PRIVACY AND ACCURACY SYSTEMS ON FINANCIAL DATABASES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ADNAN BILGEN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
MATHEMATICS

JUNE 2019

Approval of the thesis:

PRIVACY AND ACCURACY SYSTEMS ON FINANCIAL DATABASES

submitted by **ADNAN BILGEN** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Mathematics Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Yıldıray Ozan _____
Head of Department, **Mathematics**

Prof. Dr. Ersan Akyıldız _____
Supervisor, **Mathematics, METU**

Assoc. Prof. Dr. Murat Cenk _____
Co-supervisor, **IAM, METU**

Examining Committee Members:

Prof. Dr. Sevtap Ayşe Kestel
IAM, METU _____

Prof. Dr. Ersan Akyıldız
Mathematics, METU _____

Assist. Prof. Dr. Erman Ayday
Computer Engineering, Bilkent University _____

Assoc. Prof. Dr. Ali Doğanaksoy
Mathematics, METU _____

Assoc. Prof. Dr. Oğuz Yayla
Mathematics, Hacettepe University _____

Date: 13.06.2019

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: ADNAN BILGEN

Signature :

ABSTRACT

PRIVACY AND ACCURACY SYSTEMS ON FINANCIAL DATABASES

Bilgen, Adnan
Ph.D., Department of Mathematics
Supervisor : Prof. Dr. Ersan Akyıldız
Co-Supervisor : Assoc. Prof. Dr. Murat Cenk

June 2019, 53 pages

A statistical database is a collection of data which contains the sensitive information of individuals. They are extensively used for many purposes. Since the system contains sensitive data of individuals, it must be secure to protect every individual in the data set against attackers.

In this thesis, we especially work on the privacy of databases which include financial data. We use data perturbation techniques to work the accuracy and privacy balance between the original database and the perturbed one. We test the accuracy of our masked data on selected statistics. We measure the reliability of our system against existing attack techniques. We develop user-friendly software to use data sanitization by using Java and R programming languages.

Keywords: Additive Noise, Privacy, Accuracy, Attack Techniques, Disclosure

ÖZ

FİNANSAL VERİ TABANLARINDA GİZLİLİK VE DOĞRULUK SİSTEMLERİ

Bilgen, Adnan
Doktora, Matematik Bölümü
Tez Yöneticisi : Prof. Dr. Ersan Akyıldız
Ortak Tez Yöneticisi : Doç. Dr. Murat Cenk

Haziran 2019 , 53 sayfa

İstatistiksel veritabanları bireyler hakkında hassas verileri de barındırır. Bu sistemler her ne kadar bilimsel çalışmalar açısından önemli bir girdi olsalar da bireylerin hassas verilerini içerdikleri için; saldırganlara karşı, veri kümesindeki her bir bireyi koruyacak şekilde güvenli olmalıdır.

Tez çalışmalarımızda, özellikle; finansal verileri içeren veri tabanlarındaki gizliliğin korunması üzerine çalıştık. Gizlilik doğruluk dengesini sağlamak adına orijinal veriyi maskeledik. Sistemimizin mevcut ataklara karşı dayanıklılığını test ettik. Java ve R programa dillerini kullanarak, orijinal veri kümesinden maskelenmiş veri üretilmesine olanak sağlayan kullanıcı dostu bir yazılım geliştirdik.

Anahtar Kelimeler: Toplamsal Gürültü, Gizlilik, Doğruluk, Saldırı Teknikleri, İfşa

To my beloved wife Ebru and my daughter Ela Eren...

ACKNOWLEDGMENTS

I would like to thank to my supervisor Prof. Dr. Ersan Akyıldız for his encouragement and his constant support. I am grateful to Prof. Dr. Sevtap Kestel for her friendly approaches and motivating comments. I would like to thank Asist. Prof. Dr. Erman Ayday for his contributions and corrections during the progress meetings.

It is an honor for me to acknowledge my co-advisor Assoc. Prof. Dr. Murat Cenk for every intellectual or visionary contribution he has given.

I would not forget to thank to Prof. Dr. Yurdahan Güler who inspired me as a Scientist.

I thank to my friend İrem Keskinkurt Paksoy for sharing her ideas and for our cooperation.

I acknowledge the Scientific and Technological Research Council of Turkey for supporting this thesis work partially with the program BIDEB 2211.

This Thesis is granted by Central Bank of the Republic of Turkey under the project, "Protection of Confidentiality in Statistical Databases".

I would like to thank my parents, sisters and nieces for their love and endless support.

My wife Ebru and my daughter Ela Eren deserve much more than my special thanks for their love, respect, patience and understanding during this graduate work of which I hope the result is worthy.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

SF	Spectral Filtering
SVD	Singular Value Decomposition
PCA	Principle Component Analysis
SQL	Structured Query Language
PDF	Probability Density Function
CDF	Cumulative Distribution Function
GLM	Generalized Linear Model

CHAPTER 1

INTRODUCTION

A statistical database is a collection of data which contains sensitive information of individuals (patient, student, company, etc.) which are commonly used in research, planning and decision making. With the development of technology, it is easy to achieve, collect and analyze data. Collected data is used extensively by researchers and decisionmakers in different fields. Increasing amounts of such databases are provided by agents like census bureaus, universities, hospitals and, business organizations. A data collector releases the data for more analysis. Released data contain confidential information such as income, credit ratings, type of disease, or test scores of individuals. In the medical area, health record systems are constructed for exchanging medical information [1]. In e-commerce, data is collected from many activities including searching, browsing and online shopping [2]. Mobile healthcare record has been examined because of the high sensitivity of health data and disclosure risk on it [3, 4]. In 2015, researchers from MIT described that unimportant dates and places of only four pieces of consumption records are enough to identify 90 percent of the people in a dataset which obtained from credit-card activities of the users [5]. Because of the concerns about individuals privacy, researchers develop a series of privacy protection methods mainly including data distortion, data encryption, and restrictive release [6].

In the privacy context, database fields are categorized into four basic categories. The first field type is explicit identifiers. This field directly defines individuals, national security number, social security number are examples for this field. Quasi-identifiers are the second type. A single quasi-identifier does not define the individuals alone but some of them together may show the owner of the record. Date of birth, ZIP code, and

gender are an example of this field. The third type is the reason for all these works, called sensitive fields. Disease, income, and test score are examples of this field. The last type is non-sensitive attributes, favorite color is an example of this type [7].

To protect the sensitive information of individuals in healthcare data, the restricted releasing methods are proposed[8]. These methods are *k-anonimity*, *l-diversity*, and *t-closeness*. Sweeney and Samarati introduce the first restricted releasing method: *k-anonimity*[9, 10]. In *k-anonimity*, each record is indistinguishable from at least $k-1$ records, but, it may disclose privacy information[11]. Then, *l-diversity* and *t-closeness* were proposed to improve privacy protection [12]. These methods also cannot protect the privacy of individuals in a data set. The curators sometimes apply some simple anonymization techniques, but the adversary can destroy the privacy and re-identify the data set. In the early years, some researchers deanonymize a medical data set by combining with another public vote list data set [13]. The linked information is defined as background information [14]. The adversary with background information will be able to identify the individuals records with high probability.

The main purpose of this study is to make financial data sets available to researchers while ensuring the confidentiality of the data. Since financial data sets include sensitive data, releasing them is not a good idea. Instead of the original data set, the owner of the data sets will release the sanitized data set by masking the original data using either additive [15] or multiplicative [16] noise addition methods. In this thesis, we have two important aim. The first one is in the masked dataset, we want to preserve the statistical properties of the original dataset. Besides accuracy, we want to satisfy the privacy of the individuals in a dataset. We have to balance accuracy and privacy. The second one is we want to get accurate results in a masked data set for the highest number of statistical functions as possible. We want to construct a non-interactive system, which means that it would suffice to mask the data once before releasing it.

In Chapter 1 we discuss the studies on privacy and define our problem. In Chapter 2, we give brief information about some statistical methods. In Chapter 3, we explain the techniques we developed to obtain perturbed (masked) data. Also in this chapter, we test the accuracy on selected statistics (mean, standard deviation, kurtosis, skewness, simple and multiple linear regression analysis, simple and multiple logistic regression

analysis) when we work on masked data sets instead of the original data sets. We also test accuracy on log-transformed masked data. In Chapter 4, we study the known attacks from literature. We apply these techniques to our masked data sets to measure the reliability of our system against existing attack techniques. In this section, we have seen the weakness of the classical random number generator used in R-program. The necessity of improving the way of producing random numbers emerged. We made an improvement in random number generation part. In this section, we also define some possible attacks and offer some data releasing strategies to protect our system from these attacks. In Chapter 5, we give brief information about our user-friendly software which will be used to generate secure masked data sets from the original data. On the last chapter, we give a brief summary and talk about future works.

CHAPTER 2

PRELIMINARIES

2.1 Statistical Database

A statistical database is a collection of data which contains the sensitive information of the individuals. They are used in many research, planning and decision making. Increasing amounts of statistical information are provided by agents like census bureaus, universities, hospitals, and business organizations. They contain confidential information such as income, credit ratings, type of disease, or test scores of individuals. Researchers are using these databases for their scientific works. For example, they obtain Mathematical models from the analysis to predict future results. Firms use statistical databases to define strategies to increase their profit.

Financial databases are a special kind of statistical databases. They include financial data of individuals or firms, such as budget, revenue, and profit.

