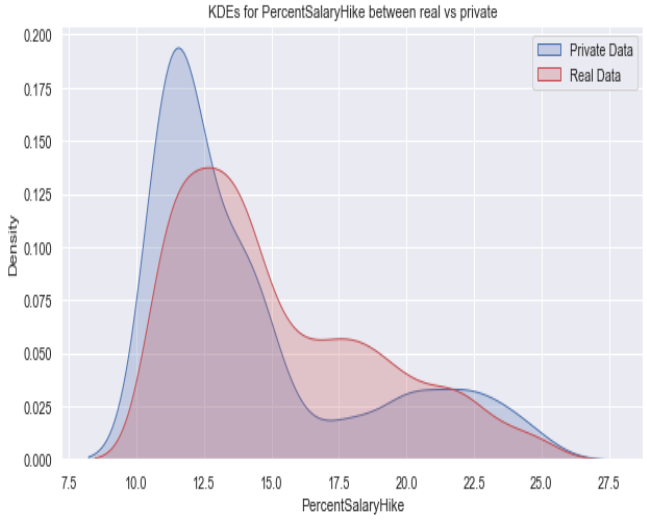

Figure 18: Univariate comparison for a randomly picked feature for HR dataset with TVAE + DbP

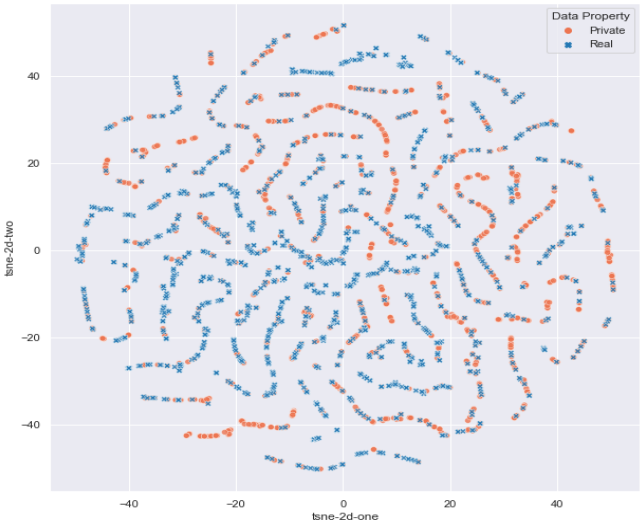

Figure 19: tSNE comparison for HR dataset with TVAE +DbP

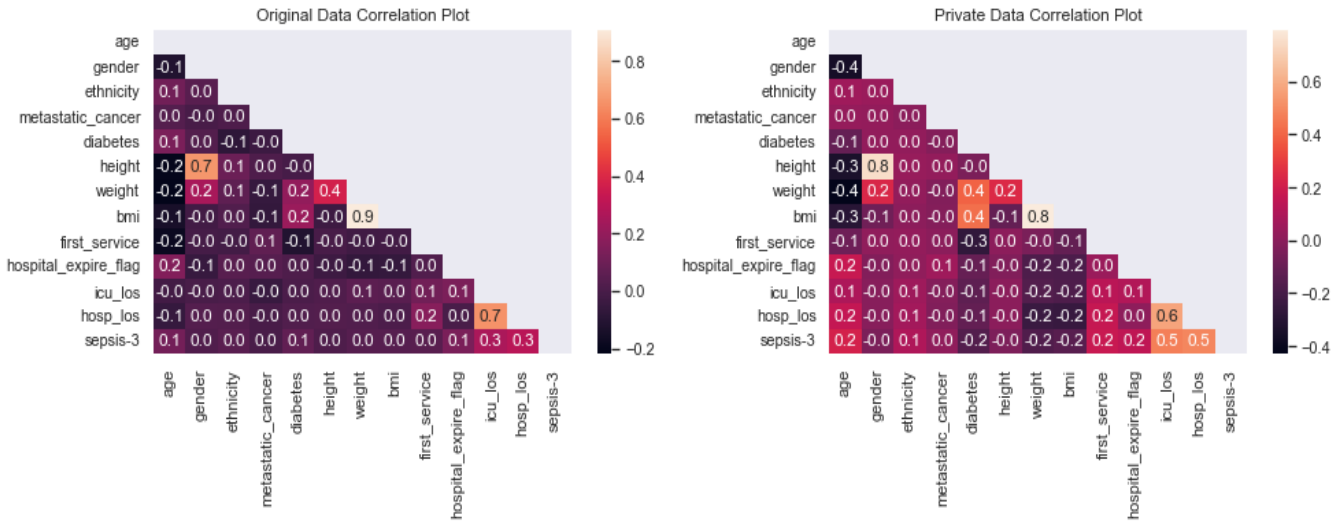

Figure 20: Correlation heatmap comparison for MIMIC-3 dataset with TVAE + DbP

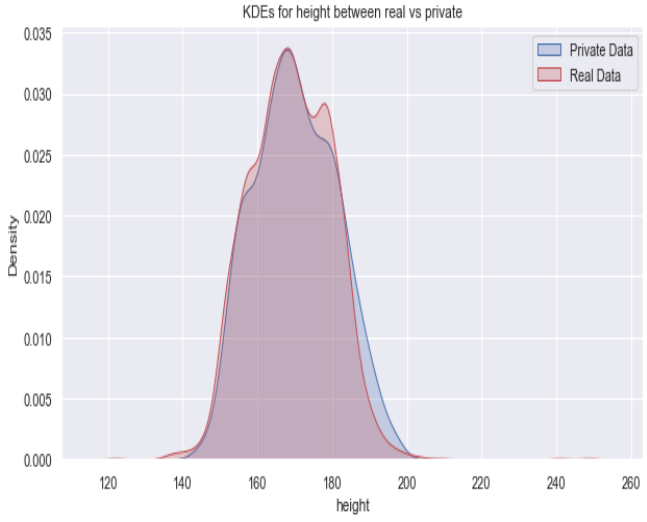

Figure 21: Univariate comparison for a randomly picked feature for MIMIC-3 dataset with TVAE + DbP

Figure 22: tSNE comparison for MIMIC-3 dataset with TVAE +DbP

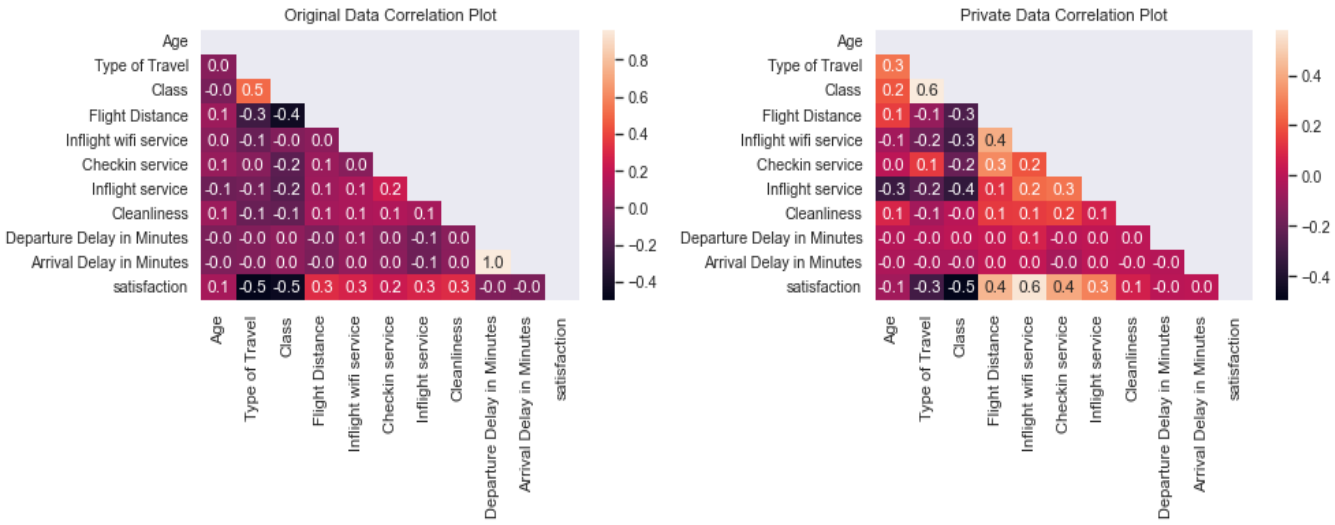

Figure 23: Correlation heatmap comparison for AIRLINE dataset with TVAE + DbP. Note: For AIRLINE dataset, we have shown the heatmap for a subset of the features as the whole image could not fit in the diagram. Doing so would make the diagram illegible.

