Differentially Private Data Cleaning and Synthetic Data Generation

Faezeh Ebrahimianghazani
Mathematics
University of Waterloo
Waterloo, Ontario
f5ebrahimianghazani@uwaterloo.ca

Bo Dong
Mathematics
University of Waterloo
Waterloo, Ontario
bo.dong@uwaterloo.ca

Amy Bhatia
Mathematics
University of Waterloo
Waterloo, Ontario
a54bhati@uwaterloo.ca

***Abstract*—Data privacy has gained attention over the years, and sharing data between owners and the public has always been challenging. The state-of-art solution has been differential privacy that offers quantitative measures against privacy leakage. A traditional solution to high dimensional dataset has been introducing a prohibitive amount of noise; recent solutions handle a type of dataset well, especially with the development of generative methods in machine learning algorithms. However, most of the algorithms assume the data is clean, which is unusual when data is obtained initially. This report will analyze the effect of data cleaning methods on differential privacy, show its sensitivity analysis, and present both theoretically and empirically that deletion leaks the least amount of information when used as a cleaning method before synthesizing using PrivBayes.**

***Index Terms*—differential privacy, data synthesis, data cleaning**

I. Introduction

With increased connectivity and access to IoT devices, more and more data is collected and shared for research and commercial activities. The data collected from various devices are used for knowledge discovery, which has countless applications and often serves a great value. Since most of the data sources contain users' sensitive information and their behaviour records, it is important to pre-process the datasets before distributing them to ensure privacy requirements. User privacy, if breached, can lead to legal obligations for the company working with the data. Multiple methodologies that utilize differential privacy [3] approaches, which guarantee that changing a record in a database has no effect on the output value of a query, have been proposed through various research topics.

As an example, when a user uses an e-commerce app to purchase products, his or her account and demographics information, interaction with the application such as the duration of the stay, whether items are purchased, will be recorded and shared with the mobile platform that has access to it. When a hospital research department wants to know the effectiveness of a particular drug on a targeted disease, the staff from the department needs to collect data from trials that involve patients' personal information. Information similar to the examples mentioned above can be used for their designed use cases; it can deliver general patterns when used in conjunction with data collected from other platforms. Therefore, direct sharing of information would pose privacy leakage, and at times prone to inappropriate use. One of the methods to reduce the privacy leakage from shared data is to perform data synthesis that keeps statistical features close to the original dataset while masking any private information. Many algorithms have been proposed for generating synthetic relational data - *PrivBayes* [8] have been focusing on generating high dimensional private data using BayesNet; *PATE-GAN* [7] modifies the discriminator in a GAN to achieve a privacy bound; and *DP-AuGM/ DP-VaeGM* [2] uses autoencoder and variational autoencoders to generate differential private synthetic data.

However, one of the common assumptions used in these data synthesis algorithms is that the dataset is complete and has no exceptional records. These assumptions are convenient and propose friendly mathematical properties; however, datasets rarely possess these properties in real life. In general, most collected data has inconsistencies like missing, improper values, or values of the same meanings but different spellings. Therefore, data cleaning is an essential pre-processing task even for data synthesis. In data analytics, data cleaning takes up a large portion of time in total work performed. Consequently, before synthesizing data, obtaining a clean dataset from the raw data must meet the assumptions.

In this project, we aim to test the effectiveness of data cleaning on synthesized data using adult income datasets, remove records with question marks in any attributes to obtain a clean dataset as ground truth. We will remove a percentage of records in selected attributes to simulate the amount of missing data and perform various cleaning algorithms to verify their differential privacy properties and find how data cleaning could pose challenges to privacy. In the end, we will also compare the synthetic data directly generated using raw data to that with ground truth and analyze its privacy-related properties and utility.

II. Related Work

A. Traditional Data Cleaning Methods

Almost all of the raw datasets are considered dirty and need cleaning. Some of the records might violate validity constraints like a data-type constraint, range constraint, mandatory fields, regular expression patterns, and more in unclean datasets. On the other hand, some values may be far from the actual value,

where others are missed. In this project, one of the focuses is on missing values since each of the detected bad values can be removed and replaced with a missing value. When encountering missing values, one approach is to drop rows containing missing values if their number is negligible. If there is a considerable number of missing values in a column, a simple way is to drop that column. The second approach is to estimate the missing values based on other existing values. There are several techniques to impute the missing values. One can use statistical values like mean and median, but this makes the dataset biased, especially if the number of missing values is high. Another technique is to use a linear or non-linear regression, but it is sensitive to outliers. Another approach is to copy other existing values into missing values; this can be done randomly or by finding the closest records or clustering data. [4]

B. Privacy and Data Cleaning

Despite the popularity of commonly used data cleaning methods, none of them provides privacy. There are a few works that study the problem of data cleaning under privacy. Talukder et al. use a cryptographic protocol to detect constraint violations in a database. Jaganathan and Wright [5] use similar cryptographic protocols for secure data imputation. However, the studied methods do not address privacy in a quantifiable measure, such as differential privacy. Krishnan et al. [6] introduce PrivateClean for data cleaning and approximate query processing with local differential privacy guarantees. They used a generalized random response and Laplace noise for categorical and continuous variables to achieve the desired effect.