2.2 Normal Distribution

Normal distribution (Gaussian distribution) or a bell curve is a continuous probability distribution. The normal distribution suits many natural phenomena such as height and blood pressure, so it is the most important probability distribution in statistics. It describes how the values of a variable are distributed. Most of the observations cluster around the central peak and probability of the values further away stands equally in both directions in normal distribution as a symmetric distribution. It is defined by its mean and standard deviation. The distribution is shifted by the mean value, either to the left or to the right on the x-axis, and the standard deviation controls

the spread. The standard deviation is the same in all directions. That is the important characteristic from other distributions. The general formula for the probability density function of the normal distribution is

$$f(x) = \frac{e^{-(x-\mu)^2/2\sigma^2}}{\sigma\sqrt{2\pi}} \tag{2.1}$$

where μ is the location parameter and σ is the scale parameter.

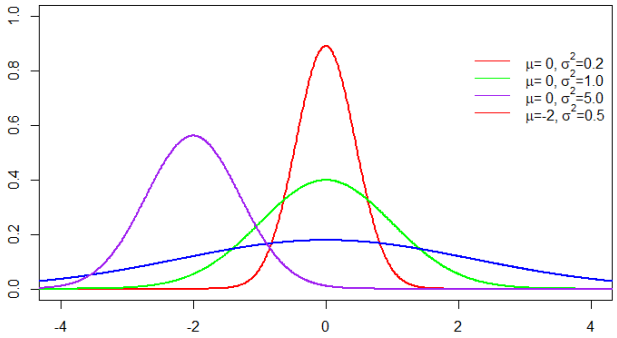

Figure 2.1: PDF of Normal Distribution

Standart normal distribution is the case where $\mu = 0$ and $\sigma = 1$. The equation for the standart normal distribution is

$$f(x) = \frac{e^{-x^2/2}}{\sqrt{2\pi}}. \tag{2.2}$$

The cumulative distribution function of the standard normal distribution is

$$F(x) = \int_{-\infty}^{x} \frac{e^{-\pi^2/2}}{\sqrt{2\pi}}. \tag{2.3}$$

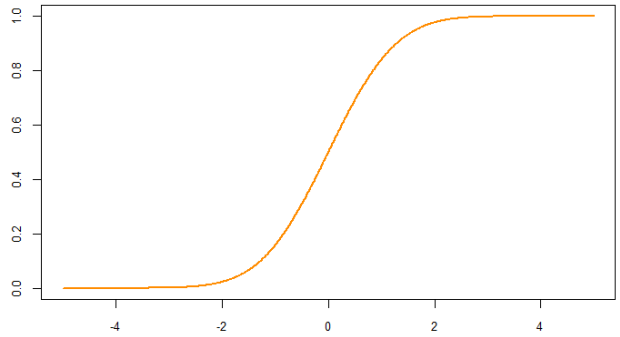

Figure 2.2: CDF of Standart Normal

2.3 Statistics

Statistics is the area that deals with developing and studying methods for collecting, analyzing, interpreting and presenting experimental data. In all scientific fields, statistical methods are used. They are used to perform technical analysis of data. During our work, we use the following statistical functions.

2.3.1 Mean

The statistical mean refers to the mean or average that is used to derive central tendency of the data in question. The mean of a sample $x_1, x_2, \ldots, x_n$ is usually denoted by $\overline{x}$, and it is the sum of the values divided by the number of the values in the sample.

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + \ldots + +x_n}{n} \tag{2.4}$$

Statistical mean has a wide range of use. We can eliminate accidental errors by calculating the mean of the result of the experiments instead of the result derived from a particular experiment. The statistical mean is popular because it includes every item in the data set and it can easily be used with another statistical measurement. In a normal distribution, the statistical mean is equal to median and mode.

The major disadvantage in using statistical mean is that it can be affected by extreme values, and therefore it might be biased.

2.3.2 Standard Deviation

In statistics, the standard deviation is a measure for a group that shows how they are spread out from average(mean). For a sample $x_1, x_2, \ldots, x_n$, the standard deviation can be calculated as follows:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{2.5}$$

The standard deviation tells us how well the mean represents all of the data. It measures the deviation from the mean and shows the central tendency. It squares and makes the negative numbers positive. The square of small numbers is smaller and the square of large numbers is larger. So it makes you ignore small deviations and see the larger one clearly.

2.3.3 Skewness

In a statistical distribution, skewness is asymmetry. The curve skewed either to the left or to the right. The graph is symmetrical in a normal distribution. On each side of the curve, the tails are exact mirror images of each other. The tail on the curve's left-hand side is longer than the tail on the right-hand side when a distribution is skewed to the left. This situation is called a negative skewness. The tail on the curve's right-hand side is longer than the tail on the left-hand side when a distribution is skewed to the right. This situation is called a positive skewness. Skewed data arises quite naturally in many situations. For example, salaries are skewed to the right because the mean can greatly be affected by even just a few individuals who earn millions of dollars, and there are no negative incomes. We see all kind of skewness on the following figure.

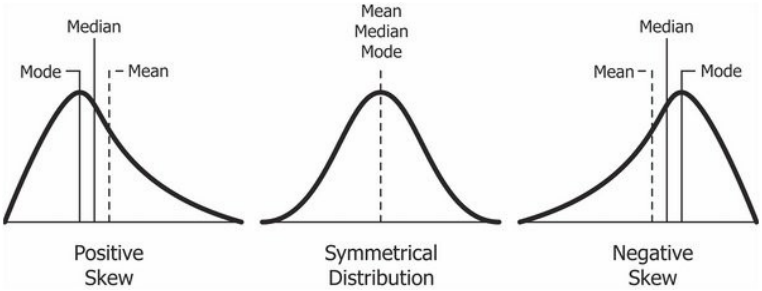

Figure 2.3: Skewness

2.3.4 Kurtosis

Kurtosis is a measure of the "tailedness" of the probability distribution of a real-valued random variable in probability theory and statistics. The kurtosis of any univariate normal distribution is 3. It is common to compare the kurtosis of distribution to this value. Distributions with kurtosis less than 3 are said to be platykurtic, greater than 3 are said to be leptokurtic and equal to 3 are said to be mesokurtic. The next figure shows us all types of kurtosis.

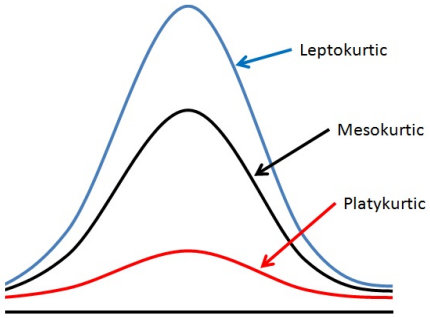

Figure 2.4: Kurtosis

2.3.5 Simple and Multiple Linear Regression

One of the important question is that, how the variables are related, if we have data with multiple variables. Regression is a set of techniques for estimating relationships. We start with **simple linear regression** in which there are only two variables of interest.

We fit our data to a line $y = \beta_0 + \beta_1 x$ in simple linear regression. x is called the independent (predictor) variable and y is called the dependent (response) variable here. β_1 is one of the most important quantity in any linear regression analysis. It is the slope of the line. If β_1 is close to zero then it indicates small to no relationship. If the value of β_1 is large, either positive or negative values, then it indicates large positive or large negative relationship respectively. β_0 is the intercept of the line.

We calculate β_0 and β_1 by using least square method as following:

$$\begin{aligned} \beta_1 &= \frac{\sum_{i=1}^{n} x_i y_i - \frac{1}{n}\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i^2 - \frac{1}{n}(\sum_{i=1}^{n} x_i)^2} \\ &= r\frac{s_y}{s_x} \end{aligned} \tag{2.6}$$

$$\beta_0 = \bar{y} - \hat{\beta}_1 \bar{x} \tag{2.7}$$

where $\bar{x}$,$\bar{y}$,s_x and s_y are the sample means and standart deviation for x values and y values, respectively. And the correlation coefficient r is defined as :

$$r = \frac{1}{n-1}\sum_{i=1}^{n}(\frac{x_i - \bar{x}}{s_x})(\frac{y_i - \bar{y}}{s_y}). \tag{2.8}$$

The **multiple linear regression** is the case when there are more than one independent variable, instead of x we have a vector $(x_1, x_2, \ldots, x_p)$ for every data point i. So, we have n data points, each with p different predictor variables. We will then try to predict y for each data point as a linear function of the different x variables :

$$y = \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p. \tag{2.9}$$

We will represent our input data in matrix form as X, an $n \times p$ matrix where each row corresponds to a data point and each column corresponds to a feature. We'll represent the collection as an n-element column vector y, since each output y_i is just a single number. Then our linear model can be expressed as

$$y = X\beta + \epsilon \tag{2.10}$$

where β is a p-element vector of coefficients, and ϵ is an n-element matrix where each element like ϵ_i earlier, is normal with mean 0 and variance σ^2. To solve the optimization problem, we can use some basic linear algebra:

$$\hat{\beta} = (X^T X)^{-1} X^T y. \tag{2.11}$$

2.3.6 Logistic Regression

A generalized linear model (GLM) logistic regression is used to model a binary categorical variable using numerical and categorical predictors.

