Secure statistical computation system on encrypted data:
An empirical study of secure regression analysis for official statistics

Satoshi Tanaka[*], Yutaka Abe[**], Satoshi Takahashi[*], Ryo Kikuchi[*], Atsushi Doi[*], Koji Chida[*], Kiyomi Shirakawa[***]

[*] NTT Secure Platform Laboratories, Tokyo, Japan, tanaka.s@lab.ntt.co.jp

[**] National Statistics Center, Tokyo, Japan, yabe3@nstac.go.jp

[***] Institute of Economic Research, Hitotsubashi University, Tokyo, Japan, kshirakawa@ier.hit-u.ac.jp

Abstract. A novel trial of on-site service to use census microdata for research purpose has been started in Japan since January 2017. The on-site service allows researchers to analyze census microdata in the limited remote areas such as satellite centers in research institutes cooperating with Japan National Statistics Center (JNSC). However, since satellite centers are far to visit and completely separated from the Internet to avoid the risk of data leakage, this trial may not be so much useful for some researchers who cannot visit. The aim of our research is to construct secure analysis environment in which researchers can analyze census microdata from the convenient places such as their office via the Internet while avoiding the risk of data leakage. In this study, we focus on the *secure computation* technology, which enables to analyze the encrypted data without decryption. This method can keep microdata secret from not only eavesdroppers but also internal illegal managers since they can see only the encrypted data, and only the analysis result is provided to the researchers. Therefore, we expect that the secure computation technology can break the restriction of on-site service using remote access.

As a first step of our study, we implemented the framework for secure statistical analysis on encrypted data by using the current secure computation system. The researchers can compute aggregation and linear regression in this framework. The linear regression cannot be computed directly in the current secure computation system, and we therefore propose another approach to compute it. We also evaluated the practicality of our framework through an experiment using Synthetic Microdata for Educational Use provided by JNSC. As a result, we confirm that our framework can compute aggregation and linear regression with the accurate results within a few seconds.

1 Introduction

Recently, the secondary use of microdata has been started. There are several approaches for providing microdata to researchers. One is providing anonymized microdata but this approach tends to have a problem about the accuracy of the analysis results. Another approach is providing microdata in the limited areas. Since January 2017, a trial of on-site service to use census microdata for research purpose has been started in Japan. The on-site service enables researchers to analyze the microdata in the limited areas such as satellite centers located in research institutes cooperating with Japan National Statistics Center (JNSC). The satellite centers are completely separated from the Internet to avoid the risk of data leakage. Although this approach allows researchers analyze microdata, the researchers need to visit the satellite centers whenever s/he wants to analyze microdata. Therefore, it may not be so much useful for some researchers who are far from any satellite centers.

We aim at the construction of secure analysis environment in which researchers can analyze the microdata from convenient places via the Internet while avoiding the risk of data leakage. To achieve this aim, we focus on *secure computation* technology, which is a method to analyze data while the data is encrypted. We consider the framework of secure statistical analysis on encrypted data by using secure computation as follows. The microdata is encrypted and registered in the servers that are connected to the Internet. When a researcher requests an analysis on microdata, the servers obtain the *encrypted* results of the analysis on each server using secure computation, and send them to the researcher. The researcher decrypts them and obtains the analysis result. Throughout of the above steps, microdata is encrypted from beginning to end. Therefore, not only eavesdroppers but also even internal illegal managers cannot obtain any information about microdata, and thus we expect that secure computation technology can break the restriction of on-site service using remote access.

On the other hand, the drawback of using secure computation technology is its efficiency. The analysis in the secure computation costs a long time compared to the case microdata is note encrypted. This drawback can be avoided by using a tailor-maid algorithm suitable for the secure computation, and basic statistics, such as average and variance, can be computed efficiently [6].

1.1 Our contribution

As a first step of our research, we implemented the framework for secure statistical analysis on encrypted data by using a current secure computation system. In this paper, we consider the analysis scenario in which frequency/magnitude table, linear regression, and Chow test are computed. Although frequency and magnitude tables can be computed in the current system, computing linear regression and the Chow test is not supported by the current system. Therefore, we explain how to compute them in secure computation.

