Differentially Private Verification of Predictions from Synthetic Data

by

Haoyang Yu

Program in Statistical and Economic Modeling
Duke University

Date: ____________________________

Approved:

__________________________
Jerome P Reiter, Supervisor

__________________________
Rebecca Steorts

__________________________
V. Joseph Hotz

Dissertation submitted in partial fulfillment of the requirements for the degree of Master of Science in the Program in Statistical and Economic Modeling in the Graduate School of Duke University

2017
Abstract

Differentially Private Verification of Predictions from Synthetic Data

by

Haoyang Yu

Program in Statistical and Economic Modeling
Duke University

Date: ______________________
Approved:

______________________________
Jerome P Reiter, Supervisor

______________________________
Rebecca Steorts

______________________________
V. Joseph Hotz

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Master of Science in the Program in Statistical and Economic Modeling in the Graduate School of Duke University 2017
Copyright © 2017 by Haoyang Yu
All rights reserved except the rights granted by the
Creative Commons Attribution-Noncommercial Licence
Differentially Private Verification of Predictions from Synthetic Data

by

Haoyang Yu

A thesis submitted in partial fulfillment for the degree of Master of Science in Statistical and Economic Modeling

December 2017
When data are confidential, one approach for releasing public use files is to make synthetic data, i.e., data simulated from statistical models estimated on the confidential data. Given access only to synthetic data, users have no way to tell if the synthetic data preserve the adequacy of their analysis. In this thesis, I present methods that can help users to make such assessments automatically while controlling subjects’ privacy. There are three verification methods presented in this thesis: differentially private prediction tolerance intervals, differentially private prediction histograms, and differentially private Kolmogorov-Smirnov tests. I use simulation to illustrate these prediction verification methods.
Acknowledgements

I would like to thank Professor Jerry Reiter, my supervisor at Duke University, for the encouragement and advice he has provided. I still remember the words he said to me at the stairs in front of the Old Chemistry building at the moment when I lost my confidence and started doubting my choice of transferring to the Statistical Science department. It was his words that made me believe in myself and keep on trying and challenging myself for the days following. I have been extremely lucky to have a supervisor who cared so much about my work, and who responded to my questions and queries so promptly.

I would also like to thank Andres Felipe Barrientos for supporting my work on the thesis. Further, I would like to thank Professor Joe Hotz. He introduced me to Econometrics, the world of frequentists. With his help, I got the chance to work with Professor Jerry Reiter. I would also like to thank Professor Rebecca Steorts, who introduced me to the world of Bayesian statistics and helped me make up my mind to fully transfer to the Statistical Science department.

And This work was support by the grant NSF SES 11-31897.
# Contents

Abstract

i

Acknowledgements

ii

1 Introduction

1

2 Review of Differential Privacy

5

3 Verification Methods for Predictions

7

3.1 DP Prediction Tolerance Intervals ................................ 8

3.2 DP Prediction Histograms ............................................ 9

3.3 DP Kolmogorov-Smirnov Test .................................... 10

4 Empirical Illustrations

13

4.1 Same Generators ..................................................... 14

4.1.1 Same Linear Model for Data Generation and Perfect Prediction Model ........................................... 15

4.1.2 Same Linear Model Generators and Imperfect Prediction Model ................................................. 16

4.1.3 Same Quadratic Model Generators and Imperfect Prediction Model .............................................. 17

4.1.4 Summary ............................................................ 18

4.2 Different Generators ................................................ 19

4.2.1 Quadratic Authentic Generator and Linear Synthetic Generator ................................................... 19

4.2.2 Linear Authentic Generator and Quadratic Synthetic Generator ................................................. 21

4.2.3 Summary ............................................................ 22

5 Conclusion

24

Bibliography

26
Chapter 1

Introduction

Nowadays, data producers seek to provide broader public access to data on individual records. They also have the obligation to protect the confidentiality of data subjects identities and sensitive attributes. Data Producers take steps to limit how analysts can access the data to protect the confidentiality.

Data Producers may suppress and anonymize datasets by removing obvious identifiers such as names and addresses. However, generally these steps are not sufficient to protect confidentiality [1–4]. Intruders may still be able to link the released data with records in external non-anonymized databases to get personal sensitive attributes, which may result in the indirect release of the individuals’ information.

For example, differential privacy has been developed to ensure that the presence or absence of any one record does not significantly change the output[5]. It can neutralize the linkage attack and decrease the risk of information leakage because the property of its data access mechanism. Differential privacy is usually described as a paradox where researchers learn useful information about a population without learning anything about an individual[6]. It guarantees that the conclusion given by the study has nothing to do with one individuals presence or absence in the data. Differential privacy promises to protect the individual from information leakage that they may face when their data is in a private dataset. However, differential privacy does not protect group privacy[5].

One differentially private approach is to use remote query control systems [7], in which users can submit queries to servers that hold the true datasets. The remote servers run
the queries and return the analysis results to the user without giving the user access to the underlying micro-data. However, it can be risky to directly return the queries results, since intruder may learn individual’s information by changing details of queries. To further reduce the disclosure risks, the analysis results can be perturbed to satisfy differential privacy.