We assume a binomial distribution produces the outcome variable and we want to model p the probability of success for a given set of predictors. We need to establish a reasonable link function that connects a linear model $\eta = \beta_0 + \beta_1 X_1 + \ldots + \beta_n X_n$ to p to complete specifying the logistic model. There are lots of options but commonly used is the so-called logit function which is described as follows:

$$logit(p) = \log(\frac{p}{1-p}), \qquad for \qquad 0 \leq p \leq 1 \tag{2.12}$$

The logit function takes a value between 0 and 1. And also maps it to a value between $-\infty$ to ∞. Inverse logit(logistic) function

$$g^{-1}(x) = \frac{exp(x)}{1 + exp(x)} = \frac{1}{1 + exp(-x)} \tag{2.13}$$

takes a value between $-\infty$ and ∞ and maps it to a value between 0 and 1.

2.3.7 Log Transformations

The log transformation can be used to transform highly skewed distribution to less skewed distribution. This is important for making patterns in the data more interpretable.

As an example how a log transform make patterns more visible; Consider brain weights of animals as a function of their body weights. In Figure 2.5 both graphs

plot the brain weight of them. On the left panel the raw weights are shown, on the right panel the log transformed weights are plotted. After log transform the pattern of the data becomes more visible.

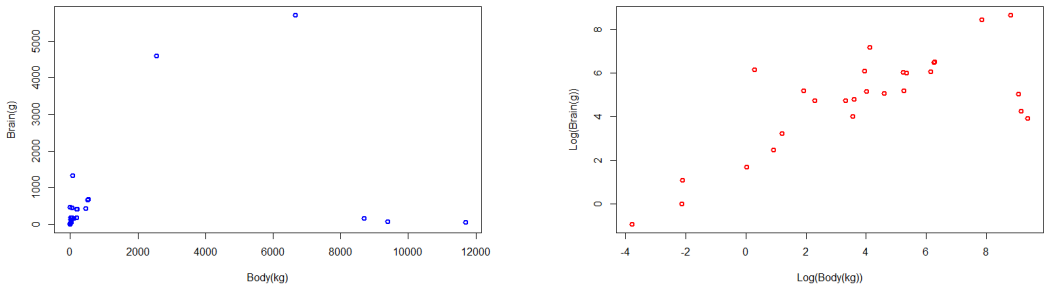

Figure 2.5: Raw and log-transformed form of skew data

2.4 Random Numbers

Random numbers are the set of numbers where the values are uniformly distributed over a defined interval and it is impossible to predict future values by using present ones. They are important in statistical analysis and probability theory.

Mostly, they are randomly derived from single-digit decimal numbers, integers in $\{0, 1, \ldots, 9\}$. Generating random digits from this set is not trivial. In lotteries this method is popular . Digits are selected one by one. After each selection, the selected ball come back to the set and the balls are mixed a while, then another ball is allowed to exit.

The existing number-generation algorithms produce future values by using past and/or current ones. Random numbers which are generated by some kinds of algorithms are called pseudo-random numbers. To generate such random numbers, we start with a seed value. Seed is the starting point. In our work we use R programming language to produce random numbers. In R the seed must be an integer between $-2^{31} + 1$ and $2^{31} - 1$. This is, of course, a small number considering today's technology, and therefore it is not secure against a brute force attack.

In our work, we generate random numbers as a noise for masking original data sets.

2.5 Sinusoidal and Triangular Data

A sinusoid is a periodic continuous wave. We use sinusoidal distributed data to verify the work on existing attack techniques which we will discuss later on the related section . We generate 10000 random numbers from $\sin$ function using the following equation :

$$y(x) = sinx \tag{2.14}$$

We see the first 250 generated number on a graph.

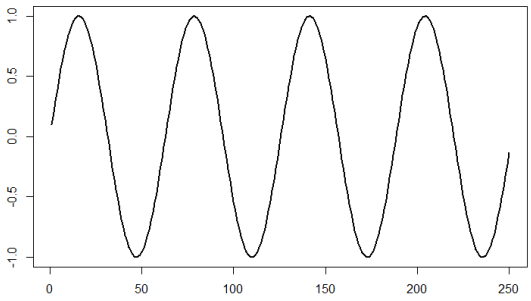

Figure 2.6: Section from sinus wave

A triangle wave is a non-sinusoidal, called triangular because of its shape. It is a periodic, piecewise linear, continuous real function. To generate triangular distributed data, we use the following equation :

$$y(x) = \frac{2}{\pi} \arcsin(\sin(\pi x)) \tag{2.15}$$

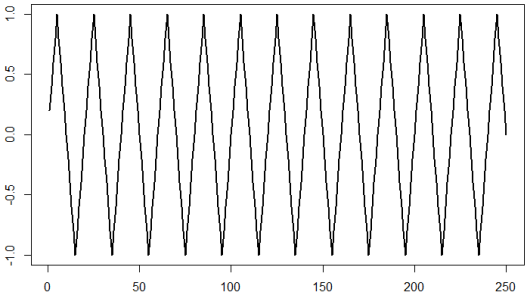

Figure 2.7: Section from triangular wave

CHAPTER 3

SANITIZATION AND ACCURACY IN STATISTICS

3.1 Sanitization

Data sanitization is the process to secure sensitive data. Noise addition works by adding or multiplying a stochastic or randomized numbers to confidential quantitative attributes.

In our work to achieve privacy without losing accuracy, we mask our original financial data before releasing. We use additive and multiplicative noise to sanitize the original data set. We generate noise from normal distribution. To generate random numbers from the normal distribution, we have to define two parameters, mean and variance. We use the mean and variance of the original data as input parameters.

3.1.1 Additive Masking

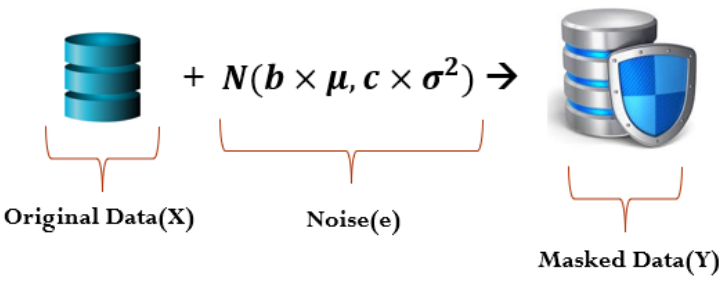

Figure 3.1: Additive Masking

When using noise to masking numerical data sets, the general assumption in the literature is that the mean of the generated noise is zero and the variance of the generated noise is proportional to the variance of original data [15]. In our study, we set the mean of the noise proportional to the mean of the original data instead of zero. Let X is the original data. We generate the noise from normal distribution as : $N(b\mu, c\sigma^2)$ where μ and σ^2 are mean and variance of original data respectively.b and c are the proportion parameters.We can use b and c to control the accuracy in desired level.Let Y and e is masked data and generated noise respectively. We release masked Y instead of original data X, as following:

$$Y = X + e \tag{3.1}$$

where

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad , \quad X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} , \quad and \qquad e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix} . \tag{3.2}$$

3.1.2 Multiplicative

Multiplying original data with generated noise might protect the confidentiality better, but there will be accuracy problems in statistical functions. Let X be the original data. We generate the noise from normal distribution with mean 1 and variance proportional to the variance of original data such that $N(1, c\sigma^2)$. We release the masked Y as following:

$$Y = X \times e \tag{3.3}$$

explicitly,

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 \times e_1 \\ x_2 \times e_2 \\ \vdots \\ x_n \times e_n \end{bmatrix}. \tag{3.4}$$

When we compare with additive noise, multiplicative noise is safer but cannot maintain the statistical properties of the data [16].

3.2 Accuracy on Statistical Methods

Accuracy refers to the closeness of the results of statistical functions computed with the original and masked data sets. In this section we make experiments to discuss the accuracy on selected statistics when we work with masked data instead the original data. For our experimental works we choose 5 variables randomly from our financial data set. Our variables consist 7951 row. Some of them includes positive, some of them negative , and some of them include both negative and positive values.

Table 3.1: Summary of variables that are used in the experiments

	Range	$(-\infty, 0)$	0	$(0, \infty)$
$var1$	$(-\infty, \infty)$	2342	7	5602
$var2$	$(0, \infty)$	-	4685	3266
$var3$	$(0, \infty)$	-	750	7201
$var4$	$(0, \infty)$	-	369	7582
$var5$	$(-\infty, 0)$	7311	64	-

3.2.1 Accuracy on Additive Masked Data

Firstly, we compute the descriptive statistics. For each variable we calculate the mean, standart deviation, kurtosis, and skewness for both the original and masked data set.

We use aditive data perturbation technique. We generate noises from normal distribution as $N(b \times \mu, c \times \sigma^2)$, where the parameters $|b|$ and $|c|$ are both in the interval

$[0, 0.05]$. We choose $|b|$ and $|c|$ in the defined interval to preserve accuracy close to desired values. If b and c choosen close to 0.05 privacy on masked data is increasing, while accuracy is decreasing. If b and c choosen close to 0 accuracy on masked data set is increasing while privacy is decreasing. We prepare the following table for descriptive analysis. In this work we take $b = c = 0.025$. We calculate each statistical function on original data, then on the masked data and take the ratio to see the accuracy. We prepare the following table for randomly selected variables :

Table 3.2: Accuracy of descriptive statistics on original and additively masked data set

	$var1$	$var2$	$var3$	$var4$	$var5$
$mean(original)/mean(masked)$	0.982	0.979	0.978	0.973	0.978
$stdev(original)/stdev(masked)$	1.001	1	0.999	0.997	1.001
$kurtosis(original)/kurtosis(masked)$	1.001	1.001	1.001	0.999	1.001
$skewness(original)/skewness(masked)$	1.001	1.001	1.001	0.997	1.002

If we use b and c from defined interval and using normal distribution for masking, we obtain desired and measurable accuracy, when we work with masked data instead original data.