C. Privacy and Data Synthesis

Many algorithms have been proposed to generate synthetic data in differentially private means. PATE-GAN [7] produce the most promising results on image-based and credit datasets. In a PATE-GAN model, the discriminator of a GAN network is replaced with a PATE model. The entire setup consists of a generator that takes in random noise as input and generates and outputs synthetic data points. The discriminator consists of k classifiers called 'teachers' that take in the generator's data or their portion of the original dataset in each iteration. The input feature vector is fed to the teachers. Each teacher model outputs the probability of its input being real, i.e. belonging to the original dataset. The outputs from all the teacher models are aggregated, and noise is added. The process preserves the differential privacy of the data. The student classifier, which follows the teachers, takes in the sample generated by the generator, which is labelled using the noisily aggregated outputs from the teacher discriminators and is trained to classify it the same way as the teachers. The generator iterations are performed until the privacy budget is available. Overall, the teachers are trained to improve their loss with respect to the generator. The generator is trained to strengthen its loss regarding the student. The student is trained to improve its loss for the generator.

We have evaluated the PATE-GAN [7] on the Adult Income dataset; however, due to the dataset composes of categorical variables and real variables that behave like discrete variables, the result from PATE-GAN algorithms is nearly unusable.

Since our project aims to study the impact of data cleaning on the private data synthesis method's output, we investigated several private data synthesis methods. Private data synthesis methods [1] protect the individuals' privacy while releasing a synthesized dataset similar to the original one used by analysts. Differentially private data synthesis methods can be divided into two categories: Non-parametric approaches and parametric approaches. Non-parametric approaches usually identify highly correlated attributes using a public data set and then generate marginals and use perturbed marginals to generate data points. Parametric approaches usually use models like Bayesian networks, graphical models, or generative adversarial models to generate data. We chose to continue our project with a *PrivBayes* [8] parametric model that uses Bayesian Networks to synthesize data and provides high privacy promises.

III. Methodology

We aim to preserve the privacy of users for data analysis tasks. We propose to set up a synthetic data generator that will yield data with similar statistical properties to the original data. We aim to test the effectiveness of the synthetic data generator on an unclean dataset named Adult Census. The team first cleans the original dataset to make a clean dataset the ground truth and then applies two types of data cleaning techniques on the artificially created unclean dataset. Finally, the team uses the synthetic data generator algorithm on the dataset cleaned by selected methods using the previously mentioned artificial unclean dataset. We aim to compare the results in both the cases stated. Ultimately, we intend to conclude on the preferred options considering the privacy budget and overhead parameters.

Many algorithms have been proposed to generate synthetic data in a differentially private manner. *PrivBayes* [8] is the algorithm on which we have focused our analysis. It constructs a low-dimensional Bayesian Network to model the relations between different attributes in the given dataset under study. Using the network, conditional distributions between the attributes are created and added with noise. Using the noisy marginals, approximate data distribution is constructed and sampled to create a synthetic dataset.

IV. Project Setup

A trusted data provider wants to share a dataset with an untrusted analyst so that the analyst can study a relationship R between the dataset's attribute values. This trusted provider performs two steps to ensure that the privacy of the original data is preserved - first, he applies a data cleaning method to make sure that the dataset is ready for the next step (this step is not necessarily private). He then generates a new dataset using private data synthesis methods and shares this synthetic data with the analyst. The dataset can include numerical-valued or discrete-valued attributes.

We assume that the analyst can perform any function on the synthesized dataset. Without the data cleaning step, we know that the process preserves individuals' privacy due to the private data synthesis model's differential privacy guarantees. However, in the presence of the data cleaning step, we suspect that the analyst can reveal information relations related to sensitive information in the original dataset even without access to the original. We will study this hypothesis by designing some experiments to see: 1) how the data cleaning step affects the quality of the output dataset and 2) whether the cleaning process decreases the privacy guarantee of the protocol.

V. Privacy Definition

We consider the well-known ε-differential privacy as our privacy definition.

Definition. *Differential Privacy. For $\varepsilon > 0$, an algorithm $\mathbf{M}$ is ε-differentially private if for any pair of neighboring datasets $\mathbf{D}$, $\mathbf{D}^{'}$ and any subset $\mathbf{S} \subseteq Range(\mathbf{M})$,*

$$Pr[\mathbf{M}(\mathbf{D}) \in \mathbf{S}] \leq e^{\varepsilon} Pr[\mathbf{M}(\mathbf{D}') \in \mathbf{S}].$$

The only difference between our work and the previous works in terms of differential privacy, is that we consider the cleaning process a part of $\mathbf{M}$. This means that the output of the process should be close for two neighboring raw datasets rather than two neighboring cleaned datasets.

Definition. *Neighbouring datasets. We call two potentially unclean datasets $\mathbf{D}$ and $\mathbf{D}^{'}$ neighbouring, if they differ in only one record. This difference can be an additional row in one of the datasets or a record that has different values for one or more than one attributes in $\mathbf{D}$ and $\mathbf{D}^{'}$. Other rows should be exactly the same. This means that even the missing values should be possessed in common. If any attribute value is missing for record c in $\mathbf{D}$, it should also be missing in the $\mathbf{D}^{'}$.*

VI. Privacy Analysis of Different Cleaning Methods

In this section, we will discuss the privacy analysis of the following cleaning methods:

1) Filling with Statistical Values: Replacing the missed values with a statistical value (mean, median, and mode) based on the other available values in that column
2) Deletion: Removing the rows that contain missed values

As we stated before, the cleaning methods that we discuss are suitable for datasets that have missed values. However, for other types of uncleanness like lack of validity, we can first remove the invalid data and then apply these methods, so the analysis would hold for datasets with invalid data too.

