Irving Fisher Committee on
Central Bank Statistics

BANK FOR INTERNATIONAL SETTLEMENTS

Ninth IFC Conference on "Are post-crisis statistical initiatives completed?"

Basel, 30-31 August 2018

Data sharing under confidentiality[1]

Erdem Başer and Timur Hülagü,
Central Bank of the Republic of Turkey,

Ersan Akyıldız, Adnan Bilgen, Murat Cenk,
İrem Keskinkurt-Paksoy and A. Sevtap Selçuk-Kestel,
Middle East Technical University

[1] This paper was prepared for the meeting. The views expressed are those of the authors and do not necessarily reflect the views of the BIS, the IFC or the central banks and other institutions represented at the meeting.

Data Sharing Under Confidentiality

Ersan Akyıldız[1], Erdem Başer, Adnan Bilgen, Murat Cenk, Timur Hülagü, İrem Keskinkurt-Paksoy and A. Sevtap Selcuk-Kestel

Abstract

Central Bank of the Republic of Turkey presents an approach to address the data sharing dilemma of maximizing the benefit for academic research while ensuring compliance with applicable data confidentiality legislations. The work in this paper compares the performance of different perturbation methods. Empirical estimates are presented over a wide range of statistical methods. The results in the paper are expected to be used to inform the design of access procedures to confidential microdata in central banks.

Keywords: Data perturbation, accuracy, privacy, financial dataset

Contents

[1] Akyıldız, Bilgen, Cenk, Keskinkurt-Paksoy and Selcuk-Kestel are affiliated with Middle East Technical University while Başer and Hülagü are affiliated with the Central Bank of the Republic of Turkey. All views expressed in this paper are ours and do not necessarily represent those of the Central Bank of the Republic of Turkey or Middle East Technical University.

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. Increasing amounts of such databases 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. Collected data is used extensively by researchers and decision-makers in different fields.

However, most collected datasets contain private or sensitive information. The curators sometimes apply some simple anonymization techniques, but the adversary can destroy the privacy and re-identify the dataset. In the early years some researchers de-anonymize a medical data set by linking with another public vote list dataset [1]. The linked information is defined as the background information [2]. 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 datasets available to researchers while ensuring the confidentiality of the data. We divide our works into three parts. In the first part, we study the techniques to obtain sanitized data and test the accuracy for selected statistical functions on masked data. In the second part, we study the existing attacks and apply them to the masked dataset. In this step, we also handle the weakness of the most widely used random number generator. In the last part, we develop a user-friendly software, to generate a secure masked dataset from the original data.

Method and Accuracy Tests

In the literature, there are a couple of existing approaches on masking techniques [3,4]. We have observed that the following additive masking techniques solves our accuracy problem: 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. Figure 1 shows the masking strategies that we used in our system:

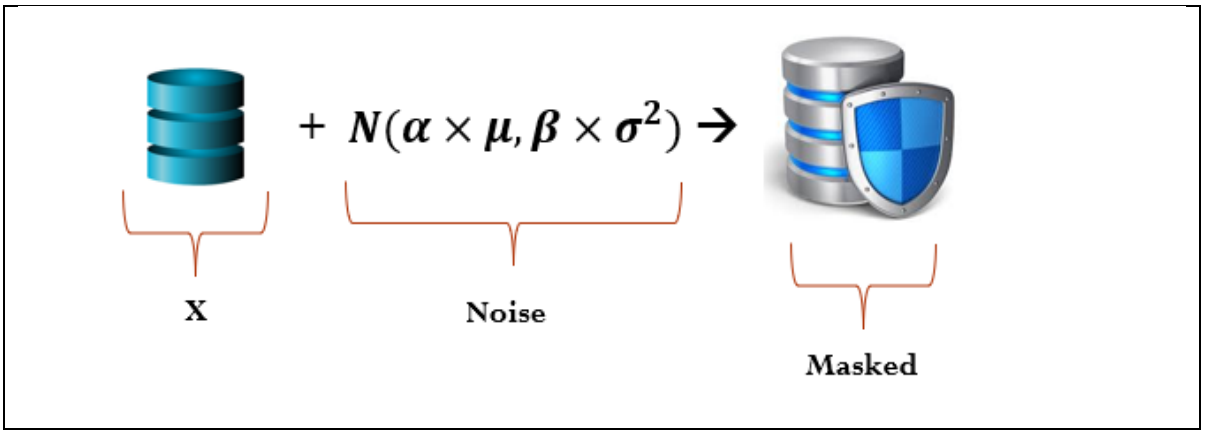


Figure 1. Masking original data set by random perturbation

Here, X denotes the original data set. 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 tables 1-5. Each table presents the accuracy obtained in implementing the additive normal perturbation to some artificial data sets. The proportion of observed (O) data series and masked (M) data series is expected to remain within a certain accuracy which is taken to be 5% in our case. Table 1 and Table 2 indicate 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 (Table 3 and Table 4) come up with the same accuracy results which verifies that the masked data yield a certain accuracy in linear modeling.