We evaluated the practicality of our framework through an experiment with Synthetic Microdata for Educational Use, provided by JNSC. As a result, we confirm that our framework can compute aggregation and linear regression with the accurate results within a few second.

1.2 Related works

Bogdanov et al. [3] proposed secure computation protocols for inverse matrix calculation using a determinant, Gaussian elimination, and conjugate gradient method. These protocols can be components of secure computation for linear regression analysis. They showed an experimental result that a secure computation for linear regression analysis using 10,000 records data with single attribute can be worked about 3.8 seconds. Lu et al. [13] proposed a secure computation protocol for inverse matrix based on fully homomorphic encryption. Their implementation system requires about 870 seconds to compute linear regression analysis using 32,651 records data with six attributes.

2 Preliminaries

Let $\mathbf{x}$ be a vector of length N, $\mathbf{X}$ be a matrix, and $||$ denote the concatenation of vectors. In this section, we introduce linear regression analysis, structural change test, and secure computation.

2.1 Linear regression

Linear regression of $\mathbf{y}$ on $\mathbf{x}_1, \ldots, \mathbf{x}_k$ predicts $\mathbf{y}$ by a linear combination of $\mathbf{x}_1, \ldots, \mathbf{x}_k$ with coefficients $w_0, ..., w_k$ as

$$\hat{\mathbf{y}} := w_0 + \sum_{i=1}^{k} w_i \mathbf{x}_i. \tag{1}$$

Although the actual value of $\mathbf{y}$ is not the same as $\hat{\mathbf{y}}$, we can choose the coefficients in order to minimize an error between $\mathbf{y}$ and $\hat{\mathbf{y}}$.

In linear regression, the sum of squared residuals (SSR) is used as the error function. SSR is defined as

$$\sum_{j=1}^{N} (\hat{y}_j - y_j)^2, \tag{2}$$

It is known that we can choose the coefficients $w_0, \ldots, w_k$ that minimize SSR. Let $\mathbf{X}$ be the matrix whose rows are $\mathbf{1}, \mathbf{x}_1, \ldots, \mathbf{x}_k$ in order; namely, $\mathbf{X} = (\mathbf{1}||\mathbf{x}_1||\mathbf{x}_2|| \cdots ||\mathbf{x}_k)$. Let $\mathbf{w} := (w_0, \ldots, w_k)$. The coefficients minimizing SSR can be computed as

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{3}$$

In the special case of $k = 1$, w_0 and w_1 can be computed by the following equations.

$$w_1 = \frac{\sigma_{xy}}{\sigma_x^2}, \qquad w_0 = \mu_y - w_1\mu_x$$

where μ_x and σ_x^2 are the average and variance of $\mathbf{x}$, and σ_{xy} is the covariance of $\mathbf{x}$ and $\mathbf{y}$. This special case of linear regression is called *simple* linear regression.

Akaike's information criterion (AIC) [1] is a criteria to measure how well $\mathbf{y}$ is explained by the linear regression. AIC can be obtained by the following equation.

$$N\left(\log{(2\pi\frac{\mathrm{SSR}}{N})}+1\right)+2(k+2) \tag{4}$$

The smaller AIC is the better linear regression is.

2.1.1 Structure break test

The structure break test is a general term of tests to check the equality of two statistical models of different datasets. The chow test [4] is a popular method of the structure break test for linear regression. Let $\mathbf{D}_1 := \mathbf{X}_1||\mathbf{y}_1$ and $\mathbf{D}_2 := \mathbf{X}_2||\mathbf{y}_2$ be datasets, where $\mathbf{D}_1$ and $\mathbf{D}_2$ contain N_1 and N_2 records and both have k attributes; $\mathbf{D}_1$ is $N_1 \times (k+2)$ matrix and $\mathbf{D}_2$ is $N_2 \times (k+2)$ matrix.