First, let us specify some notations that we will use in the rest of the paper:

Notation 1. *For an attribute $\mathbf{A}$, we show the value of that attribute in row k by A_k.*

Notation 2. *We use ϕ to show the missed values. We show the set of values that are not ϕ in column A by $\hat{S_A}$.*

Notation 3. *We show the size of the dataset by n and the number of rows with missed values with m. We also show the number of rows with missed values for a specific attribute $\mathbf{A}$ by m_A.*

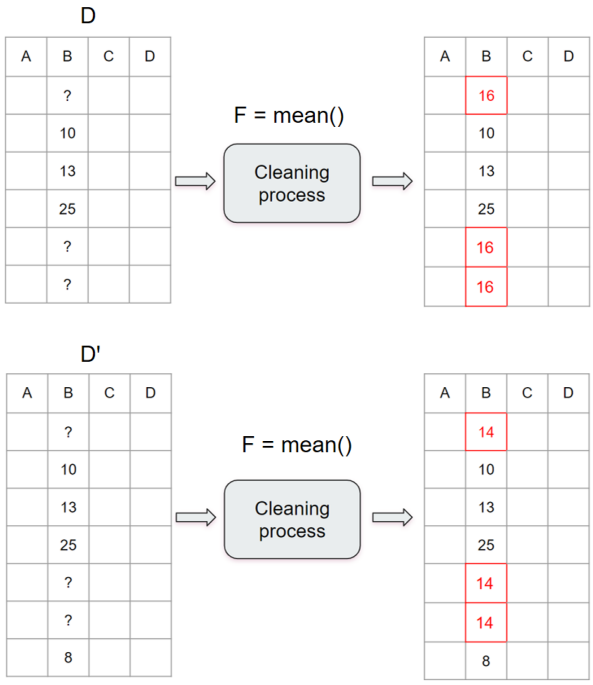

Fig. 1. An example to show that how m_A+1 rows of the output dataset can be changed for a neighbouring dataset that only has one additional row.(cleaning method $\mathbf{F} = \mathbf{mean}$)

A. Filling with Statistical Values

In this approach, if A_k is a missed value, we fill it with $A_k^{'} = \mathbf{F}(\hat{S_A})$. $\mathbf{F}$ is one of the mentioned statistical values.

$$\mathbf{F} \in \{mean, mode, median\}$$

Lemma 1. *If we assume that only one of the columns of the input dataset (column $\mathbf{A}$) contains missed values, the stability of the cleaning methods which fill in the missing values with mean/median/mode of available values is proportional to the number of missing rows (m_A) in that column, more accurately we have:*

$$\mathcal{S}_{stat} = m_A + 1.$$

Proof. Without loss of generality, we consider adding a new row to the input dataset and feed it to the cleaning method as a neighboring dataset. Since we assumed that only column A has missed values, the output of the cleaning method will not change for records that $A_k \neq \phi$. We also know that in this cleaning approach, all of the missing values are filled in with the same new value. Therefore, if that single value ($\mathbf{F}(\hat{S_A})$) changes in the new neighboring dataset, all of the rows containing missing values will change in the output of

the cleaning process. This means that if there is a possibility that $\mathbf{F}(\hat{S_A})_{D'} \neq \mathbf{F}(\hat{S_A})_D$, then the stability of the cleaning process is $m_A + 1$. Now we will show that for each statistical function it is possible that $\mathbf{F}(\hat{S_A})_{D'} \neq \mathbf{F}(\hat{S_A})_D$:

- **mean**: The only scenarios that $\mathbf{mean}(\hat{S_A})_{D'}$ can be equal to $\mathbf{mean}(\hat{S_A})_D$ are 1) when the value of column $\mathbf{A}$ for the added row (A_{new}) is equal to $\mathbf{mean}(\hat{S_A})_D$, or 2) when the added row has also missed value for column $\mathbf{A}$. Otherwise, we have:

  $$\mathbf{mean}(\hat{S_A})_{D'} = (n * \mathbf{mean}(\hat{S_A})_D + A_{new})/n + 1$$

  $$(n+1) * \mathbf{mean}(\hat{S_A})_{D'} = n * \mathbf{mean}(\hat{S_A})_D + A_{new}$$

  $$\mathbf{mean}(\hat{S_A})_D \neq A_{new} \rightarrow \mathbf{mean}(\hat{S_A})_{D'} \neq \mathbf{mean}(\hat{S_A})_D$$