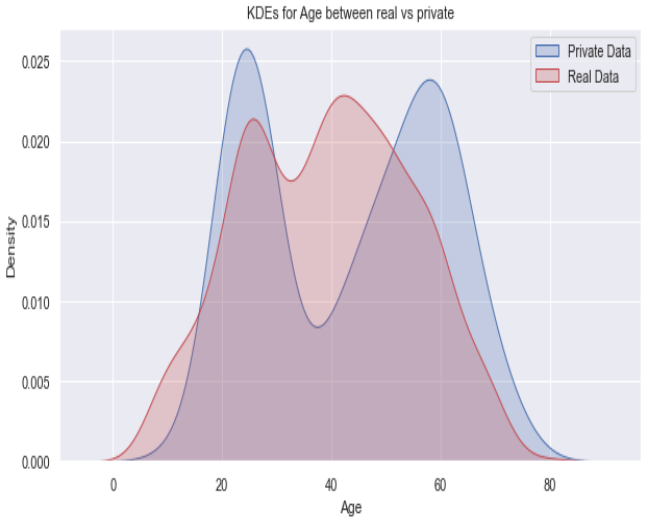

Figure 24: Univariate comparison for a randomly picked feature for AIRLINE dataset with TVAE + DbP

Figure 25: tSNE comparison for AIRLINE dataset with TVAE +DbP

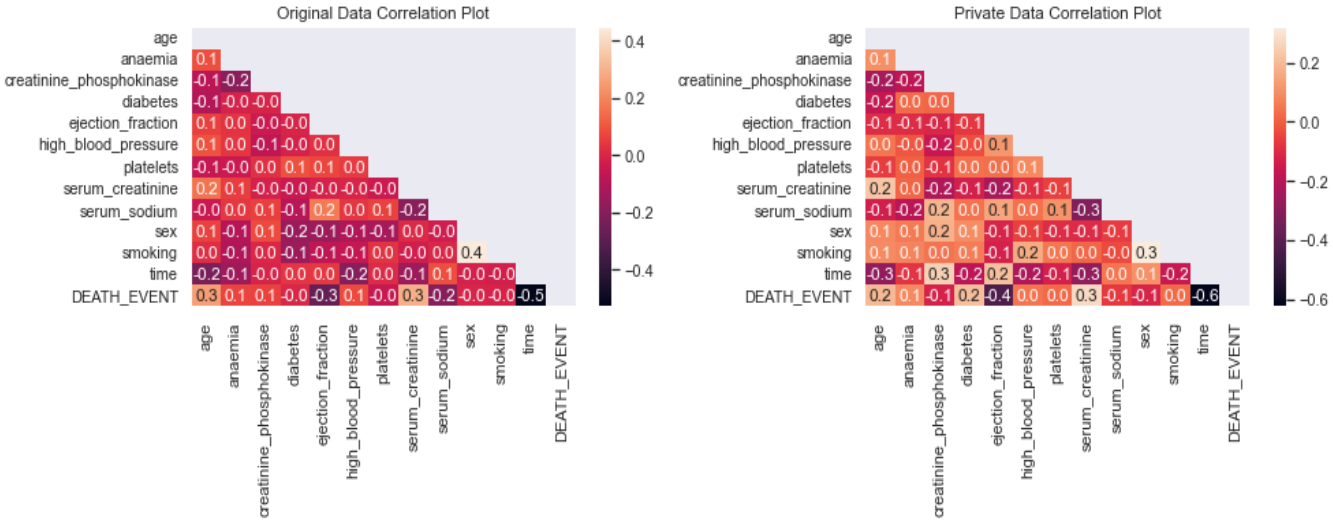

Figure 26: Correlation heatmap comparison for HEART dataset with TVAE + DbP

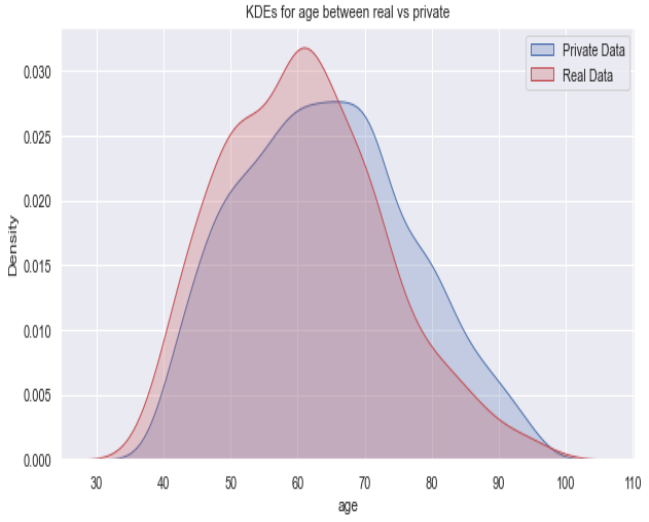

Figure 27: Univariate comparison for a randomly picked feature for HEART dataset with TVAE + DbP