We continue with simple and multiple linear regression analysis. For simple linear regression analysis, we choose $var1$ as dependent variable and $var2, var3$ as independent variable respectively. Our first result is for the dependent variable $var1$ and the independent variable $var2$:

Table 3.3: Simple linear regression analysis on original data set

Coefficients :	Estimate	Std.Error	t Value	Pr(>|t|)
Intercept	-135264.30982	143950.58319	-0.94	0.35
var2	0.0305	0.05895	0.05	0.96
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual Standart error : 12800000 on 7949 degrees of freedom				
Multiple R-squared : 3.36e-07, Adjusted R-squared: -0.000125				
F-static: 0.00267 on 1 and 7949 DF, p-value: 0.959				

Table 3.4: Simple linear regression analysis on masked data set

Coefficients :	Estimate	Std.Error	t Value	Pr(>|t|)
Intercept	-137729.1288	143899.0628	-0.96	0.34
var2M	0.032	0.0589	0.05	0.966
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 12800000 on 7949 degrees of freedom				
Multiple R-squared : 3.71e-07, Adjusted R-squared: -0.000125				
F-static: 0.00295 on 1 and 7949 DF, p-value: 0.957				

We repeat simple linear regression analysis for dependent variable $var1$ and independent variable $var3$:

Table 3.5: Simple linear regression analysis on original data set

Coefficients :	Estimate	Std.Error	t Value	Pr(>|t|)
Intercept	-110287.1128	144131.0597	-0.77	0.444
var3	-0.0567	0.0328	-1.73	0.084 .
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 12800000 on 7949 degrees of freedom				
Multiple R-squared : 0.000376, Adjusted R-squared: 0.000251				
F-static: 2.99 on 1 and 7949 DF, p-value: 0.0837				

Table 3.6: Simple linear regression analysis on masked data set

Coefficients :	Estimate	Std.Error	t Value	Pr(>|t|)
Intercept	-113210.9379	144089.9644	-0.79	0.432
var3M	-0.0543	0.0327	-1.66	0.097 .
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 12800000 on 7949 degrees of freedom				
Multiple R-squared : 0.000346, Adjusted R-squared: 0.000221				
F-static: 2.76 on 1 and 7949 DF, p-value: 0.097				

We repeat simple linear regression analysis many times for randomly selected pairs, chosen from our financial data set. We always get accurate results.

Multiple lineer regression analysis is the next. We choose $var1$ as dependent variable, $var2$ and $var3$ as independent variables. We obtain the following results :

Table 3.7: Multiple linear regression analysis on original data set

Coefficients :	Estimate	Std.Error	t Value	Pr(>|t|)
Intercept	-111732.05583	144570.77643	-0.77	0.440
var2	0.00764	0.05900	0.13	0.897
var3	-0.05687	0.03280	-1.73	0.083 .
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 12800000 on 7948 degrees of freedom				
Multiple R-squared : 0.000379, Adjusted R-squared: 0.000127				
F-static: 1.5 on 2 and 7948 DF, p-value: 0.222				

Table 3.8: Multiple linear regression analysis on masked data set

Coefficients :	Estimate	Std.Error	t Value	Pr(>|t|)
Intercept	-114684.31793	144548.09886	-0.79	0.428
var2M	0.00763	0.05897	0.13	0.897
var3M	-0.05449	0.03275	-1.66	0.096 .
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 12800000 on 7948 degrees of freedom				
Multiple R-squared : 0.000349, Adjusted R-squared: 9.7e-05				
F-static: 1.39 on 2 and 7948 DF, p-value: 0.25				

We increase the number of independent variables and repeat multiple linear regression analysis for selected variables. We choose $var4$ as dependent variable and $var1$,$var2$,$var3$, and $var5$ as independent variables. We obtain the following result:

Table 3.9: Multiple linear regression analysis on original data set

Coefficients :	Estimate	Std.Error	t Value	Pr(>|t|)
Intercept	24233079.8079	353862.8486	68.48	< 0.0000000000000002 ***
var1	0.5097	0.0315	16.19	< 0.0000000000000002 ***
var2	0.1635	0.1402	1.17	0.24
var3	0.3576	0.0788	4.54	< 0.0000058 ***
var5	-1.0484	0.0524	-20.01	< 0.0000000000000002 ***
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 30300000 on 7946 degrees of freedom				
Multiple R-squared : 0.0602, Adjusted R-squared: 0.0597				
F-static: 127 on 4 and 7946 DF, p-value<0.0000000000000002				

Table 3.10: Multiple linear regression analysis on masked data set

Coefficients :	Estimate	Std.Error	t Value	Pr(>|t|)
Intercept	24922342.6657	355669.4495	70.07	< 0.0000000000000002 ***
var1M	0.5128	0.0316	16.23	< 0.0000000000000002 ***
var2M	0.1636	0.1406	1.16	0.24
var3M	0.3558	0.0790	4.50	< 0.0000068 ***
var5M	-1.0538	0.0526	-20.04	< 0.0000000000000002 ***
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 30400000 on 7946 degrees of freedom				
Multiple R-squared : 0.0603, Adjusted R-squared: 0.0599				
F-static: 128 on 4 and 7946 DF, p-value<0.0000000000000002				

When we work with masked data set instead of original data set we also obtain accurate result for simple and multiple linear regression analysis.

The next statistics is logistic regression analysis. For this statistics we define a threshold value and using this value we convert the dependent variable to categorical binary form. We use original data set to create dependent variable. Then we use masked independent variables and test the accuracy. We also check the effect of selected threshold.

We choose $var1$ as dependent variable. We set threshold value to 100000. We convert it to a categorical binary variable. For this threshold there are 4566 1's and 3385 0's. We choose arbitrarily $var4$ as independent variable. We have the following result for

simple logistic regression analysis:

Table 3.11: Simple logistic regression analysis on original data set

Coefficients :	Estimate	Std.Error	z Value	Pr(>|z|)
Intercept	-0.23642578983	0.03145121709	-7.52	0.000000000000056 ***
var4	0.00000002296	0.00000000103	22.20	< 0.0000000000000002 ***
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
(Dispersion parameter for binomial family taken to be 1)				
Null deviance :10846 on 7950 degrees of freedom				
Residual deviance :10180 on 7949 degrees of freedom				
AIC : 10184				
Number of Fisher Scoring iterations : 4				

Table 3.12: Simple logistic regression analysis on masked data set

Coefficients :	Estimate	Std.Error	z Value	Pr(>|z|)
Intercept	-0.25059155827	0.03189880025	-7.86	0.000000000000004 ***
var4M	0.00000002284	0.00000000103	22.17	< 0.0000000000000002 ***
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
(Dispersion parameter for binomial family taken to be 1)				
Null deviance :10846 on 7950 degrees of freedom				
Residual deviance :10181 on 7949 degrees of freedom				
AIC : 10185				
Number of Fisher Scoring iterations : 4				

We set threshold value to 10000000. For this threshold there are 211 1's and 7740 0's. We repeat the previous case for this threshold and obtain the following result for simple logistic regression analysis:

Table 3.13: Simple logistic regression analysis on original data set

Coefficients :	Estimate	Std.Error	z Value	Pr(>|z|)
Intercept	-4.63761299024	0.11586773811	-40.0	0.0000000000000002 ***
var4	0.00000002431	0.00000000146	16.7	< 0.0000000000000002 ***
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
(Dispersion parameter for binomial family taken to be 1)				
Null deviance :1947.9 on 7950 degrees of freedom				
Residual deviance :1702.2 on 7949 degrees of freedom				
AIC : 1706				
Number of Fisher Scoring iterations : 7				

Table 3.14: Simple logistic regression analysis on masked data set

Coefficients :	Estimate	Std.Error	z Value	Pr(>|z|)
Intercept	-4.65297663670	0.11657350341	-39.9	0.0000000000000002 ***
var4M	0.00000002422	0.00000000145	16.7	< 0.0000000000000002 ***
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
(Dispersion parameter for binomial family taken to be 1)				
Null deviance :1947.9 on 7950 degrees of freedom				
Residual deviance :1702.2 on 7949 degrees of freedom				
AIC : 1706				
Number of Fisher Scoring iterations : 7				

We see that the change of threshold doesn't effect the accuracy on analysis. Working with perturbed independent variable gives accurate result for simple logistic regression analysis. Now we increase the number of independent variables. We will use $var2$,$var3$,$var4$,and $var5$ as independent variable. We set the threshold to 1000000. We obtain the following result for multiple logistic regression analysis:

Table 3.15: Multiple logistic regression analysis on original data set

Coefficients :	Estimate	Std.Error	z Value	Pr(>|z|)
Intercept	−4.636587728160	0.115992292071	−39.97	0.0000000000000002 ***
var2	0.000000007678	0.000000021184	0.36	0.72
var3	0.000000000973	0.000000011395	0.09	0.93
var4	0.000000024407	0.000000001480	16.49	0.0000000000000002 ***
var5	0.000000003168	0.000000007371	0.43	0.67
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
(Dispersion parameter for binomial family taken to be 1)				
Null deviance :1947.9 on 7950 degrees of freedom				
Residual deviance :1701.9 on 7946 degrees of freedom				
AIC : 1712				
Number of Fisher Scoring iterations : 7				

Table 3.16: Multiple logistic regression analysis on masked data set

Coefficients :	Estimate	Std.Error	z Value	Pr(>|z|)
Intercept	−4.65187866939	0.11670117734	−39.86	0.0000000000000002 ***
var2M	0.00000000766	0.00000002119	0.36	0.72
var3M	0.00000000110	0.00000001134	0.10	0.92
var4M	0.00000002431	0.00000000147	16.49	0.0000000000000002 ***
var5M	0.00000000324	0.00000000738	0.44	0.66
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
(Dispersion parameter for binomial family taken to be 1)				
Null deviance :1947.9 on 7950 degrees of freedom				
Residual deviance :1701.9 on 7946 degrees of freedom				
AIC : 1712				
Number of Fisher Scoring iterations : 7				

For simple and multiple logistic regression analysis, working with pertubed data set instead original data gives accurate results.