  Thus, it is possible that $\mathbf{mean}(\hat{S_A})_{D'} \neq \mathbf{mean}(\hat{S_A})_D$ and $m_A + 1$ rows in the output dataset are modified. You can find an example in Figure 1.
- **median**: Without loss of generality, we assume that n is odd, and if we sort the $\hat{S_A}$, we call the $\frac{n+1}{2}^{\mathbf{th}}$ element med and we show the element before med by pre_med and the element after med by $post\_med$. Now after adding the new row, one of the following cases can happen:
  - $A_{new} < pre\_med$: in this case if $pre\_med \neq med$ then $\mathbf{median}(\hat{S_A})_{D'} \neq \mathbf{median}(\hat{S_A})_D$.
  - $pre\_med \leq A_{new} \leq post\_med$: in this case if $A_{new} \neq med$ then $\mathbf{median}(\hat{S_A})_{D'} \neq \mathbf{median}(\hat{S_A})_D$.
  - $post\_med < A_{new}$: in this case if $post\_med \neq med$ then $\mathbf{median}(\hat{S_A})_{D'} \neq \mathbf{median}(\hat{S_A})_D$.
- **mode**: Consider a case that the most frequent value in $\hat{S_A}$, $mode$, has been repeated $f_{mode} = f$ times and the second most frequent value in $\hat{S_A}$, $\hat{mode}$, has been repeated $f_{\hat{mode}} = f - 1$ times. In this case, instead of adding a new row, we change one of the rows of $\mathbf{D}$ to create the neighboring dataset, $\mathbf{D}^{'}$. We only replace the value of column $\mathbf{A}$ for row k that $A_k = mode$ by $\hat{mode}$. Now, the number of times that $mode$ has been repeated, f_{mode}, is decreased by 1 ($f_{mode} = f - 1$) and frequency of $\hat{mode}$ is increased by 1 ($f_{\hat{mode}} = f$). Therefore, $\mathbf{mode}$ of $\mathbf{D}^{'}$ is not the same as $\mathbf{D}$ and we have $\mathbf{mode}(\hat{S_A})_{D'} \neq \mathbf{mode}(\hat{S_A})_D$.

□

Theorem 1. *The stability of the cleaning methods which fill in the missing values with mean/median/mode of available values is proportional to the number of missing values in the whole dataset (m), more accurately we have:*

$$\mathcal{S}_{stat} = m + 1.$$

Proof. As stated in Lemma 1, if we only apply statistical filling to one column (e.g. column $\mathbf{A}$), the stability of the process is $m_A + 1$. In general, we can have missing values in all columns. In the worst case scenario, we assume that each row has only one missing value. This is the worst case because otherwise, if we have for example only two missing values and they are in the same row, then by applying the cleaning method on both columns only one record will change in the output, but if they were in different rows two records would have been modified in the output. Considering the worst case, we have m rows that each of them are modified during applying the cleaning method on a column. Thus the stability in general is $m + 1$.

□

B. Deletion

In this approach, if A_k is a missed value, we remove the $k^{\mathbf{th}}$ row of the dataset.

Lemma 2. *If we assume that only one of the columns of the input dataset (column $\mathbf{A}$) contains missed values, the stability of the deletion method is 1.*

Proof. We consider all three types of neighboring datasets (adding a row, removing a row, and changing a row) and show that for each of them the maximum number of rows that can be changed in the output of deletion is equal to 1.

- adding a row: In this case $\mathbf{D}^{'}$ has one additional row compared to $\mathbf{D}$ and all of the other rows are exactly the same. If the added row has a missing value in column $\mathbf{A}$, then the output of deletion for both neighbouring datasets would be the same. The only case that we have different outputs is when the added row has not any missing values. In that case, the output of the process for $\mathbf{D}^{'}$ has an additional row and all of the other rows are exactly the same as the output for $\mathbf{D}$.
- deleting a row: In this case $\mathbf{D}^{'}$ has one row less than $\mathbf{D}$ and all of the other rows are exactly the same. Similar to the previous case, maximum number of rows that can be modified in the output dataset is 1.
- changing a row: In this case one of the rows in $\mathbf{D}^{'}$ is different from the corresponding row in $\mathbf{D}$ and all of the other rows are exactly the same. If the modified row in both $\mathbf{D}$ and $\mathbf{D}^{'}$ contains a missing value, then the output of the process is the same for both datasets because that row will be deleted every time. Otherwise, the output datasets for the neighboring datasets $\mathbf{D}$ and $\mathbf{D}^{'}$ will be different in exactly one row.

As you can see, the maximum number of rows that can be different in the output of deletion process for two neighboring datasets is equal to 1. You can find an example in Figure 2. □

Theorem 2. *The stability of the deletion method is 1.*

Proof. Similar to the proof of theorem 1. □

VII. Experiments and Results

We evaluated the quality of synthesized data using different cleaning processes in two ways. First, we compared the distribution of single attributes between synthesized data and the original data. And in another set of experiments, we evaluated the relationship between attributes using SVM-classifiers.

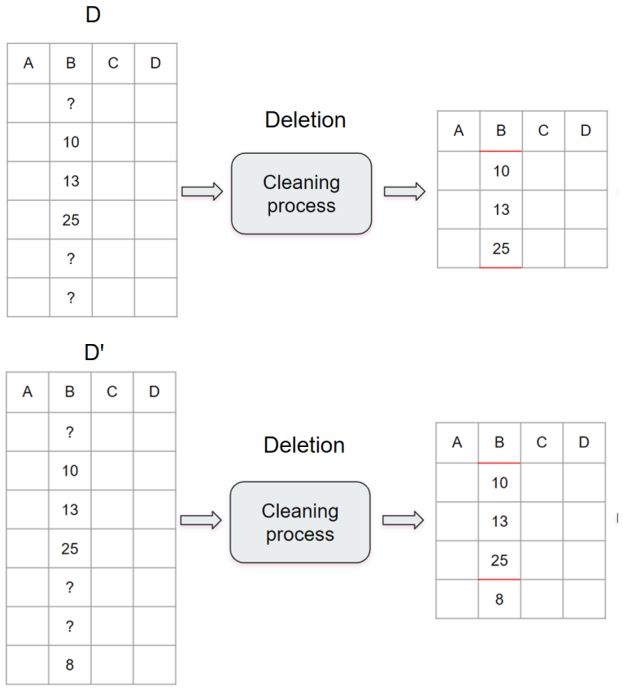

Fig. 2. An example to show the stability of deletion process.