Descriptive statistic results of original series — Table 1

Variable	B59	B42	G62
Mean(O/M)	0.9777	0.9754	0.9756
Standard Deviation(O/M)	1	0.9995	1
Skewness(O/M)	1	1.001	1.002
Kurtosis(O/M)	1.001	1.001	1.006

Table 2: Descriptive statistics results for log-transformed data

Variable	B30	B50	G600
Mean(O/M)	0.9756	0.9756	0.9756
Standard Deviation(O/M)	0.9997	0.9995	0.9998
Skewness(O/M)	1.001	1.003	1.002
Kurtosis(O/M)	1.002	1.008	1.004

Given the ratios presented in Table 1 and 2 (original / masked), it was observed that the values ranged from 0.95 to 1.05. This means that the statistics of the log transformed variables remain within the specified accuracy limits.

Table 3: Simple linear regression applied to original data

Dependent Variable	Independent Variable			
G69	B40			
Original				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	1471556.82	124927.49	11.8	<0.0001
B40	0.8733	0.0156	55.9	<0.0001
Multiple R-Squared: 0.2820	Adjusted R-Squared: 0.2820			
F-statistic: 0.0031	p-value: <0.0001			
Masked				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	1509476.22	125047.53	12.1	<0.0001
MB40	0.8731	0.0156	55.9	<0.0001
Multiple R-Squared: 0.2820	Adjusted R-Squared: 0.2820			
F-statistic: 0.0031	p-value: <0.0001			

Table 4: Multiple Linear Regression applied to original data

Dependent Variable	Independent Variable			
G69	G600, G601, B300, B40, G66, B590			
Original				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	40987.73	135028.06	0.30	0.7615
G600	0.02230	0.00343	6.50	<0.0001
G601	0.02839	0.00748	3.80	0.0002
B300	-0.06936	0.00808	-8.58	<0.0001
B40	-0.06358	0.00228	-27.85	<0.0001
G66	0.63973	0.01628	39.30	<0.0001
B590	1.12325	0.02196	51.15	<0.0001
Multiple R-Squared: 0.5230	Adjusted R-Squared: 0.5230			
F-statistic: 0.0015	p-value: <0.0001			

Masked				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	41896.84	136612.71	0.31	0.75909
MG600	0.02225	0.00343	6.49	<0.0001
MG601	0.02839	0.00748	3.80	0.0002
MB300	-0.06858	0.00808	-8.49	<0.0001
MB40	-0.06347	0.00228	-27.85	<0.0001
MG66	0.64025	0.01627	39.35	<0.0001
MB590	1.12128	0.02197	51.04	<0.0001
Multiple R-Squared: 0.5230	Adjusted R-Squared: 0.5230			
F-statistic: 0.0015	p-value: <0.0001			

In addition to linear regression, binary response variable case is considered and given a presumed threshold value the simple and multiple logistic regression analyses are repeated with the same approach. Tables 5 and 6 show that the estimates and the tests statistics are close and the masked data can be used in logistic regression modeling.

Table 5: Simple Logistic Regression applied to original data

Dependent Variable	Independent Variable			Threshold = 200000
B590	G600			
Original				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	0.3622	0.0070	51.8	<0.0001
G600	0.4753e-8	0.0171e-8	27.8	<0.0001
Multiple R-Squared: 0.0885	Adjusted R-Squared: 0.0884			
F-statistic: 772	p-value: <0.0001			
Masked				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	0.3591	0.0070	50.9	<0.0001
MG600	0.4753e-8	0.0171e-8	27.8	<0.0001
Multiple R-Squared: 0.0885	Adjusted R-Squared: 0.0884			
F-statistic: 772	p-value: <0.0001			

Table 6: Multiple Logistic Regression applied to original data

Dependent Variable	Independent Variable			Threshold = 200000
B590	G600, G601, G67			
Original				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	0.3389	0.0071	47.91	<0.0001
G600	0.4674e-8	0.0170e-8	27.49	<0.0001
G601	0.5681e-8	0.0385e-8	14.77	<0.0001
G67	0.0196e-8	0.1179e-8	0.17	0.8700
Multiple R-Squared: 0.1130	Adjusted R-Squared: 0.1130			
F-statistic: 337	p-value: <0.0001			
Masked				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	0.3354	0.0071	46.90	<0.0001
MG600	0.4672e-8	0.0170e-8	27.48	<0.0001
MG601	0.5666e-8	0.0385e-8	14.74	<0.0001
MG67	0.0212e-8	0.1179e-8	0.18	0.8700
Multiple R-Squared: 0.1130	Adjusted R-Squared: 0.1120			
F-statistic: 337	p-value: <0.0001			

Most of the econometric and financial data require logarithmic transformations. For such case, we test if the proposed approach gives the same accuracy. The log transformed original series is masked and the same statistical analyses are performed. As it is presented in Table 2 the descriptive statistics are found to remain within the accuracy level. To apply the linear regression models, we assume the regression model is as follows:

$$\log\left(Dependent\ Variable\right) = \alpha_0 + \alpha_1 \log(Independent\ Variable_1) + \cdots + \alpha_k \log(Independent\ Variable_k) + \varepsilon$$

The simple linear regression (Table 7) and multiple linear regression (Table 8) analyses show that after masking the log-transformed original data, the regression models stay in the accuracy bounds compared to the results obtained using original-log-transformed data sets.

