SORT Special issue: Privacy in statistical databases, 2011, 77-98

The Microdata Analysis System at the U.S. Census Bureau*

Jason Lucero, Michael Freiman, Lisa Singh, Jiashen You,
Michael DePersio and Laura Zayatz

Abstract

The U.S. Census Bureau has the responsibility to release high quality data products while maintaining the confidentiality promised to all respondents under Title 13 of the U.S. Code. This paper describes a Microdata Analysis System (MAS) that is currently under development, which will allow users to receive certain statistical analyses of Census Bureau data, such as cross-tabulations and regressions, without ever having access to the data themselves. Such analyses must satisfy several statistical confidentiality rules; those that fail these rules will not be output to the user. In addition, the *Drop q Rule*, which requires removing a relatively small number of units before performing an analysis, is applied to all datasets. We describe the confidentiality rules and briefly outline an evaluation of the effectiveness of the *Drop q Rule*. We conclude with a description of other approaches to creating a system of this sort, and some directions for future research.

MSC: 62P20, 62Q05, 68P20

Keywords: Statistical disclosure limitation, Microdata analysis, Information retrieval, Cross-tabulations.

1. Introduction

The U.S. Census Bureau collects its survey and census data under Title 13 of the U.S. Code, which prevents the Census Bureau from releasing any data "...whereby the data furnished by any particular establishment or individual under this title can be identified." In addition to Title 13, the Confidential Information Protection and Statistical Efficiency Act of 2002 (CIPSEA) requires the protection of information collected or acquired for exclusively statistical purposes under a pledge of confidentiality. However, the agency

[*] This report is released to inform interested parties of ongoing research and to encourage discussion of work in progress.

Disclaimer: The views expressed are those of the authors and not necessarily those of the U.S. Census Bureau.

Received: November 2010
Accepted: March 2011

also has the responsibility of releasing data for the purpose of statistical analysis. In common with most national statistical institutes, our goal is to release as much high quality data as possible without violating the pledge of confidentiality, as described in Duncan et al. (2001) and Kaufman et al. (2005).

This paper discusses a Microdata Analysis System (MAS) that is under development at the U.S. Census Bureau. Much of the framework for the system was described in Steel and Reznek (2005) and Steel (2006). The system is designed to allow data users to perform various statistical analyses (regressions, cross-tabulations, correlation coefficients, etc.) on confidential survey and census microdata without seeing or downloading the underlying microdata.

In Section 2, we give some background on the MAS and the motivation for its development. In Section 3, we discuss the current state of the prototype system, including its capabilities and the rules that protect confidentiality. In Section 4, we briefly summarize a study of the effectiveness of the *Drop q Rule*, one of the disclosure avoidance measures taken within the system. In Section 5, we examine some other approaches to the problem of creating a remote access system such as the MAS. In Section 6, we conclude with remarks on future research and the further development of the system.

2. Background on the MAS

The Census Bureau conducts reidentification studies on our public use microdata files. In these studies, we attempt to link our public use files to external files that contain identifiers. It is reasonable to expect that with more publicly available data and expanded use of data mining tools, there will be an increase in the number and complexity of confidentiality threats. There is some concern that in order to meet the confidentiality requirements under which the Census Bureau operates, we may have to reduce the detail available in our data products and use more perturbation techniques to protect them, thus degrading the quality of the data.

This problem of data confidentiality—at the Census Bureau and other statistical agencies around the world—has motivated the creation of *remote access systems* which allow the user to request a statistical analysis and receive the result without having direct access to the underlying microdata. Common to almost all remote access systems is that the ability to receive desired results is not absolute: in some instances, the result might be based on perturbed data, and most proposals for remote access systems include the rejection of some queries to preserve confidentiality. The idea of a remote access system goes back at least to Keller-McNulty and Unger (1998), although the concept of allowing customized queries was proposed much earlier; see the description of the Geographically Referenced Data Storage and Retrieval System in Fellegi et al. (1969). Fellegi (1972) anticipates the need to screen the query results to ensure that confidentiality is adequately protected. Adam and Worthmann (1989) describe several restrictions on systems that release counts of numbers of people with particular characteristics. These include suppressing counts if the numbers are too close to 0 or to the full size of the database; requiring that multiple queries from the same user have only limited overlap; and keeping a log of each user's queries and checking each new query against the log to verify nondisclosure. However, they acknowledge that the last of these is sufficiently time consuming and storage intensive as to be unfeasible. They also consider the possibility of partitioning the data into indivisible units of two or more observations each and allowing only queries that operate on unions of the units, rather than on arbitrary sets of observations.