As a last experiment we take log transform of the data. For this experiment we choose nonnegative data sets. We control the data sets and replace all 0's with 1. Then, we take the log of original data, then mask it. We compare results of primitive statistics and regression analysis on original and masked log transformed data.

Table 3.17: Ratio of descriptive statistics on original and additively masked log transformed data set

	$var2$	$var3$	$var4$
$mean(original)/mean(masked)$	0.9679	0.9576	0.975
$stdev(original)/stdev(masked)$	1.005	1.002	1
$kurtosis(original)/kurtosis(masked)$	1.008	1.001	1.002
$skewness(original)/skewness(masked)$	0.9977	1.001	1.001

For multiple lineer regression analysis on log transformed data, we choose $var4$ as dependent variable, $var2$ and $var3$ as independent variables. We obtain the following results:

Table 3.18: Multiple linear regression analysis on original log transformed data set

Coefficients :	Estimate	Std.Error	z Value	Pr(>|z|)
Intercept	15.27487	0.05515	276.98	$<$ 0.0000000000000002 ***
var2	0.05729	0.00896	6.40	0.00000000017 ***
var3	0.04183	0.01089	3.84	0.00012 ***
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 3.82 on 7948 degrees of freedom				
Multiple R-squared : 0.00777, Adjusted R-squared: 0.00752				
F-static: 31.1 on 2 and 7948 DF, p-value<0.0000000000000343				

Table 3.19: Multiple linear regression analysis on masked log transformed data set

Coefficients :	Estimate	Std.Error	z Value	Pr(>|z|)
Intercept	15.6547	0.0560	279.53	$<$ 0.0000000000000002 ***
var2M	0.0575	0.0090	6.39	0.00000000018 ***
var3M	0.0416	0.0109	3.81	0.00014 ***
Signif. codes : 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				
Residual standart error : 3.82 on 7948 degrees of freedom				
Multiple R-squared : 0.00772, Adjusted R-squared: 0.00747				
F-static: 30.9 on 2 and 7948 DF, p-value<0.0000000000000416				

Besides shared experiments, we made many other experiments with randomly selected variables. We obtained measurable accuracy for the statistics we mentioned.

3.2.2 Accuracy on Multiplicative Masked Data

We start with computing the descriptive statistics. For each variable we calculate the mean, standart deviation, kurtosis, and skewness both for original and masked data set.

In this section, we use the multiplicative data perturbation technique. We generate noises from normal distribution as $N(1, c \times \sigma^2)$. In the following work, we choose c as 0.025. We calculate each statistical function on original data, then on the masked data and take the ratio to see the accuracy. We obtain the following results:

Table 3.20: Accuracy of descriptive statistics on original and multiplicatively masked data set

	$var1$	$var2$	$var3$	$var4$	$var5$
$mean(original)/mean(masked)$	0.017	0.508	0.818	0.011	0.165
$stdev(original)/stdev(masked)$	0.026	0.15	0.838	0.01	0.043
$kurtosis(original)/kurtosis(masked)$	0.825	0.652	0.917	1.222	0.868
$skewness(original)/skewness(masked)$	0.727	0.395	0.557	0.811	0.733

During our experimental works we saw that if we use multiplicative masked data, we cannot guarantee accuracy. To solve this problem we have to choose c close to 0, then we face the privacy problem. Since we deal with the balance on accuracy and privacy, in the rest of our work we focus additively masked data.

CHAPTER 4

PRIVACY ANALYSIS

4.1 Attacks

We study some of existing attacks from the literature which are applicable to our masked data sets[17]. Researchers assumed the role of an attacker and developed methods for estimating the original data from the sanitized data. Their work shows the vulnerabilities of this type of data perturbation.

We define the original data set as an $p \times q$, real valued matrix X. The owner of the data perturbs X by using additive perturbation methods and release Y. The attacker uses Y and find an estimation for X, denoted by $\hat{X}$. Our first assumption is that each record of the original data set arose as an independent sample. Let $\sum_X$ denote the covariance matrix of X. The second assumption is that $\sum_X$ has all distinct and non-zero eigenvalues.

The data owner replaces the original data set X with

$$Y = X + R \tag{4.1}$$

where R is a noise matrix with each column generated independently from a p-dimensional random vector R with mean vector zero. We assume throughout that $\sum_R$ equals $\sigma^2 I$. In our case we use normal distribution.

In this chapter we describe three different attacks against additive peturbation. These are Spectral Filtering (SF) [18], Singular Value Decomposition (SVD) [19] and Principal Component Analysis (PCA) [20]. For each method, we apply the attack to our masked data sets and obtain so-called estimated data sets. Then, we measure the distance between the estimated and original values shown as $d(O, E)$ and the dis-

tance between masked and original shown as $d(O, M)$. A comparison indicator, m, is defined as:

$$m = \left[\frac{d(O, E)}{d(O, M)}\right] \tag{4.2}$$

where $d(A, B)$ is defined as:

$$d(A, B) = \sum_{i=1}^{n} |a_i - b_i| \tag{4.3}$$

If m is in $(0, 1)$, after attacking we come close to original data. If m is 1 we find masked data itself. The methods used to check the privacy are explained in detail in [17], [18], [19], and [20]. Each method is applied on a hypothetical data set which is generated using triangular and sinusoidal functions to check that it works. We then apply these methods to financial data set to show how much privacy is preserved in case of such attacks.

4.1.1 Spectral Filtering

This technique, developed by Kargupta et al. [18], utilizes the fact that the eigenvalues of a random matrix are distributed in a fairly predictable manner. The steps in applying spectral filtering are as follows:

- We calculate the covariance matrix of masked data.
- We calculate the eigenvalues and the corresponding eigenvectors.
- We calculate the boundaries for eigenvalues by using the following equations:

$$\lambda_{min} = \sigma^2(1 - \frac{1}{\sqrt{\theta}})^2, \qquad \lambda_{max} = \sigma^2(1 + \frac{1}{\sqrt{\theta}})^2 \tag{4.4}$$

where σ^2 is the variance of noise matrix and $\theta = \frac{p(rownumber)}{q(columnnumber)}$.

The attack is done by using the corresponding eigenvectors of eigenvalues greater than λ_{max}. For estimating the masked observation we use the following equation,

$$E = M \times A_0 \times A_0^T \tag{4.5}$$

where E stands for the estimation, M is released masked data set, and A_0 is the matrix includes calculated eigenvectors as columns. Afterwards, we compare the closeness to exact data set by using the measure m we mentioned.

To verify the method we generate sinusoidal and triangular data and mask these according to the proposed model. We attack the masked data sets by using spectral filtering approach whose results are presented in Table 4.1 and Figure 4.1. In these graphs, red circles are the absolute difference between original and masked data points. The black circles are the difference between the original and estimated data points. We see that after attacking we come close to original data sets.

Table 4.1: Spectral filtering method tested on sinusoidal and triangular data sets

	Sinusoidal	Triangular
λ_{min}	0.01121	0.007627
λ_{max}	0.06104	0.04152
$p \times q$	250×40	250×40
$d(O, M)$	1410	1163
$d(O, E)$	319.6	205.2
m	0.227	0.176

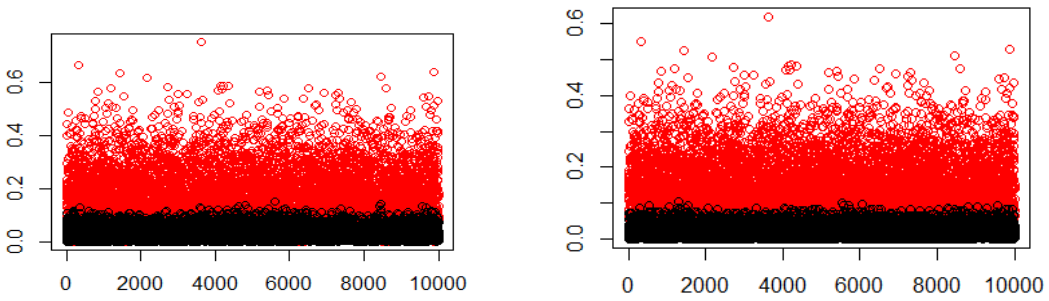

Figure 4.1: Spectral filtering method tested on sinusoidal and triangular data sets

Application of this attack method to masked financial data set yields the results presented in Table 4.2 and Figure 4.2. We see that, after attacking with spectral filtering to masked financial data set, we obtain the masked data itself. In other words, there is no disclosure of our financial data by applying this attack.