There are plenty of ways that are developed to do the differentially private regression diagnostics\[8–14\] based on this approach. Another approach is to release full synthetic datasets \[15–18\]. Synthetic dataset can be generated through predictive distribution estimated with the confidential dataset, which can be called synthetic builders \[19\]. Fully synthetic datasets are generated by sampling new values from predictive distributions for every confidential value. Approach for generating multiply-imputed, partially synthetic datasets handling disclosure limitation and missing data simultaneously is developed \[20\]. Winkler \[21\] applied EM algorithms to estimate the weights subject to convex constraints when generating synthetic data. There are other imputation strategies developed to generate the synthetic datasets.

However, there are drawbacks to the two approaches. For the first approach, users can only submit queries on regression models, and the returned results are limited to the relevant differentially private diagnostics algorithms. For the second approach, even though users can do analysis on synthetic dataset, they have no way to tell the adequacy of their analysis.

The previous DP t-test on regression coefficients algorithm \[22\] is then developed to help users to compare the signs and significance levels of their privately-computed regression coefficients computed from the synthetic dataset with the ones calculated on the confidential dataset. It is designed to partition the dataset into M disjoint bins, estimate the regressions using data in each bin separately to get the t-statistics, truncate the t-statistics based on the threshold given by users, average the statistics, and add noise to the results to satisfy the differentially private criterion. However, this t-test algorithm does not tell the adequacy of user-computed analysis on the synthetic data such as the accuracy of their prediction models as a whole instead of the individual
coefficient’s comparison. Thus, we may need new algorithms to tell users the accuracy of their privately-computed prediction results from synthetic data.

In this thesis, I establish some new prediction verification algorithms helping users test the accuracy of their privately-computed prediction results from the synthetic data. Users can send queries to the remote server, along with their regression models and the privately-computed regression results from the synthetic datasets. Then the remote verification server will calculate the predicted mean values $\mu_i$ for each data point $x_i$ based on the prediction models from the synthetic data given by users, by plugging in data points from authentic dataset. Based on the predicted values, the remote server will then return users with feedback of the three verification algorithms.

In the first algorithm, users can choose the tolerance interval size. For each data point $y_i$, the remote servers can compute the number of the tolerance intervals that contain the true data points, denote as $S(D)$. Laplace noise is added to $S(D)$ to satisfy differential privacy. The remote server will return to the user the percentage of intervals that contain the true data points, which is differentially private. This method is built on needs of the user to choose the tolerance levels based on their empirical model assumption or other information on the analysis. It can give the user a sense of the accuracy of their prediction results on the synthetic dataset.

In the second algorithm, we assume the response data points $i$ are centered at the prediction value $\hat{\mu}_i$ with the standard deviation $\hat{\sigma}$ estimated by the user. The remote server calculates the cumulative distribution function (CDF) for each data points. If the standard errors calculated in the synthetic datasets are similar to the authentic datasets, the model is close to the true underlying model, and the errors are normally distributed, then the frequency histogram of the CDF presented should be flat. Laplace noise is added to each bar in the histogram, which is then differentially private. This method gives the user a direct visual understanding of the accuracy of their privately-computed prediction results. We call this the differentially private prediction histogram algorithm.

The third algorithm uses the Kolmogorov-Smirnov(KS) statistics to quantify the distance between the empirical distribution of the predicted values for the response data
points and the empirical distribution of the true response data points. We add Laplace noise to the KS test to satisfy the differential privacy criterion.

The remaining part of this thesis is organized as follows: In Section 2, we review differential privacy and some techniques used to design differentially private algorithms. In Section 3, we present the three differentially private prediction verification algorithms. In Section 4, we illustrate the performance of the algorithms with simulations of different settings with a goal of giving users a general idea how these algorithms work. In Section 5, we conclude with suggestions for implementation of these techniques, and discuss future research on other verification methods when complex models are applied.
Chapter 2

Review of Differential Privacy

Differential privacy is developed to ensure that the presence or absence of any one record does not significantly change the output. Its goal is that researchers learn useful information about a population without learning anything about any individual.[5]

Let $D$ be the confidential dataset comprising individuals. For each $i$th individual, the record is composed of a $p \times 1$ vector of explanatory variables, $x_i = (x_{i1}, x_{i2}, ..., x_{ip})$, and a response variable $y_i$. We define a neighborhood relation $Q$ on datasets. $D$ and $D'$ are considered neighboring datasets ($D, D' \in Q$), since they only differ in one row such that there exists a $j$th row, where $D$ has $(x_j, y_j)$, but $D'$ has $(x'_j, y'_j)$. Let $A$ be the algorithm that takes $D$ as the input. $A$ is said to be differentially private if its output are statistically similar on neighboring datasets [6]. The following definition is about the $\epsilon$-differential privacy.

**Definition 1** ($\epsilon$-differential privacy) An algorithm $A$ satisfies $\epsilon$-differential privacy if for any pair of neighboring datasets ($D, D' \in Q$), and any output $S \in range(A)$, $\Pr(A(D) = S) \leq e^\epsilon \cdot \Pr[A(D') = S]$. Differentially private algorithms also satisfy the following composition properties [6].

The $\epsilon$, known as the privacy budget, controls the degree of privacy given by Algorithm $A$, and limits how the intruders can distinguish one dataset from another neighboring dataset given the output. The smaller the epsilon, the more privacy is protected. Differential Privacy is known to satisfy two key theorems, which we state here:
**Theorem 1** (Sequential Composition) Suppose $A_1(D)$ and $A_2(D)$ be $\epsilon_1$- and $\epsilon_2$-differentially private algorithms separately: If releasing the outputs of $A_1(D)$ and $A_2(D)$, the outputs satisfies $(\epsilon_1 + \epsilon_2)$-differentially private.