A. *Distribution of Attributes*

In the experiment, we have used the Adult Income dataset as the source of unclean data and performed deletion of rows that contain question marks to remove any missing values. We treat the obtained dataset as ground truth or a clean dataset. After the clean version of the dataset is prepared, the team then applies a percentage of missing value on the selected attribute of the dataset to simulate unclean data from sources. We have chosen age and races as the attribute of removal to observe the effect of cleaning on data synthesis after deterministically filling the missing value using mean or median and deleting the records from the dataset completely.

We measure the divergence between the distribution of a selected attribute in the ground truth dataset and the synthesized dataset obtained after applying data cleaning. The divergence formula we chose was the KL divergence. It measures the "distance" between two distributions. The formula is as follows

$$D(P \parallel Q) = \sum_{x}(P(x)log(\frac{P(x)}{Q(x)}))$$

The divergence calculation shows higher value when the two distributions are less similar to each other.

In the experiment, we can see that the divergence measures steadily increase as the percentage of value in the selected attribute is made missing and replaced with mean or median calculated with the rest of the values, as shown in figure 3 and 4. The observation is straightforward to understand -

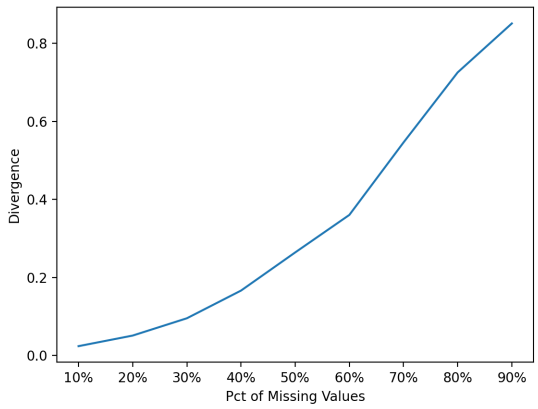

Fig. 3. Distribution divergence measurement on age attribute between ground truth and synthesized data after filling the missing values with mean given percentage of missing

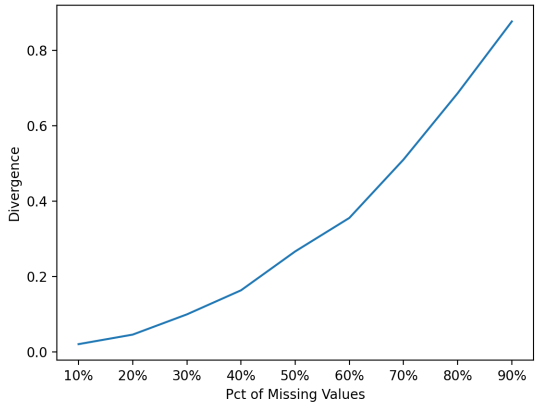

Fig. 4. Distribution divergence measurement on age attribute between ground truth and synthesized data after filling the missing values with median given percentage of missing

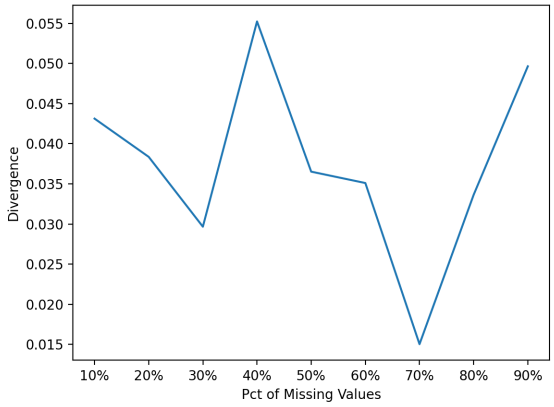

Fig. 5. Distribution divergence measurement on age attribute between ground truth and synthesized data after filling the missing values with median given percentage of missing

increasingly, the number of values is turned into a fixed value, hence disturbing the true distribution with a biased distribution towards mean and median. Therefore, when synthesizing on the filled dataset, the generated data will show similar trends to the cleaned dataset.

Shown in the figure 6, where the original distribution is shown on top left, the top right, bottom left and bottom right represents 30%, 60%, and 90% of missing value. The more missing value, the more concentrated the frequency is due to filling with a deterministic value. Although the bin that possesses the high frequency may change, the observation of greater distribution deviation from the original is true.