The Microdata Analysis System will allow the U.S. Census Bureau to provide a controlled, cost-effective setting in which data users have access to more detailed and accurate information than is currently available in our public use microdata files. The data accessible through the MAS can identify smaller geographic areas and show more detail in certain variables where our public use files would be coarsened. Our goal for the MAS is to allow access to as much high quality data as possible. An advantage of the MAS is that it lessens the need for data to be released in less secure or more expensive manners, such as those described in Weinberg et al. (2007). A predecessor of the MAS is discussed in Rowland and Zayatz (2001).

Unlike the proposal in Schouten and Cigrang (2003), our plan is to make the MAS available to anyone who wishes to use it. In a sense, the MAS will serve as a Research Data Center for the entire public, although there will be restrictions in place that a qualified researcher would not encounter at an established Research Data Center. The MAS will allow access to data from demographic surveys and decennial censuses, with the goal of eventually including economic survey and census data, as well as linked datasets. We will initially make available regression analyses and cross-tabulations, with other analyses to be added in the future. Currently, we intend to keep a record of all of the queries entered into the system, but not the identities of the users making the queries. Although the record will not directly affect the output that the system provides, it will allow us to see how the system is being used. Our goal in doing this is to improve the user experience and enhance the disclosure avoidance techniques if necessary.

Our current plan—as described in Chaudhry (2007)—is to offer the MAS through the Census Bureau's free DataFERRETT service with the intention that the system will be used by people needing fairly simple statistical analyses: news media, some policy makers, teachers, students, etc. The MAS has a graphical interface that allows users to select variables of interest from a list. In the case of regression, variables can be dragged into equations and, with a few clicks, users may create variable interactions and transformations of selected variables. Some users may feel the need to use the underlying confidential microdata for more exploratory data analysis, but it is not apparent how to allow this within the MAS without violating confidentiality. These users may find our public use files, when available, meet their needs if they account for the decreased accuracy inherent in our disclosure avoidance procedures. Having a limited range of allowable analyses is a weakness of the MAS, but, other than expanding the number of off-the-shelf analyses the system offers, it is difficult to see how to remedy it.

3. Overview of the MAS Confidentiality Rules

In 2005, the Census Bureau contracted with Synectics to develop an alpha prototype of the MAS using the SAS language. We also contracted with Dr. Jerome Reiter of Duke University to help in developing the confidentiality rules of the system and with Dr. Stephen Roehrig of Carnegie Mellon University to help in testing these rules. Some rules were developed and modified as a result of the testing. The beta prototype of the MAS implements a Java interface within DataFERRETT, which submits requested analyses to an R environment. We are using the publicly available data from the Current Population Survey March 2008 Demographic Supplement to test the system.

The MAS software is programmed with several confidentiality rules and procedures that uphold disclosure avoidance standards. The purpose of these rules and procedures is to prevent data intruders from reconstructing the microdata records of individuals within the underlying confidential data through submitting multiple queries. The confidentiality rules discussed in this section are quite complex, and this discussion does not delve into the complexities. More detail can be found in Lucero (2009, 2010a). All analyses are subjected to two logical checks, referred to as the *No Marginal 1 or 2 Rule* and the *Universe Gamma Rule*, which ensure that no query is answered if the universe is too small or if the universe can be used to carry out differencing attacks by comparing results of similar universes. Regression analyses are further subjected to restrictions on the use of predictor and response variables. We plan to explore whether additional rules are necessary for correlation coefficients.