**Theorem 2** (Post-processing Composition) Suppose $A_1(D)$ be $\epsilon_1$-differentially private algorithms for any algorithm $A_3(\cdot)$, releasing $A_3(A_1(D))$ still satisfies $\epsilon_1$-differentially private.

The global sensitivity of a function $f$ plays an important role in the design of many algorithms. It captures the maximum magnitude change in $f$ by changing only one row in dataset.

**Definition 2** (Global Sensitivity) The global sensitivity of a function $f: D \rightarrow \mathbb{R}^n$, denoted $\mathcal{G}_S(f)$, is defined to be the maximum $L_1$ distance of the output from two neighboring datasets $D$ and $D'$, $\mathcal{G}_S(f) = \max_{(D,D')} \|f(D) - f(D')\|_1$

The Laplace mechanism is designed based on the global sensitivity regarding the target function. And the Laplace mechanism satisfies the differential privacy.

**Definition 3** (Laplace Mechanism (LM)), For any function $f: D \rightarrow \mathbb{R}^n$, $A(D) = f(D) + \delta$, $\delta$ is a vector of independent random variables drawn from a Laplace distribution with the probability density function $p(\mathcal{G}_S(f) | \epsilon) = \frac{1}{2\epsilon} e^{-|\mathcal{G}_S(f)|/\epsilon}$

We now preview some examples using Laplace Mechanism: In some counting queries, such as calculating the total number of elements that have property $\mathcal{P}$, sometimes in fractional form or with weights, we then add Laplace noise drawn from Lap$(1/\epsilon)$. In some histogram queries, we partition the results into cells, and the queries ask how many elements contained in each cell. Because cells are disjoint, adding or removing only one element may affect only one cell, so the sensitivity is bounded by 1. Then the results can be returned by adding independent draws from Lap$(1/\epsilon)$ to the actual count for each cell. There are other differentially private mechanisms such as exponential mechanisms[23] and iterative construction mechanism[24].
Chapter 3

Verification Methods for Predictions

Three differentially private verification methods are proposed: differentially private prediction tolerance intervals, differentially private prediction histograms, and differentially private Kolmogorov-Smirnov test. These three algorithms are all established to test the accuracy of the privately-computed prediction results without re-running the regressions on the authentic dataset or doing any partitions, which is computationally easier compared with other verification methods.

The differentially private prediction tolerance intervals algorithm provides the user with the freedom to choose the size of the prediction tolerance intervals either based on the prediction tolerance level or other information. The differentially private prediction histogram algorithm provides the user with a visual understanding of how their privately-computed results may work on the authentic dataset. As for the differentially private Kolmogorov-Smirnov test, it directly quantifies the maximum distance between the cumulative distribution function of the authentic data points and the cumulative distribution function of the predicted values. We now describe each algorithm.
3.1 DP Prediction Tolerance Intervals

In our settings, we present the methods in the context of ordinary regression models. Users can submit their queries based on their needs, along with their regression models and the privately-computed regression results, \((\tilde{\alpha}, \tilde{\beta}_1, \ldots, \tilde{\beta}_p, \tilde{\sigma})\), from the synthetic data. Then the remote verification server will calculate the predicted mean values \(\tilde{\mu}_i\) for each data point \(x_i\) from the authentic dataset and user-computed coefficients from synthetic data based on the prediction model.

Users can choose the prediction tolerance intervals \([l_i, u_i]\). For example, they may want to check the prediction tolerance intervals with a fixed width \(2\Delta\). Then, the predicted tolerance intervals would be \([\tilde{\mu}_i - \Delta, \tilde{\mu}_i + \Delta]\). Or they could choose the intervals with lower and upper bounds proportional to the predicted mean values. Then the predicted tolerance intervals would then be \([a\tilde{\mu}_i, b\tilde{\mu}_i]\), for some constants \(a\) and \(b\).

As an example of additive \(\Delta\), we can use the \((1 - \alpha) \cdot 100\%\) prediction intervals for each data point \(y_i\) in the authentic dataset. By setting \(\Delta = t_{1-\alpha/2}SE(y_i)\), where \(SE(y_i) = \sqrt{\hat{\sigma}^2(1 + x_i^T(X^TX)^{-1}x_i)}\), we use Student t distribution to approximate the 95% prediction tolerance intervals for each data point \(y_i\) in the authentic dataset, where \(X\) refers to the regression data matrix from the synthetic data. We obtain the 95% prediction intervals \([l_i, u_i] = [\mu_i - \Delta, \mu_i + \Delta]\).

The remote servers can count how many predicted intervals \([l_i, u_i]\) contain the true response values \(y_i\), denoted as \(S(D)\). The global sensitivity of \(S(D)\) is 1. By removing or changing only one row (the \(j\)th row) in dataset \(D\), the change will be the tolerance interval \([\tilde{\mu}_j' - \Delta, \tilde{\mu}_j' + \Delta]\), which may or may not contain the corresponding response data point \(y_j'\). By adding Laplace noise to \(S(D)\), we get \(S^*(D)\) which is equal to the round value of \((S(D) + \delta)\), where \(\delta\) is an independent random variable drawn from a Laplace distribution with the probability density function \(p(G_S(S(D))|\varepsilon) = \frac{1}{2\varepsilon}e^{-1/\varepsilon}\).