Table 4.2: Spectral filtering method tested on financial data set

	Financial Variable
λ_{min}	45870034800
λ_{max}	194658444176
$p \times q$	250×30
$d(O, M)$	1964402616
$d(O, E)$	1964402616
m	1

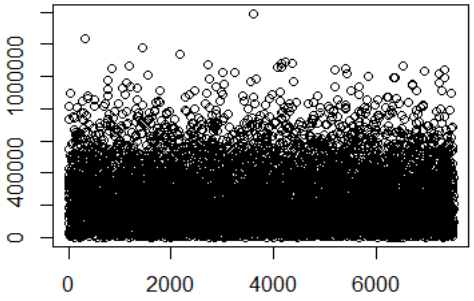

Figure 4.2: Spectral filtering method tested on financial data set

4.1.2 SVD Filtering

Guo et al. [19] proposed a singular value decomposition-based data reconstruction approach and proved the equivalence of this approach to spectral filtering. SVD is applied as following :

- We apply SVD to masked data matrix. We decompose M as $M = \tilde{L}\tilde{D}\tilde{R}^T$
- We find the singular values, $\tilde{\sigma}_1 \geq \tilde{\sigma}_2 \geq \tilde{\sigma}_3 \geq \ldots$
- We apply SVD to noise matrix and find the largest singular value, σ_v.
- We find k, $k = min\{i : \{(\tilde{\sigma}_i < \sqrt{2}\sigma_v) - 1\}\}$

• We attack by using the following equation:

$$E = \sum_{i=1}^{k} \tilde{\sigma}_i \times \tilde{l}_i \times \tilde{r}_i^T$$

We compare the closeness to exact data set by using the measure m we mentioned.

Similar to the first method, we first generate sinusoidal and triangular data and mask them. We attack the masked data sets by using SVD. The results of this attack method is shown in Table 4.3 and Figure 4.3. We see that the red circles in the graph are the absolute difference between original and masked data points. The black circles are the difference between the original and estimated data points. We see that after attacking we come close to original data sets.

Table 4.3: SVD filtering tested on sinusoidal and triangular data sets

	Sinusoidal	Triangular
$p \times q$	250×40	250×40
$d(O, M)$	1410	1163
$d(O, E)$	319.5	205.4
m	0.227	0.177

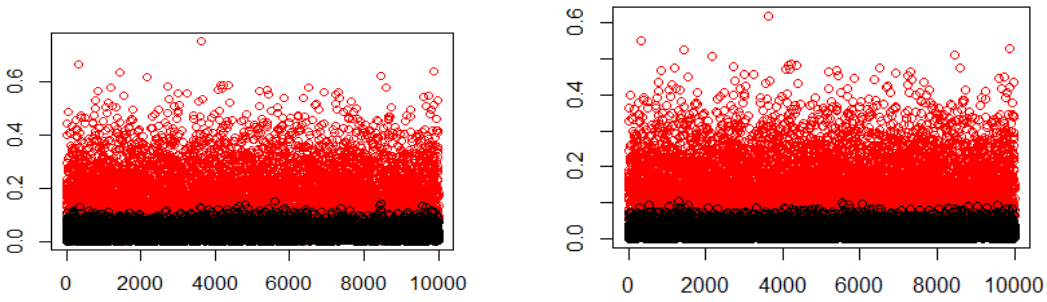

Figure 4.3: SVD filtering tested on sinusoidal and triangular data sets

Application of SVD attack method to masked financial data set yields the results presented in Figure 4.4. We see that, after attacking with SVD to masked financial data set, we obtain the masked data itself. In other words, there is no disclosure of our financial data by applying this attack.

Table 4.4: SVD filtering tested on financial data sets

	Financial Variable
$p \times q$	250×30
k	30
m	1

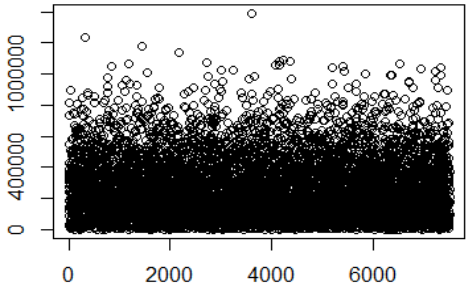

Figure 4.4: SVD filtering tested on financial data set

4.1.3 PCA Filtering

Huang et al. [20] proposed a filtering technique based on PCA(Principle Component Analysis). A major difference with spectral filtering, is that PCA filtering does not use matrix perturbation theory and spectral analysis to estimate dominant PCs of original data. PCA can be applied as following:

• We compute the mean of each column of masked matrix, then subtract it from calculated column.

• We calculate the covariance matrix of the new form of masked matrix. We produce:

$$\sum \hat{X} = \sum \hat{Y} - \sigma^2 I$$

an estimate of $\sum X$.

•We calculate the eigenvalues of $\sum \hat{X}$ and count the number of dominant eigenvalues and denote it as k.

• Using the k dominant eigenvalues, we calculate the corresponding eigenvectors.

$$\hat{V}_x = [\hat{v}_x^1 \dots \hat{v}_x^k]$$

• We attack by using the following equation

$$\hat{X} \approx Y \hat{V}_x \hat{V}_x^T$$

As a last step we compare the closeness to the exact data set by using the measure function that we mentioned.

Implementation of PCA on experimental functions are presented in Table 4.5 and Figure 4.5. We observe similar result as in other two methods. The red circles are the absolute difference between original and masked data points. The black circles are the difference between the original and estimated data points. We see that after attacking we come close to original data sets.

Table 4.5: PCA filtering tested on sinusoidal and triangular data sets

	Sinusoidal	Triangular
$m \times n$	250×40	250×40
$d(O, M)$	1408	1173
$d(O, E)$	312.2	263.2
m	0.228	0.1224

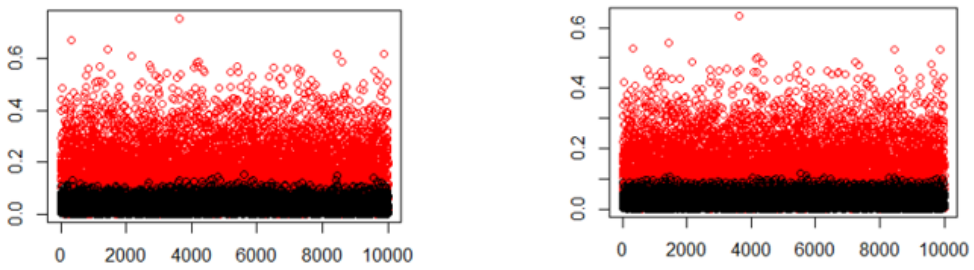

Figure 4.5: PCA filtering tested on sinusoidal and triangular data sets

Application of PCA on financial data set yields no disclosure of original data as presented in Table 4.6 and Figure 4.6.

Table 4.6: PCA filtering tested on sinusoidal and triangular data sets

	Financial variable
$p \times q$	250×30
k	30
m	1

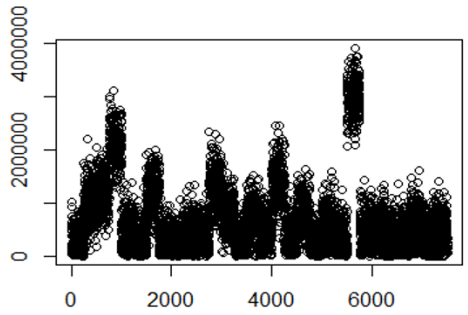

Figure 4.6: PCA filtering tested on financial data set

4.2 Some More Privacy for Data Releasing

We defined some restrictions on data releasing because some kind of requests may threat the privacy of the data.

4.2.1 Every variable must have the same masked form

We start with the following question. If we have two different masked set of same data, is there any disclosure risk?

Let X be the original data set such that:

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{4.6}$$

Assume that we have two different masked form of the same data set.

$$X^1 = \begin{bmatrix} x_1^1 \\ x_2^1 \\ \vdots \\ x_n^1 \end{bmatrix} = \begin{bmatrix} x_1 + \epsilon_1^1 \\ x_2 + \epsilon_2^1 \\ \vdots \\ x_n + \epsilon_n^1 \end{bmatrix}$$

and

$$X^2 = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \vdots \\ x_n^2 \end{bmatrix} = \begin{bmatrix} x_1 + \epsilon_1^2 \\ x_2 + \epsilon_2^2 \\ \vdots \\ x_n + \epsilon_n^2 \end{bmatrix}$$

By using X^1, and one record reduced version of X^2, we construct S and T as following:

$$\begin{aligned} S &= x_1^1 + \ldots + x_n^1 \\ &= x_1 + \ldots + x_n + \epsilon_1^1 + \ldots + \epsilon_n^1 \end{aligned}$$

and

$$\begin{aligned} T &= x_1^2 + \ldots + x_{n-1}^2 \\ &= x_1 + \ldots + x_{n-1} + \epsilon_1^2 + \ldots + \epsilon_{n-1}^2 \end{aligned}$$

We calculate $S - T$,

$$S - T = x_n + \underbrace{\epsilon_1^1 + \ldots + \epsilon_n^1}_{n \times b \times \overline{X}} - \underbrace{(\epsilon_1^2 + \ldots + \epsilon_{n-1}^2)}_{(n-1) \times b \times \overline{X}} \tag{4.7}$$

Then we simplify equation (4.7) and find the following equality,

$$x_n = S - T - b\overline{X} \tag{4.8}$$

where S, T and average of masked data are known.