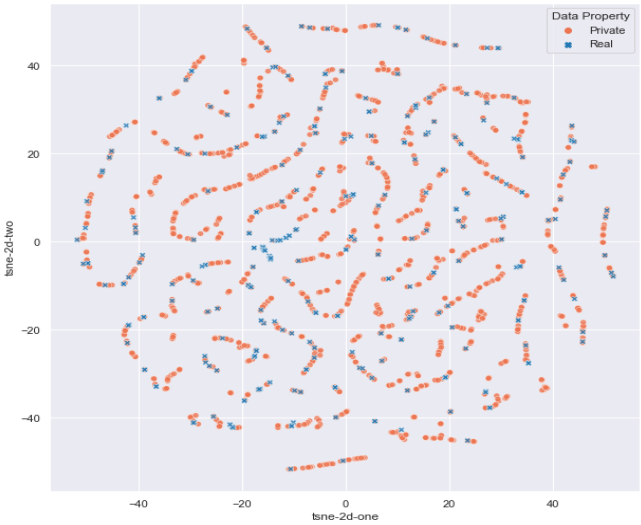

Figure 28: tSNE comparison for HEART dataset with TVAE +DbP

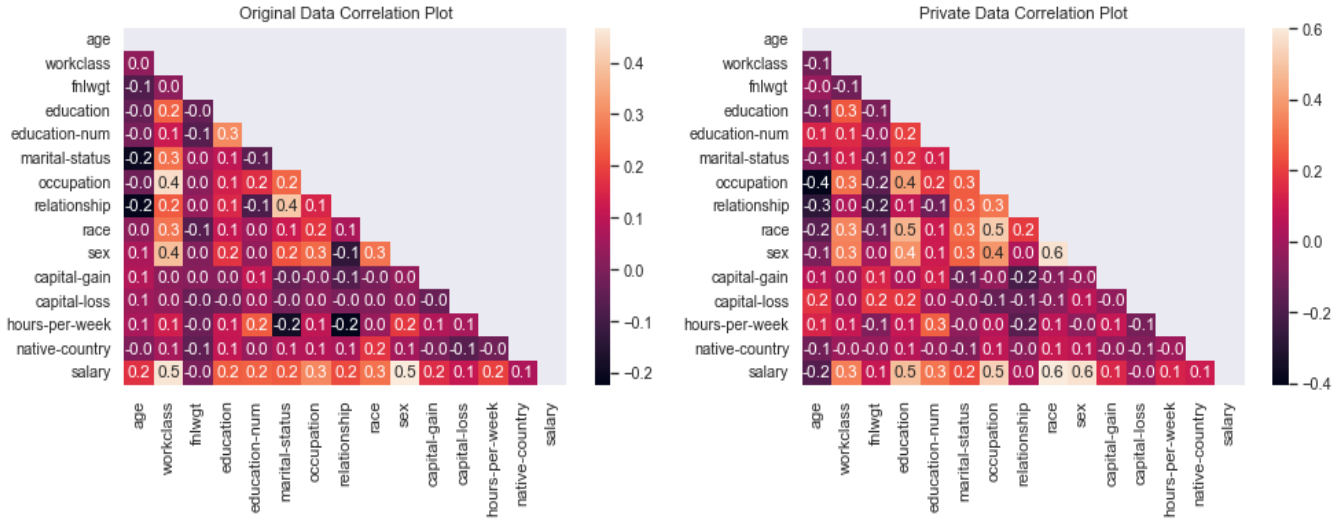

Figure 29: Correlation heatmap comparison for ADULT dataset with TVAE + DbP

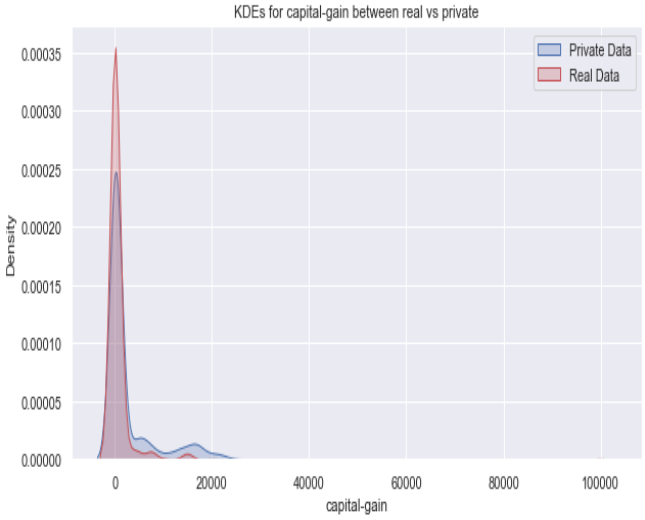

Figure 30: Univariate comparison for a randomly picked feature for ADULT dataset with TVAE + DbP

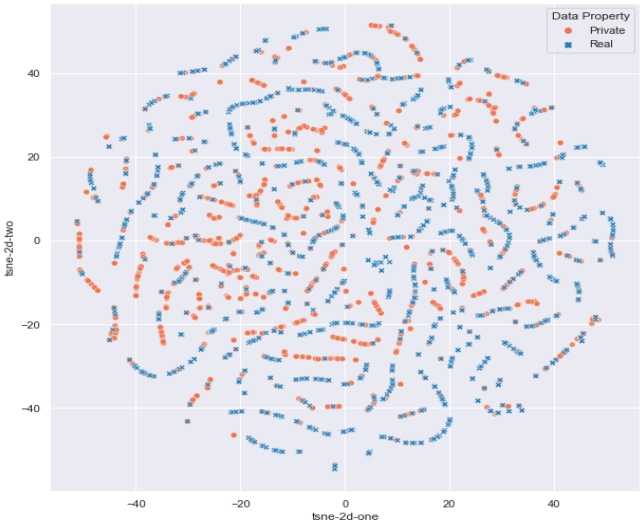

Figure 31: tSNE comparison for ADULT dataset with TVAE +DbP

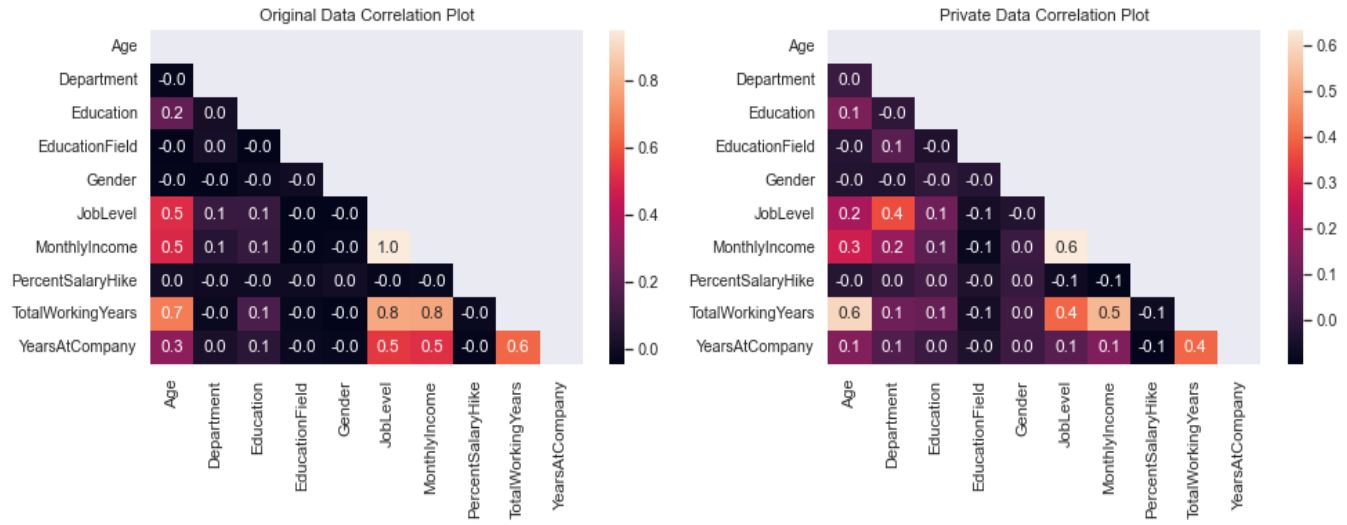

Figure 32: Correlation heatmap comparison for HR dataset with GC + DbP

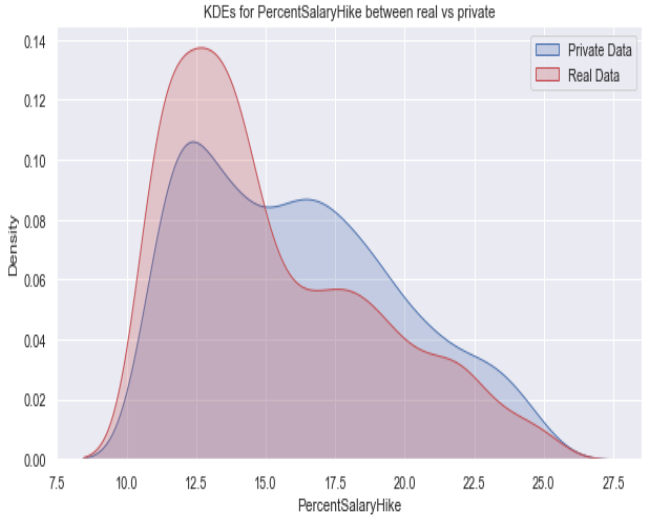

Figure 33: Univariate comparison for a randomly picked feature for HR dataset with GC + DbP