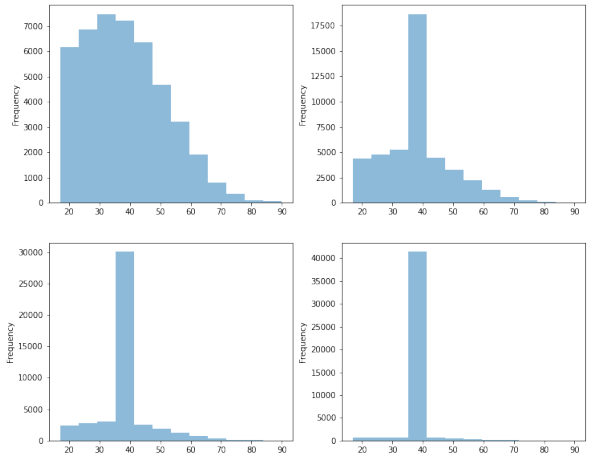

Fig. 6. Distribution of the dataset after filling the median value on missing records

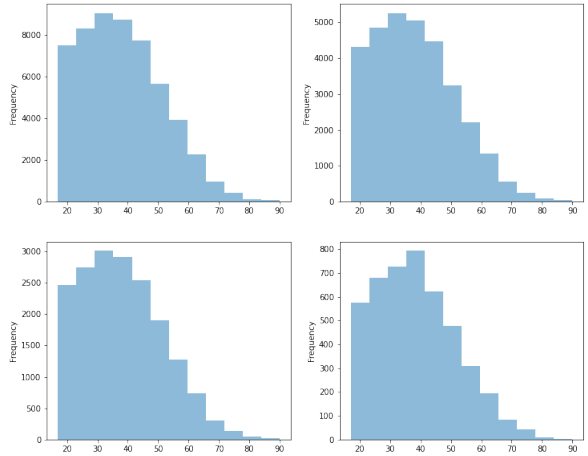

Fig. 7. Distribution of the dataset after performing deletion on missing records

As for the deletion, because the number of missing values are selected independently, the overall distribution of the attribute remains very similar to that of the original distribution before data synthesis, shown in figure 7. As shown in the y-axis, the divergence is smaller than 0.01 without dependence on the percentage of missing values deleted. This observation may not be the case in reality. For example, the number of age missing usually happens with females than males; hence deletion may disturb distribution.

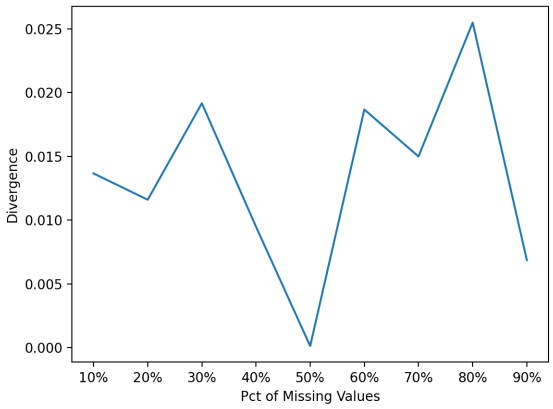

Fig. 8. Distribution divergence measurement on race attribute between ground truth and synthesized data after filling the missing values in age with mean given percentage of missing

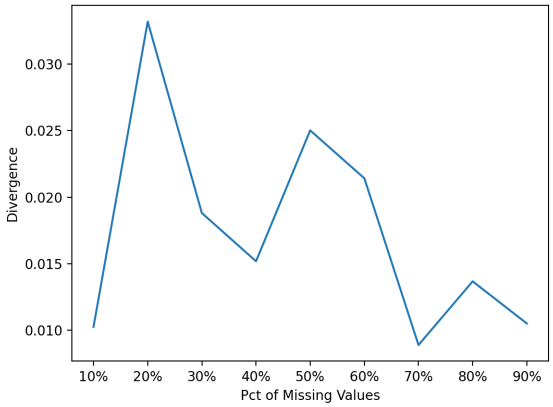

Fig. 9. Distribution divergence measurement on race attribute between ground truth and synthesized data after filling the missing values in age with median given percentage of missing

We have also compared the divergence of distribution of non-selected variables when the selected attribute is increasingly missing. The result is that most of the columns are unaffected due to the assumption of conditional independence, shown in figure 8 and 9. Later on, there are SVM evaluations that are used on the interrelations of attributes.

B. Attribute Correlations

We evaluated the quality of the synthesized data in maintaining the correlation between attributes by training several SVM classifiers. An SVM classifier predicts the value of one attribute (target attribute or $\mathbf{Y}$) using the other attributes. We borrowed this method for utility measurement from [8]. In this approach, four different SVM classifiers are trained simultaneously for four different target attributes:

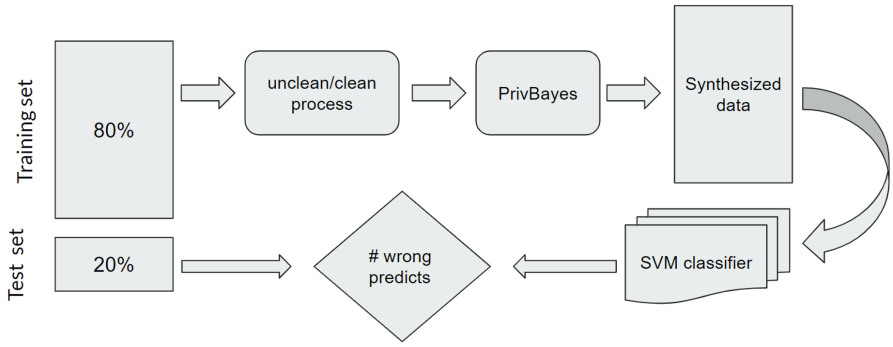

Fig. 10. Different steps to evaluate the correlation between attributes using SVM classifiers.