Consequently, we see that if we have two different masked set of same data, there might be a disclosure risk. Therefore we propose to mask a data once for all requests, the response of the same request must be exactly same.

4.2.2 One record difference between two request

We start this section with the following question.Is there any disclosure risk, if we response two query with one record difference?

We have two query such that the result sets have only one record difference. We have the following result sets.

$$X_1 = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad and \quad X_2 = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \end{bmatrix} \tag{4.9}$$

We masked and release this two data set as following:

$X^1 = X_1 + N(b \times \mu_1, c \times \sigma_1^2)$ and $X^2 = X_2 + N(b \times \mu_2, c \times \sigma_2^2)$ where μ_1, μ_2 are the mean and σ_1^2, σ_2^2 are the variance of X_1, X_2 respectively. In vector form we have the following masked data sets,

$$X^1 = \begin{bmatrix} x_1 + \epsilon_1^1 \\ x_2 + \epsilon_2^1 \\ \vdots \\ x_n + \epsilon_n^1 \end{bmatrix} \quad and \quad X^2 = \begin{bmatrix} x_1 + \epsilon_1^2 \\ x_2 + \epsilon_2^2 \\ \vdots \\ x_{n-1} + \epsilon_{n-1}^2 \end{bmatrix} \tag{4.10}$$

Now, we define S and T such that :

$$S = x_1 + \ldots + x_n + \epsilon_1^1 + \ldots + \epsilon_n^1 \tag{4.11}$$

and

$$T = x_1 + \ldots + x_{n-1} + \epsilon_1^2 + \ldots + \epsilon_{n-1}^2 \tag{4.12}$$

We calculate $S - T$,

$$S - T = x_n + \underbrace{(\epsilon_1^1 + \ldots + \epsilon_n^1)}_{n \times b \times \mu_1} - \underbrace{(\epsilon_1^2 + \ldots + \epsilon_{n-1}^2)}_{(n-1) \times b \times \mu_2} \tag{4.13}$$

Then we substitute μ_1 and μ_2 into equation (4.13)

$$\begin{aligned} S - T = x_n + n \times b \times \frac{x_1 + \ldots + x_n}{n} \\ - (n-1) \times b \times \frac{x_1 + \ldots + x_{n-1}}{(n-1)} \end{aligned} \tag{4.14}$$

We simplify equation (4.14) and obtain the following equality.

$$S - T = x_n(1 + b) \quad \rightarrow \quad x_n = \frac{S - T}{1 + b} \tag{4.15}$$

where S and T is known and b is predictable.

Consequently, if the result set of new query has only one different record from one of the result set of the past queries we don't response.

4.3 The Inclusion of Random Number Generation in Privacy

In R-software system there are totally 2^{30} different seed values. If we use a seed value to generate all random numbers producing the noises, an attacker can recover the original data by constructing 2^{30} tables. In order to construct a table from a possible seed, the attacker generates the noises from this seed and then they are subtracted from the masked data. The tables from all other possible seeds are built similarly. Note that one of these tables is the original data. If there are n values of data, then the total size of the tables is $n2^{30}$. This amount of data can be efficiently stored in practice for the values of n used in practical applications. Therefore, while generating a masked data, a different seed value must be used for each value in the data in order to avoid such an attack. Moreover, if a masked data that was generated before is requested, the system must generate the same masked data, i.e., the same noises should be employed for generating the masked data. Otherwise, the system would be vulnerable against collusions. Under these requirements, we propose the following method described in Table 14 for noise generation. In this method, k is a key that must be kept secret by the authority generating noise. We use a function f to generate the seeds. The seed values are dependent on the original value of the data so that whenever the system gets a request of generating a masked data produced before, the same masked data will be generated. In the proposed system, we chose a nonlinear function $f(x) = \mu x^3 + \sigma$ where μ and σ are the mean and the standard deviation of the original data, respectively.

Table 4.7: Privacy algorithm using proposed random number generation

Original data	Seed	Noise	Masked data
x_1	$s_1 = f(k + x_1) \mod 2^{30}$	$\epsilon_1 = RNG(s_1)$	$x'_1 = x_1 + \epsilon_1$
x_2	$s_2 = f(x_2 + x'_1) \mod 2^{30}$	$\epsilon_2 = RNG(s_2)$	$x'_2 = x_2 + \epsilon_2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_n	$s_n = f(x_n + x'_{n-1}) \mod 2^{30}$	$\epsilon_n = RNG(s_n)$	$x'_n = x_n + \epsilon_n$

It should be remarked that μ and σ are also uncertain for the attacker. If it is easy to estimate those values for an attacker, then several more keys can be used in order to increase the privacy. In this case, we use $s_i = f(k_i + x_i) \mod 2^{30}$ for $i = 1, 2, \ldots, t$ and $s_i = f(x_i + x'_{n-1}) \mod 2^{30}$ for $i = t + 1, \ldots, n$ where t is a privacy parameter.

In practice, selecting a master key of size about 1200-bit, splitting it in 40 equal parts having each 30-bit (that is t = 40) and assigning each 30-bit to a subkey k_i will be more than enough to provide approximately a privacy level of 100-bit.

CHAPTER 5

SOFTWARE FOR MASKING FINANCIAL DATA SETS

We develop user-friendly software that enables raw data in the database to be shared with external users by perturbing it with a secure and easy to use transformation. In this software, a secure masking technique resistant to possible attacks from external users is aimed to be developed such that some specific statistical analysis results on masked data are close to the original within the defined accuracy limits $(\pm 5\%)$.The application masks the data in such a way that the researchers' results are reasonably close to the original results.

The software applies masking on the original data and/or on the logarithm of the original data. It also enables to define the dependent variable that is necessary for logistic regression analysis. Masking operation can be defined in two ways: Automatic or Manuel. While Automatic masking option masks all cells of selected data set, the manual masking option enables users to choose a subset of data, they can execute their own SQL. Moreover, masking can be performed according to two different methods: (i) Additive; (ii) Multiplicative. The additive masking technique adds generated noise to data and the multiplicative masking technique multiplies data by generated noise. For both methods, two parameters, the coefficient of the mean (b) and the coefficient of (c) must be selected. These parameters take values in the interval (0-0,05), but when these values get closer to zero the probability of occurrence of a privacy flaw rises.

When application starts User Login Screen appears. It is mandatory to be a registered user to be able to use the application. Two roles as Admin and Operator are defined.

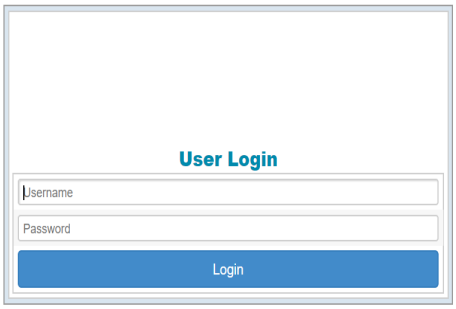

Figure 5.1: Login Page

After login by filling the User Name and Password fields, they can start to use this program.

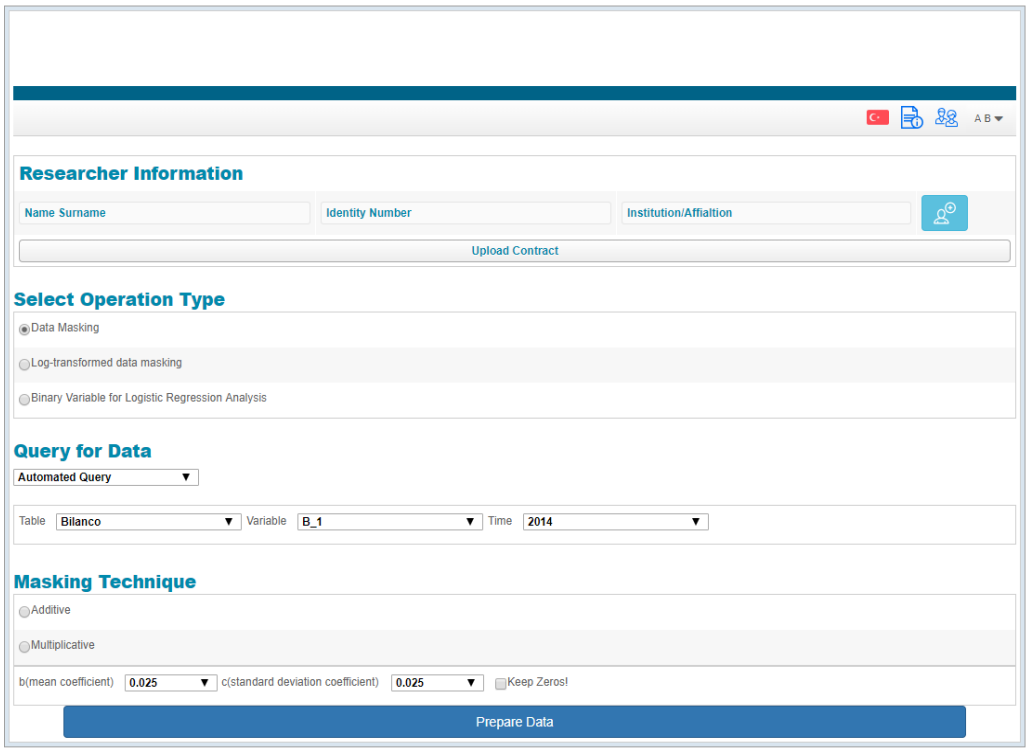

Figure 5.2: Home Page

In the Home Page on Figure 5.2, primarily the information about the researcher who made a request for data is entered and saved to database. The scanned ‘jpeg’ version of the document signed by the researcher requesting the data, is uploaded to database by using the Upload Contract button.

