National Institute for Applied Statistics Research Australia

University of Wollongong

Working Paper

15-14

Density Approximant Based on Noise Multiplied Data: MaskDensity 10.R and its Applications

Yan-Xia Lin and Mark James Fielding

National Institute for Applied Statistics Research Australia, University of Wollongong, Wollongong NSW 2522. Phone +61 2 4221 5435, Fax +61 2 4221 4845. Email: anica@uow.edu.au

Density Approximant Based on Noise Multiplied Data: MaskDensity10.R and its Applications

Yan-Xia Lin
National Institute for Applied Statistics Research Australia
School of Mathematics and Applied Statistics
University of Wollongong, NSW 2500, Australia
yanxia@uow.edu.au

Mark James Fielding
DHI Water & Environment Singapore
1 Cleantech Loop
♯03-05 CleanTech One, Singapore 637141
Email:mjf@dhi.com.sg

Abstract

A framework, the sample-moment-based density approximant, for estimating the probability density function based on noise multiplied data was proposed in Lin (2014). Based on the framework, an R package, MaskDensity10.R, is built in this paper. The package is available from

http://www.uow.edu.au/ ∼yanxia/Confidential_data_analysis/.

The framework is developed for continuous univeriates (see Lin, 2014). With the techniques of nonparametric smoothing and K-means clustering integrated, MaskDensity10.R can be used for estimating the mass functions of categorical variables.

The same R package, MaskDensity10.R, can be used by the data agency to create the masked data set, as well as used by the end-user to obtain the approximation of the density function of the original data set based on masked data.

Simulation studies and real life data applications of MaskDensity10.R are presented in this paper. The risk of disclosure in the application of the R package to microdata is discussed, particularly for category data.

1 Introduction

Confidential data are not allowed to be issued to public without certain levels of protection. Many protection methods, including microaggregation of sensitive data, local suppression of unique data cells, top and bottom coding of continuous variables, rank swapping, rounding, additive noise, imputation and multiplicative noise, have been recommended and used in practice. More information on the discussions of those protection methods can be found in Duncan and Lambert (1986 and 1989), Willenborg and De Waal (2001), Oganian (2010), Shlomo (2010), and references therein.

The multiplicative noise method is one type of data protections. Kim and Jeong (2008) classified the multiplicative noise scheme into two schemes, Multiplicative Noise Scheme I and Multiplicative Noise Scheme II. In this paper, only the Multiplicative Noise Scheme I is considered. The Multiplicative Noise Scheme I can be briefly described as follows. Let $y_1, y_2, \cdots, y_N$ (original data) be a sample from a sensitive random variable Y. Let C be a positive random variable, independent of Y. When we say the original data $y_1, y_2, \cdots, y_N$ were masked by C, it means their masked data have the form $y_i^* = y_i \times c_i$, where $\{c_i\}$ is an independent sample from C. In literature, sometimes C is imposed to have $E(C) = 1$. With this restriction, y^* is an unbiased estimator of y given y. This restriction does not apply to this paper. Therefore, the unbiased estimator of y will be $y^*/E(C)$, given y.

The purpose of releasing protected data sets to public is to provide the end-user with an opportunity to obtain the statistical information of the original data sets without breaking confidentiality. However, data perturbation may destroy unbiasedness and other properties of estimators (see Nayak *et al.*, 2011; Sinha, *et al.*, 2011). In general, standard statistical inference methods might not be appropriate for analyzing data perturbed.

Instead of issuing perturbed data to public, issuing synthetic data is another approaches. The main differences between the approach of synthetic data and the approach of noise multiplied data could be summarized as follows: (i) for synthetic data approach, the data agency needs to do data pre-analysis on the original data for end-users. The subsequent inference analysis on the original data will rely on the quality of the synthetic data provided by the data agency. It could be the case that one type of synthetic data set is only for one particular inference purpose and it might be complex to update an issued synthetics data set when its original data set is updated; (ii) for the approach of the noise multiplied data, the data agency provides the end-user with masked data. Basically, the data agency is not

necessarily to do any pre-analysis on the original data for the end-user. The end-user has his/her own right to choose a correct technique to analyze the original data based on masked data. Thus, the inference analysis results on the original data will be strongly affected by the inference techniques applied to the masked data. Comparing with the approach of synthetic data, noise multiplied data sets can be easily updated if their original data sets are updated. The two approaches, "synthetic data" and " noise multiplied data" treat confidentiality analysis from two different angles. Basically, their are not comparable.

The properties of the multiplicative noise method, including evaluation disclosure risk, confidential protection, moment estimation, linear regression parameter estimation, properties of balanced noise distribution and effects on data quality and privacy protection in context of tabular magnitude data, have been deeply discussed and investigated in literature (Evans, 1996; Evans *et al.*, 1998; Hwang, 1998; Kim and Winkler, 2003; Kim and Jeong, 2008; Oganian, 2010; Krsinich and Piesse, 2002; Nayak, *et al.*, 2011; Sinha, *et al.*, 2011; Lin and Wise, 2012 and Klein and Sinha, 2013).

For noise multiplied data, developing appropriate data analysis methods for different inference purposes is necessary, for example, Kim and Jeong (2008) for domain estimation, Sinha, *et al.* (2011) for quantile estimation, Lin and Wise (2012) for linear regression parameters estimation and Lin (2014) for a frame work on the sample-moment-based approximation of the density function.

To understand the probability distribution of the underlying confidential data, basic statistical information on the data, including the summary statistics, the histogram and the plot of the probability density, are helpful. Directly releasing the summary statistics of a confidential data set could sometimes lead to the exact or approximate disclosure of the confidential data of single individuals (Malvestuto and Moscarini, 2003). To reduce the risk of disclosure, it becomes a standard process that the outcomes of summary statistics of confidential data have to be adjusted before they are issued to public, especially the values of maximum and minimum. If the end-user is allowed to submit multiple queries on summary statistics of subsets of the underlying data set, the risk of disclosure on the underlying confidential data might be increased. Sometimes, individual data could be exactly calculated or accurately estimated from the information of summary statistics of multidimensional database enquired by the end-user. Discussions on the issues of avoiding revealing (directly or indirectly) such individual data can be found from Malvestuto and Moscarini (2003).

To reduce the risk of disclosure of the underlying confidential data, the data

agency either does not allow the end-user to have multiple queries on the summary statistics of subsets of the data set or has to set a strategy which could enable the data agency to protect the data set if multiple queries meet certain regulation. Strategies for achieving this purpose were suggested and discussed in Malvestuto and Moscarini (2003). Malvestuto and Moscarini (2003) argued when answering queries that ask for summary statistics, a query-system of a multidimensional database should guard confidential data and the query-system should be provided with an auditing procedure. Each time a new query is processed, the system will check that its answer does not allow a (knowledgeable) user to disclose any sensitive data.

The key issues of introducing the query-system are: (1) the original confidential data are not allowed to be accessed by end-users; (2) the summary statistics provided need to be audited. One of the consequences of adopting the query-system is that the auditing procedure and relevant records have to be maintained for every valid issued confidential data set. It might lead to a high cost of maintaining the service. Furthermore, the restriction on accessing confidential data might bring inconvenience to the end-user, including many tedious administration processes.

Different from the query-system approach, Sinha *et al.* (2011) introduced a Bayesian method to infer about a quantile of a microdata set based on noise multiplied data. The inference procedure proposed is strongly related to the probability distribution of the multiplicative noise. Four types of noise, uniform distribution, Gamma distribution, Log-normal distribution and normal distribution, are considered by Sinha *et al.* (2011). The inference procedure might become complex or invalid if the probability distribution of the multiplicative noise is complex.

Lin (2014) proposed a different approach, sample-moment-based density approximant, for approximating the density function of the underlying sensitive variable Y based on its masked data. Let $\{y_i\}_1^N$ be a sample drawn from a random variable Y. The sample were masked by a noise C and yielded masked data $\{y_i^*\}_1^N$. Let $\{c_i\}_1^N$ be another independent sample drawn from C. The sample-moment-based density approximant of the density function f_Y of Y is defined as

$$f_{Y,K|\{y_i^*,c_i\}_1^N}(y) = \sum_{k=0}^{K} a_k(y) \frac{\overline{(Y^*)^k}}{\overline{C^k}} \tag{1}$$

where $\overline{(Y^*)^k} = \sum_{i=1}^N (y_i^*)/N$ and $\overline{C^k} = \sum_{i=1}^N c_i^k/N$; $a_k(y) = a_k(y; a, b)$ is a continuous function of y, where a and b used in Lin (2014) are $\max_{1\le i\le N}\{y_i\}$ and $\min_{1\le i\le N}\{y_i\}$, respectively (the details on (1) see Lin, 2014). Lin (2014) showed that $f_{Y,K|\{y_i^*,c_i\}_1^N}$ is able to well present the density function of Y given the size N of the sample and the upper order K are appropriate. Thus, f_Y can be approximated without accessing the original data $\{y_i\}_1^N$.

The approach of the sample-moment-based density approximant has no restriction on the type of distribution of the multiplicative noise. It gives the data agency more flexibility on the decision of the noise used to mask the underlying data and, in the meanwhile, creates more difficulties for the intruder in identifying the probability distribution of the noise, consequently, provides more protection on the underlying data.