Table 7: Simple Linear Regression on log-transformed data

Dependent Variable	Independent Variable			
G590L	G64L			
Original				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	9.4150	0.1748	53.87	<0.0001
G600	-0.0190	0.0143	-1.33	0.1800
Multiple R-Squared: 0.0002	Adjusted R-Squared: 9.78e-5			
F-statistic: 1.78	p-value: 0.1830			
Masked				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	9.6540	0.1784	54.10	<0.0001
MG600	-0.0191	0.0143	-1.34	0.1800
Multiple R-Squared: 0.0002	Adjusted R-Squared: 9.97e5			
F-statistic: 1.79	p-value: 0.1810			
We found that the results applied to the original log transformed and masked data are close within 5% limits and are compatible in statistical tests.				

Table 8: Multiple Linear Regression applied to log-transformed

Dependent Variable	Independent Variable			
B590L	G64L, B32L, B400L			
Original				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	5.5555	0.3996	13.90	<0.0001
G64L	-0.0207	0.0141	-1.47	0.1400
B32L	0.3401	0.0250	13.60	<0.0001
B400L	-0.0874	0.0098	-8.95	<0.0001
Multiple R-Squared: 0.0295	Adjusted R-Squared: 0.0292			
F-statistic: 80.6	p-value: <0.0001			
Masked				
Coefficients:				
	Estimate	Std.Error	t-value	p-value
(Intercept)	5.6829	0.4093	13.88	<0.0001
G64L	-0.0207	0.0141	-1.47	0.1400
B32L	0.3408	0.0250	13.63	<0.0001
B400L	-0.0871	0.0098	-8.91	<0.0001
Multiple R-Squared: 0.0296	Adjusted R-Squared: 0.0292			
F-statistic: 80.7	p-value: <0.0001			
We found that the results are within the 5% limits of the original log transformed and masked cases and that they are also compatible with the statistical tests.				

Privacy Tests

For privacy, we study some of existing attacks in the literature which are applicable to our datasets [5]. These are spectral filtering, singular value decomposition and principal component analysis. 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 to original values shown as d(O,E) and the distance between masked to original shown as d(O,M). A comparison indicator, m, is defined as

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

where

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

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. Each method is applied firstly a hypothetical data set which are generated using triangular and sinusoidal functions based on the study done in [5] for justification of the methods to be functional in detecting the attacks. Afterwards, these methods are applied to the financial data set to show how much privacy is preserved in case of such attacks.

Spectral Filtering

This technique, developed by Kargupta et al. [6], 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:

i. We calculate the covariance matrix of masked data.

ii. We calculate the eigenvalues and the corresponding eigenvectors.

iii. We calculate the boundaries for eigenvalues by using the following equations:

$$\boldsymbol{\lambda_{min}} = \boldsymbol{\sigma^2}\left(\mathbf{1} - \frac{\mathbf{1}}{\sqrt{\boldsymbol{\theta}}}\right)^2 , \qquad \boldsymbol{\lambda_{max}} = \boldsymbol{\sigma^2}\left(\mathbf{1} + \frac{\mathbf{1}}{\sqrt{\boldsymbol{\theta}}}\right)^2 \text{ where } \boldsymbol{\theta} = \frac{\boldsymbol{m}(\textbf{row numbers})}{\boldsymbol{n}(\textbf{column numbers})}$$

The attack is done by using the corresponding eigenvectors of eigenvalues greater than $\boldsymbol{\lambda_{max}}$. For estimating the masked observation

$$\boldsymbol{E = M \times A_O \times A_O^T}$$

where E stands for is the estimation, M is the released masked data set, and A_O is the matrix with calculated eigenvectors as columns. Afterwards, we compare the closeness to the exact dataset by using

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

To verification of 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 9 and Figure 2. 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 9: Spectral filtering method tested on sinusoidal and triangular data sets

	Sinusoidal	Triangular
λ_{min}	0,01121	0,007627
λ_{max}	0,06104	0,04152
m x n	250x40	250x40
$d(O,M)$	1410	1163
$d(O,E)$	319,6	205,2
m	0,227	0,176

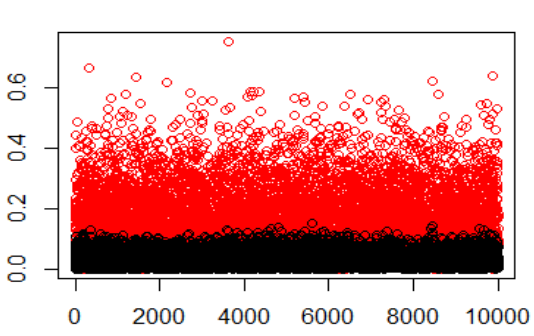
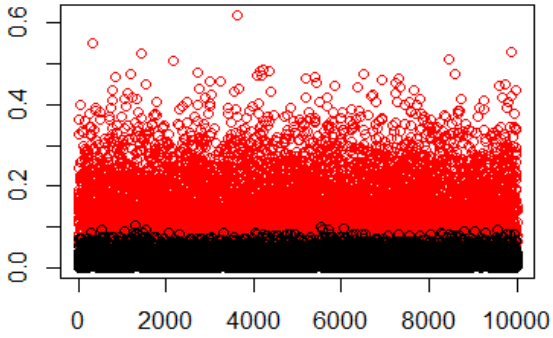
Sinusoidal	Triangular
	

Figure 2: 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 10 and Figure 3. We see that, after attacking with spectral filtering to our masked financial dataset, we obtain the masked data itself. In other words, there is no disclosure of our financial data by applying this attack.