The percentage of the tolerance intervals containing the authentic response data points is denoted as \(P(D)\), which equals \(S^*(D)/N\), where \(N\) is the total numbers of the rows in dataset \(D\). It is the statistic the remote verification server returns to the user. According to the postprocessing composition property, \(P(D)\) satisfies \(\varepsilon\)-differential privacy.
This method informs the user of the differences that may exist between the synthetic dataset and the authentic dataset. If the authentic dataset and synthetic dataset have similar information regarding to the response variable and explanatory variables, and the user are using the $(1 - \alpha) \cdot 100\%$ prediction intervals, then $\mathcal{P}(D)$ should be in the proximity of $(1 - \alpha) \cdot 100\%$. Most importantly, this method is built on the needs of the user, since different research topics may require varying levels of tolerance. If the user choose a smaller $\Delta$, which implies lower tolerance level, but reported $\mathcal{P}(D)$ is relatively high, then we can say the prediction models given by the user have sufficient accuracy. On the contrary, if the user chooses a higher $\Delta$, a higher tolerance level, but reported $\mathcal{P}(D)$ is relatively small, then we can say the prediction models given by the user have a low accuracy.

### 3.2 DP Prediction Histograms

For each response data point, we can calculate the cumulative distribution function (CDF) of the authentic response data points $y_i$ using a normal distribution $N(\tilde{\mu}_i, \tilde{\sigma}^2)$, centering at the predicted mean value $\tilde{\mu}_i$, with standard deviation $\tilde{\sigma}$. Then, we divide the range of CDF into 10 pieces from $[0, 0.1]$, $(0.1, 0.2]$, ..., $(0.9, 1]$ to derive the histogram. For each data point $y_i$, the CDF of $y_i$ centered at $\tilde{y}_i$ will fall into one of the intervals.

The histogram shows the frequency of each of the intervals $M_1, \ldots, M_{10}$. Because they are disjoint, adding or removing only one element affects only one piece of the intervals, so the global sensitivity of the whole prediction histogram is bounded by 1, where $G_S(H(D)) = 1$. This is similar to the histogram query mention in chapter 2. Then, we add Laplace noises to the frequency of each $k$th interval: $M_k^* = M_k + \delta_k$, where $\delta_k$ is an independent random variable drawn from a Laplace distribution with the probability density function $p(G_S(H(D))|\epsilon) = \frac{1}{2\epsilon} e^{-1/\epsilon}$ According to Laplace mechanism, this algorithm is differentially private.

This differentially private prediction histogram can provide users with a visual understanding of the accuracy of their predictions. If the correlation of their prediction model is close to the true underlying correlation, and the synthetic dataset and authentic
dataset are similar, then the histogram should be flat. Also, when the prediction CDF is centered at 0.5, we can say that the prediction model is close to the underlying model and can capture the trend of the data points, even though the standard deviation estimated is larger than the true underlying standard deviation. It is because information differences may exist between the authentic and synthetic datasets. Combined with other algorithms mentioned in this chapter, users can eliminate the differences between synthetic and authentic dataset as a factor in consideration and thus have an accurate understanding of the accuracy of their own prediction models.

3.3 DP Kolmogorov-Smirnov Test

The Kolmogorov-Smirnov (KS) statistic quantifies a distance between between the empirical distribution functions of two samples. In our settings, we use the KS test to check whether two underlying one-dimensional probability distributions differ [25].

For each response data point $y_i$ in the authentic dataset, the predicted value $\tilde{y}_i$ can be drawn from $N(\tilde{\mu}_i, \tilde{\sigma}^2)$. $\tilde{\mu}_i$ is the predicted mean value, calculated based on the prediction models from synthetic data, for each data point $y_i$ on authentic dataset. $\tilde{\sigma}$ is the estimated standard deviation from synthetic data. Even though we derive the prediction intervals by using Student t distribution, we sample the predicted $\tilde{y}_i$ from normal distribution, since t distribution converges to normal distribution when sample size is large enough. We then have an empirical distribution for actual authentic response data points $Y = \{y_i\}_{i=1,...,N}$ and an empirical distribution for predicted values of the authentic response data points, $\tilde{Y} = \{\tilde{y}_i\}_{i=1,...,N}$.