Let $\mathbf{D}_{12}$ be the horizontal concatenation of $\mathbf{D}_1$ and $\mathbf{D}_2$; namely, $\mathbf{D}_{12}$ is an $N_1N_2 \times (k+2)$ matrix. Let SSR_1, SSR_2 and SSR_{12} be SSRs of $\mathbf{D}_1$, $\mathbf{D}_2$ and $\mathbf{D}_{12}$, respectively. The Chow test computes the statistics F as

$$F = \frac{\mathrm{SSE}_{1+2} - (\mathrm{SSE}_1 + \mathrm{SSE}_2)}{\mathrm{SSE}_1 + \mathrm{SSE}_2}\left(\frac{N-2k}{k}\right). \tag{5}$$

The null hypothesis of the Chow test is that both $\mathbf{D}_1$ and $\mathbf{D}_2$ have the same linear regressions. If the null hypothesis holds, F follows the F-distribution.

2.2 Secure computation

2.2.1 Secret sharing

Secret sharing is a kind of "encryption." The data is encrypted into n ciphertexts called *shares*. When a qualified subset of n shares are gathered, one can reconstruct the data from the subset of shares. In addition, any information of the data cannot be obtained even if an adversary obtains an unqualified subset of n shares are gathered. We use (t, n) secret sharing in which one can reconstruct the data from any t shares.

Let $[a]_1$, …, $[a]_n$ be n shares of a. In addition, let $[\mathbf{x}]$ and $[\mathbf{X}]$ be a vector and matrix of shares. We use Shamir's secret sharing [15] as a (t, n)-secret sharing.

2.2.2 Secure computation based on secret sharing

There are n parties $P_1, \ldots, P_n$, and a client C. The secure computation is a technique to compute the function f on data a while a is kept secret to the parties. It can be realized by using secret sharing [2, 5].

Roughly speaking, secret-sharing-based secure computation proceeds as follows.

1. Dataset is encrypted through secret sharing and shared among the parties.

2. A client C requests the parties to compute some function f.

3. The parties compute f on the dataset by interacting with each other, and obtain the encrypted result.

4. The parties send the encrypted result to C, and C decrypts the result.

The security of the above steps is discussed as follows. We assume an adversary does not corrupt more than $t-1$ parties, i.e., at most $t-1$ parties may collude. In this case, no information of the dataset is leaked from the encrypted dataset. In addition, throughout of the whole procedures, the dataset is not reconstructed so any information of the dataset does not leak except the resultant linear regression.

2.3 Computing basic functions

The difficulty of computing f depends on what the f is. It is known that the following basic functions can be computed in secure computation.

- Addition: On input $[a]_i$ and $[b]_i$ for P_i and $1 \leq i \leq n$, P_i can obtain $[a+b]_i$ by computing $[a]_i + [b]_i$.

- Product-sum: On input $[\mathbf{x}]_i := ([x_1], \ldots, [x_k])$ and $[\mathbf{y}]_i := ([y_1], \ldots, [y_k])$ for P_i and $1 \leq i \leq n$, P_i obtains $[\sum_{1 \leq j \leq k} x_j y_j]_i$ by conducting the product-sum protocol [11].

- Shuffle: On input $[\mathbf{x}]_i := ([x_1], \ldots, [x_k])$ for P_i and $1 \leq i \leq n$, P_i obtains $[\tilde{\mathbf{x}}]_i := ([x_{\pi(1)}], \ldots, [x_{\pi(k)}])$ by conducting the shuffling protocol [11], where π is a permutation in $\{1, \ldots, k\}$.

- Filtering: On input $[\mathbf{x}]_i := ([x_1], \ldots, [x_k])$ and some condition for P_i and $1 \leq i \leq n$, P_i obtains the subset of $[\mathbf{x}]_i$ consists of the elements satisfying the condition, by conducting the filtering protocol.