Table 10: Spectral filtering method tested on financial data set

	G69
$\boldsymbol{\lambda_{min}}$	45870034800
$\boldsymbol{\lambda_{max}}$	194658444176
m x n	250x30
$\mathbf{d(O, M)}$	1964402616
$\mathbf{d(O, E)}$	1964402616
m	1

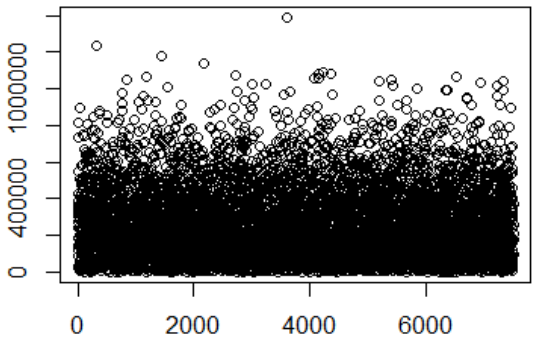


Figure 3: Spectral filtering method tested on sinusoidal and triangular data sets

Singular Value Decomposition(SVD)

Guo et al. [6] proposed a singular value decomposition-based data reconstruction approach and proved the equivalence of this approach to spectral filtering. SVD is applied as following:

a. We apply SVD to masked data matrix

b. $\mathrm{M} = \tilde{\mathrm{L}}\tilde{\mathrm{D}}\tilde{\mathrm{R}}^{\mathrm{T}}$

c. and we find the singular values, $\widetilde{\sigma_1} \geq \widetilde{\sigma_2} \geq \widetilde{\sigma_3} \geq \cdots$

d. We apply SVD to noise matrix and find the largest singular value, σ_V .

e. We find k,

f. $k = \min\{i : \{(\tilde{\sigma}_i < \sqrt{2}\sigma_v) - 1\}\}$

g. We attack by using the following equation,

h. $E = \hat{O}_k = \sum_{i=1}^{k} \tilde{\sigma}_i \times \tilde{l}_i \times \tilde{r}_i^T$

i. Then we compare the closeness to the exact dataset

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

Similar to the first method, we first generate sinusoidal and triangular data and maske them. We attack the masked data sets by using SVD. The results of this attack method is shown in Table 11 and Figure 4. 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 11: Singular value decomposition method tested on sinusoidal and triangular data sets

	Sinusoidal	Triangular
m x n	250x40	250x40
$\boldsymbol{d(O,M)}$	1410	1163
$\boldsymbol{d(O,E)}$	319,5	205,4
m	0,227	0,177

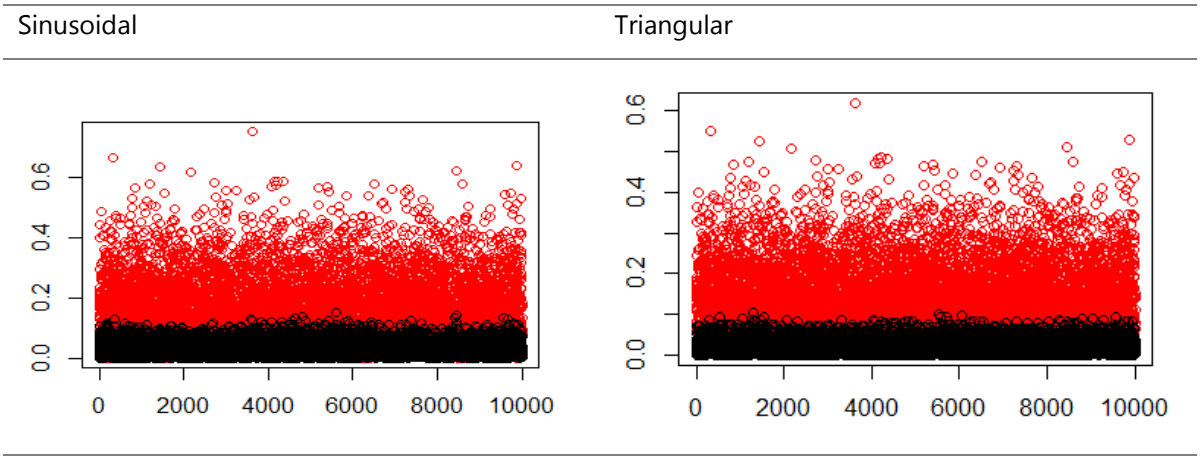


Figure 4: Singular value decomposition method tested on sinusoidal and triangular data sets

Then, we apply SVD to our masked financial dataset and see that attacking with SVD approach to our masked financial data set, we obtain masked data itself. There is no disclosure on our financial data

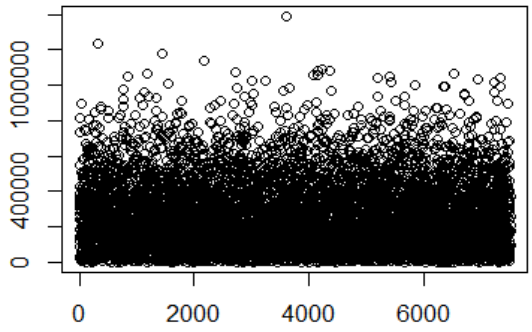


Figure 5: Singular value decomposition method tested on sinusoidal and triangular data sets

.