Lin (2014) only gives a framework of the approach of the sample-moment-based density approximant. When the approach is implemented in practice, some technical issues need to be fixed, including the determination of the upper order K without the reference of f_Y and the boundaries a and b for the density approximant of a subset of data without accessing the original data of the subset.

The aim of this paper is to build an R package for the approach of the sample-moment-based density approximant proposed in Lin (2014). In the meanwhile, some issues related to the risk of disclosure of the approach are discussed. With the R package built, estimating the density function of categorical variables based on their masked data becomes feasible, though the approach of the sample-moment-based density approximant proposed in Lin (2014) is based on continuous univariate random variables.

The remainder of the paper is organized as follows. In Section 2, the determination of the upper order K and the boundaries a and b are investigated. The R package, MaskDensity10.R, is described in Section 3. Simulation studies and real life data applications of the Package are presented in Section 4. The final section is for discussion.

2 Evaluating the sample-moment-based density approximant without accessing the original confidential data

The sample-moment-based density approximant $f_{Y,K|\{y_i^*,c_i\}_1^N}$ is evaluated based on the masked data $\{y_i^*\}_1^N$, a independent noise sample $\{c_i\}_1^N$, the upper order K, boundaries $a = \min\{y_i\}$ and $b = \min\{y_i\}$ (see Lin, 2014). To well present f_Y through $f_{Y,K|\{y_i^*,c_i\}_1^N}$, the upper order K in Lin (2014) is determined by comparing the plots of $f_{Y,K|\{y_i^*,c_i\}_1^N}$ and f_y. using this way to determine K is impracticable in practice as the end-user has no right to access the original data and, consequently has no plot of f_Y as reference. Furthermore, if $\min\{y_i\}$ and $\max\{y_i\}$ are confidential

in some scenarios, it will be inappropriate to use their exact values in $f_{Y,K|\{y_i^*,c_i\}_1^N}$. A method for determining K and boundaries a and b without directly employing the information of the original data is desirable.

2.1 Determination of the upper order K in $f_{Y,K|\{y_i^*,c_i\}_1^N}$

Provost (2005) pointed out that, if an inappropriate upper order of moment K is used in the density approximant $f_{Y,K}$, it may cause $f_{Y,K}$ taking negative values. Determining the appropriate K for $f_{Y,K}$ is easy by inspecting the plot of f_Y if the plot is available. Due to confidentiality reasons, the plot of f_Y is not available for the end-user. It is a challenge to determine an appropriate K for $f_{Y,K|\{y_i^*,c_i\}_1^N}$ without the reference of the plot of f_Y. Simulation studies carried in Lin (2014) show that it is not necessarily that, the larger the K is, the more accurate the density approximant will be.

Lin (2014) demonstrated that the larger the value of the correlation between $f_{Y,K|\{y_i^*,c_i\}_1^N}$ and f_Y is, the better approximate the $f_{Y,K|\{y_i^*,c_i\}_1^N}$ to f_Y will be. Motivated by this fact, we suggest the following steps in determining the appropriate K for $f_{Y,K|\{y_i^*,c_i\}_i^N}$, without directly using the information of the original data $\{y_i\}_1^N$.

Step 1. Set an initial order of moment, $K = 1$ and a maximum order of moment we want to test. The maximum order is set as 100 in the R package built in this paper.

Step 2. Independently simulate a sample $\{c_i\}$ from C, then, evaluate $f_{Y,K|\{y_i^*,c_i\}_1^N}(y)$ using (1). The sample $\{c_i\}$ is not necessarily the sample used to mask $\{y_i\}$, though both of them simulated from the same population C.

Step 3. Simulate sample $\{y_j'\}_{j=1,\ldots,N}$ from $f_{Y,K|\{y_i^*,c_i\}_1^N}(y)$.

Step 4. Independently simulate a second sample $\{c_j'\}$ from C. Mask $\{y_j'\}$ by this new sample of noise and yield a new set of masked data $\{y_j'^*\}$.

Step 5. Evaluate the correlation $Cor(K)$ between $\{y_i'^*\}$ and $\{y_i^*\}$. Keep track of the optimum number of moments such that $Cor(K_{\text{opt}}) = \max_{k \le K} Cor(k)$.

Step 6. Update K to $K+1$ and return to Step 2 if $K+1 \le 100$. Stop when $Cor(K)$ drops below a threshold taken as $Cor(K) < 1 - 10\,(1 - Cor(K_{\text{opt}}))$ or $K+1 > 100$.

Step 7 Report K_{opt} as the optimum number of moments used.

Remarks:

(1) Step 5 is the key step in identifying an appropriate upper order of moment for the approximant of f_Y. The logic we used here is that, if the approximant determined by $\{y_i^*, c_i\}$ is close to the true density function f_Y, then $\{y_i'\}$ can be considered as an independent sample from Y and $\{y_i'^*\}$ will be an independent sample from $Y \times C$. Thus, the smoothed density functions determined by $\{y_i^*\}$ and $\{y_i'^*\}$, respectively, should be more likely strongly correlated.

In the R package build in this paper, the correlation between the smoothing density functions determined by $\{y_i^*\}$ and $\{y_i'^*\}$, respectively, will be reported. The higher the value is, the relatively better approximation between $f_{Y,K|\{y_i^*,c_i\}}(y)$ and $f_Y(y)$ will be.

(2) We set the maximum order of moment to be test is 100. However, to save time, we will not like to have the test procedure go from $K = 1$ to $K = 100$. In Step 6, our experience (also see examples in Lin, 2014) shows that, when K becomes too large, $Cor(K)$ will decrease quite rapidly as a result of poor estimates of high order moments. If $Cor(K)$ decreases too lower and

$$1 - Cor(K_{\text{opt}}) < \frac{1}{10}(1 - Cor(K)),$$

according to our empirical testing, it is not necessarily to further carry out the testing procedure. Therefore, we set the threshold $1 - 10\,(1 - Cor(K_{\text{opt}}))$ in Step 6.

The Steps proposed above are integrated in the R package, *MaskDensity10.R*. Examples how well the above Steps work are presented in Section 4.

2.2 The boundaries a and b used in MaskDensity10.R

Following the discussion in Lin (2014), the boundaries a and b used in $f_{Y,K|\{y_i^*,c_i\}}$ are $a = min_{1\le i\le N}\{y_i\}$ and $b = \max_{1\le i\le N}\{y_i\}$, respectively. Therefore $[a, b]$ is the domain of $f_{Y,K|\{y_i^*,c_i\}}$ and the values of a and b can be identified from the plot of $f_{Y,K|\{y_t^*,c_i\}}$ in R. Directly let $a = min_{1\le i\le N}\{y_i\}$ and $b = \max_{1\le i\le N}\{y_i\}$ in $f_{Y,K|\{y_i^*,c_i\}}$ is not appropriate, particularly if the values are confidential.

Example 1 below shows the impact of the values of a and b on the plot of $f_{Y,K|\{y_i^*,c_i\}}$.

Example 1. Simulate a sample $\{y_i\}_1^{2000}$ from $N(5, 3^2)$. To purely focus on the impact of the boundaries a and b on the performance of $f_{Y,K|\{y_i^*,c_i\}_1^{2000}}$ without any interference from the noise C, we let $C = 1$.

Seven pair-boundary (a, b)s: PB(j) $= (\min\{y_i\} + (2 - j + 1)s, \max\{y_i\} - (2 - j + 1)s)$, $j = 1, 2, 3, 4, 6, 7, 11$, are considered, where s is the sample standard error given by $\{y_i\}_1^{2000}$.

The domains determined by the pair-boundaries are subsets of the others in order. The shortest domain is $[\min\{y_i\} + 2s, \max\{y_i\} - 2s]$ and the longest one is $[\min\{y_i\} - 8s, \max\{y_i\} + 8s]$. The pair-boundary PB(3) is determined by $a = min\{y_i\}$ and $b = \max\{y_i\}$, used as a reference.

The plots of $f_{Y,K|\{y_i^*,c_i\}}(y)$ based on the seven pair-boundaries are presented in Figure 1.

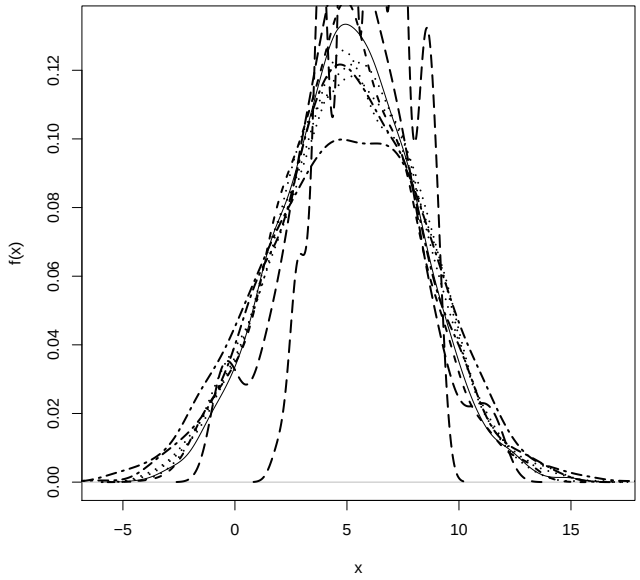

Figure 1: The smoothing density function for $N(5, 3^2)$ is in bold line. The approximant smoothing density function $f_{\tilde{Y},K|\{y_i^*,c_i\}}$ given by PB(1) and PB(2) are in longdash line; given by PB(3) (i.e. $a = \min\{y_i\}$ and $b = \max\{y_i\}$) in dashed line; given by PB(4) to PB(6) in dotted line; given by PB(7) and PB(8) in twodashed line.