3 Statistical analysis in secure computation

If we want to perform statistical analysis in secure computation, the parties compute specific function f, such as aggregation, linear regression, etc. Some statistical analysis, such as computing average, can be computed in secure computation by combining the basic functions. However, some statistical analysis are difficult to be computed by just combining the basic functions. Therefore, the difficulty of computing statistical analysis in secure computation heavily depends on what the analysis is.

In this paper, we consider the following analysis scenario. An analyst wants to evaluate the effect on the income/outcome due to the number of workers in household. The dataset is the table in which each record corresponds to a household and contains the number of members and workers in a household, its income and outcome, and sampling weight. An example is shown in Table 1. We assume an

	# of members	# of workers	income	outcome	weight
foo	3	1	30	25	2.5
⋮	⋮	⋮	⋮	⋮	⋮

Table 1: Example of dataset

analyst computes the following four computations.

1. Frequency table in the number of members and workers in microdata.
2. Magnitude table of weights in the number of members and workers.
3. (Simple) linear regression between outcome and income.
4. Chow test in the different number of workers.

It is trivial that the frequency table and magnitude table can be computed in secure computation by combining basic functions. A classified table with the number of members and workers and the summation can be obtained by the filtering protocol and addition, respectively. However, computing linear regression and Chow test is a non-trivial task and we therefore explain how to compute them in secure computation.

3.1 Secure linear regression

The linear regression requires computing the matrix inversion as shown in Equation 3. However, it is complex to compute the matrix inversion in secure computation, and we therefore explored another approach.

We observed that $X^T X$ in Equation 3 is

$$X^T X = \begin{pmatrix} N & S_{\mathbf{x}_1} & \cdots & S_{\mathbf{x}_n} \\ S_{\mathbf{x}_1} & S_{\mathbf{x}_1\mathbf{x}_1} & \cdots & S_{\mathbf{x}_1\mathbf{x}_n} \\ \vdots & \vdots & \ddots \vdots & \\ S_{\mathbf{x}_n} & S_{\mathbf{x}_1\mathbf{x}_n} & \cdots & S_{\mathbf{x}_n\mathbf{x}_n} \end{pmatrix}, \tag{6}$$

where $S_{\mathbf{x}_i} := \sum_{\ell=1}^{N} x_{i,\ell}$ and $S_{\mathbf{x}_i\mathbf{x}_j} := \sum_{\ell=1}^{N} x_{i,\ell}x_{j,\ell}$ for $1 \leq i, j \leq k$. In addition, $X^T\mathbf{y}$ holds $X^T\mathbf{y} = (S_{\mathbf{y}}, S_{\mathbf{x}_1\mathbf{y}}, \ldots S_{\mathbf{x}_n\mathbf{y}})^T$. Therefore, each element in Equation 3 can be computed by addition and product-sum. In addition, these elements are statistics and they can be public.

From the above observation, our strategy is computing not the linear regression directly in the secure computation but each elements in Equation 3. After a client C obtains those elements, C can compute the linear regression locally.

3.2 Secure Chow test

From Equations 2 and 1, we observe

$$SSR = Nw_0^2 + 2\sum_{s=1}^{k} S_{\mathbf{x}_s} w_0 w_s + \sum_{s=1}^{k}\sum_{t=1}^{k} S_{\mathbf{x}_s\mathbf{x}_t} w_s w_t - 2S_{\mathbf{y}} w_0 - 2\sum_{s=1}^{k} S_{\mathbf{x}_s\mathbf{y}} w_s + S_{\mathbf{yy}}. \quad (7)$$

Therefore, SSR can be computed from the outputs of the secure linear regression and statistics, and the Chow test can be computed from SSR. This means that we can employ the same approach as the linear regression: A client C obtains the statistics and the output of linear regression, and then conducts the Chow test locally. We show secure Chow test in Appendix B.

4 Empirical analysis

4.1 Experimental environment

We implemented our protocols on the system developed by Kiribuchi et. al. [12], and show that the analysis can be computed efficiently and the analysis results are almost the same as those in the non-encrypted case.