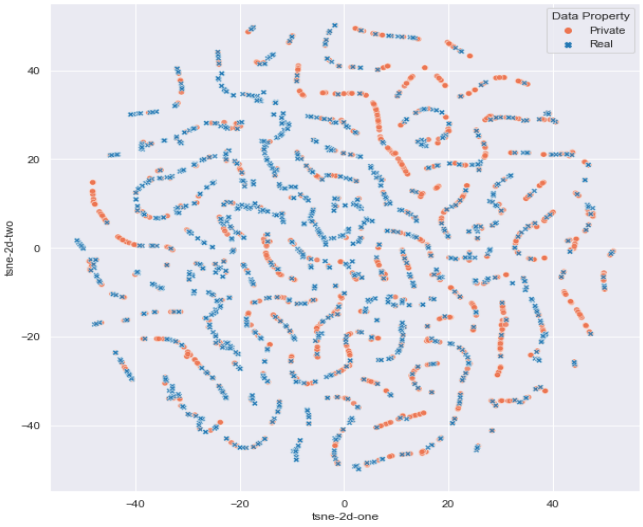

Figure 34: tSNE comparison for HR dataset with GC +DbP

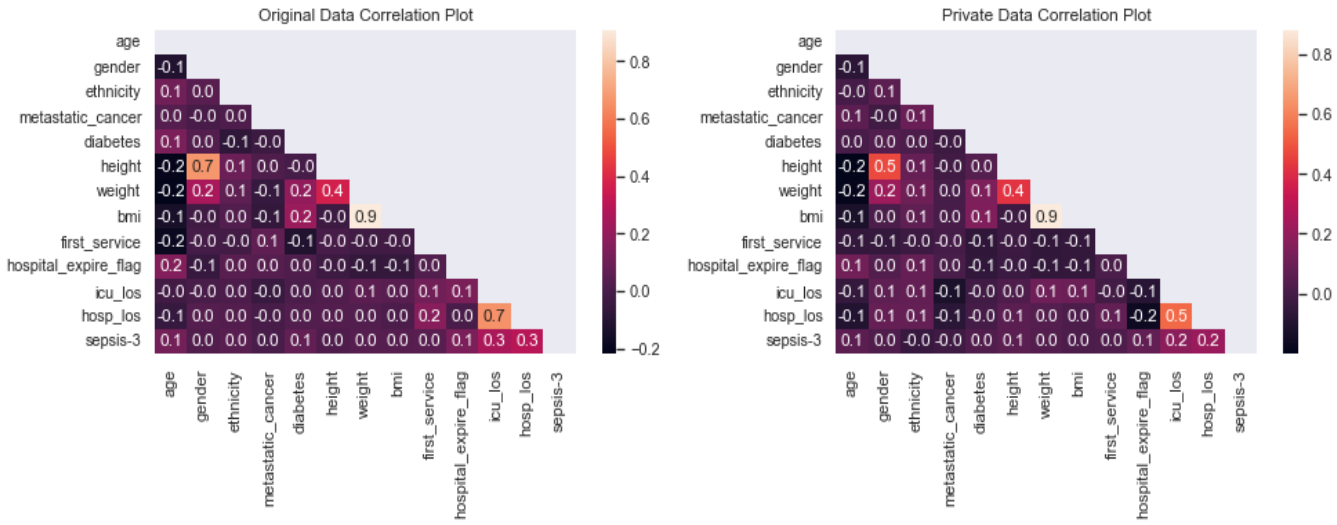

Figure 35: Correlation heatmap comparison for MIMIC-3 dataset with GC + DbP

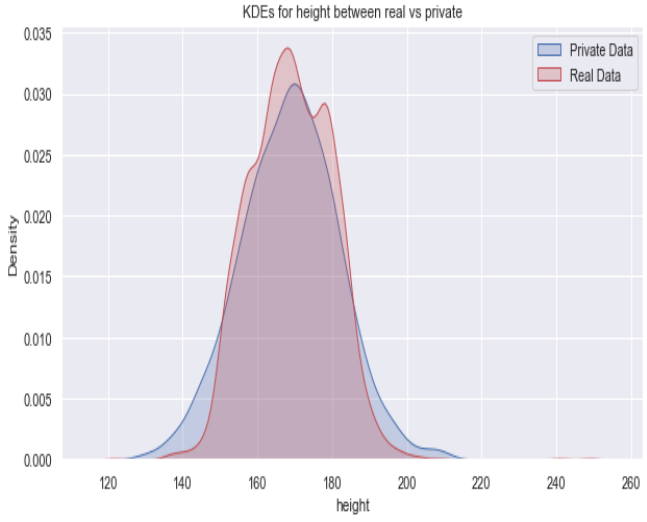

Figure 36: Univariate comparison for a randomly picked feature for MIMIC-3 dataset with GC + DbP

Figure 37: tSNE comparison for MIMIC-3 dataset with GC +DbP

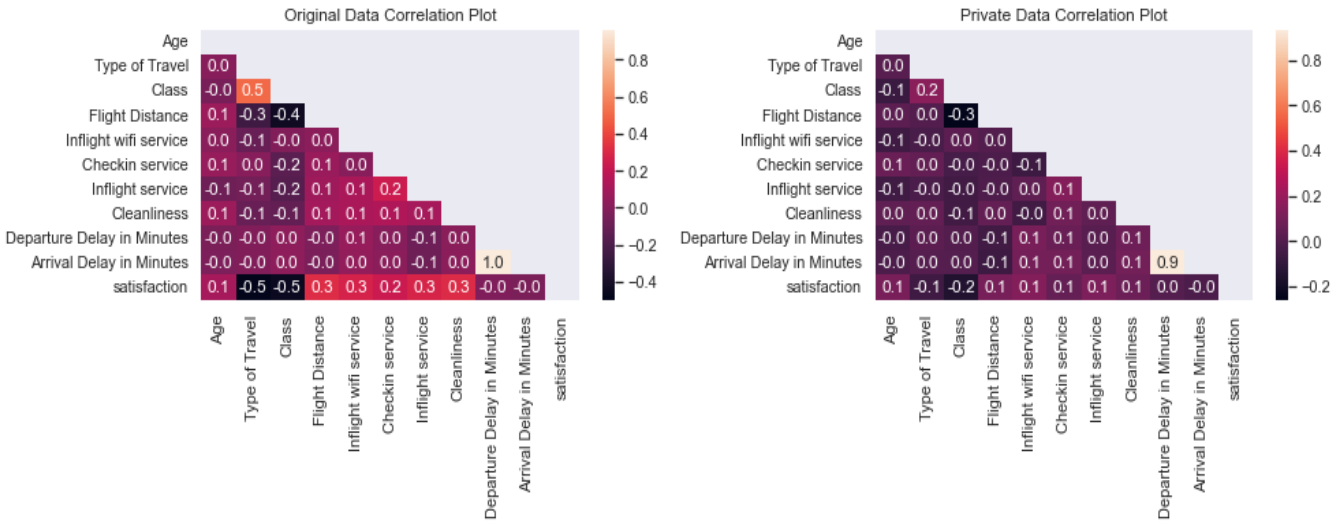

Figure 38: Correlation heatmap comparison for AIRLINE dataset with GC + DbP. Note: For AIRLINE dataset, we have shown the heatmap for a subset of the features as the whole image could not fit in the diagram. Doing so would make the diagram illegible.

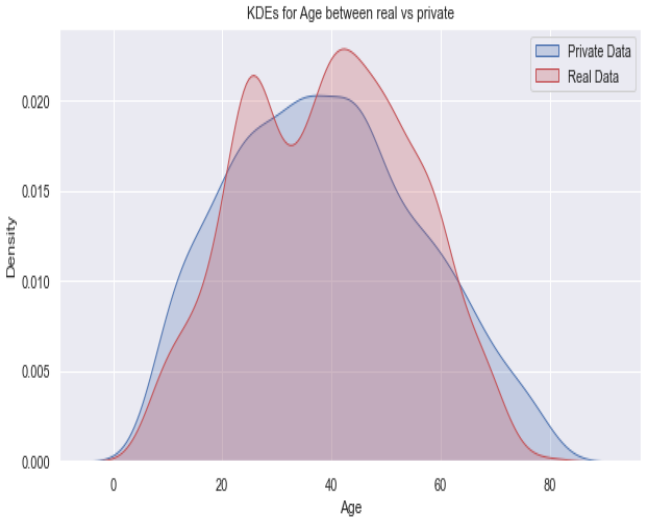

Figure 39: Univariate comparison for a randomly picked feature for AIRLINE dataset with GC + DbP

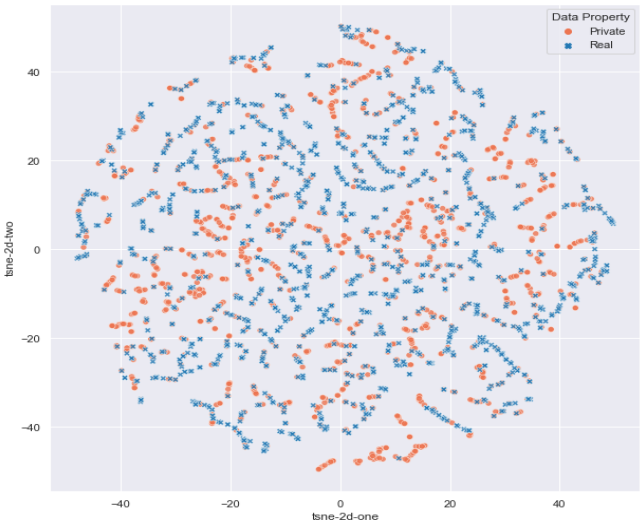

Figure 40: tSNE comparison for AIRLINE dataset with GC +DbP

private with the following plots in figure 41, 42 and 43.

Results on ADULT dataset The qualitative results are presented for ADULT dataset with the GC + DbP generated private with the following plots in figure 44, 45 and 46.

4.4 Sensitivity Analysis on Privacy Threshold

With the proposed privacy mechanism we utilize a cut-off value θ on the minimum HMJD to determine which are the 'risky' samples. The choice of this cut-off is pertinent as it decides the samples to discard in-order to improve privacy. As discussed we utilized a chi-saured distribution and utilized the cardinality (i.e. number of features) as the degrees of freedom to obtain the threshold. Additionally, we also performed a sensitivity analysis to explore the threshold vs privacy and utility trade-off.