Figure 1 shows that the plots of the density approximants given by PB(1) and PB(2) are very different from the plot of the true density function; the plots given by PB(4) and PB(6) are reasonable. The $Cor(K_{opt})$ based on PB(4) and PB(6) are all around 0.9997, which are higher than the $Cor(K_{opt})$s based on PB(1) and PB(2) (0.9873 and 0.9973). It confirms that, based on a same sample $\{y_i\}$, a density approximant with a relatively higher value of $Cor(K_{opt})$ should give a better approximation on f_Y. As the size of the interval $[a, b]$ increases, the plot of the

corresponding density approximant tends to be flat and gradually run away from the plot of the true density function (see the plots given by PB(7) and PB(8)).

Based on the simulation studies carried out in Example 2 and other more simulations (to save space, not shown here), the impact of a and b on $f_{Y,K|\{y_i^*,c_i\}}$ can be summarized as follows:

1. If $[a,b]$ is a subset of $[\min\{y_i\}, \max\{y_i\}]$ with a size much smaller than the size of $[\min\{y_i\},\ \max\{y_i\}]$, $f_{Y,K|\{y_i,c_i\}}$ might have less chance to be a good approximation of f_Y as $f_{Y,K|\{y_i,c_i\}}$ has to squeeze all the information provided by $\{y_i\}$ into a smaller interval $[a,b]$.

2. If the domain $[a,b]$ is close to $[\min\{y_i\}, \max\{y_i\}]$ (either a subset or a supperset), $f_{Y,K|\{y_i^*,c_i\}}$ is able to give a good approximate of f_Y. Particularly, the difference between the approximants of the density of Y based on domain $[\min_{1\le i\le N} y_i, \max_{1\le i\le N}\{y_i\}]$ and $[a,b] \supseteq [\min_{1\le i\le N}\{y_i\}, \max_{1\le i\le N}\{y_i\}]$ is not significant, because both approximants are evaluated based on the same sample $\{y_i\}_1^N$ and no $\{y_i\}_1^N$ drop within intervals $[a, \min_{1\le i\le N} y_i)$ and $(\max_{1\le i\le N}\{y_i\}, b]$. The smoothing density function defined on the interval $[a,b]$ will not add too much weights on $[a, \min_{1\le i\le N}\{y_i\})$ and $(\max_{1\le i\le N}\{y_i\}, b]$.

3. As the size of the interval $[a,b] \supseteq [\min_{1\le i\le N}\{y_i\}, \max_{1\le i\le N}\{y_i\}]$ increases, the normalized smoothing function $f_{Y,K|\{y_i^*,c_i\}}$ based on the pair-boundary (a,b) has to spread more weights to the whole interval $[a,b]$ and the plot of the $f_{Y,K|\{y_i^*,c_i\}}$ will be flattened, comparing to the plot of the $f_{Y,K|\{y_i^*,c_i\}}$ based on the pair-boundary $(\min_{1\le i\le N}\{y_i\}, \max_{1\le i\le N}\{y_i\})$.

Denote $\{y_{sub,j}\} \subseteq \{y_i\}_1^N$ a subset of $\{y_i\}_1^N$ from Y. Denote the population of $\{y_{sub,j}\}$ by Y_{sub}. Both samples were masked by noise C and yield masked data sets $\{y_i^*\}_1^N$ and $\{y_{sub,j}^*\}$, respectively. In general, the probability distributions of Y and Y_{sub} might not be the same. The density approxiamnts of f_Y and $f_{Y_{sub}}$ can be obtained from the masked data $\{y_i^*\}_1^N$ and $\{y_{sub,j}^*\}$, respectively, by the approach of the sample-moment-based density approximant with their appropriate pair-boundaries. The appropriate pair-boundaries for f_Y and $f_{Y_{sub}}$ might not be the same. Having the full knowledge on the original data $\{y_i\}_1^N$, the data agency has no problem to provide the end-user with an appropriate pair-boundary (a,b) for $f_{Y,K|\{y_i^*,c_i\}_1^N}$. However, it is impossible for the data agency to provide an appropriate pair-boundary (a,b) for $f_{Y,K|\{y_i^*,c_i\}_{sub}}$ without knowing which subset $\{y_{sub,j}\}$ the end-user might be interested.

To provide the end-user more opportunities to explore the probability density functions of subsets of $\{y_i\}_1^N$ by **himself/herself**, MaskDensity10.R should have a function to automatically justify the values of boundaries for the underlying subset of data based on the masked data $\{y_i^*\}_1^N$.

Taking into account all the above discussions, a standard procedure for determining a and b is suggested and adopted by MaskDensity10.R.

The standard procedure for determining a and b for $f_{Y,K|\{y_{sub,j},c_j\}}$:

(1) If Y is a categorical variable taking values $1, 2, \cdots, M$, let $a = 0$, and $b = M+1$ (see Section 3 for the discussion of the density approximant of a categorical variable).

(2) If Y is not a categorical variable, the values of a and b are determined by the following way.

Step 1 Let a_{basic} and b_{basic} be the boundaries determined by the data agency based on $\{y_i\}_1^N$ and $[a_{basic}, b_{basic}]$ is a superset of $[\min_{1\le i\le N}\{y_i\}, \max_{1\le i\le N}\{y_i\}]$.

Step 2 For each $\alpha = 0.01$ to 0.05 with increment 0.01 calculate [1], let

$$a_\alpha = \max\left\{a_{basic}, \frac{\overline{y^*_{subset}}}{\bar{c}} - \sqrt{1/\alpha}\sqrt{\frac{\overline{y^{*2}_{subset}}}{\overline{c^2}} - (\frac{\overline{y^*_{subset}}}{\bar{c}})^2}\right\} \quad (2)$$

$$b_\alpha = \min\left\{b_{basic}, \frac{\overline{y^*_{subset}}}{\bar{c}} + \sqrt{1/\alpha}\sqrt{\frac{\overline{y^{*2}_{subset}}}{\overline{c^2}} - (\frac{\overline{y^*_{subset}}}{\bar{c}})^2}\right\} \quad (3)$$

where $\overline{y^*_{subset}}$ and $\overline{y^{*2}_{subset}}$ are the sample mean and the sample second moment of $\{y^*_{sub,j}\}$, respectively, and $\bar{c}$ and $\overline{c^2}$ are the sample mean and the sample second moment of the noise C, respectively;

Step 3 For each pair-boundary (a_{basic}, b_{basic}), (a_α, b_α), $\alpha = 0.01, \cdots, 0.05$, determine the optimal order K for $f_{Y,K|\{y^*_{sub,j},c_j\}}$ and record $Cor(K_{opt})$, denoted by $Cor(K_{opt,basic})$ and $Cor(K_{opt,\alpha})$, $\alpha = 0.01, \cdots, 0.05$, respectively;

Step 4 Let $a = a_{\alpha_0}$ and $b = b_{\alpha_0}$, $\alpha_0 \in \{basic, 0.01, 0.02, \cdots, 0.05\}$ such that

$$Cor(K_{opt,\alpha_0}) = \max\{Cor(K_{opt,basic}), Cor(K_{opt,\alpha}), \alpha = 0.01, \cdots, 0.05\}.$$

Remarks: The logic used to support the standard procedure above is explained as follows.

By noting the following two facts:

[1] To save the time in running program, we only consider these five different values of α in MaskDensity10.R.

(i) given $[a_{basic}, b_{basic}]$ is a superset of $[\min_{1\le i\le N}\{y_i\}, \max_{1\le i\le N}\{y_i\}]$ and $\{y_{sub,j}\} \subset \{y_i\}$, we have $a_{basic} \le \min\{y_{sub,j}\} \le \max\{y_{sub,j}\} \le b_{basic}$;

(ii) from Tchebichev inequality, we have

$$P(L_\alpha \le Y_{sub} \le U_\alpha) > 1 - \alpha. \tag{4}$$

where

$$L_\alpha = E(Y_{sub}) - \sqrt{Var(Y_{sub})/\alpha} = \frac{E(Y^*_{sub})}{E(C)} - \sqrt{\frac{1}{\alpha}}(\frac{E(Y^{*2}_{sub})}{E(C^2)} - (\frac{E(Y^*_{sub})}{E(C)})^2)$$

and

$$U_\alpha = E(Y_{sub}) + \sqrt{Var(Y_{sub})/\alpha} = \frac{E(Y^*_{sub})}{E(C)} + \sqrt{\frac{1}{\alpha}}(\frac{E(Y^{*2}_{sub})}{E(C^2)} - (\frac{E(Y^*_{sub})}{E(C)})^2).$$

Ignoring the probability α, $\{y_{sub,j}\}$ will be bounded by

$$[\max\{a_{basic}, L_\alpha\}, \min\{b_{basic}, U_\alpha\}]. \tag{5}$$

By taking into account the information $Cor(K_{opt,basic})$ and $Cor(K_{opt,\alpha})$, and replacing the means by their sample means in L_α and U_α, we should expect that $[a_{\alpha_0}, b_{\alpha_0}]$ is a reasonable domain to replace $[min\{y_{sub,j}\}, \max\{y_{sub,j}\}]$ based on the information of $\{y_{sub,j}, c_j\}$. There might be other ways for determining the appropriate pair-boundary (a, b) for $f_{Y,K|\{y^*_{sub,j},c_j\}}$. We leave it as an open question.