3.1. Confidentiality Rules for Universe Formation

MAS users are allowed to run their statistical analyses on a universe, or sub-population, of interest. Users are presented with a set of variables and category levels from which they can define a universe using condition statements on the variables. For example, if the user selects $gender = 2(female)$ from the metadata, the universe is defined to be the sub-population of all females. A slightly more complicated universe is $gender = 1(male) \vee employment\ status = 0(unemployed)$. One of the confidentiality rules requires that all variables used to define universes must be categorical.

Since a user may want to define a universe based on variables that are not inherently categorical (i.e., those that are continuous), raw numerical variables are presented to the user as categorical recodes based on output of a separate binning routine. This cutpoint program, outlined in Lucero et al. (2009b), creates bins of numerical values and ensures a pre-specified minimum number of observations between any two cutpoint values. Section 3.1.3 describes possible ways to generate cutpoints.

To define a universe using a numerical variable, a user is forced to choose from a predetermined list of ranges the range that best meets her goal. For example, if a user wished to run analysis on people with $income$ = \$46,000, the user would select the metadata $income = 4$, which is the range $(\$45,000, \$53,000]$ on the variable $income$

Table 1: Table representation of the universe defined from (1) and (2).

	income				
gender	\$0 to \$28,000	\$28,000 to \$39,000	\$39,000 to \$45,000	\$45,000 to \$53,000	Total
male	$n_{1,1}$	$n_{1,2}$	$n_{1,3}$	$n_{1,4}$	$n_{1,.}$
female	$n_{2,1}$	$n_{2,2}$	$n_{2,3}$	$n_{2,4}$	$n_{2,.}$
Total	$n_{.,1}$	$n_{.,2}$	$n_{.,3}$	$n_{.,4}$	$n_{.,.}$

and defines the universe as the sub-population of all individuals whose income is between \$45,000 and \$53,000. Note that a user cannot define the universe to be the range $\mathit{income} = (\$39,000, \$46,000]$ unless \$39,000 and \$46,000 are among the predetermined cutpoints. The user must choose a range of values consistent with the cutpoints that are given. This is a crucial restriction on what a user can do, since allowing arbitrary universe formation on continuous data could lead to a differencing attack disclosure, as described in Section 3.1.2.

3.1.1. Confidentiality by Minimum Universe Size Requirements

To define a universe in the MAS, the user would first select m recoded variables from the metadata, then select up to j bins for each of the m recoded variables. Universe formation on the MAS is performed using an implicit table server. For example, suppose a data user defines the universe as the union:

$$\mathit{gender} = \text{female AND } \$45,000 < \mathit{income} \leq \$53,000 \tag{1}$$

OR

$$\mathit{gender} = \text{male AND } \$28,000 < \mathit{income} \leq \$45,000 \tag{2}$$

This universe is represented as selected cells from a two-way table of counts for *gender* and *income*, as shown in *Table 1*. Note that there are $n_{2,4} + n_{1,2} + n_{1,3}$ total observations in this universe. For convenience, we will use the notation $\mathrm{U}(n)$ to denote a universe with n observations. In most cases, it should be clear from the context which n observations lie in the universe. In this example, the universe defined as the union of (1) and (2) will be referred to as $\mathrm{U}(n_{2,4} + n_{1,2} + n_{1,3})$.

In describing universes, we make a distinction between a simple universe and a complex universe. A simple universe is one that can be described using variable categories and the intersection set operator. A complex universe is constructed as the union of multiple simple universes.

All universes formed on the MAS must pass two confidentiality rules: the *No Marginal 1 or 2 Rule* and the *Universe Gamma Rule*. If a universe violates either of these rules, the MAS will reject the universe query and prompt the user to modify his selections. These rules are tested prior to performing the user's selected statistical analysis on the defined universe.