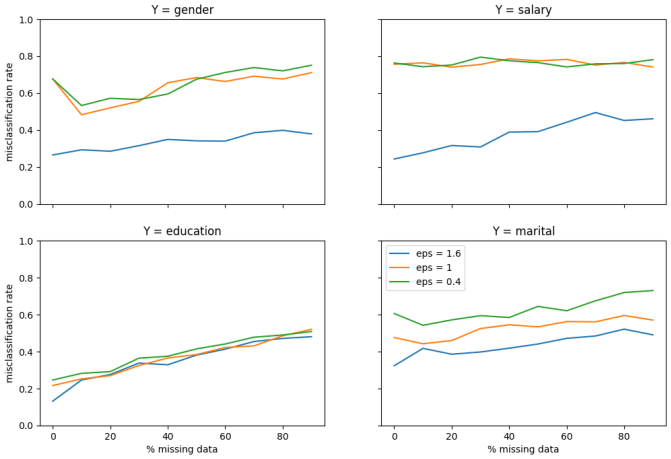

Fig. 11. $Misclassification\ rate$ for each SVM classifier when the cleaning method is filling with statistical values (mean)

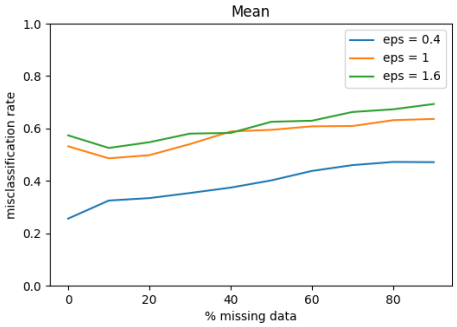

Fig. 12. Average $misclassification\ rate$ when the cleaning method is filling with statistical values (mean)

1) Gender: whether a person is male or female.
2) Income: whether a person makes over 50K a year.
3) Education: whether a person has post-secondary education.
4) Marital: whether a person has ever been married.

For each of the above classification tasks, we used 80% of the rows as the training set and 20% as a test set. We make the training set unclean by randomly replacing a percentage of values in Column "hours-per-week" as missed values and then apply a cleaning method (Mean or Deletion) to prepare the data for the next step. Now we apply PrivBayes on the cleaned training dataset to generate a synthetic dataset. Finally, we use the synthetic data to train four SVM classifiers. The quality of the synthetic data is measured based on the quality of the SVM classifiers that we introduced on them. We use $misclassification\ rate$ on the test dataset as a metric to measure the quality of classifiers. It means that we calculate the percentage of the records in the test dataset that are classified incorrectly by the SVM classifiers. Thus, for high-quality synthetic data, $misclassification\ rate$ should be small. You can find a flowchart of the process in Figure 10.

In our experiment set, we performed the above-mentioned tasks for different percentage of missing values, different privacy budgets (ε) used in the PrivBayes algorithm, and two cleaning methods: 1) filling with statistical value - mean, and 2) deletion.

Figure 11 shows the $misclassification\ rate$ of the four SVM classifiers using three different ε values for mean as the cleaning method. Figure 12 shows the average of $misclassification\ rates$ on four classifiers. The $misclassification\ rate$ generally increases by the number of missing values. We can say that for all ε values if we use the mean to fill the missing values, the percentage of missing values directly affects the synthesized data quality. Another observation is that the $misclassification\ rate$ for $\varepsilon = 1$ and $\varepsilon = 1.6$ are close together.

Figure 13 shows the $misclassification\ rate$ of the four SVM classifiers using three different ε values when the cleaning method is deletion. Figure 14 shows the average $misclassification\ rate$ on four classifiers. In this case, there is no meaningful relation between the $misclassification\ rate$ and the number of missing values.

VIII. Comparison between Different Cleaning Methods

This section will discuss each of the cleaning methods that we used in our project. Table I is a comparison table to conclude the section.

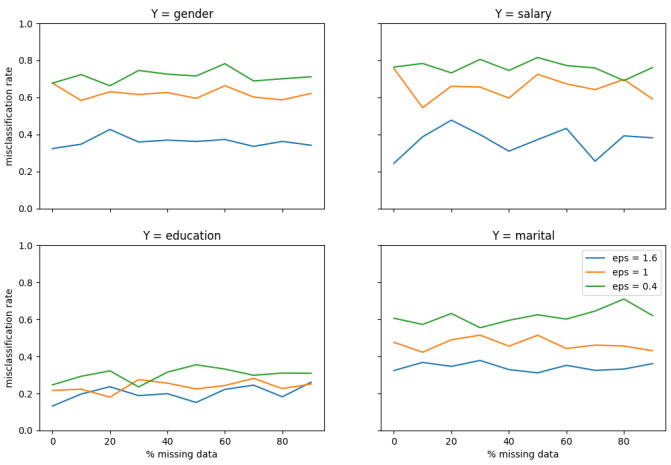

Fig. 13. $Misclassification\ rate$ for each SVM classifier when the cleaning method is deletion