3 Description of the *MaskDensity10.R* package

MaskDensity10.R, implementing the sample-moment-base density approximant, is available at

http://www.uow.edu.au/∼yanxia/Confidential_data_analysis/

The same package used by the data provider yielding masked data is also used by the end-user in making statistical analyses. The analyses are also possible when taking subsets of a corresponding set of masked data $\{y^*_i\}_{1\le i\le N}$.

Masked data is output from a *mask* function along with a binary file, named *noisefile*, which contains **the independent sample of noise $\{c'_j\}_{1\le i\le n'}$, the values of a_{basic} and b_{basic}, and the information whether the underlying data are numerical or categorical**. The binary file is recognisable only by the package *MaskDensity10.R*. By default the values of a_{basic} and b_{basic} in the binary file are set as $\min\{y_i\}_1^N$ and $\max\{y_i\}_1^N$, respectively. The sample of noise $\{c'_j\}_{1\le i\le n'}$ in the

binary file is not the same one used to produce the masked data $\{y_i^*\}_{1\leq i\leq N}$. This offers additional security while if it was possible to uncover the information, the original data remains protected. Any one-to-one matching with the noise applied is lost. The sample size of noise n' differs from and can be much larger than N to best represent the noise distribution. By default this noise is from a mixture of Gaussian distributions with means that are randomly generated. The data provider also has the option of providing the means of the noise or providing their own set of pair-boundary (a_{basic}, b_{basic}) and their own set of noise.

The end-user is able to view the contents of the *mask* function used by the data provider but this gives little information about the actual noise applied. Literature, including Kim and Jeong (2008), Sinha, *et al.* (2011), Oganian (2010), Lin and Wise (2012) and references therein, have discussed the risk of disclosure on noise multiplied data and point out that the original data can be well protected by an appropriate noise. By using *MaskDensity10.R*, noise information is concealed from the end-user. It will provide an extra protecting layer on the original data.

The end-user can use the *unmask* function to obtain the plot of the density approximant and estimated summary statistics based on the density approximant.

Categorical data is the main type of data commonly considered in confidential micro date sets. The technique for the approximant of density function proposed by Lin (2014) is applied to the density function of a continuous random variable. However, *MaskDensity10.R* can still work for categorical variables. The basic treatment built in *MaskDensity10.R* for categorical data is briefly described as follows: (i) masking the underlying categorical variable by a continuous noise. Thus, the masked data are no longer categorical data; (ii) applying the approach of sample-moment-based density approximant to the masked data to obtain the density approximant of the smoothing density function of the categorical variable. Obviously, this density approximant will have multiple centers at the levels of the categorical variable; (iii) finally, using the existing K-means clustering R package to convert the density approximant back to the mass function of the underlying categorical variable.

4 Applications of *MaskDensity10.R*

4.1 Simulation studies I: density approximant based on noise multiplied data

In this subsection, we show how to use *MaskDensity10.R* to produce the masked data set from an original data set and how to use the same R package to obtain the plot of the density approximant of the original data based on their masked data, in the meanwhile, to obtain the summary statistics based on the density approximant. Two types of data sets are studies in this subsection. One set of data were simulated from a continiuous random variable and the other were from a categorical variable.

Example 2. Let $\{y_i\}_1^{10000}$ be a sample drawn from a random variable $Y = I_{(w=0)}Y_1 + I_{(w=1)}Y_2$, where I is an indicator function, $Y_1 \sim N(30, 4^2)$, $Y_2 \sim N(50, 2^2)$ and w is Bernoulli distributed with $P(w = 0) = 0.3$. Let $C = I_{(v=0)}C_1 + I_{(v=1)}C_2$ be the multiplicative noise used to mask $\{y_i\}$, where v has Bernoulli distribution with $P(v = 0) = 0.6$; $C_1 \sim N(80, 5^2)$ and $C_2 \sim N(100, 3^2)$.

The R code used to simulated $\{y_i\}$ and $\{c_i\}$ is listed below:

```
set.seed(123)
n=10000
rmulti <- function(n, mean, sd, p)
{
  x <- rnorm(n)
  k<-length(mean)
  u <- sample(1:k, size=n, prob=p, replace=TRUE)
  for(i in 1:k)
    x[u==i]<-mean[i]+sd[i]*x[u==i]
  return(x)
}
y <- rmulti(n=10000, mean=c(30, 50), sd=c(4,2), p=c(0.3, 0.7))
      # y is a sample drawn from Y.
noise<-rmulti(n=10000, mean=c(80, 100), sd=c(5,3), p=c(0.6, 0.4))
      # noise is a sample drawn from C.
```

The R code used to generate the masked data of $\{y_i\}$ is:

```
library(MaskDensity10)
ymask<-mask(y, noisefile="noise.bin", noise, a1=min(y), b1=max(y))
write(ymask$ystar, "ystar.dat")
```

where $a1$ and $b1$ are the a_{basic} and b_{basic} introduced in Section 2.2 and the values can be decided by the data agency. In this example, we simply let $a1 = \min\{y_i\}$ and $b1 = \max\{y_i\}$.

After running the above R code, two files "ystar.dat" and "noise.bin" are generated and ready for the end-user. File "ystar.dat" is a readable file containing masked data. File "noise.bin", readable by *MaskDensity10.R* only, is a binary file containing the information of noise and others mentioned in Section 3. The true sample $\{y_i\}$ is concealed from the end-user.

Saving the files "ystar.dat" and "noise.bin" into a same R working directory, the end-user can use the following code to obtain the plot of the density approximant of $Y = I_{(w=0)}Y_1 + I_{(w=1)}Y_2$:

```
library(MaskDensity10)
ystar <- scan("ystar.dat")
y1 <- unmask(ystar, noisefile="noise.bin")
plot(density(y1), main="density(ymask)", xlab="y")
# the plot of the approximant of $f_Y$
```

In Figure 2, the left-top panel is the plot of the density function of the multiplicative noise C; the right-top panel is the plot of the smoothing density function of the original data Y. These two plots cannot be obtained by the end-user, as the end-user only has masked data. The left-bottom panel is the plot of the smoothing density function of the masked data, which can be produced from "ystar.dat". There is no obvious connection between the plots of the smoothing density function of the original data Y and the smoothing density function of the masked data. The plot of the density approximant based on the masked data, produced by *MaskDensity10.R*, is presented at the right-bottom panel (titled density(ymask)). This plot is close to the plot of the smoothing density function of the original data. The R output reports $Cor(K_{opt}) = 0.9998$ (See Section 2.1 for the definition of $Cor(K_{opt})$.). A professional statistician should be able to guess the summary statistics of Y from the plot of the density approximant. Since "$y1$" were simulated from the density approximant, the summary of statistics of the data "$y1$" will provide the information of the summary statistics of the original data. For comparison purpose, we report the summary statistics given by "$y1$" and "y" in Table 1. The corresponding values in the summary statistics are close to each other as expected.

Sinha *el at.* (2011) introduced a Bayesian approach for quantile estimation from noise multiplied data. We apply the method proposed by Sinha *et al.* (2011) and

Table 1: The summary of statistics given by "y_1" and "y".

data	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
y	16.34	33.63	48.83	43.90	50.74	57.70
$y1$	16.59	34.15	48.39	43.73	50.82	55.98

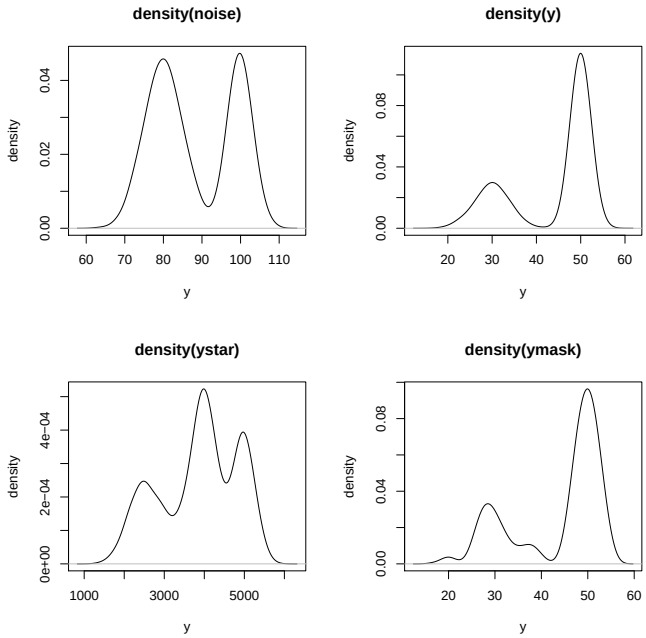

Figure 2: The top left and right panels present the smoothing density functions of noise and the original data, respectively. The smoothing density function of masked data is given by the bottom left panel, while the approximant of the density function of Y is in the bottom right panel.

the method proposed by Lin (2014) to the original sample $\{y_i\}_1^{10000}$ in Example 2, and compare the estimations of summary statistics given by the two methods.