Principle Component Analysis (PCA)

Huang et al. [7] proposed a filtering technique based on PCA. 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:

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

b. We calculate the covariance matrix of modified masked matrix. We produce:

c. $\sum \hat{X} = \sum \hat{Y} - \sigma^2 I$ an estimate of $\sum X$

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

e. From the k dominant eigenvalues, we calculate the corresponding eigenvectors.

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

g. We attack by using the following equation

h. $\hat{X} \approx Y\hat{V}_X\hat{V}_X^T$

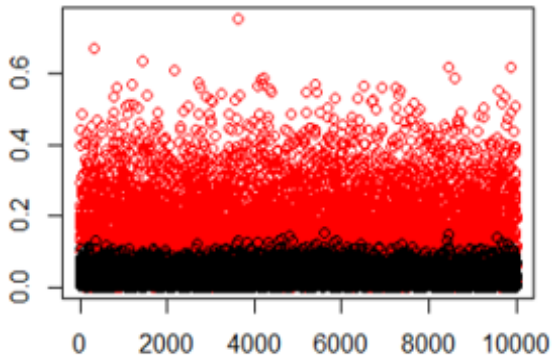
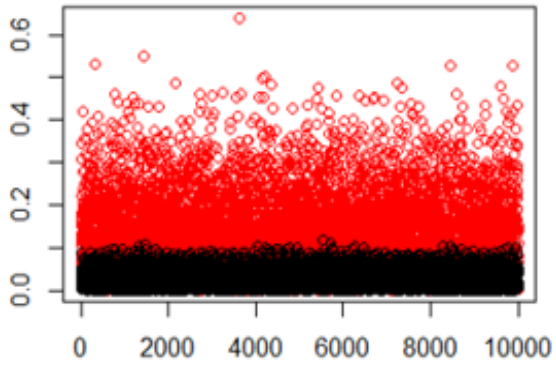
i. Then we compare the closeness to the exact dataset

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

Implementation of PCA on experimental functions are presented in Table 12 and Figure 6. 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 12: Principal component analysis method tested on sinusoidal and triangular data sets

	Sinusoidal	Triangular
m x n	250x40	250x40
$\boldsymbol{d(O,M)}$	1408	1173
$\boldsymbol{d(O,E)}$	312,2	263,2
m	0,228	0,1224

Sinusoidal	Triangular
	

.

Figure 6: Principal component analysis method tested on sinusoidal and triangular data sets

Application of PCA on financial data set yields no disclosure of original data as presented in Table 13.

Table 13: Principal component analysis method tested on sinusoidal and triangular data sets

	G69
m x n	250x30
m	1

The Inclusion of Random Number Generation in Privacy

In R-software system that we plan to use in the implementation, 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 14: Privacy algorithm using proposed random number generation

Original data	Seed	Noise	Masked data
x_1	$s_1 = f(k + x_1) \bmod 2^{30}$	$\mathcal{E}_1 = RNG(s_1)$	$x'_1 = x_1 + \mathcal{E}_1$
x_2	$s_2 = f(x_2 + x'_1) \bmod 2^{30}$	$\mathcal{E}_2 = RNG(s_2)$	$x'_2 = x_2 + \mathcal{E}_2$
□	□	□	□
x_n	$s_n = f(x_n + x'_{n-1}) \bmod 2^{30}$	$\mathcal{E}_n = RNG(s_n)$	$x'_n = x_n + \mathcal{E}_n$

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 security. In this case, we use $s_i = f(k_i + x_i) \bmod 2^{30}$ for $i=1,2,\ldots,t$ and $s_i = f(x_i + x'_{n-1}) \bmod 2^{30}$ for $i=t+1,..n$ where t is a security 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 security level of 100-bit.

Conclusion and Future Works

We achieve measurable accuracy on masked data. We observed that our system is secure for the attacks we studied. As a result, for the statistical functions we mentioned, we satisfy the privacy-accuracy balance.

As future work, we study two new attacks and the accuracy of some new statistical functions on our masked data set. Finally, we will produce a user-friendly software product to produce confidential data.

References

[1] P. Samarati and L. Sweeney Generalizing data to provide anonymity when disclosing information. page 188. 1998.
[2] L. Sweeney k-anonymity : A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(5):557-570, 2002.

[3] Josep Domingo-Ferrer francesc Seb'e and Jordi Castell'a-Roca On the Security of Noise Addition for Privacy in Statistical Databases, LNCS 3050, pp. 149–161, 2004.