With high value of threshold we tend to identify more samples as 'risky' and discard them to improve privacy. On the flip side, this ends up diluting the utility. For the results reported in this study (main paper table 1 and 2) we utilize significance level of 99% for the chi-squared distribution as a solid balance between privacy and utility. More results with different significance levels are provided here for the HR dataset (cardinality is 10, sample size n 1500). Please note that we start with generating 5 times more synthetic samples than what we want our private dataset to have (for experiments we kept the same number of private samples as original therefore we started with 5n synthetic samples where n is the number of samples in original dataset).

Significance Level (α)	**Cut-off**	**RIA (%)**	**ML Utility**
0.995	2.16	92	0.81
0.99*	**2.56***	**95**	**0.78**
0.95	3.94	96	0.74
0.90	4.87	98	0.69
0.80	6.18	100	0.52

Table 3: Sensitivity analysis on Privacy cut-off for DbP. In the main paper results are provided for the second selection (cut-off value of 2.56.

We observe that with lower significance level (i.e. probability of exceeding the critical value) the threshold value gets higher and we lose more samples leading to higher privacy estimation but lower utility.

4.5 Hyper-parameter Selection

The primary CTGAN hyper-parameters that are tuned during data synthesis, are generator and discriminator learning rates $2e^{-4}$, batch size between $[32, 128]$ and epoch size between $[300, 3000]$. The only other hyper-parameter in the DbP setup is the privacy threshold value θ. In our experiments θ is derived from the chi-squared table with a significance level of 0.99. Next, details are provided on the attacks. In attribute inference attack (AIA), regression and classification tasks are utilized and the accuracy (A) is estimated with the adjusted R-squared/ F1-score. For the re-identification attack (RIA) technically any distance metric can be used. In this paper the HMJD is used for the same. Similarly, β value also can be chosen empirically, however we used a significance level of 95% to get the cut-off $\beta = \chi^2(0.95, p)$ from the chi-squared table, with p as the number of features. Finally, for the membership inference attack (MIA), the $R1$ and $R2$ split is done randomly in a 80:20 proportion. More details can be found in the supplementary materials.

4.6 Privacy vs Utility Regression Analysis:

In the final section of the main manuscript we present a regression analysis. Here we fit the privacy values (average privacy against three types of attacks) as the predictor (i.e. X) and the ML utility values as the target (y). We have 4 data-points for the first 4 datasets as shown in table 1 (main paper). Next, we fit a simple regression line between these points first for DP-CTGAN and then for DbP-CTGAN. The resultsa re provided in figure 47.

The plot (and all the results in main paper) are based on 5 independent runs and the privacy/ utility values reported are all average of these 5 runs. From the figure we see negative slopes for both regression lines (DP and DbP). This indicates the decreasing utility with increasing privacy. Finally, we also observe that the slope (absolute value of the co-efficient) for the DP regression line is much higher than DbP. This further indicates to a more drastic dilution of utility with privacy for DP mechanism.

5 Implementation Details

Few other details that could not be provided in the main manuscript for the constraint of space are listed here,

- **Start with more synthetic samples than required:** As we iteratively discard risky samples with DbP therefore we should start with more synthetic samples. Please note, generating synthetic samples is a very low expense process for all of the aforementioned generators (once they are trained). After a thorough experiments we have concluded that, if we need to generate n number of 'private' samples, we should start with at least $5 * n$ number of synthetic samples to be on the safer side. If we end up with more than n records after the DbP process, we can always sample the required number of records from that.
- **Choice of Distance Metric:** We have utilized the HMJD in the main manuscript as the datasets are often mixed type. However, one can stick to a Minkowski or Mahanalobish distance if they had only numerical columns in the dataset. We even observed decent results with Minkowski distance (after integer encoding the categorical columns)
- **The privacy threshold :** θ is chosen from a chi-squared table in the main manuscript. However, the same can be done by an expert user as well depending on the requirement for privacy. A higher threshold value genrally results in removal of more samples therefore more privacy but lower utility.
- **Re-identification Inference Attack :** The main idea behind RIA is as follows: if an adversary has access to the the synthetic data-set (publicly open), can they re-identify an original record using distance based similarity? To simulate this attack, each of the synthetic record

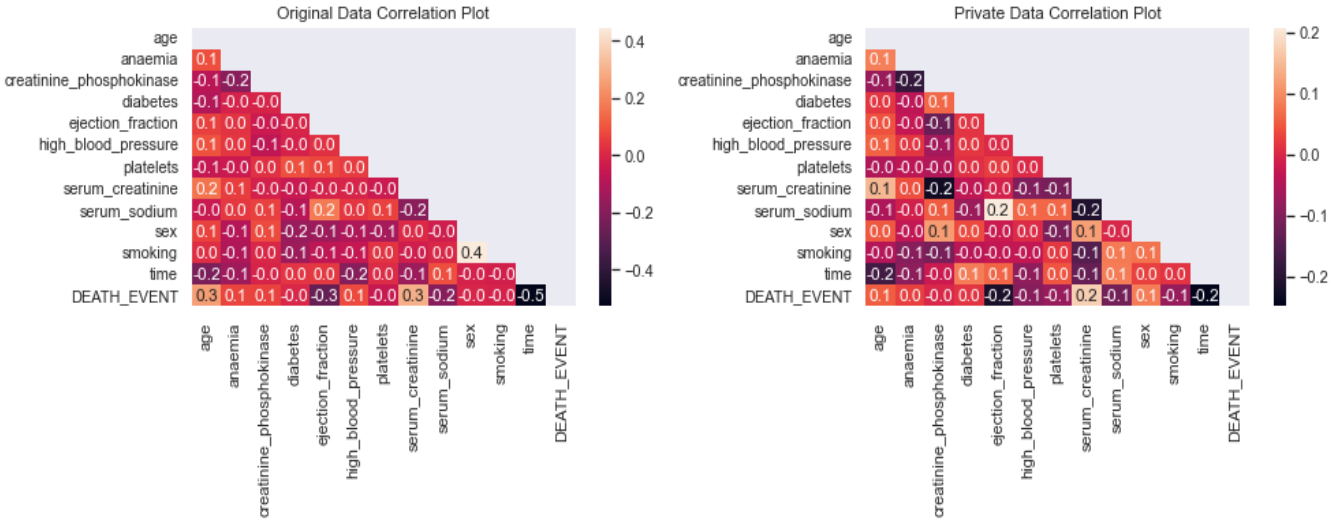

Figure 41: Correlation heatmap comparison for HEART dataset with GC + DbP

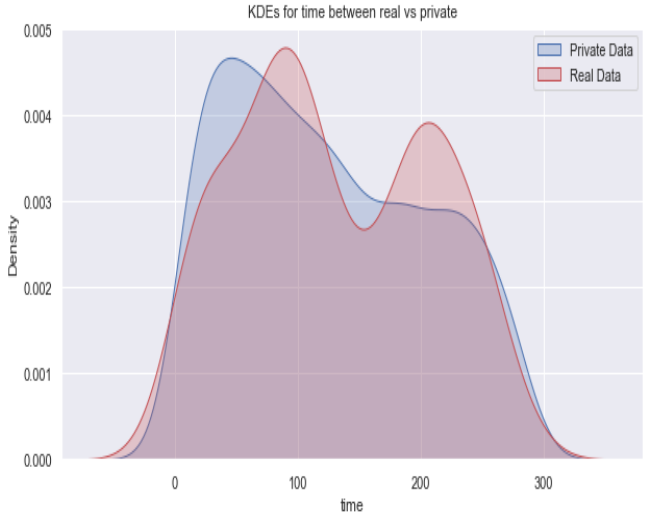

Figure 42: Univariate comparison for a randomly picked feature for HEART dataset with GC + DbP