This system consists of three servers, one management server, and clients. The management server connects to every servers and clients for the convenience of implementation, while the management server cannot obtain any secret information; every transactions are encrypted by end-to-end encryption. We show the image of the system in Figure 1. In this system, clients use the R software as interface. Encryption and decryption are computed in the behind of R, and clients can obtain statistics without being aware of the detail of secure computation.

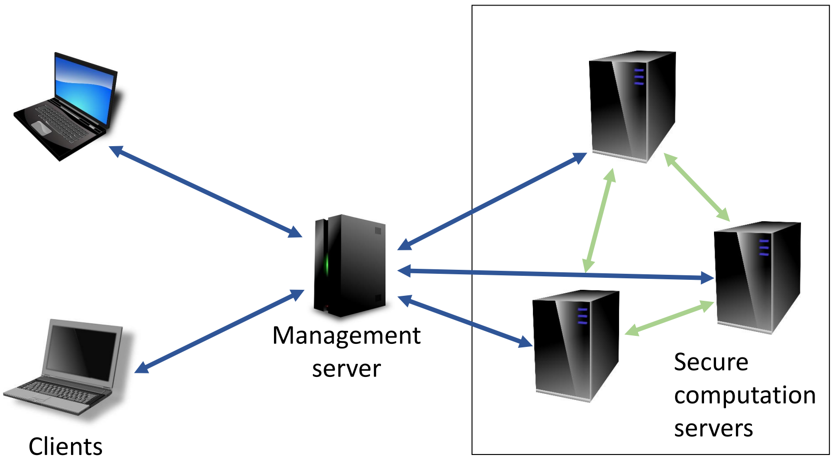

Figure 1: System of secure computation

4.2 Dataset

We used Synthetic Microdata for Educational Use provided by JNSC [17] for experiments. It is pseudo-individual data based on National Survey on Family Income and Expenditure in 2004 [16], which contains 32,027 households and 197 attributes.

In the experiments, we assumed that the logarithm of income/outcome are encrypted and registered as well as other attributes. Those logarithms are used in the linear regression.

4.3 Experiment 1: Aggregation

We aggregated the frequency and magnitude of the dataset by the number of members and workers. The frequency table represents the aggregation of samples, while the magnitude table represents the estimated aggregation of whole population. Details of the frequency table and the magnitude table are shown in Appendix C.

Both tables were exactly the same as those of the computation on non-encrypted data. Furthermore, the frequency table was generated in 2.8 seconds and the magnitude table was aggregated in 11.0 seconds in secure multi-party computation.

4.4 Experiment 2: Simple linear regression

We computed the simple linear regression, i.e, coefficients (w_0, w_1), of the logarithms of income and outcome, and AIC. In this experiment, we compute those in four cases: Households that have one, two, three workers, and overall households. We ignore the households more than three workers and unknown since such households are few. The precise result of the linear regression appear in Appendix C.

We compared the accuracy of the results in the secure computation and those in the non-encrypted data. We use the relative error $\frac{a^*-a}{a}$, where a is the result in the non-encrypted data and a^* is the one in the secure computation. In the simple linear regression, there are two coefficients, (w_0, w_1), and we therefore compute the relative error in each coefficient. As a result, the relative errors are under 2×10^{-8} and we conclude that our implementation has the sufficient accuracy for the simple linear regression. In addition, each computation of the linear regression was about a second, which is enough efficient.

4.5 Experiment 3: Structural break tests

We computed structural break test using the Chow test for each group of the number of workers. The significance level is 5%. As a result, the cases of a single worker versus two and three workers exceed the significance level. Therefore, the statistical model of single worker is different from the ones of two and three workers.

The relative errors of statistics F are under 2×10^{-7}. In addition, each Chow test was computed about five seconds. Therefore, our secure Chow test is enough efficient for both of accuracy and efficiency.

5 Discussion

5.1 Performance of our implementations

From the above experimental results, the results in secure computation have sufficient accuracy compared with those in non-encrypted data. Since even the largest relative error is 2×10^{-7}, our implementation can yield the accurate result. In addition, all experiments were finished within few seconds. Even in the worst case, computing the magnitude table, the computation costs only 11 seconds. Therefore, our implementation is efficient.