[4] Jay J. Kim and William E. Winkler Multiplicative Noise for Masking Continuous Data, Article Research Report Series, January 2003

[5] Liu, K., Giannella, C., & Kargupta, H. A survey of attack techniques on privacy-preserving data perturbation methods, Privacy-Preserving Data Mining, 359-381, 2008.

[6] Hillol K., Souptik D., Qi W., Krishnamoorthy S. On the privacy Preserving Properties of Random Data Perturbation Techniques, Baltimore, Maryland, USA, 2008.

[7] Zhengli Huang, Wenliang Du and Bia Chen Deriving Private Information from Randomized Data, Baltimore, Maryland, USA, 2005.

Irving Fisher Committee on
Central Bank Statistics

BANK FOR INTERNATIONAL SETTLEMENTS

Ninth IFC Conference on "Are post-crisis statistical initiatives completed?"

Basel, 30-31 August 2018

Data sharing under confidentiality[1]

Erdem Başer and Timur Hülagü,
Central Bank of the Republic of Turkey,

Ersan Akyıldız, Adnan Bilgen, Murat Cenk,
İrem Keskinkurt-Paksoy and A. Sevtap Selçuk-Kestel,
Middle East Technical University

[1] This presentation was prepared for the meeting. The views expressed are those of the authors and do not necessarily reflect the views of the BIS, the IFC or the central banks and other institutions represented at the meeting.

TÜRKİYE CUMHURİYET MERKEZ BANKASI

Data Sharing Under Confidentiality: CBRT Case

Timur Hülagü, Ph. D.

AUGUST 30, 2018 | BASEL

Disclaimer

This is a joint project between CBRT and METU. All views expressed here are those of authors and do not necessarily reflect those of the two institutions.

Motivation

- **Main Goal**: Address the growing need of accessing micro data for academic research

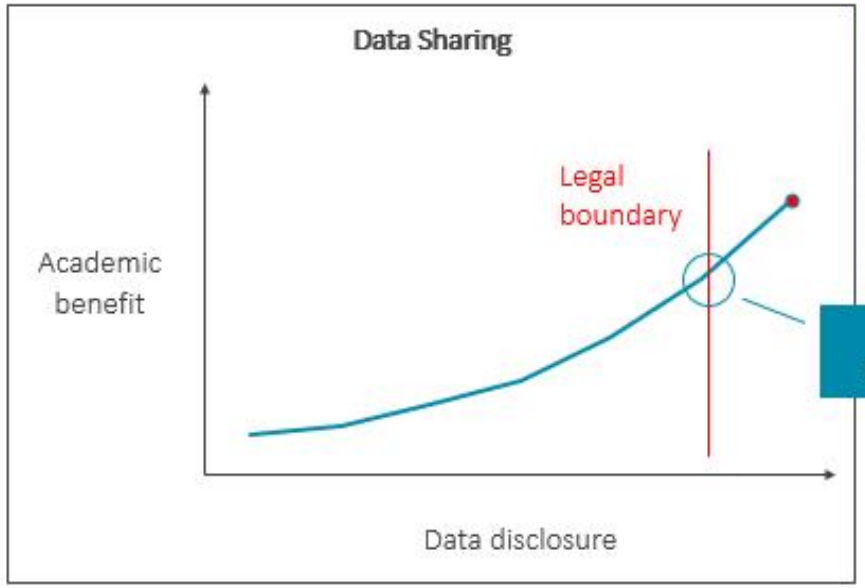
- **G-20 Data Gaps Initiative 2, Recommendation II.20:** Promotion of Data Sharing by G-20 Economies

Share information and ideas on ways to apply confidential rules/arrangements in a manner that would allow sharing of more granular data

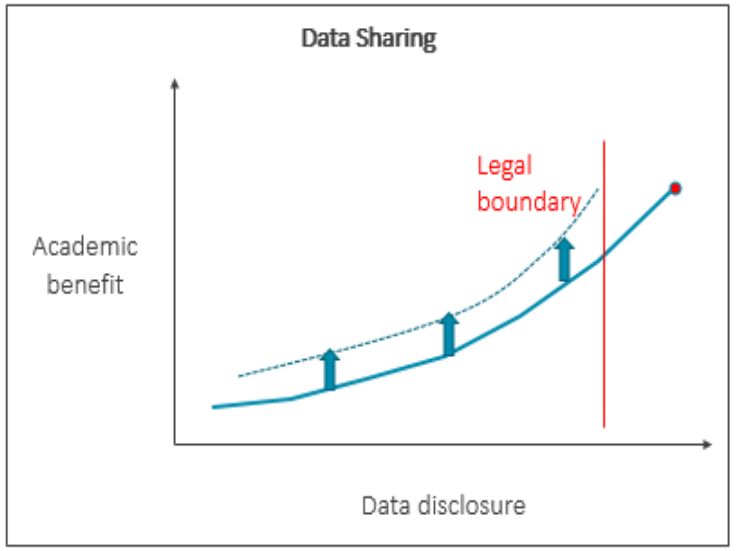
- **Eurostat Peer review report on the compliance with the Code of Practice and the coordination role of the National Statistical Institute in Turkey**