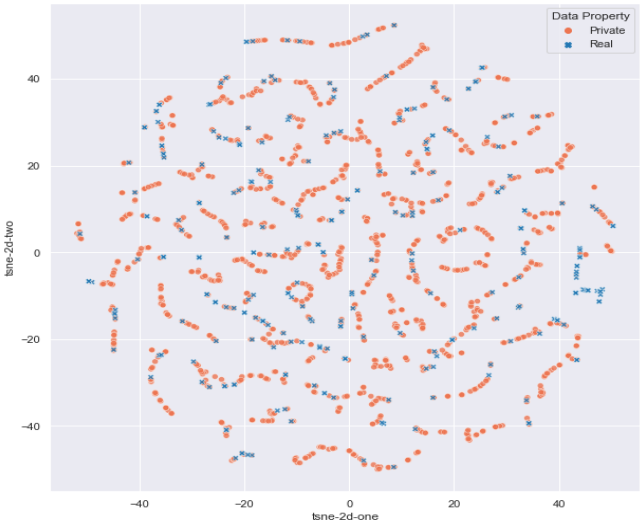

Figure 43: tSNE comparison for HEART dataset with GC +DbP

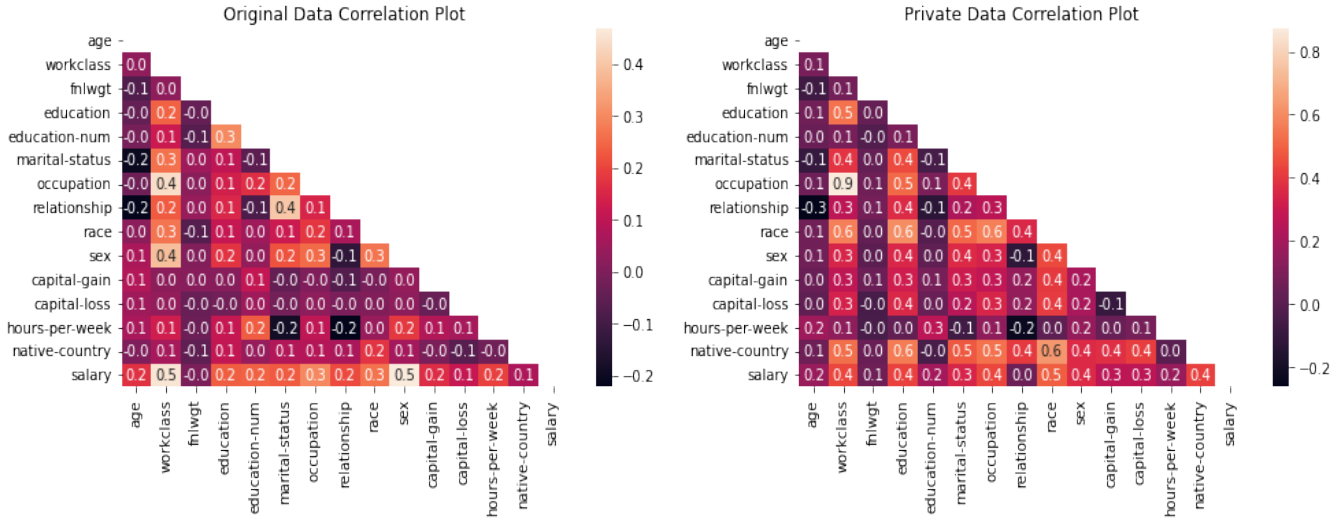

Figure 44: Correlation heatmap comparison for ADULT dataset with GC + DbP

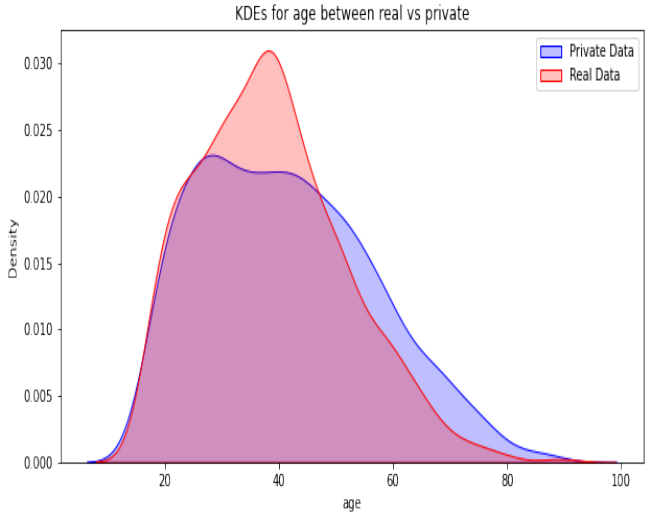

Figure 45: Univariate comparison for a randomly picked feature for ADULT dataset with GC + DbP

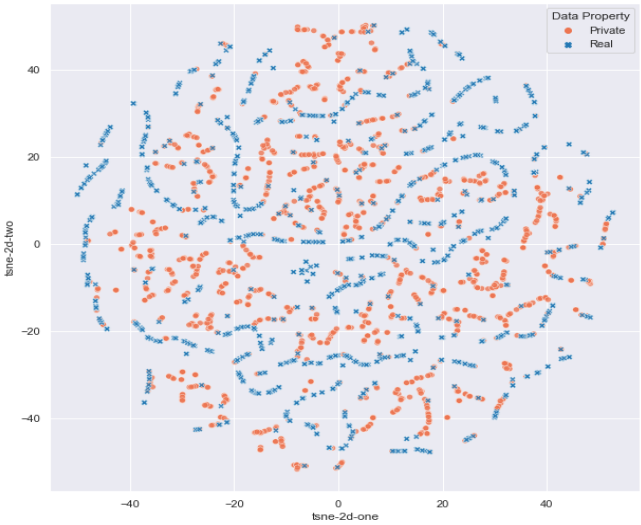

Figure 46: tSNE comparison for ADULT dataset with GC +DbP

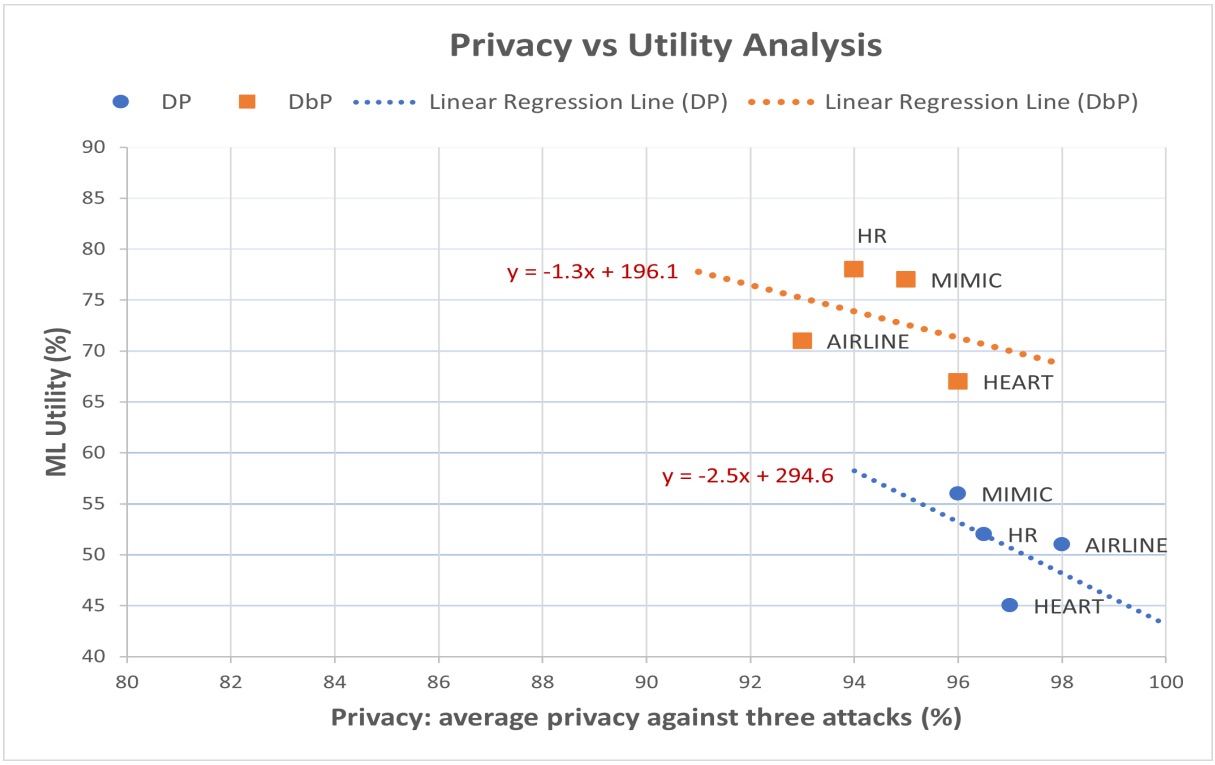

Figure 47: Privacy vs Utility Analysis (Regression Lines)

is picked and its closest match in the original record is found, using a distance based mechanism (any type of suitable distance metric can be utilized here [Yoon *et al.*, 2020]). If the corresponding distance is found to be lower than the specified cut-off then it is concluded that the adversary has successfully re-identified the real record. The privacy against RIA i.e. P_R is quantified using the proportion of the data-set for which the attack is not successful i.e. proportion of 'non-risky' samples in the data-set,

$$P_R = (1 - \frac{|\{y_j \in S | D_j \leq \beta\}|}{|S|}) * 100\% \qquad (1)$$

where $|.|$ denotes the number of samples, D_j is the distance between j^{th} synthetic sample and its closest original match, and β is the cut-off that determines the power of the attack.