The software offers three different operations. These are :

• Data masking: Masking the original data

• Logarithmic data masking: Masking the logarithmic transformation of the original data

• Preparing Binary Categorical Variable for Logistic Regression: Transforming data into the values 0 and 1 according to a threshold value.

If you are admin user you can also manage the users of the program from interface. You can add, delete or update a user.

You use the user interface to prepare the data. Processes on data will be done by an R-script on the backend. In the end, you get the data as an Excel.

CHAPTER 6

CONCLUSION

In this work, we have studied additive and multiplicative masking techniques to understand the accuracy and privacy of some statistical algorithms applied to the data set. Since the multiplicative masking does not provide a reasonable accuracy we have focused on additive masking and study its properties in detail. For this purpose, we first generate our noise from a normal distribution. In doing this, the random numbers are derived from the original data using its mean and the standard deviation. We generate noise from a normal distribution and mask the original data additively which is aimed to be shared with researchers.

There are two dimensions of the problem of applying this methodology: accuracy and privacy of the masked data have to be at desired levels. For the first, we observe that the accuracy is satisfied for original and log-transformed masked data over the following statistical analyses such as descriptive statistics (mean, standard deviation, skewness, and kurtosis), simple and multiple linear regression analysis, simple and multiple logistic regression analysis on masked datasets. Some experimental results for accuracy are done and illustrated in figures. Each figure presents the accuracy obtained in implementing the additive normal perturbation to some artificial data sets. The proportion of observed data series and masked data series is expected to remain within a certain accuracy which is taken to be 5% in our case. We obtain that the results of descriptive statistics at which the mean, standard deviation, skewness and kurtosis of original data and log-transformed data remain within the target accuracy limit, respectively. Simple and multiple linear regression applied to original and masked data sets come up with the same accuracy results which verify that the masked data yield a certain accuracy in linear modeling. We repeat all experiments

many times for randomly selected variables. In these works, we obtain measurable accuracy for the statistics we mentioned.

After accuracy analysis, we deal with privacy. We study the possibility of getting the original data from the masked data. We work as an attacker. We apply the attacks to our masked data from the literature in section 3.1, which are suitable for our system. First, we verify the attacks as studied in the literature than apply our masked datasets. We see that our system is reliable against existing attacks that we discussed in detail. During privacy works, we saw a weakness in our system. Brute force attacks are a threat to our system. To get rid of this, we improved the noise generation part of our system, which we discuss in details on the section 4.3. We see also some threats, we discussed on section 4.2.1 and 4.2.2. For threats, we define some data releasing strategies. As the last part of our works, we develop a user-friendly software which includes all constraint check at the backend to prepare reliable masked data.

As future work, we can search for some new attacks and the accuracy of some new statistical functions on our masked data set.

REFERENCES

[1] A. K. Jha, C. M. DesRoches, E. G. Campbell, K. Donelan, S. R. Rao, T. G. Ferris, A. Shields, S. Rosenbaum, and D. Blumenthal, "Use of electronic health records in us hospitals," *New England Journal of Medicine*, vol. 360, no. 16, pp. 1628–1638, 2009.

[2] Y. Xu, K. Wang, A. W.-C. Fu, and P. S. Yu, "Anonymizing transaction databases for publication," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 767–775, ACM, 2008.

[3] F. K. Dankar and K. El Emam, "The application of differential privacy to health data," in *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, pp. 158–166, ACM, 2012.

[4] T. Francis, M. Madiajagan, and V. Kumar, "Privacy issues and techniques in e-health systems," in *Proceedings of the 2015 ACM SIGMIS Conference on Computers and People Research*, pp. 113–115, ACM, 2015.

[5] Y.-A. De Montjoye, L. Radaelli, V. K. Singh, *et al.*, "Unique in the shopping mall: On the reidentifiability of credit card metadata," *Science*, vol. 347, no. 6221, pp. 536–539, 2015.

[6] Y. Li, W. Wen, and G.-Q. Xie, "Survey of research on differential privacy," *Jisuanji Yingyong Yanjiu*, vol. 29, no. 9, 2012.

[7] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 571–588, 2002.

[8] H. Ye and E. S. Chen, "Attribute utility motivated k-anonymization of datasets to support the heterogeneous needs of biomedical researchers," in *AMIA Annual Symposium Proceedings*, vol. 2011, p. 1573, American Medical Informatics Association, 2011.

[9] L. Sweeney, *Computational disclosure control: a primer on data privacy protection*. PhD thesis, Massachusetts Institute of Technology, 2001.

[10] P. Samarati, "Protecting respondents identities in microdata release," *IEEE transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010–1027, 2001.

[11] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," in *22nd International Conference on Data Engineering (ICDE'06)*, pp. 24–24, IEEE, 2006.

[12] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *2007 IEEE 23rd International Conference on Data Engineering*, pp. 106–115, IEEE, 2007.

[13] P. Samarati and L. Sweeney, "Generalizing data to provide anonymity when disclosing information," in *PODS*, vol. 98, p. 188, Citeseer, 1998.

[14] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[15] W. E. Winkler, "Masking and re-identification methods for public-use microdata: Overview and research problems," in *International Workshop on Privacy in Statistical Databases*, pp. 231–246, Springer, 2004.

[16] A. Shah and R. Gulati, "Evaluating applicability of perturbation techniques for privacy preserving data mining by descriptive statistics," in *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 607–613, IEEE, 2016.

[17] K. Liu, C. Giannella, and H. Kargupta, "A survey of attack techniques on privacy-preserving data perturbation methods," in *Privacy-Preserving Data Mining*, pp. 359–381, Springer, 2008.

[18] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar, "On the privacy preserving properties of random data perturbation techniques," in *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pp. 99–106, IEEE, 2003.

[19] S. Guo, X. Wu, and Y. Li, "On the lower bound of reconstruction error for spectral filtering based privacy preserving data mining," in *Knowledge Discovery in Databases: PKDD 2006*, pp. 520–527, Springer, 2006.

[20] Z. Huang, W. Du, and B. Chen, "Deriving private information from randomized data," in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pp. 37–48, ACM, 2005.

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Bilgen, Adnan
Nationality: Turkish (TC)
Date and Place of Birth: 1975, Berlin
Phone: +90 532 333 72 10
e-mail: adnan_bilgen@hotmail.com

EDUCATION

Degree	Institution	Department	Year of Graduation
MSc	Çankaya University	Mathematics and Computer	2006
BSc	Çankaya University	Computer Engineering	2004
BSc	Çankaya University	Mathematics and Computer	2003

PROFESSIONAL EXPERIENCE

Year	Place	Enrollment
2019-	Ministry of Interior	Senior Java Software Developer
2015-19	Turkish Statistical Institute	Java Software Developer
2012-15	Private Sector	Java Software Developer
2004-12	Çankaya University Mathematics and Computer	Research Assistant

TEACHING AND TUTORING EXPERIENCE (2004-2012)

Course Code	Course Name
MATH 105	Mathematics for Business and Economics I
MATH 106	Mathematics for Business and Economics II
MATH 155	Calculus for Engineering I
MATH 156	Calculus for Engineering II
MATH 151	Calculus for Math Students I
MATH 152	Calculus for Math Students II
MATH 252	Advanced Calculus II
MATH 228	Mathematics for Industrial Engineering
MATH 244	Numerical Analysis
MATH 245	Differential Equations
MATH 258	Introduction to Differential Equations
MATH 315	Partial Differential Equations
MATH 473	Differential Geometry
MATH 281	Computer Programming II
MATH 382	Data Structures

INTERNATIONAL CONFERENCE AND WORKSHOP PARTICIPATION

Year	Event	Place
2012	The 11th Int. Workshop on Dynamical Sytems and Applications	Ankara/Turkey
2010	3rd Int. Conference on Nonlinear Science and Complexity	Ankara/Turkey
2010	New Trends in Nanotechnology and Nonlinear Dynamical Systems	Ankara/Turkey
2008	3rd Int. IFAC Workshop on Fractional Differentiation and its Applications	Ankara/Turkey
2006	Mathematical Methods in Engineering	Ankara/Turkey

PUBLICATIONS

1. E. Akyıldız, E. Başer, A. Bilgen, M. Cenk, T. Hülagü, İ. Keskinkurt-Paksoy, A. S. Selçuk-Kestel, 2019. "Data sharing under confidentiality," IFC Bulletins chapters, in: Bank for International Settlements (ed.), Are post-crisis statistical initiatives completed?, volume 49 Bank for International Settlements. <https://ideas.repec.org/h/bis/bisifc/49-34.html>

PROJECTS

1. Protection of Confidentiality in Statistical Databases, Central Bank of the Republic of Turkey, R&D Project, 2016-2018.