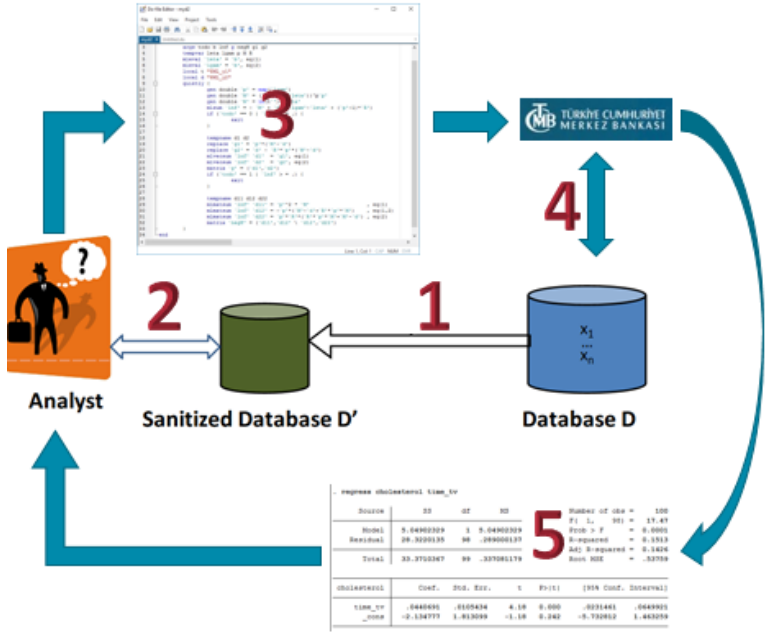
Recommendation 22: TurkStat should introduce remote access facilities for researchers, who are permitted to use its anonymized microdata for research purposes (European Statistics Code of Practice, indicator 15.4)

Data Sharing Trade-off

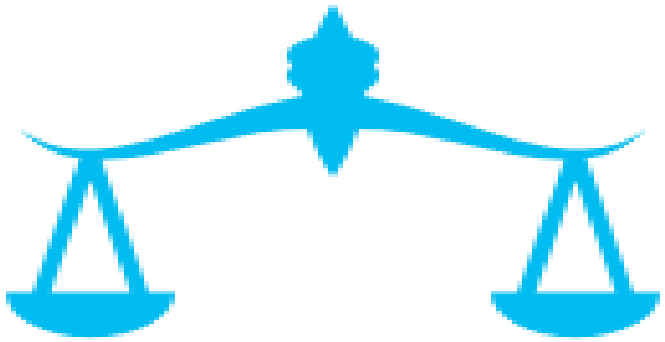


CBRT Practice





Main Aspects



Accuracy **Privacy**

Accuracy

- Descriptive Analysis ☑
- Univariate Regression Analysis ☑
- Multivariate Regression Analysis ☑
- Logistic Regression ☑
- Logarithmic Regression ☑

Accuracy

Descriptive analysis

Variable	B30	B50	B15	G600
Seed	123000	234000	345000	456000
Mean (O/M)	0.9756	0.9756	0.9756	0.9756
Standart Deviation (O/M)	0.9997	0.9995	0.9999	0.9998
Skewness (O/M)	1.001	1.003	1.002	1.002
Kurtosis (O/M)	1.002	1.008	1.004	1.004

Multiple Linear Regression Analysis

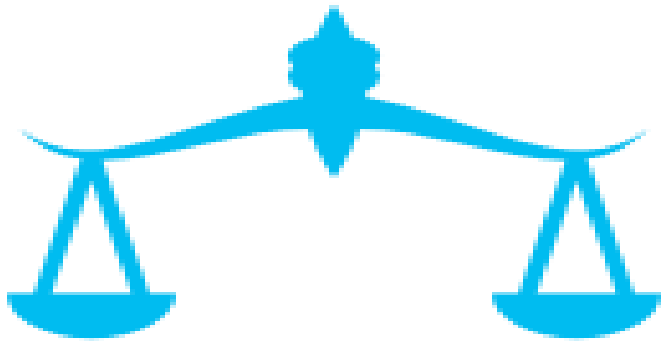
Original

```
Coefficients:
            Estimate Std. Error t value             Pr(>|t|)
(Intercept)  8.08888    0.11373   71.13 < 0.0000000000000002 ***
B2L          0.04716    0.00616    7.66    0.000000000000022 ***
B10L         0.21006    0.00486   43.22 < 0.0000000000000002 ***
B32L         0.32332    0.00583   55.44 < 0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.856 on 7342 degrees of freedom
Multiple R-squared:  0.511,     Adjusted R-squared:  0.511
F-statistic: 2.56e+03 on 3 and 7342 DF,  p-value: <0.0000000000000002
```

Masked

```
Coefficients:
            Estimate Std. Error t value             Pr(>|t|)
(Intercept)  8.28657    0.11670   71.01 < 0.0000000000000002 ***
B2L_M        0.04777    0.00617    7.75    0.000000000000011 ***
B10L_M       0.20994    0.00486   43.16 < 0.0000000000000002 ***
B32L_M       0.32314    0.00584   55.36 < 0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.857 on 7342 degrees of freedom
Multiple R-squared:  0.51,      Adjusted R-squared:  0.51
F-statistic: 2.55e+03 on 3 and 7342 DF,  p-value: <0.0000000000000002
```