Example: A P_R value of 90% means, 90% of the samples are not vulnerable to RIA. A high P_R is desired.

In RIA we measure a how easy it is for the adversary to re-identify an original sample from a private synthetic sample using distance based matching. The attack is simulated by comparing the minimum HMJD for each private sample to find the closest original match. If the match is found to be smaller than a certain threshold β we conclude that the adversary has been successful in his RIA. We perform this for all the synthetic samples and find out the proportion of samples in the private dataset that was re-identified. This proportion is reported as the measurement of privacy against RIA. The β value is chosen from a chi-squared table with a significance level of 95%. This basically means, that if we have 95% confidence in the RIA-privacy estimation.

- **Attribute Inference Attack :** Here, an adversary tries to reveal a particular sensitive field for a real record [Kunar *et al.*, 2021],[Mendelevitch and Lesh, 2021]. This attack is modified based on the hypothesis of differential privacy. According to DP, the inclusion of a single record to the synthetic data-pool should not change the overall attack accuracy significantly. To simulate this attack, random original samples are attached with the synthetic data one at a time, to track the average change e in AIA accuracy. The privacy against AIA is denoted with P_A,

$$P_A = (1-e) * 100\% = (1 - |A_S - \bar{A}_{SR}|) * 100\% \quad (2)$$

where A_S is the accuracy of the attribute attack when the adversary has access to only the synthetic data S. $\bar{A}_{SR}$ on the other hand, denotes the average attack accuracy when each of the real sample is added to S one by one e.g. $\bar{A}_{SR} = \frac{\sum(A_{Sr})}{|R|}$ for each $r \in R$.

Example: A P_A value of 90% means, while revealing an attribute the probability of making error by the adversary is 90% (accuracy is 10%). A high P_A is desired.

The AIA attack is further summarized with the help of schematic diagram in figure 48. The diagram shows how to compute the difference in the AIA attack accuracy i.e. e for a synthetic and a private (DbP) dataset. The privacy metric is computed in a percentage from i.e.

$P_A = (1 - e) * 100\%$. From the diagram we observe that with private data the e value is much lower than the synthetic data as expected.

- **Membership Inference Attack :** T With MIA, an adversary tries to infer if a synthetic sample can be matched to an original sample that belongs to the training data-set that was used to create the synthetic data-set in the first place [Mendelevitch and Lesh, 2021] i.e. if its membership can be inferred. This can have significant impact. For example: If an adversary is aware that the synthetic data-set S is cancer related and is also able to find out that a synthetic sample can be matched to the training set R_1, then they can confidently infer that the victim has cancer.

  To simulate this attack, first the real data R is divided into two parts randomly, $R1$: which is used to create the synthetic data and $R2$ which is kept aside. Next, each synthetic record is tested against all of the records to find 'risky' samples (very similar to the original) in R. Four cases arise,

  - **True Positive (TP):** When adversary finds a 'risky' sample that is part of R1.
  - **False Positive (FP):** When adversary finds a 'risky' sample that is part of R2.
  - **True Negative (TN):** When adversary finds a 'non-risky' sample that is part of R2.
  - **False Negative (FN):** When adversary finds a 'non-risky' sample that is part of R1.

  Using the information above a confusion matrix can be built. The F1-score (Harmonic mean of Precision and Recall) is used to measure the attack accuracy of the adversary. Privacy against membership attack i.e. P_M is,

$$P_M = (1-F1)*100\% = (1-\frac{TP}{TP + 0.5(FP + FN}) * 100\% \quad (3)$$

  Example: A P_M value of 90% means, while trying to infer the membership the adversary is only 10% accurate (i.e. 90% error probability). A high P_M is desired.

  he MIA attack is also summarized with the help of the schematic diagram in figure 49 (taken from [Mendelevitch and Lesh, 2021]). The diagram here demonstrates how to compute the membership of a synthetic sample and the privacy is computed in percentage with $P_M = (1 - F1 - score) * 100\%$. Please note we can divide R into R1 and R2 in any proportion we many want. We did a 80:20 random split in order to make the split imbalanced so that the attack simulation is more powerful.

- **Privacy Budget for DP-CTGAN :** In general, DP based methods in literature rely on the privacy budgets for the guarantee of privacy. A low value of ϵ indicates to a stricter privacy guarantee and vice versa. In our experiment, when we built our own DP-CTGAN we utilized a $\epsilon = 0.35$ for a very strong guarantee of privacy. The same is reflected in table 1 of main manuscript as we see it consistently outperforming the DbP in terms of the privacy attacks. However, at the same time, DbP showed much better utility. We have experimented with different privacy budgets (i.e. 0.35, 0.5, 1) however for each of those cases DP-CTGAN showed sub-par privacy against the attack simulations (although utility is better retained than $\epsilon = 0.35$). Therefore for our experiments we stuck to $\epsilon = 0.35$ for a fair comparison on privacy.

- **Noise-Level (pseudo) Approximation in DbP:** In DP we have a privacy budget (ϵ) that we can use to control the amount of noise injected into the system. Generally lower ϵ corresponds to more noise and therefore a stricter guarantee of privacy. In DbP however, we do not add any noise so we do not have a privacy budget. Therefore, for a meaningful comparison with the SOTA (in table 2) we required an analogous of this privacy budget for the proposed mechanism.

  To achieve this, first we project the datasets (original and DbP) into a continuous only feature space (as the raw data is mixed-type, it is harder to estimate noise, henceforth we do this projection). We have used a Eigen value decomposition based PCA approach to achieve this transformation. Next, we look at the distribution difference in an univariate level between the original and DbP first. The difference in standard-deviation can be utilized to estimate the scale of the noise (assuming Normal as used in DP-auto-GAN) e.g.

  $var[P_{F_i}] = var[O_{F_i}] + var[N_{F_i}] - 2cov[O_{F_i}, N_{F_i}]$
  $var[P_{F_i}] = var[O_{F_i}] + var[N_{F_i}]$
  $\sigma[P_{F_i}] = \sqrt{var[O_{F_i}] + var[N_{F_i}]}$

  where, $\sigma[P_{F_i}], \sigma[O_{F_i}], \sigma[N_{F_i}]$ represents the standard-deviation of the private and original feature i and noise on that feature. The covariance between the original and the noise is 0 as they are independent. Given, standard-deviation of a Normal noise is σ we can compute the scale parameter. Finally, privacy budget ϵ_i can be approximated by $\Delta F_i/\sigma$ where ΔF_i is the L1-sensitivity of the feature (estimated with the ratio of the variance between the private and original feature). A median value is taken for all the ϵ_i over all the features to obtain the final proxy value of the privacy budget ϵ in our DbP mechanism.

- **Benchmarking :** In table 2 (in main paper) we benchmark DbP's utility against different SOTA models for three different privacy budgets on the ADULT dataset. The results are quoted from literature. Generally we see that the highest privacy for the SOTA models occur with the low privacy budget of 0.36 and for that we see the utility to be lowest (high JSD). To report the utility scores (JSD/ MLU) for DbP and make a fair comparison, first we have approximated a proxy-ϵ based on the assumption of pseudo-noise to match the ϵ values in literature. We observe that DbP retains better utility (low JSD and high MLU) for each ϵ values.

6 Conclusions

In this supplementary document first we detailed the generative models, their architecture and hyperparameter set-ups.