The *No Marginal 1 or 2 Rule* requires that for a universe defined using m variables, there may not be an $m-1$ dimensional marginal total equal to 1 or 2 in the m-way contingency table induced by the chosen variables. The universe $\mathrm{U}(n_{2,4}+n_{1,2}+n_{1,3})$ passes the *No Marginal 1 or 2 Rule* if:

$$(n_{i,.} \geq 3 \text{ OR } n_{i,.} = 0, \text{ for } i = 1,2) \text{ AND } (n_{.,j} \geq 3 \text{ OR } n_{.,j} = 0, \text{ for } j = 1,\ldots,4)$$

The *Universe Gamma Rule* requires that a universe must contain at least Γ observations; otherwise no statistical analysis will be performed. The value of Γ is not given here since it is Census confidential.

The way this rule is checked is dependent on whether the universe is disjoint or joint. A universe is classified as *disjoint* if its individual pieces do not share cell counts in common. For example, pieces (1) and (2) for the universe $\mathrm{U}(n_{2,4}+n_{1,2}+n_{1,3})$ are disjoint. Since $\mathrm{U}(n_{2,4}+n_{1,2}+n_{1,3})$ is a disjoint universe, the MAS would check that piece (1) and piece (2) each contain at least Γ observations. Note that the cutpoint bins of *income* are combined within piece (2) prior to performing the test; however, bins representing different classes of an inherently categorical variable would not be combined. In this case, since the $n_{1,2}$ and $n_{1,3}$ bins differ from each other only by a cutpoint variable, they are combined, and the MAS checks:

$$n_{2,4} \geq \Gamma \text{ AND } (n_{1,2}+n_{1,3}) \geq \Gamma$$

A universe is classified as *joint* if at least one of its individual pieces shares cell counts in common with at least one other piece. For example, suppose the user defines the universe $\mathrm{U}(n_{2,.}+n_{1,3}+n_{1,4}) = (3)$ OR (4), where (3) and (4) are given by

$$[\textit{gender} = \text{ female}] \tag{3}$$

$$[\$39,000 < \textit{income} \leq \$53,000] \tag{4}$$

In this case, the observations in $n_{2,3}$ and $n_{2,4}$ — females with income in the interval (\$39,000 , \$53,000] — are included in both pieces (3) and (4). See *Table 2*. Since $\mathrm{U}(n_{2,.}+n_{1,3}+n_{1,4})$ is a joint universe, the *Universe Gamma Rule* would first check that pieces (3) and (4) contain at least Γ observations, following the disjoint universe scenario. Next, the intersection $I = (3) \cap (4) \neq \{\}$ would be checked to determine that I contains at least Γ^* observations, where $\Gamma^* \leq \Gamma$ is another Census confidential parameter. In this example, the MAS checks that the following inequalities are satisfied before any results will be returned:

$$n_{2,.} \geq \Gamma \text{ AND } (n_{.,3}+n_{.,4}) \geq \Gamma \text{ AND } (n_{2,3}+n_{2,4}) \geq \Gamma^*$$

Once again, the cutpoint bins of income are first combined within piece (4) and within I prior to the testing of the *Universe Gamma Rule*. In general, when a joint universe

Table 2: Table representation of the universe defined from (1) and (2).

	income				
gender	\$0 to \$28,000	\$28,000 to \$39,000	\$39,000 to \$45,000	\$45,000 to \$53,000	Total
male	$n_{1,1}$	$n_{1,2}$	$n_{1,3}$	$n_{1,4}$	$n_{1,\cdot}$
female	$n_{2,1}$	$n_{2,2}$	$n_{2,3}$	$n_{2,4}$	$n_{2,\cdot}$
Total	$n_{\cdot,1}$	$n_{\cdot,2}$	$n_{\cdot,3}$	$n_{\cdot,4}$	$n_{\cdot,\cdot}$

$$\begin{array}{c|cc} \mathrm{T}_n & ES_1 & ES_2 \\ \hline G_1 & n_{1,1} & n_{1,2} \\ G_2 & n_{2,1} & n_{2,2} \end{array} - \begin{array}{c|cc} \mathrm{T}_{n-1} & ES_1 & ES_2 \\ \hline G_1 & n_{1,1} & n_{1,2}-1 \\ G_2 & n_{2,1} & n_{2,2} \end{array}$$

$$= \begin{array}{c|cc} \mathrm{T}_1 & ES_1 & ES_2 \\ \hline G_1 & 0 & 1 \\ G_2 & 0 & 0 \end{array}$$