The inference procedure proposed by Sinha *et al.* (2011) strongly depends on the probability distribution of the noise C. Only four types of noise distributions are considered in Sinha *et al.* (2011), including Gamma distribution. In this comparison study, we let the probability distribution of C have Gamma distribution $Gamma(\theta = 0.025, k = 1/0.025)$. After the original data $\{y_i\}_1^{10000}$ were masked by the noise C and yielded the masked data $\{y_i^*\}_1^{10000}$, we applied *MaskDensity10.R* and Sinha's method to $\{y_i^*\}_1^{10000}$, respectively.

When we applied Sinha's approach to $\{y_i^*\}_1^{10000}$, the number of independent replicates used in the method is 1000. The outputs of summary statistics based

Table 2: The estimate of summary statistics.

Data source	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
based on original $\{y\}$	16.34	33.63	48.83	43.90	50.74	57.70
The method proposed in this paper						
based on masked data	16.45	31.96	47.58	43.27	51.14	55.67
Sinha's method (with Gamma distribution)						
based on masked data	11.23328 (1.077836)	35.39753 (0.124766)	45.70855 (0.1181506)	46.01877 (0.0722413)	55.39265 (0.1473023)	117.0882 (8.245374)

on masked data are reported in Table 2 .

Table 2 shows that both methods, Sinha's method and the method proposed by Lin (2014), provide reasonable information on the summary statistic of $\{y_i\}$ (excluding the maximum and minimum given by the Sinha's method, as the method does not provide estimations of the maximum and minimum). Both of the methods have their merits in applications. One obvious advantage of the method proposed by Lin (2014) is that the method has no restriction on the probability distribution of the noise. Therefore, the method offers the data agency with a wide range of choices on the type of multiplicative noise and gives the intruder less chances in correctly identifying the type of noise used to mask the data.

Example 3 below demonstrates how MaskDensity10.R works for categorical variables.

Example 3. Let Y be a random variable with probability distribution $Bernoulli(0.5)+1$ and the multiplicative noise C the absolute value of a random variable with distribution $N((a+b)/2, 1+(a-b)^2/4)$, where $a = 170$ and $b = 80$. The following R code is used to obtain the samples from Y and C, respectively. Both of them have size 2000.

```
set.seed(124)
n<-2000
a<-170
b<-80
y<-rbinom(n, 1, 0.5)+1
noise<-(a+b)/2+ sqrt(1+(a-b)^2/4)*rnorm(n, 0,1)
noise[noise<0]<- - noise[noise<0]
```

Since Y is a categorical variable taking values 1 and 2, the boundaries a_{basic} and b_{basic} used in its density approximant are 0 and 3, respectively. The R code used to

mask "y" by the "noise" is:

```
library(MaskDensity10)
ymask<-mask(factor(y), noisefile="noise.bin", noise, a1=0,b1=3)
              # using factor(y) because y is a categorical variable
write(ymask$ystar, "ystar.dat")
```

Then, the files "ystar.dat" and "noise.bin" are ready for the end-user. To obtain the plot of the density approximant of Y, the following code is used:

```
library(MaskDensity10)
ystar<-scan("ystar.dat")
y1 <- unmask(ystar, noisefile="noise.bin")
plot(table(y1), lwd=5, main="hist(ymask)", col="grey30")
```

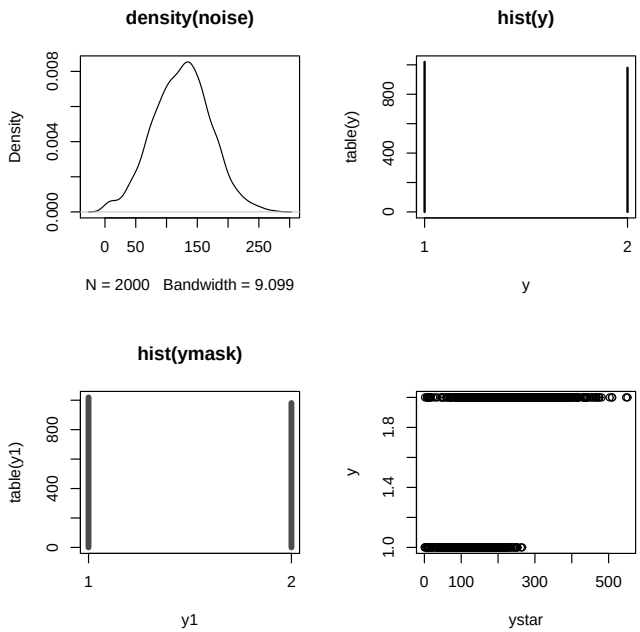

Figure 3: Top left panel: the smoothing density function of noise C. Top right panel: the frequency of the original data. Bottom left panel: the estimate of the frequency based on masked data. Bottom right panel: the plot of original data vs masked data.

In Figure 3, the plot of the smoothing density function of the noise is in the top left panel. The plot of the frequency of the original data "y" is in the top right panel (1020 "1"s and 980 "2"s). Both of plots cannot be observed by the end-user.

The masked data received by the end-user are no longer categorical data. The plot "y" against "*ystar*" (masked data) is presented in the bottom right panel in Figure 3. The values "$y = 1$" are mapped to any values between 0 and 300 and the values of "$y = 2$" are mapped to any values between 0 and 600. Obviously, the larger the value of "*ystar*" is, the higher the chance of the values of "*ystar*" corresponding to original value "$y = 2$" will be. Given no information on the noise C, the end-user might have difficulty to figure out a general rule that, beyond which value, a value of "*ystar*" will definitely correspond to the original value "$y = 2$" and, under which value, a value of "*ystar*" will definitely correspond to the original value "$y = 1$" (More studies on this concern see Section 4.4.). Applying the technique of cluster analysis, *MaskDensity10.R* converts "*ystar*" to "$y1$" which is categorical. The end-user can obtain the plot of "$y1$" (in the bottom left panel of Figure 3). The estimates of the frequency of "y" (1019 "1"s and 981 "2"s) based on masked data can be given by R code "summary(y1)". In this example, the estimate of the mass function of Y based on masked data is very close to the true mass function of given by the original data.

4.2 Simulation studies II: empirical evaluation on the approximant density function

Lin (2014) shows that, for each y fixed, $f_{Y,K|\{y_i^*,c_i\}_1^N}(y)$ is approximately unbiased about $f_{Y,K}(y)$ when Y is bounded. It becomes complex to show the unbiasedness in theory when Y is not bounded. From practical point view, it is of interest to know the impact of the independent samples of noise C on the performance of $f_{Y,K|\{y_i^*,c_i\}_1^N}$ given the original data $\{y_i\}$ is kept the same.

Example 4. Consider the same Y and the same noise C in Example 2. Draw a sample $\{y_i\}_{1\le i\le 10000}$ from Y and apply this sample to the following two cases of simulation studies. The simulation studies are described below:

Case (i) Use the code in Example 2 to create files "ystar.dat" and "noise.bin" for the end-user. Then, the end-user independently applies the "unmask" function in *MaskDensity10.R* to the "ystar.dat" 100 times and plots all the density approximants out.

Mentioned in Section 3, when the "unmask" function is applied to "ystar.dat", an independent sample of the noise will be drawn from the noise sample stored in "noise.bin" and used in the evaluation of the density approximant. The samples drawn at each time might not be the same, therefore, the outcomes of the density approximant might not be the same, though the "ystar.dat" is the same. In this

Table 3: The means of summary statistics.

Data source	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
based on original $\{y\}$	16.34	33.63	48.83	43.90	50.74	57.70
Case (i)						
based on masked data	16.418 (0.07893227)	34.4208 (0.4516987)	48.4292 (0.06279162)	43.7677 (0.08808691)	50.8294 (0.04352684)	56.5176 (0.2299324)
Case (ii)						
based on masked data	17.0327 (0.6858967)	34.0106 (0.9291908)	48.4872 (0.1161337)	43.7877 (0.1362103)	50.8609 (0.08506795)	56.7782 (0.409842)

Table 4: The means of summary statistics.

Data source	$P(Y=1)$	$P(Y=2)$
based on original $\{y\}$	0.5	0.5
Case (i)		
based on masked data	0.5133485 (0.01701319)	0.4866515 (0.01701319)
Case (ii)		
based on masked data	0.496948 (0.02723025)	0.503052 (0.02723025)

simulation study, we empirically show the differences between each unmasked outputs and compare them with the plot of the density function of the original data in Figure 2;

Case (ii) Independently simulated 100 sets of sample from C. Then apply them to $\{y_i\}_{1\leq i\leq 10000}$, respectively, and yield 100 masked data sets (100 "ystar.dat" files). Finally, apply *MaskDensity10.R* to each of "ystar.dats", respectively, and produce the plots of all the density approximants. We want to empirically show the differences among the outputs when the original data were masked by independent samples from the same noise.

The summary statistics given by the original data and density approximants are reported in Table 3. The plots for Cases (i) and (ii) are presented in Figures 4 and 5.

Example 5. A sample $\{y_i\}_{0<i\leq 2000}$ were simulated from the categorical variable Y studied in Example 3 and the noise C is the same as in Example 3. We carry out the same simulation studies described in Example 4.

The sample means and sample standard errors of the estimates of $P(Y=1)$ and $P(Y=2)$ for Case (i) and (ii) are reported in Table 4.