Consequently, our experiment showed that aggregation, simple linear regression, and the Chow test can be computed accurately and efficiently on the data containing 32,027 households, while the data is encrypted.

5.2 Statistical disclosure control in our implementations

In our implementation, we assumed that statistics, such as $S_{\mathbf{x}}$, do not leak privacy information about data. However, the assumption does not hold in some cases. For example, a mean of $\mathbf{x}$ leaks x_i when the variance of $\mathbf{x}$ is 0.

One of countermeasures for such risk is statistical disclosure control (SDC). SDC prohibits researchers to obtain the results containing privacy information. For example, in Eurostat's guideline [8], the linear regression coefficients are regarded as safe if

- each attribute has at least 10 degrees of freedom,
- the linear regression is not computed by only category attributes, and
- the linear regression is not the analysis of a single unit.

Moreover, [8] indicates that SDC subjects must be constructed by 10 or more than units.

However, such naïve SDC might not work to encrypted subjects. Threfore, one of our future works is a study of secure SDC, which judges disclosure or not over encrypted data.

6 Conclusion

In this work, we implemented data aggregation and linear regression for the first step of achieveing secure computation to statistical analysis. Our implementaions reduced steps in protocols of seucre multi-party computations by only calculating statistical values in secure and processing core algorithms in analysis over plaintext using statistical values. Every experiments show that our secure protocols are practical for both of speed and accuracy. Our future work is establishment of methods for evaluation and overcoming risk of leakage microdata from result of statistical analysis.

Acknowledgement

This work is supported by Grants in Joint Usage and Research Center Programs from Institute of Economic Research, Hitotsubashi University, Japan.

References

[1] Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. Proceedings of the 2nd International Symposium on Information Theory, Petrov, B. N., and Caski, F. (eds.), Akadéimiai Kiado, Budapest, 1973, 267-281.

[2] Ben-Or, M., Goldwasser, S., and Wigderson, A. (1988). Completeness theorems for non-cryptographic fault-tolerant distributed computation, Proceedings of the 20th Annual ACM Symposium on Theory of Computing, STOC'88, 1-10.

[3] Bogdanov, D., Kamm, L., Laur, S., and Sokk, V. (2016). Rmind: a Tool for Cryptographically Secure Statistical Analysis. IEEE Transactions on Dependable and Secure Computing, IEEE, 2016.

[4] Chow, G. C., (1960). Tests of equality between sets of coefficients in two linear regressions. Econometrica: Journal of the Econometric Society, 1960, 591-605.

[5] Chaum, D., Crepeau, C., and Damgård, I. (1988). Multiparty unconditionally secure protocols. Proceedings of the 20th Annual ACM Symposium on Theory of Computing, STOC'88, 11–19.

[6] Chida, K., Morohashi, G., Fuji, H., Magata, F., Fujimura, A., Hamada, K., Ikarashi, D. and Yamamoto, R., (2014). Implementation and evaluation of an efficient secure computation system using 'R' for healthcare statistics. Journal of the American Medical Informatics Association, JAMIA, 21(2), 115–130.

[7] Damgård, I., Fitzi, M., Kiltz, E., Nielsen, J. B., and Toft. T. (2006). Unconditionally secure constant-rounds multi-party computation for equality, comparison, bits and exponentiation. S. Halevi and T. Rabin eds., TCC, Springer, Lecture Notes in Computers Science, (3876), 285–304.

[8] A Network of the Excellence in the European Statistical System in the field of Statistical Disclosure Control. (2010). Guidelines for the checking of output based on microdata research. Available at: `http://neon.vb.cbs.nl/casc/ESSnet/guidelines_on_outputchecking.pdf` (accesed 30 June 2017).