The KS test results should be: $KS(Y, \tilde{Y}) = sup|F_{n,Y}(y) - F_{n,\tilde{Y}}(\tilde{y})|$, where $F_{n,Y}$ and $F_{n,\tilde{Y}}$ are empirical distribution functions for $Y$ and $\tilde{Y}$, respectively. By changing or removing one line in the authentic dataset $D$, the changed response data points are $Y' = \{y_i\}_{i=1,...,j',...,N}$. Let $\tilde{Y}' = \{\tilde{y}_i\}_{i=1,...,j',...,N}$ be the predicted values for the changed data points. Then the KS test result for the changed dataset $D'$ should be: $KS(Y', \tilde{Y}') = sup|F_{n,Y'}(y) - F_{n,\tilde{Y}'}(\tilde{y})|$, where $F_{n,Y}$ and $F_{n,\tilde{Y}}$ are empirical distribution functions for $Y'$ and $\tilde{Y}'$, respectively. The KS statistic between the two neighboring data $Y$ and $Y'$
can be derived as: \( \text{KS}(Y, Y') = \sup \left| F_{n,Y}(y) - F_{n,Y'}(y) \right| \leq 1/N \), which can be easily proved by considering the extreme case—moving one point of the set from the largest value to the smallest value. Similarly, \( \tilde{Y} \) and \( \tilde{Y}' \) are also neighboring data that the only difference is the \( j^{th} \) element. So the KS statistic between the two neighboring data \( \tilde{Y} \) and \( \tilde{Y}' \): \( \text{KS}(\tilde{Y}, \tilde{Y}') \) should also be smaller than or equal to \( 1/N \). Then, the global sensitivity for our KS test between the data points \( Y \) and the predicted values \( \tilde{Y} \) for the data points can be bounded \( 2/N \) on the authentic dataset. This can be shown as following:

\[
\text{KS}(Y', \tilde{Y}') = \sup \left| F_{n,Y'}(y) - F_{n,\tilde{Y}'}(\tilde{y}) \right|
\leq \sup \left| F_{n,Y'}(y) - F_{n,Y}(y) + F_{n,Y}(y) - F_{n,\tilde{Y}}(\tilde{y}) + F_{n,\tilde{Y}}(\tilde{y}) - F_{n,\tilde{Y}'}(\tilde{y}) \right|
\leq \sup \left| F_{n,Y'}(y) - F_{n,Y}(y) \right| + \sup \left| F_{n,Y}(y) - F_{n,\tilde{Y}}(\tilde{y}) \right| + \sup \left| F_{n,\tilde{Y}}(\tilde{y}) - F_{n,\tilde{Y}'}(\tilde{y}) \right|
\leq \frac{2}{N} + \text{KS}(Y, \tilde{Y})
\]

\[
\mathcal{G}_S(\text{KS}) = \left| \text{KS}(Y', \tilde{Y}') - \text{KS}(Y, \tilde{Y}) \right| \leq \frac{2}{N}
\]

Thus, we use \( \mathcal{G}_S(\text{KS}) = 2/N \). In the Laplace Mechanism, we add Laplace noise to the test result, \( \text{KS}^*(Y, \tilde{Y}) = \text{KS}(Y, \tilde{Y}) + \delta \), where \( \delta \) is an independent random variable drawn from a Laplace distribution with the probability density function \( p(\mathcal{G}_S(\text{KS})|\epsilon) = \frac{1}{2\epsilon}e^{-|\mathcal{G}_S(\text{KS})|/\epsilon} = \frac{1}{2\epsilon}e^{-2/(N\epsilon)} \), which is differentially private. The KS statistic quantifies a maximum distance between the empirical distribution of the authentic response data points and the empirical distribution of predicted values of the response data points.

To get the Laplace noised KS test reference distribution, we want to add Laplace distribution to the empirical KS statistic distribution. However, convenient simulation routines for the Kolmogorov distribution can be only evaluated using cumulative distribution function (CDF) [26]. All the computational methods available can only calculate empirical CDF for a given KS statistic value and sample size. And the empirical KS distribution depends on the sample size of the testing data.

Our empirical method to get the Laplace noised KS test reference distribution has four steps. The first step is is to first create a sequence of grids from 0 to 1, and compute the
empirical CDF value of the KS statistic for each grid. The second step is to subtract the CDF between each grid and get an approximation to the discrete PDF for each grid. The third step is to sample from the grids using the discrete probability calculated in the second step, add δ to the grid value, and save it into the reference empirical distribution, where δ is an independent draw from the Laplace distribution with the probability density function $p(G_S(KS)|\epsilon) = \frac{1}{\pi}e^{-2/(N\epsilon)}$. The fourth step is to repeat the third step $M$ times, and rank the grids from highest to lowest values. The P value for a particular differentially private KS statistic in our test can be computed by dividing the rank of the statistic in the grids by the total number of grids in our Laplace noised KS reference distribution, which should be $\frac{R}{M}$.

We can get differentially private P value for the differentially private KS statistic by referring to the Laplace noised KS reference distribution we constructed. By looking into the above mentioned test result, we can check whether the privately-computed results from the synthetic dataset can make good predictions on the authentic dataset. If the prediction model is different from the underlying model in the authentic dataset, then the trend of the prediction values of the data points should be different from the actual data points, which means the KS statistic should be relatively large.
Chapter 4

Empirical Illustrations

We illustrate the performance of the three algorithms by using simulated datasets under different scenarios.

For each scenario, the authentic dataset is generated by an underlying model, which includes the actual correlation between the response variable and the explanatory variables. The model is denoted as the authentic dataset generator. The model we used to generate the synthetic dataset is called the synthetic dataset generator. It may differ from the authentic dataset generator. Meanwhile, users may propose different prediction models. The prediction model is the one used by users to do analysis on the synthetic dataset.