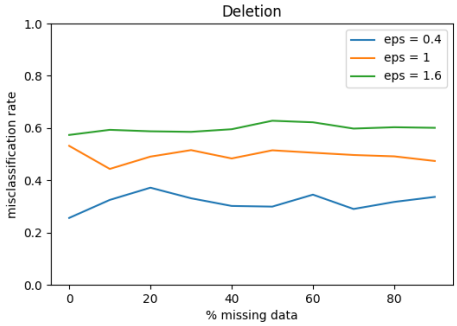

Fig. 14. Average $misclassification\ rate$ when the cleaning method is deletion

A. Filling with statistical values

In terms of applicability of these methods, mean and median only can be used on numerical data and are most preferred for continuous numerical data. On the other hand, mode can be applied to numerical and categorical data and is most preferred for discrete numerical or categorical data.

In terms of privacy cost, as stated in Section VI, the stability of these methods is a function of the number of missing values, m. This means that the amount of noise that should be added to preserve the privacy of individuals and make the whole process a ε-differentially private algorithm increases linearly by the number of missing values. Especially for the large datasets, even if only a small percentage of all records contain missing values, their number can be huge and lead to adding a large amount of noise, which can totally hide the primary value and make the output result useless.

As mentioned in Section VII, applying these kinds of cleaning methods on the unclean dataset leads to low-quality synthesized data according to attribute distribution and attribute correlation. The quality of data decreases with the percentage of missing values.

B. Deletion

There is no limit to the applicability of this method. One can easily remove rows containing missing values independent of the type of data in each column.

In terms of privacy loss, according to Section VI, the stability of deletion is one. The small value in stability measure means that we don't need to add additional noise to ensure privacy.

According to Section VII, the distribution of each column in the synthesized data change very little by removing the rows containing missing values compared to the original dataset. However, we made the unclean datasets by randomly replacing some of the records with missing values in a clean dataset. This way, removing the rows with missing values can only be considered a process in which we randomly selected a subset of a dataset. Therefore, it should not change the distribution of each column. The independence and the uniform assumption are not necessarily the case with real unclean datasets. For example, in a dataset that is collected based on a survey, it is possible that the number of women refusing to enter their age in the questionnaire be more than the number of men that do not enter the age. Thus by removing the rows containing missing values, we remove more entries from women than men, and the distribution of gender is modified.

IX. Conclusion

In conclusion, if the missing values in a dataset result from data corruption during aggregating of the data, we can assume that missing values are distributed randomly through the data, and deletion would not affect the quality of synthesized data in terms of attribute distribution.

X. Future work

Our current analysis focuses only on one category of uncleanliness, i.e. missing values in a dataset. To our understanding, this work can also be adapted to cases where uncleanness is in the form of invalid/garbage values. We plan to experiment with such different types of unclean datasets and try to apply some differentially private data cleaning algorithms before passing on the datasets to PrivBayes. Additionally, we aim to test PrivBayes by giving different parameters on the missing value columns.

Acknowledgment

Our team would like to thank Xi He for her generosity, help and guide during this project. Code of PrivBayes was also a shared work of her students.

References

[1] Claire McKay Bowen and Fang Liu. Comparative study of differentially private data synthesis methods. *Statistical Science*, 35(2), May 2020.

[2] Qingrong Chen, Chong Xiang, Minhui Xue, Bo Li, Nikita Borisov, Dali Kaafar, and Haojin Zhu. Differentially private data generative models. *CoRR*, abs/1812.02274, 2018.

[3] Cynthia Dwork. Differential privacy. In *Proceedings of the 33rd International Conference on Automata, Languages and Programming - Volume Part II*, ICALP'06, page 1–12, Berlin, Heidelberg, 2006. Springer-Verlag.

Cleaning Method	Applicable data types	Stability	Quality of the synthesized data (distribution)	Quality of the synthesized data (correlation)
Filling with statistical values (mean)	Continuous Numerical	m+1	Poor, diverging data when more missing	Poor, increased misclassification rate when more missing
Filling with statistical values (median)	Continuous Numerical	m+1	Poor, diverging data when more missing	Poor, increased misclassification rate when more missing
Filling with statistical values (mode)	Discrete Numerical/ Categorical	m+1	Poor, diverging data when more missing	Poor, increased misclassification rate when more missing
Deletion	All	1	Does not diverge, reasonable quality	misclassification rate does not increase by missing values

TABLE I
COMPARISON BETWEEN DIFFERENT CLEANING METHODS

[4] Omar Elgabry. The ultimate guide to data cleaning, Mar 2019.

[5] Geetha Jagannathan and Rebecca N. Wright. Privacy-preserving imputation of missing data. *Data Knowl. Eng.*, 65(1):40–56, April 2008.

[6] Sanjay Krishnan, Jiannan Wang, Michael J. Franklin, Ken Goldberg, and Tim Kraska. Privateclean: Data cleaning and differential privacy. In *Proceedings of the 2016 International Conference on Management of Data*, SIGMOD '16, page 937–951, New York, NY, USA, 2016. Association for Computing Machinery.

[7] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. PATE-GAN: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*, 2019.

[8] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. Privbayes: Private data release via bayesian networks. *ACM Trans. Database Syst.*, 42(4), October 2017.