In Examples 4 and 5, the means of each element in summary statistics and

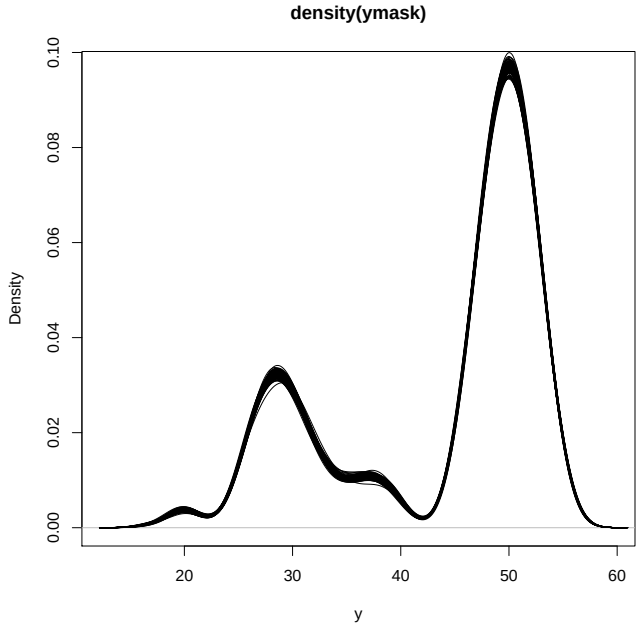

Figure 4: Plots of the 100 density approximants studied in case (i).

the functional mean of the density approximants are close to their corresponding elements in summary statistics and the density function based on the original data, respectively. Independently applying MaskDensity10.R to the same "ystar.data" with "noise.bin" provided, we found from Figure 4 that there are not much difference between the density approximants. Figure 5 shows that, in general, the functional mean of the density approximants is close to the true density function, but some sets of masked data might lead to less accurate density approximants. It means the "quality" of the sample of noise, Used to mask the original data, has certain level of impact on the accuracy of the density approximant. This phenomenon is understandable and it might give the data agency a caution on the quality of the sample of noise. Checking the satisfactory of the density approximant before releasing associated "ystar.data" and "noise.bin" might be necessary.

In practice, different distribution of noise might lead to different level of accuracy of the density approximant. Given an original data set, it is of interest to identify an appropriate noise such that the original data can be well protected through the noise and, in the meanwhile, the density approximant well presents the density function of the original data. This topic is beyond the purpose of this paper and not discussed here.

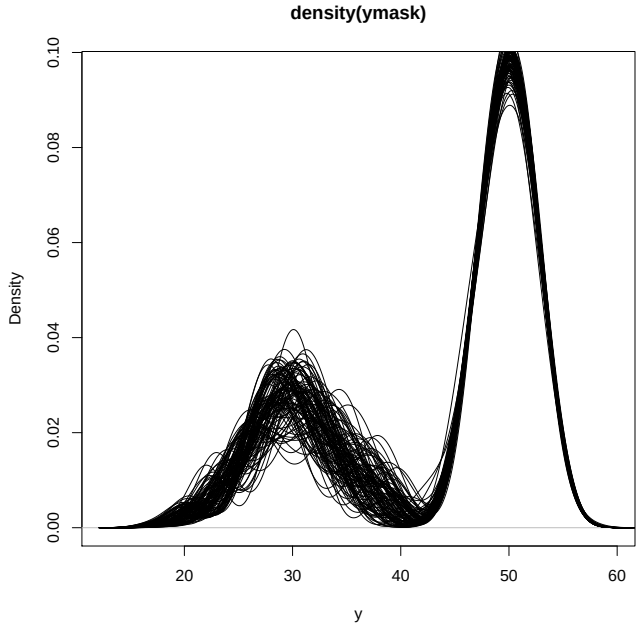

Figure 5: Plots of the 100 density approximants studied in case (ii).

4.3 Simulation studies III: the density approximant of subset data

In this subsection, we demonstrate that it is possible the end-user is able to obtain the information of the density function of a subset of the original data set by using the package *MaskDensity10.R*, based on the masked data set created from the full original data set.

Example 6: Use the same Y and noise C in Example 2, but assume that the observations $\{w_i\}_1^{10000}$ of the variable w used to conduct the sample $\{y_i\}_1^{10000}$ are not confidential.

Assume that the data agency used the R code in Example 2 to yield files "ystar.dat" and "noise.bin", sending them to the end-user along with the observations of $\{w_i\}_1^{10000}$. Based on the values of $w_i = 0$ or $w_i = 1$, the masked data set "ystar.dat" can be partitioned into two subsets, denoted by "ystar1.dat" and "ystar2.dat", corresponding to the samples drawn from Y_1 and Y_2, respectively.

Apply the "unmask" function in *MaskDensity10.R* to "ystar2.dat", using the "noise.bin" received. The plot of the density approximant based on the subset of masked data "ystar2.dat", along with the plots of the density function of the corresponding original subset, is reported in Figure 6. Table 5 reports their summary of statistics.

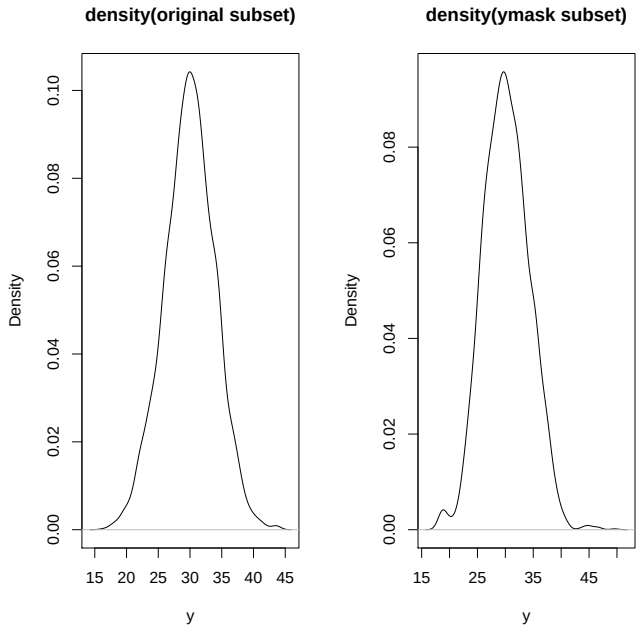

Figure 6: The left panel is the plot of the density function of Y_2 and the right panel is plot of density approximant given by "ystar2.dat"

The outputs show that using the subset of masked data to estimate the density function and summary statistics of its corresponding the subset of original data is very promising. More applications of *MaskDensity10.R* to subset data are given in next subsections.

Table 5: Analysis outputs for Example 7

Data source	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
based on original subset $\{y\}$ (drawn from Y_2)	16.34	27.34	29.96	29.92	32.54	43.78
Apply the MaskDensity10.R 1 time to "ystar2.dat" .						
based on masked data	17.94	27.24	29.89	30.06	32.75	46.30
Independently apply MaskDensity10.R 500 times to "ystar2.dat"						
mean, based on masked data	18.01 (0.2580)	27.23 (0.1286)	29.96 (0.1052)	30.13 (0.1041)	32.88 (0.1565)	50.23 (3.6337)

4.4 The risk of disclosure and the sample-moment-based density approximant method

Once a masked data set was issued to public, the data agency has no control how end-users work on the data. A data intruder has right to apply whatever available methods to the issued masked data for his/her purposes. It is of interest whether, with MaskDensity10.R, the intruder is able to gain more chances to successfully identify the values of the original data based on their released masked data. For this issue, only categorical data are under consideration in the following study as a categorical variable only takes finite possible values, thus the values of a categorical variable might be are easily identified.

Let Y be a categorical variable taking values $1 < 2 < \cdots < M$, masked by a noise C. Assume that $\{y_i\}$ is a sample from Y and its masked data set is $\{y_i^*\}$. Since $C > 0$, the larger the value of y^* is, the higher the likelihood of the y^* corresponding to the larger value of y will be. Particularly, the largest value of y^* is more likely corresponding to $y = M$. Therefore, some values of "M"s in the original data could be easily identified from their masked values. It is of interest to know the percentage of those values.

For a given masked data set, different intruders might have different strategies to attack the data set based on their own knowledge on the original data. Also the level of the risk of disclosure might be affected the nature of the underlying data and the nature of the multiplicative noise, including the probability distribution of noise, the variance of noise (see Lin and Wise, 2012), the size of sample and so on. It is impossible for us to evaluate the approach of sample-moment-based density approximnt against all the possible attacking strategies. We only use the following example to demonstrate that, in general, the method of sample-moment-based density approximant and MaskDensity.R do not provide any significant advantage to intruders, if the noise used to mask the original data is appropriate.

Example 7. Consider a categorical variable $Y \sim Bernoulli(0.6) + 1$ and noise $C = I_{(v=0)}C_1 + I_{(v=1)}C_2$, where v has Bernoulli distribution with $P(v = 1) = 0.6$; $C_1 \sim N(\mu_1, s_1)$ and $C_2 \sim N(\mu_2, s_2)$. Denote $\theta = (\mu_1, \mu_2, s_1, s_2)$ the parameter for C.

Simulate a sample $\{y_i\}$ of size 2000 from Y and apply this sample to all the studies discussed in this example.