We have tried different simulations and repeated each simulation for more than 50 times, and the conclusions we get for our verification methods do not vary with different simulations or repetitions. Thus, in order to simplify the presentation, we only show results from one simulation per scenario, and consider models with two explanatory variables. Both $X_1$ and $X_2$ in authentic and synthetic datasets are randomly generated from the same $X$ generator with normal distributions centered at 0 to 9, with standard deviation of 1. The authentic dataset generator’s coefficients are $(\alpha, \beta_1, \beta_2)$. Then we can calculate the mean value $\mu_i$ for each response data points $y_i$ based on the underlying model, and sample the disturbance term $\tau_i$ from $N(0, \sigma^2)$, where $\sigma$ equals to 1. Then $y_i = \mu_i + \tau_i$. By running regression on the authentic dataset, we can get the synthetic
dataset generators’ coefficients, denoted as \((\hat{\alpha}, \hat{\beta}_1, \hat{\beta}_2)\), and the estimated standard deviation \(\hat{\sigma}\). The mean value \(\hat{\mu}_i\) for each response data point \(\hat{y}_i\) in the synthetic dataset is calculated by plugging in the explanatory variables \(x_{i1}\) and \(x_{i2}\) together with the relevant coefficients such as \(\hat{\mu}_i = \hat{\alpha} + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2}\) when the synthetic generator is a linear model. Then the disturbance term \(\hat{\tau}_i\) in the synthetic dataset can be sampled from \(N(0, \hat{\sigma}^2)\). Thus the response data point \(\hat{y}_i = \hat{\mu}_i + \hat{\tau}_i\). The prediction models’ coefficients \((\tilde{\alpha}, \tilde{\beta}_1, \tilde{\beta}_2)\) and estimated standard deviation \(\tilde{\sigma}\) are the privately-computed results given by users based on the synthetic dataset.

We discuss the problem in two settings. In each setting, we consider different scenarios with varying underlying models and different prediction models. In the first setting, we present the case when we use the same model (linear/quadratic) to generate the authentic and synthetic dataset, and use different prediction models. In the second setting, we focus on using a different model to generate the synthetic datasets based on the authentic datasets.

For each case under different settings, assuming we do not know at the beginning either how the synthetic datasets are generated or whether the prediction models are the same as the underlying model, what we have are simply the synthetic dataset and our prediction model. By sending queries regarding our prediction results to the remote server, we can get feedback constructed by the remote server based on the authentic confidential dataset. By looking into that feedback, we can make analysis on the accuracy of our predictions. We can check our conclusions with the true authentic dataset and its underlying model generator to see whether the differentially private verification methods are sensitive enough to detect the information differences between the synthetic and authentic datasets and provide us with useful information on the prediction’s accuracy.

4.1 Same Generators

In this setting, we present the scenarios when we use the same model to generate the authentic and synthetic datasets.
4.1.1 Same Linear Model for Data Generation and Perfect Prediction Model

In the first scenario, we are using simple linear models to generate the authentic and synthetic datasets. The prediction model is also a simple linear regression model, which matches the true generative mechanism. We have:

Authentic dataset generator: \( Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \tau \)

Synthetic dataset generator: \( \hat{Y} = \hat{\alpha} + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \hat{\tau} \)

Perfect prediction Model: \( \tilde{Y} = \tilde{\alpha} + \tilde{\beta}_1 X_1 + \tilde{\beta}_2 X_2 + \tilde{\tau} \)

\[\text{Figure 4.1: The 95\% Confidence Interval and DP Prediction Histogram in Scenario 1}\]

In the first algorithm, we are testing the 95\% prediction tolerance intervals\(^1\). The returned percentage is around 95.1\%. In Figure 4.1, the DP prediction histogram is relatively flat. From the first and second algorithms, we can say that the difference between the authentic and synthetic datasets may not be significant. The KS test result is around 0.02 for the perfect prediction model, and the Pvalue is around 0.988. We conclude that the privately-computed results on the synthetic dataset is accurate in making predictions on the authentic dataset, which may indicate that our prediction model is close to the underlying model and the information difference between the two models are negligible.

\(^1\)The 95\% prediction tolerance intervals figure can not be published to users and satisfy differential privacy. We list the figures here to have a better understanding of the DP prediction tolerance intervals algorithm
4.1.2 Same Linear Model Generators and Imperfect Prediction Model

Now we use the same simple linear model in generating the authentic and synthetic datasets. However, the prediction models we use are imperfect. One only uses one of the explanatory variables, the other uses a quadratic model. We have

Authentic dataset generator: \( Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \tau \)

Synthetic dataset generator: \( \hat{Y} = \hat{\alpha} + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \hat{\tau} \)

Imperfect prediction Model 1: \( \tilde{Y} = \tilde{\alpha}_1 + \tilde{\beta}_{1,1} X_1 + \tilde{\tau}_1 \)

Imperfect prediction Model 2: \( \tilde{Y} = \tilde{\alpha}_2 + \tilde{\beta}_{2,1} X_1^2 + \tilde{\beta}_{2,2} X_2^2 + \tilde{\tau}_2 \)

![Figure 4.2: The 95% Confidence Interval and DP Prediction Histogram in Scenario 2](image)

In the prediction tolerance interval algorithm, the returned percentages are around 95% for both prediction models under the 95% confidence intervals tolerance limits. We would infer that the information differences are negligible between the synthetic and authentic datasets. If not, the returned percentage should not be around 95%. In Figure 4.2, the
DP prediction histogram is relatively flat for the one-explanatory variable prediction model, but rather bumpy for the quadratic prediction model. We would infer that the quadratic prediction model may not reveal the true correlations between the response variable and explanatory variables. In the KS test algorithm, the returned statistic for the first prediction model is around 0.017, and the P-value is around 0.995. The returned statistic for the second prediction model is around 0.106 and the P-value is around 0. Then definitely, we can say the quadratic regression model is not accurate in making predictions on the authentic dataset, since the model is not accurate. For the one explanatory regression model, we think it can partially reveal the linear correlation between the variables.