Main Aspects



Accuracy **Privacy**

Privacy

- Deeper Focus on Privacy and Security

i. Spectral Filtering
ii. Singular Value Decomposition
iii. Principal Component Analysis

Attacks

$$\mathrm{m} = \left[\frac{d(O,E)}{d(O,M)}\right] \quad \text{where} \quad d(A,B) = \sum_{i=1}^{n} |a_i - b_i|$$

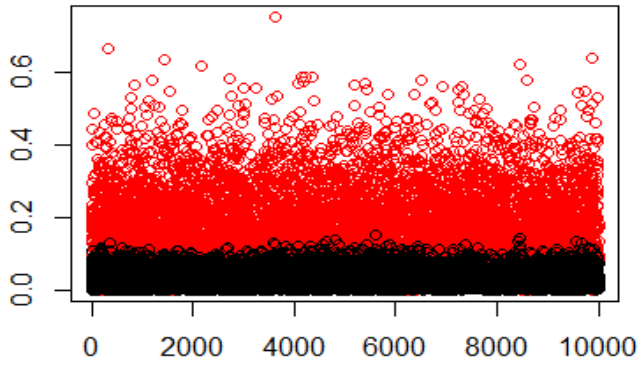
Privacy

Spectral Filtering

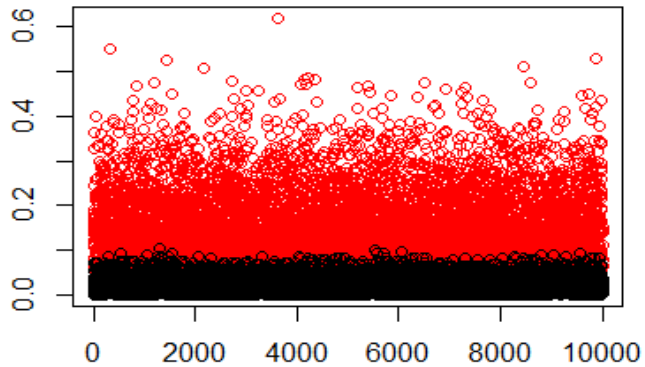
	Sinusoidal	Triangular
λ_{min}	$0,01121$	$0,007627$
λ_{max}	$0,06104$	$0,04152$
m x n	250x40	250x40
k	2	1
$d(O,M)$	1410	1163
$d(O,E)$	$319,6$	$205,2$
m	$0,227$	$0,176$

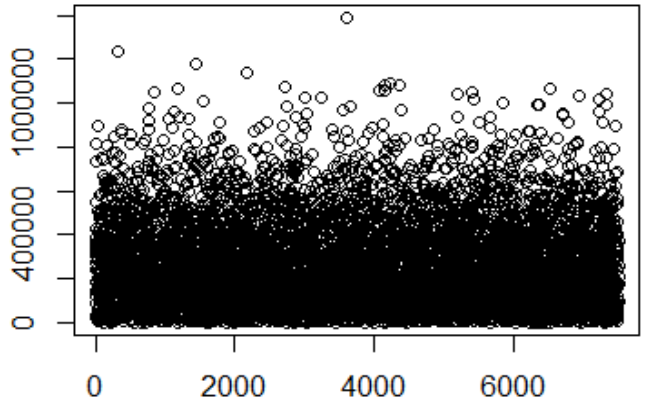
	G69
λ_{min}	45870034800
λ_{max}	194658444176
m x n	250x30
k	30
$d(O,M)$	1964402616
$d(O,E)$	1964402616
m	1

Sinusoidal



Triangular





No disclosure

Privacy

Random Number Generation In Our System

We can use 2^{30} different seed for generating random numbers. So that, brute force attacks may be a threat. To solve this problem we offer the following algorithm.

Original Data	Seed	Noise	Masked Data
x_1	$s_1 = f(\mathrm{IV} + x_1) mod\ 2^{30}$	$\varepsilon_1 = \mathrm{RNG}(s_1)$	$x'_1 = x_1 + \varepsilon_1$
x_2	$s_2 = f(x_2 + x'_1) mod\ 2^{30}$	$\varepsilon_2 = \mathrm{RNG}(s_2)$	$x'_2 = x_2 + \varepsilon_2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_n	$s_n = f(x_n + x'_{n-1}) mod\ 2^{30}$	$\varepsilon_n = \mathrm{RNG}(s_n)$	$x'_n = x_n + \varepsilon_n$

We choose nonlinear f(x), such that : $f(x) = \mu x^3 + \sigma$

Review

What is done

- We achieve, measurable accuracy on masked data.
- We observed that our system is secure for the attacks we mentioned.

Future Works

- We will study two new attacks called map estimation and distribution analysis.
- In the masked data set, we will check the accuracy of other statistical functions.
- Finally, we will produce a user friendly software product developed by using Java and R.