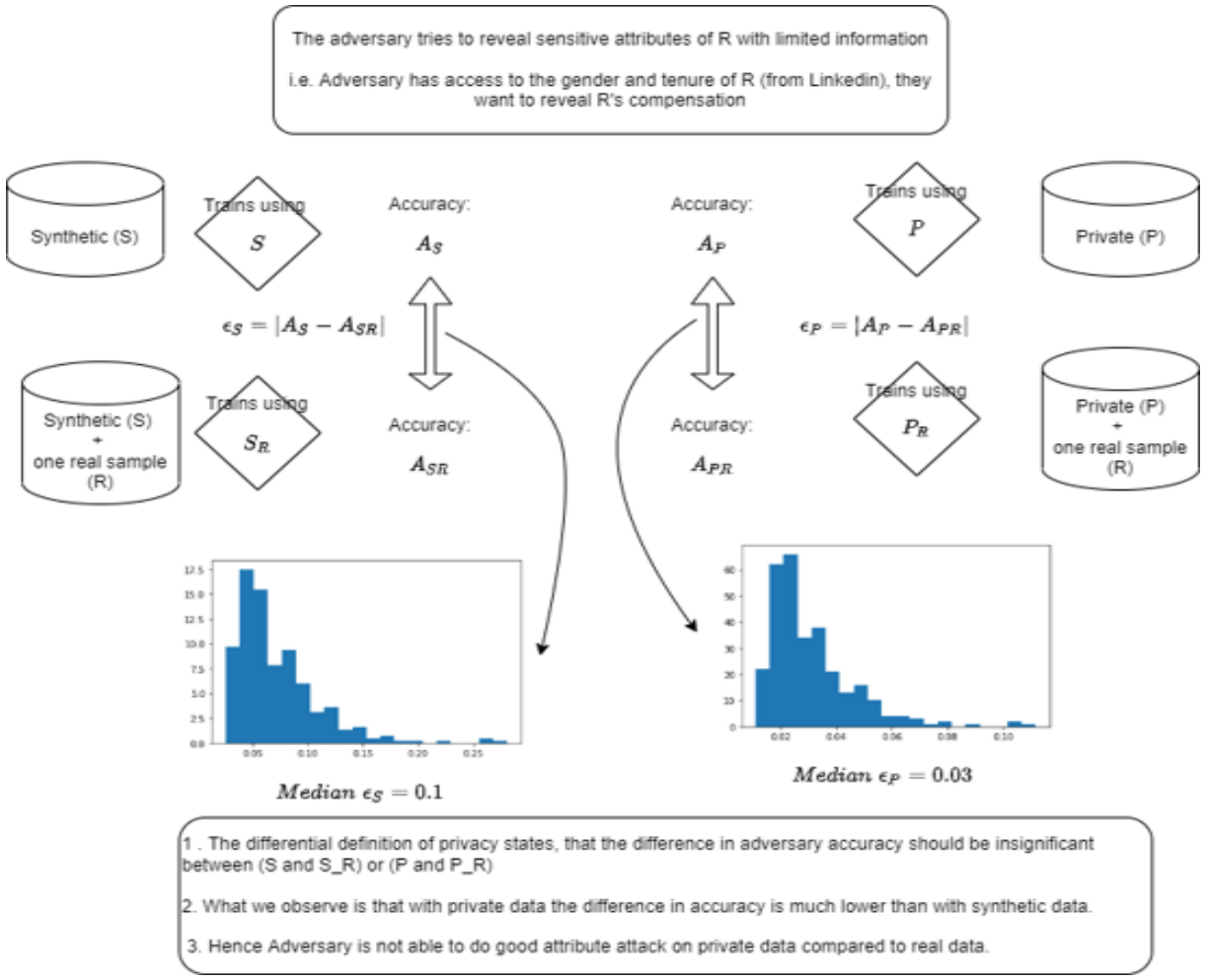

Figure 48: Schematic diagram for AIA

Next, we went over the datasets and their characteristics and sources. We also performed a detailed run-time analysis to showcase the speeds of the various generative models. Finally, we provided all the qualitative analysis to showcase the utility retained with the proposed DbP mechanism with different generators on different dataset. We added some implementation details as well. This supporting document is produced to provide empirical evidence in support of two statements we make in the main paper, (1). that the proposed mechanism is agnostic of the generative model and (2). that DbP can retain higher utility whilst providing strong privacy as DP.

References

[Johnson *et al.*, 2016] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-Wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

[Kunar *et al.*, 2021] Aditya Kunar, Robert Birke, Zilong Zhao, and Lydia Chen. Dtgan: Differential private training for tabular gans. *arXiv preprint arXiv:2107.02521*, 2021.

[Mendelevitch and Lesh, 2021] Ofer Mendelevitch and Michael D Lesh. Fidelity and privacy of synthetic medical data. *arXiv preprint arXiv:2101.08658*, 2021.

[Tantipongpipat *et al.*, 2021] Uthaipon Tao Tantipongpipat, Chris Waites, Digvijay Boob, Amaresh Ankit Siva, and Rachel Cummings. Differentially private synthetic mixed-type data generation for unsupervised learning. In *2021 12th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pages 1–9. IEEE, 2021.

[Xu *et al.*, 2019] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional gan. *arXiv preprint arXiv:1907.00503*, 2019.

[Yoon *et al.*, 2020] Jinsung Yoon, Lydia N Drumright, and Mihaela Van Der Schaar. Anonymization through data synthesis using generative adversarial networks (ads-

gan). *IEEE journal of biomedical and health informatics*, 24(8):2378–2388, 2020.

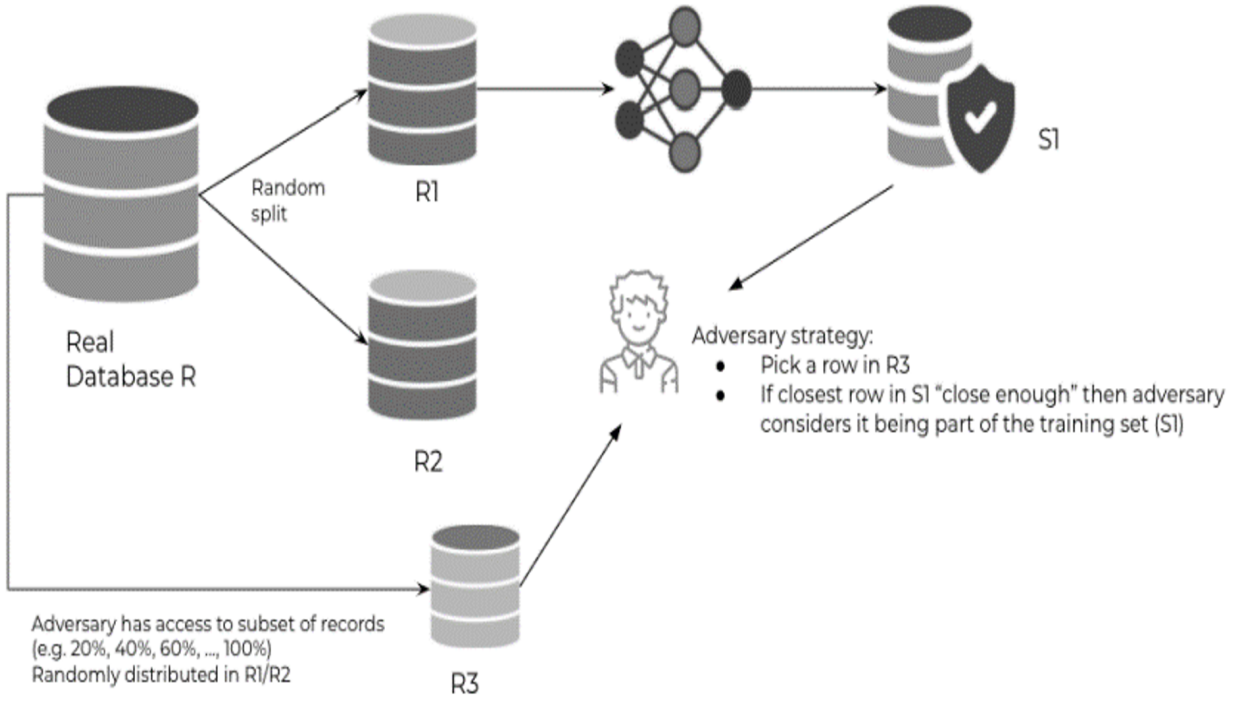

Figure 49: Schematic diagram for MIA