Denote $\{y_i^*\}$ the masked data set of $\{y_i\}$ and $\{\tilde{y}_j^*\}$ be the sorted data set of $\{y_i^*\}$, in ascending order. Let j_0 be the position in $\{\tilde{y}_j^*\}$ such that $\tilde{y}_{j_0}^*$ is the largest masked value corresponding to $y = 1$. In this example, if j_0 can be correctly estimated, then the original data of "$y = 2$", corresponding to $\{\tilde{y}_j^*\}_{j>j_0}$ can be correctly identified;

Table 6: The impact of the variance of C.

parameters (a, b, s_1, s_2)	sd. for C	j_0	$P(Y=2)$
(90, 100, 5, 3)	6.62	787	0.606
(80,100,5,3)	10	787	0.6245
(90, 100, 20, 20)	21.22	1212	0.596
(95, 100, 25, 25)	25.02	1316	0.6075
(95, 100, 40, 35)	37.80996	1533	0.616

if j_0 is close to $2000 \times 0.4 = 800$, it will mean that almost all the original data "$y = 1$", corresponding to $\{\tilde{y}_j^*\}_{j \le j_0}$, can be correctly identified. Therefore, the position j_0 plays an important role in identifying the values of the original data based on their masked data.

Based on simulation studies, the following conclusions are held.

(a) *The variance of C has a big impact on the position of j_0.*

In this study, five sets of parameters for C are considered. For different values of the variance of C, the values of j_0 and estimates of $P(Y = 2)$ based on the data masked by C are reported in Table 6.

Simulation studies show that, as the variance of C increases, the value of j_0 increases and more masked "$y = 2$" mix up with masked "$y = 1$". It means that, though some of "$y = 2$", corresponding to larger values of $\tilde{y}^*$, have higher risk to be identified, the risk can be reduced by using the approach of increasing the variance of the noise C. Interesting thing as well as a good thing is that the estimate of $P(Y = 2)$ is not significantly effected by the variance of C.

(b) *MaskDensity10.R is not helpful in estimating the value of j_0.*

The end-user only receives "ystar.dat" and "noise.bin". The value of j_0 is unknown.

In this example, if the value of j_0 could be correctly estimated, at least those "$y = 2$", corresponding to $\{\tilde{y}_j^*\}_{j > j_0}$, could be directly identified.

To estimate the j_0, it might expect that, an accurate estimation of "j_0" is the position s such that the estimate of the probability $P(Y = 2)$ given by the subset $\{\tilde{y}_j^*\}_{s+1}^{2000}$ is higher than those given by other subsets $\{\tilde{y}_j^*\}_t^{2000}$, given the fact that all $\tilde{y}_j^*$, $j > j_0$, correspond to "$y = 2$". In the following, we empirically show that *MaskDensity10.R* provides no help in estimating the "j_0" based on the logic above.

Consider the masked data set $\{y_i^*\}_1^{2000}$ studied in (a), masked by the noise C with parameter $(95, 100, 40, 35)$. The true value of j_0 was reported in Table 6.

Now, we carry out the following tests and apply *MaskDensity10.R* to the subsets $\{\tilde{y}_j^*\}_s^{2000}$, where s = 1300, 1400, 1500, 1534, 1600, 1700, and 1800. The estimations of $P(Y = 2)$ given by those subsets are 0.272857, 0.6666, 0.426, 0.32976, 0.3675, 0.3800, and 0.435. The subset $\{\tilde{y}_j^*\}_{1400}^{2000}$ gave the highest estimate of $P(Y = 2)$. From Table 6, the true value of j_0 is 1533, which is related to the subject $\{\tilde{y}_j^*\}_s^{2000}$ with $s = 1534$.

Why the subset $\{\tilde{y}_j^*\}_{1534}^{2000}$ did not yield the highest value of the estimation of $P(Y = 2)$ or show that the estimation of $P(Y = 2)$ is close to 1? One of possible reasons is that the size of the subset $\{\tilde{y}_j^*\}_{1534}^{2000}$ is too small, only 467, and the method of sample-moment-based density approximant is only valid for data sets with large size.

The above simulation study indicates that, with an appropriate noise C, the data agency is able to reduce the risk of correctly identifying $Y = M$. In the meanwhile, with the carefully selected noise, the value of j_0 might be difficultly correctly estimated.

How to determine an appropriate C for a given confidential micro dataset is an important issue, but it is beyond the purpose of this paper.

4.5 Real data applications

Example 8. In this example, we apply *MaskDensity10.R* to a real life data set taken from the United States Energy Information Authority. The data can be found in the R package *sdcMicro*, and also available from the United States Energy Information Authority website http://www. eia.doe.govcneaf/electricity /page/eia826.html under year 1996. The data set consists 15 variables generally concerning income and sales data and each of them has 4092 observations. For the sake of simplicity, we do not concern how to protect the value "0" in this paper. All the values of "0" will stay in their original data set without any pre-protections.

A categorical variable, named UTILNAME, is considered in this example. UTILNAME takes values from one of the 258 categories (utilities). We grouped these 258 utilities into 5 categories: "Council", "Central", "Power","State" and "Other", and defined a new categorical variable, denoted by "UTILITY". Presume "UTILITY" a sensitive variable. "UTILITY" takes values from one of the 5 categories based on its corresponding values of UTILNAME. We use integers 1 to 5 to replace the names of

categories "Council", "Central", "Power","State" and "Other", respectively. Thus, "UTILITY" will take 5 possible values from 1 to 5. The noise C used to mask "UTILITY" is $C = I_{(v=0)}C_1 + I_{(v=1)}C_2$, where v has Bernoulli distribution with $P(v = 0) = 0.6$; $C_1 \sim N(80, 5)$ and $C_2 \sim N(100, 3)$.

The following code is used to generate a sample of noise from C.

```
set.seed(123)
n=length(UTILITY)
rmulti <- function(n, mean, sd, p)
    {
      x <- rnorm(n)
      u <- sample(1:length(mean), n, prob=p, replace=T)
      for(i in 1:length(mean)) x[u==i]=mean[i]+sd[i]*x[u==i]
      return(x)
    }
noise<-rmulti(n, mean=c(80, 100), sd=c(5,3), p=c(0.6, 0.4))
```

To generate the files "ystar.dat" and "noise.bin", the following code is used.

```
library(MaskDensity10)
ymask <- mask(as.factor(UTILITY), noisefile="noise.bin", noise,
              a1=0, b1=6)
write(ymask$ystar, "ystar.dat")
```

To estimate the mass function of "UTILITY" based on masked data and obtain the plot of the estimated mass function, the following code will do.

```
library(MaskDensity10)
ystar <- scan("ystar.dat")
y1 <- unmask(ystar, noisefile)
plot(table(y1), lwd=5, xlab="UTILITY", ylab="",
main="Simulated Data Frequencies", axes=FALSE)
axis(1, at=1:5, labels=c("Council", "Central", "Electric",
"State Level", "Other"))
axis(2)
```

The plots of the mass function of the original data "UTILITY" and the density function of its masked data are given in Figure 7 the top panel. Because the noise is a continuous random variable, "ystar" is no longer a categorical variable. The plot

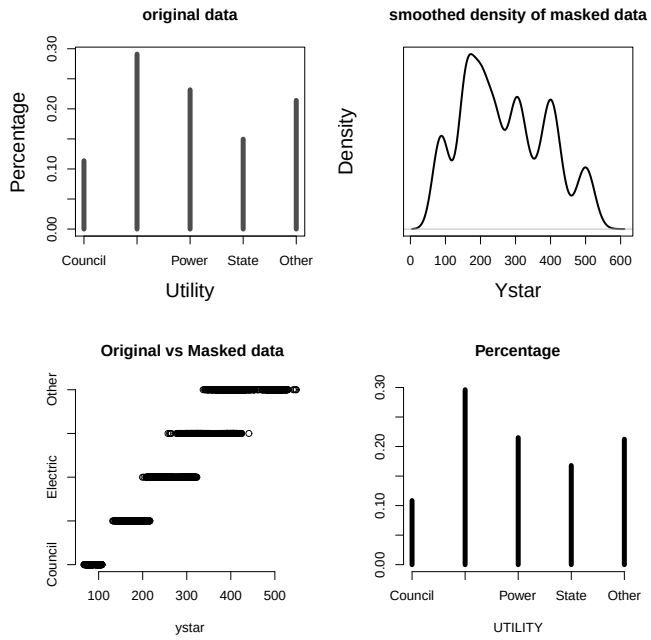

Figure 7: Top left panel: the plot of the true mass function of "UTILITY". Top right panel: the plot of the smoothing density function of masked data. Bottom left panel: the plot of original data vs masked data. Bottom right panel: the estimate of the mass function of "UTILITY" based on masked data.

Table 7: The percentage of each category calculated based on the original data and masked data, respectively

data source	Central	Council	Power	State	Other
y	0.1107143	0.2835714	0.2257143	0.1457143	0.2342857
$y1$	0.1114286	0.2776190	0.2309524	0.1654762	0.2145238

of the smoothing density function of "ystar" gives no information on "y". The plot "original vs masked", unobservable from the end-user, indicates that identifying the original values of "UTILITY" based on masked data is not straightforward, particularly if the noise information is not accessable. The plot of the approximant of the mass function (the bottom-right panel in Figure 7) is almost the same as the plot of mass function based on the original data. The data "$y1$", obtained by applying the "unmask" function to "$ystar$", is categorical. The end-user is able to obtain the mass function of "$y1$" by R. For comparison purpose, Table 7 lists the mass functions of "y" and "$y1$". The mass function of the original data is well estimated based on the masked data.

If the end-user wishes to obtain the information of the mass function of "UTIL-

Table 8: The percentages of each category calculated from the original data and masked data given by the first seven months.

data source	Central	Council	Power	State	Other
Original data	0.1165090	0.2815993	0.2377472	0.1534824	0.2106621
Masked data	0.1160791	0.2858985	0.2429063	0.1547721	0.2003439

ITY" given by particular months, this can be done by applying *MaskDensity10.R* to corresponding subset of the masked "UTILITY".