4.1.3 Same Quadratic Model Generators and Imperfect Prediction Model

We are using the quadratic models to generate both the authentic and synthetic datasets. One prediction model is only using one of the explanatory variables with quadratic form. The other prediction model is linear.

Authentic dataset generator: \( Y = \alpha + \beta_1 X_1^2 + \beta_2 X_2^2 + \tau \)

Synthetic dataset generator: \( \hat{Y} = \hat{\alpha} + \hat{\beta}_1 X_1^2 + \hat{\beta}_2 X_2^2 + \hat{\tau} \)

Imperfect prediction Model 1: \( \tilde{Y} = \tilde{\alpha}_1 + \tilde{\beta}_{1,1} X_1^2 + \tilde{\tau}_1 \)

Imperfect prediction Model 2: \( \tilde{Y} = \tilde{\alpha}_2 + \tilde{\beta}_{2,1} X_1 + \tilde{\beta}_{2,2} X_2 + \tilde{\tau}_2 \)

In the prediction tolerance intervals algorithm, the returned percentage is around 94.2% for the regression only using one explanatory variable in the quadratic form. It is around 94.5% for the simple linear prediction model. In the second algorithm, both the histograms are very bumpy in Figure 4.3. The KS statistic for the first prediction model is around 0.095, and the P-value returned is around 0.0004. The KS statistic for the second prediction model is around 0.177, and the P-value is around 0.

For the prediction tolerance intervals method, the percentages are still around 95%, which we would infer that the synthetic and authentic dataset are similar. From the bumpy DP prediction histograms in Figure 4.3, we would infer that the disturbance
term $\bar{\tau}$ for both the prediction models may contain information not detected by the prediction models. Also from the KS statistic returned, we can say the first prediction model is more accurate than the second prediction model.

### 4.1.4 Summary

Generally speaking, in this setting, the percentage returned by the 95% prediction tolerance intervals is around 95%, from which we may conclude the information difference between the authentic dataset and synthetic dataset is not very significant, compared with the following setting. It is because even though the prediction model may perform poorly, the standard error is large enough to create wide enough prediction confidence intervals that very likely contain the authentic response data points, as long as the two
datasets are similar. If the returned value is significantly different from 95%, we can conclude that there must exist information differences between the two datasets. However, it is not true in the other way around. In our simulation, the 95% prediction intervals algorithm is not useful enough to provide information on the accuracy of predictions from synthetic data. However, in the empirical analysis, users may prefer to choose the \( \Delta \) to construct the prediction tolerance intervals based on their empirical information, or choose the tolerance interval with lower and upper bounds proportional to predicted mean values so that they can have a better understanding of their analysis on a certain research topic.

The DP prediction histogram can not only detect the information differences but also provide a good visual detector for whether the prediction model can catch the underlying trend of the authentic dataset. When there exist significant information differences between the authentic and synthetic datasets, the DP prediction histogram will be rather bumpy. KS test performed well, directly quantifying the accuracy of the prediction models so that users can make comparisons between different models.

### 4.2 Different Generators

In this setting, we use different models to generate the authentic and synthetic datasets. Typically, data producers are not able to fit the true underlying model in the authentic datasets, so that they use some imperfect models to generate the synthetic datasets based on the authentic datasets.

#### 4.2.1 Quadratic Authentic Generator and Linear Synthetic Generator

We are using the quadratic model to generate the authentic dataset. The synthetic dataset is generated from simple linear model based on the authentic dataset. We are using both the quadratic model and the linear model as the prediction models. In the first case, both the authentic dataset generator and prediction model are quadratic, but the synthetic dataset generator is linear. In the second case, both the synthetic dataset generator and prediction model are linear, but the underlying model is quadratic.
Authentic dataset generator: \( y = \alpha + \beta_1 x_1^2 + \beta_2 x_2^2 + \tau \)

Synthetic dataset generator: \( y = \hat{\alpha} + \beta_1 x_1 + \beta_2 x_2 + \hat{\tau} \)

Prediction Model 1: \( \tilde{Y} = \tilde{\alpha}_1 + \tilde{\beta}_{2,1} X_1^2 + \tilde{\beta}_{2,2} X_2^2 + \tilde{\tau}_2 \)

Prediction Model 2: \( \tilde{Y} = \tilde{\alpha}_2 + \tilde{\beta}_{1,1} X_1 + \tilde{\beta}_{1,2} X_2 + \tilde{\tau}_1 \)

**Figure 4.4:** The 95% Confidence Interval and DP prediction Histogram in Scenario 5

For the first prediction model, the returned percentage from the DP prediction tolerance intervals is around 1. The DP prediction histogram is centered at 0.5, and the KS statistic returned is around 0.104 with Pvalue around 0. For the second prediction model, the returned percentage from the DP prediction tolerance intervals is around 95.4%. The DP prediction histogram is symmetric at 0.5 with higher frequencies at [0,0.1] and [0.9,1]. The KS statistic returned is around 0.155, and Pvalue is close to 0.