[9] Gennaro, R., Rabin, M. O., and Rabin, T. (1988). Simplified VSS and fast-track multiparty computations with applications to threshold cryptography. Proceedings of the seventeenth annual ACM symposium on Principles of distributed computing. ACM, 1998.

[10] Ikarashi, D., Kikuchi, R., Hamada, K., and Chida, K. (2013). $O(\ell)$ Bits Communication Bit Decomposition and $O(|p'|)$ Bits Communication Modulus Conversion for Small k Secret-sharing-based Secure Computation (in Japanese). Computer Cecurity Symposium 2013, SCIS2013, 785–792.

[11] Ikarashi, D., Kikuchi, R., Hamada, K., and Chida, K. (2014). Actively Private and Correct MPC Scheme in $t < n/2$ from Passively Secure Schemes with Small Overhead. Cryptology ePrint Archive: Report 2014/304.

[12] Kiribuchi, N., and Ikarashi, D. (2015). Design of a Database System Processable Under Keeping Data Condentiality (in Japanese). In: Computer Security Symposium 2015, CSS2015, 419–426.

[13] Lu, W., Kawasaki, S., Sakuma, J. (2017). Using Fully Homomorphic Encryption for Statistical Analysis of Categorical, Ordinal and Numerical Data, The 24th annual Network and Distributed System Security Symposium, NDSS17, 2017.

[14] Little, M. A., McCharry, P. E., Roberts, S. J., Costello, D. AE., and Moroz, I. M. (2007). Exploiting Nonlinear Recurrence and Fractal Scaling Properties for Voice Disorder Detection, BioMedical Engineering OnLine, 6–23.

[15] Shamir, A. (1979). How to share a secret. Communications of the ACM, 22.11: 612–613.

[16] Statistics Bureau of Japan. (2005). 2004 National Survey of Family Income and Expenditure Overview. Available at: `http://www.stat.go.jp/english/data/zensho/2004/cgaiyo.htm` (accesed 29 June 2017).

[17] Makita, N., Ito, S., Horikawa, A., Goto, T., and Yamaguchi, K. (2013). Development of Synthetic Microdata for Educational Use in Japan. Paper Presented at 2013 Joint IASE / IAOS Satellite Conference, Macau Tower, Macau, China, 1-9 (2013)

A Secure linear regression

In this section, we show our secure linear regression protocol as protocol 1. In the same way, we implement secure simple linear regression as protocol 2. Note that we assume the number of records, N, is public and known to C.

B Secure chow test

We show secure chow test in protocol 3.

Protocol 1 Secure linear regression

```
Require: $P_i$ has $[\mathbf{X}]_i$ and $[\mathbf{y}]_i$ for $1 \le i \le n$
Ensure: $C$ has $\mathbf{w}$
for $1 \le s \le k$ do
    for $1 \le t \le k$ do
        $P_i$ computes $[S_{\mathbf{x}_s\mathbf{x}_t}]_i$ via ProdSum
    $P_i$ computes $[S_{\mathbf{x}_s\mathbf{y}}]_i$ via ProdSum and $P_i$ computes $[S_{\mathbf{x}_s}]_i = \sum_{\ell=1}^{N} [x_{s,\ell}]_i$
$P_i$ computes $[S_\mathbf{y}]_i = \sum_{\ell=1}^{N} [y_\ell]_i$
$P_i$ sends every shares to $C$
$C$ reconstructs all statistics
$C$ solves equation 3 locally and obtains $\mathbf{w}$
```

Protocol 2 Secure simple linear regression

```
Require: $P_i$ has $[\mathbf{x}]_i$ and $[\mathbf{y}]_i$ for $1 \le i \le n$
Ensure: $C$ has $(w_0, w_1)$.
$P_i$ computes $[S_{\mathbf{xx}}]_i$ and $[S_{\mathbf{xy}}]_i$ via ProdSum.
$P_i$ computes $[S_\mathbf{x}]_i = \sum_{\ell=1}^{N} [x_\ell]_i$ and $[S_\mathbf{y}]_i = \sum_{\ell=1}^{N} [y_\ell]_i$.
$P_i$ sends every shares to $C$
$C$ reconstructs all statistics
$C$ computes $w_0 = \frac{1}{N}(S_\mathbf{y} - w_1 S_\mathbf{x})$ and $w_1 = \frac{S_{\mathbf{xy}} - S_\mathbf{x} S_\mathbf{y}}{N S_{\mathbf{x}^2} - S_\mathbf{x}^2}$
```

C Experimental result

In this section, we show our experimental result of the frequency table in the table 2 and the magnitude table in the table 3. In the same way, we show result of simple regression in the table 4 and structural break test in table 5.