As an example, we consider the following scenario where the end-user wishes to estimate the mass function of "UTILITY" determined by the first seven months. Assume that the variable "MONTH" (denoted by "$x1$" in the R code below) in the original data set is not confidential. After received the masked data file ("ystar.dat"), noise file ("noise.bin") and the information for "MONTH", the end-user can use the following R code to create a subset of masked data for "UTILITY" given by the first seven months. Applying *MaskDensity10.R* to the subset, the end-user will obtain the approximant of the mass function for "UTILITY" conditional on the first seven months:

```
xx<-cbind(ystar,x1)
xx.sub<-subset(xx, x1<=7)
y2 <- unmask(xx.sub[,1], noisefile="noise.bin")
plot(table(y2)/length(y2), lwd=5, xlab="UTILITY", ylab="",
    main="Percentage (month<=7) based on masked data",
    axes=FALSE)
axis(1, at=1:5, labels=c("Central", "Council", "Other", "Power",
    "State" ))
axis(2)
```

The plots of the mass functions of "UTILITY" based on the first seven months data given by masked data and original data are presented in Figure 8 the top panel. The bottom panel in Figure 8 shows the percentage of the categories of "y" randomly mapped to the different categories of "$y1$" and the values of "y" are well protected through this manner. Table 8 gives the estimate mass functions based on masked data and original data. Two mass functions are close to each other.

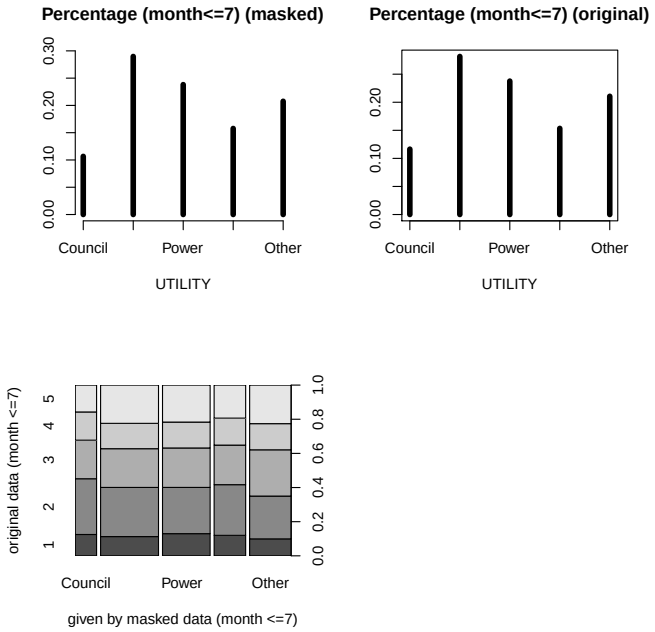

Figure 8: Top-left panel: the plot of the estimate of mass function based on masked data. Top-right panel: the plot of the mass function based on original data. Bottom panel: the percentage of each category mapping to different categories after data were masked.

5 Discussion

An R package, *MaskDensity10.R* used to implement the method of sample-moment-based density approximant proposed by Lin (2014), is built in this paper. By using the R package, the estimation of the summary statistics of the original data can be obtained based on noise multiplied data.

The advantages of the method of sample-moment-based density approximant and the R package built in this paper can be summarized as follows.

1. *One package for both parties*

   The same R package can be used by the data agency for generating masked data as well as by the end-user for estimating the density function based on masked data. The package is supported by free software R. It is convenient for both parties.

2. *No restriction on the type of distribution of the multiplicative noise*

   The method of sample-moment-based density approximant is independent of the probability distribution of the noise. It means the data agency has a wide

range of choices on the multiplicative noise.

3. *Obtaining density approximants for subsets of the original data*

   The R package provides the end-user an opportunity to explore the probability distribution of subsets of the original full data based on the information of the masked full data set.

   The inference results for the subset data are sensitive to the size of the underlying subset. The results are always reasonable accurate when the underlying variable is a categorical variable and the size of the underlying subset is reasonable larger. For continuous random variable, sometimes the density of approximant of a subset might be less accurate due to the boundaries determined by the R package, which might be too far away from the true boundaries given by the subset data.

Simulation studies carried out in this paper and Lin and Wise (2012) pointed out that, as long as the type of the multiplicative noise is appropriate, the level of protection on the underlying original data can be maintained regardless whether the probability distribution of the noise is publicly available. In this paper, the information of noise is encrypted into a binary file. With this manner, an extra protection is created for the underlying original data. A question might be raised why we do not encrypt the underlying original data into a binary file directly and provide the end-user an R package for producing the plot of the density function of the underlying original data by running the binary file in background. Our concern is that it might be too risk to issue confidential data to public directly in a binary file as no one is able to ensure binary files can be 100% safe from hackers. Under the approach proposed in this paper, only noise information is encrypted and underlying original data are masked through the noise. Therefore, the values of masked data cannot be 100% identified by the data intruder even if the data intruder is able to decrypt the binary noise file.

Simulation studies show the accuracy of the sample-moment-based density approximant might be affected by outliers in the original data and the sample of noise used to mask the original data. Many ways could be used to reduce the impact of the outliers or the sample of noise. The possible methods might include eliminating the outliers from the original data before the data were masked or using the functional mean of the sample-moment-based density functions to estimate the density function of the original data.

A key issue in the R package built in this paper is about the decision of the upper order of moment K in the density approximant. In this paper, the K is determined

based on the correlation between the masked data and the reproduced masked data. The accuracy of the density approximant can be further improved if a better way of determining the upper order K is found.

The method of the sample-moment-based density approximant proposed by Lin (2014) is for univeriate random variables, though method can be further developed for multivariate case. The R package built in this paper is based on the work in Lin (2014) and is built for univeriate. However, the package can be applied to multivariate categorical variables as the mass function of a multivariate categorical variable can be converted to a mass function of the univeriate categorical variable.

References

[1] Duncan, G. T. and Lambert, D. (1986). Disclosure limited data dissemination (with comment), *Journal of the American Statistical Association*, **81**, 1-28.

[2] Evans, T.(1996). Effects on Trend Statistics of the Use of Multiplicative Noise for Disclosure Limitation, US Bureau of the Census, *http* : *//www.census.gov/srd/sdc/papers.html*, accessed 5/12/2008.

[3] Evans, T., Zayatz, L. and Slanta, J. (1998). Using Noise for Disclosure Limitation of Establishment Tabular Data, *Journal of Official Statistics*, **14**, 537-551.

[4] Hwang, J. T. (1986). Multiplicative errors-in-variables models with applications to recent data released by the U.S. Department of Energy, *Journal of the American Statistical Association*, **81**, 680-688.

[5] Kim, J. J. and Winkler, W.E. (2003). Multiplicative Noise for Masking Continuous Data, Research Report Series (Statistics ♯2003-01), Statistical Research Division, U.S. Bureau of the Census, Washington D.C. 20233.

[6] Kim, J.J. and Jeong, D. M. (2008) Truncated triangular distribution for multiplicative noise and domain estimation, Section on Government Statistics - JSM 2008, 1023-1030.

[7] Klein, M. and Sinha, B (2013). Statistical analysis of noise-multiplied data using multiple imputation. *Journal of Official Statistics*, **29**, 425-465.

[8] Krisinich, F. and Piesse, A. (2002). Multiplicative Microdata Noise for Confidentialising Tables of Business Data: Application to AES99, Data with a Comparison

to Cell Suppression, Research and Analytical Report 2002 #19, Statistics New Zealand.

[9] Lin, Y.-X. and Wise, P. (2012). Estimation of regression parameters from noise multiplied data, *Journal of Privacy and Confidentiality*, **4**, 55-88.

[10] Lin, Y.-X. (2014) Density approximant based on noise multiplied data. In J. Domingo-Ferrer (ed.), PSD 2014, LNCS **8744**, 89-104, Springer International Publishing Switzerland.

[11] Malvestuto, F. and Moscarini, M.(2003), Privacy in Multidimensional Databases, IDEA GROUP PUBLISHING, http://www.irma-international.org/viewtitle/26973/

[12] Nayak, T. K., Sinha, B. and Zayatz, L.(2011). Statistical properties of multiplicative noise masking for confidentiality protection, *Journal of Official Statistics*, **27**, 527-544.

[13] Oganian, A. (2010). Multiplicative noise protocals. In J. Domingo-Ferrer and E. Magkos (eds.), PSD 2010, LNCS **6344**, 107-117, Springer, Heidelerg.

[14] Provost, S. B. (2005). Moment-Based Density Approximants, *The Mathematica Journal*, **9**, 728-756

[15] Shlomo, N. (2010). Releasing Microdata: Disclosure Risk Estimation, Data Masking and Assessing Utility, *Journal of Privacy and Confidentiality*, **2**, 73-91.

[16] Sinha, B. Nayak, T.K. and Zayatz, L. (2011). Privacy protection and quantile estimation from noise multiplied data, Sankhya B, **73**, 297-315.

[17] Willenborg, L and de Waal, T. (2001). Elements of Statistical Disclosure Control, vol. 155 of *Lecture Notes in Statistics*. New York: Springer-Verlag.