From the result above, the users can determine the information loss exists when we generate the synthetic dataset from linear model instead of the quadratic model. The
KS statistic returned for both the datasets are large and with a very small $P$-value. But by comparing the results, we can say that still the first prediction model can give a more precise prediction on the authentic dataset. Understandably, we know that the first prediction model is the same as the underlying authentic dataset’s model. But the bumpy histogram does reveal that the information loss is not negligible under this scenario.

4.2.2 Linear Authentic Generator and Quadratic Synthetic Generator

We use the simple linear model to generate the authentic dataset. The synthetic dataset is generated from a more complex model based on the authentic dataset, which not only include the linear form but also include the quadratic form for each variable. The first prediction model is the same as the underlying authentic dataset generator. Both are linear. The second prediction model is quadratic. We take this scenario into consideration because sometimes the agency may generate the synthetic data using over-fitting model.

Authentic dataset generator: $Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \tau$

Synthetic dataset generator: $\hat{Y} = \hat{\alpha} + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_1^2 + \hat{\beta}_3 X_2 + \hat{\beta}_4 X_2^2 + \hat{\tau}$

Prediction Model 1: $\hat{Y} = \hat{\alpha}_1 + \hat{\beta}_{1,1} X_1 + \hat{\beta}_{1,2} X_2 + \hat{\tau}$

Prediction Model 2: $\hat{Y} = \hat{\alpha}_2 + \hat{\beta}_{2,1} X_1^2 + \hat{\beta}_{2,2} X_2^2 + \hat{\tau}$

For the first prediction model, the returned percentage from the DP prediction tolerance intervals is around 95%. The DP prediction histogram is flat. The DP KS statistic is around 0.026 with $P$-value around 0.8585. From the percentage returned in the first and the flat histogram, we can infer that the authentic dataset and the synthetic dataset provide similar information on this prediction model. By looking into the results return in the KS test, we can conclude that the prediction results are accurate and the prediction model match with the underlying generative mechanism on the authentic dataset. For the second prediction model, he returned percentage from the DP prediction tolerance intervals is around 94.4%. The DP prediction histogram is rather bumpy. The KS statistic is around 0.084, and $P$-value is around 0.002. We can say the quadratic model is not accurate in making predictions on the authentic dataset.
4.2.3 Summary

In the real world setting, the data producer may not know the underlying model so that the synthetic datasets generated have some information differences from the authentic datasets. They may use an over-fitting model or an under-fitting model to fit the authentic dataset to generate the synthetic dataset. In the case when data agency use an under-fitting model, such as the first scenario when the underlying model is quadratic but the synthetic data is generated from a linear model, then the DP prediction histogram will not be flat. However, if the DP histogram is centered around 0.5, we still can infer that our prediction model be reasonable useful at capturing the underlying trend of the dataset. By looking into the DP KS test statistic and its P-value, we have a better understanding of the accuracy of predictions from synthetic data. In the case when the data-agency use an over-fitting model to generate the synthetic data,
such as the second scenario we discussed above, the prediction results can be accurate, as long as the prediction model is close to the generative mechanism. By looking into the DP prediction histogram and the DP KS test results, we can easily tell how accurate the prediction model is.
Chapter 5

Conclusion

This thesis provides a general strategy that users can rely on to do analysis without having direct access to a confidential dataset. By releasing a synthetic dataset and allowing users to send verification queries to a remote server, users can have a sense of the accuracy of their privately-computed results from synthetic dataset. The prediction tolerance intervals method may detect the information differences that exist between the authentic database and synthetic database. However, in our simulations, we find that the 95% prediction tolerance intervals may not be useful. However, it provides users with freedom to choose the tolerance levels that should be more useful in empirical settings. The DP prediction histogram method can give the user a visual understanding about their privately-computed prediction model’s accuracy on the authentic database whether or not the predictions can capture the trend of the true data points. If the prediction model is close to the underlying model on the authentic dataset, then the histogram should be either flat or concentrated at 0.5. The DP KS test directly quantifies the magnitude between the two empirical distributions of the predicted values and the true data points. By implementing these three algorithms together, the user can have a general understanding of the accuracy of their predictions from synthetic dataset. The privacy budget of publishing results of the three algorithms are 3ε in total. Thus, in empirical settings, we may require users to choose only one of the algorithms to test their prediction results from synthetic dataset to satisfy the limited privacy budget.
There are other verification methods for users to do model comparison or selection such as R-square, adjusted R-square, Mallow’s Cp, Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), F-test, t-test. However, all the methods mentioned above may need to re-run the models on the authentic data set. If we want to use the large authentic data set, we may have to do partitions on the data set to guarantee the differentially private criterion. One thing good about the KS statistic is it can directly test the distribution differences between the predicted values and the true data points. However, we can further develop some differentially private model selection or averaging methods. Thus, the user can do analysis based on these algorithms and have more precise predictions.

There are other complex models like generalized regression models such as Poisson regression models, exponential regression models or logistic regression models. But sometime we may require new algorithms to test the prediction’s accuracy. For example, for the logistic regression models, area under the curve (AUC) might be a better way for checking the model accuracy instead of KS statistics, since the prediction values should only be binary. But for Poisson regression or exponential regressions, our verification methods still can work very well. In the future research on this thesis, I may plan to focus on testing the verification methods on empirical datasets with more complex models.
Bibliography