Protocol 3 Secure chow test

```
Require: $P_i$ has $[\mathbf{D}_1]_i$, $[\mathbf{D}_2]_i$, and $[\mathbf{D}_{1+2}]_i$ for $1 \le i \le n$.
Ensure: $C$ obtains the result of chow test between $\mathbf{D}_1$ and $\mathbf{D}_2$.
$C$ and $P_i$ conduct Protocol 1 on each input $\mathbf{D}_1, \mathbf{D}_2$, and $\mathbf{D}_{1+2}$, and $C$ obtains
    statistics as well as the coefficients of those linear regressions.
$C$ computes $\mathrm{SSR}_1$, $\mathrm{SSR}_2$, and $\mathrm{SSR}_{1+2}$ by using equation 7.
$C$ computes statistics $F$ by using equation 5 with $\mathrm{SSR}_1$, $\mathrm{SSR}_2$, $\mathrm{SSR}_{1+2}$, $k$, and
    $N := N_1 + N_2$.
$C$ tests the null hypothesis of $\mathbf{D}_1$ equals $\mathbf{D}_2$ using $F$.
```

Table 2: Frequency table of Synthetic Microdata for Educational Use [17] by numbers of members and workers in a household.

Member in		Workers in a household						
a household	total	1	2	3	4	5	6	unknown
total	32,027	13,913	13,459	2,950	691	40	6	968
2	7,438	4,124	3,239	0	0	0	0	75
3	8,537	3,908	3,391	1,035	0	0	0	203
4	9,944	4,132	4,201	1,031	324	0	0	256
5	4,405	1,436	1,943	559	220	27	0	220
6	1,214	256	494	232	104	6	3	119
7	390	51	162	84	31	7	3	52
8	81	6	29	6	12	0	0	28
9	15	0	0	3	0	0	0	12
10	3	0	0	0	0	0	0	3

Table 3: Magnitude table of Synthetic Microdata for Educational Use [17] by numbers of members and workers in a household.

Member in		Workers in a household						
a household	total	1	2	3	4	5	6	unknown
total	495,465	219,743	205,723	44,041	10,168	570	99	15,120
2	115,835	64,691	50,138	0	0	0	0	1,006
3	132,005	61,284	51,523	15,912	0	0	0	3,287
4	155,850	66,299	64,868	15,615	4,851	0	0	4,216
5	67,565	22,740	29,616	7,963	3,345	383	0	3,519
6	17,161	3,813	6,801	3,302	1,354	88	49	1,761
7	5,611	831	2,336	1,137	422	108	51	727
8	1,197	84	441	76	195	0	0	401
9	207	0	0	36	0	0	0	170
10	33	0	0	0	0	0	0	33

Table 4: Simple linear regression of logarithm of household expenditure per capita on logarithm of household income per captia.

		Estimated values				time
Workers	Household	Income	Intercepts	SSR	AIC	(sec.)
Overall	32,027	0.852	0.981	2313.3	6731.2	1.02
1	13,913	0.851	1.012	992.9	2762.1	1.07
2	13,459	0.864	0.831	947.0	2480.7	1.05
3	2,950	0.854	0.953	224.9	786.1	0.46

Table 5: Structural break test for each group of the number of workers in a household.

Workers		statistics F	p value	time(sec.)
1	2	22.059	2×10^{-10}	5.15
1	3	6.729	0.0012	3.65
2	3	0.353	0.7025	3.58