Figure 1: *An Example of a Differencing Attack Disclosure.*

is considered, all of the non-empty intersections of the pieces of the universe must be checked to make sure they are sufficiently large.

3.1.2. Confidentiality by Random Record Removal

While the preceding rules provide some protection of the confidential data in the MAS, they do not completely prevent differencing attack disclosures. A *differencing attack disclosure* occurs when a data intruder attempts to reconstruct a confidential microdata record by subtracting the statistical analysis results obtained through two queries on similar universes. Suppose a data intruder first creates two universes on the MAS, $\mathrm{U}(n)$ and $\mathrm{U}(n-1)$ (a proper subset of $\mathrm{U}(n)$), where both contain the same n observations less one unique observation, i.e., $|\mathrm{U}(n)\backslash\mathrm{U}(n-1)| = 1$. The difference $\mathrm{U}(n)\backslash\mathrm{U}(n-1) = \mathrm{U}(1)$ is a manipulated universe that contains the single target observation. For illustration, suppose a data intruder has prior knowledge of demographics in a small geographic area, and in particular is aware of individuals, households or establishments with unique characteristics within that area. It may be the case that there is only one non-citizen among the n residents of the area. Then the intruder may create $\mathrm{U}(n)$ and $\mathrm{U}(n-1)$, where $\mathrm{U}(n)$ is the full universe of people in the area and $\mathrm{U}(n-1)$ is the universe consisting of citizens who live in the area. Suppose the data intruder then requests two separate cross-tabulations for gender by employment status on these universes, T_n and T_{n-1}, as shown in *Figure 1*. Since $\mathrm{U}(n)$ and $\mathrm{U}(n-1)$ differ by a unique observation, T_{n-1} will be exactly the same as T_n, less one unique cell count.

We may perform the matrix subtraction $\mathrm{T}_n - \mathrm{T}_{n-1} = \mathrm{T}_1$, where T_1 is a two-way table of gender by employment status built upon the one unique observation contained in

$\mathrm{U}(n)\backslash\mathrm{U}(n-1)=\mathrm{U}(1)$. As shown in *Figure 1*, T_1 contains a cell count of 1 in the male non-employed cell with zeros in the remaining cells, which tells the data intruder that the one unique observation contained in U(1) is an unemployed male. By performing differencing attacks similar to the one just described, a data intruder can successfully rebuild the confidential microdata record for the one unique observation contained in U(1).

A differencing attack may also be a concern if there are two observations within an area that have a certain characteristic, particularly if the intruder is himself one of these two. Suppose, for example, that the universe contains only two non-citizens, one of whom is the intruder. The intruder could then construct the full universe $\mathrm{U}(n)$ and the portion of the universe consisting solely of citizens $\mathrm{U}(n-2)$. Since the intruder knows his own personal characteristics, he may manually remove himself from $\mathrm{U}(n)$ to get $\mathrm{U}(n-1)$ and then perform a differencing attack as above by comparing $\mathrm{U}(n-1)$ and $\mathrm{U}(n-2)$ to obtain information on the other non-citizen in the area.

To help protect against differencing attacks, the MAS implements a universe subsampling routine called the *Drop q Rule*. Traditionally, subsampling has usually been used to estimate parameters when a population is too large to analyze in an efficient manner and a (usually small) subset can give approximately the same results as the full population. Our aims are very different here: the *Drop q Rule* is intended to remove just enough observations from the dataset to thwart a differencing attack. In most cases, a differencing attack performed while the *Drop q Rule* is in place will not lead to a meaningful outcome, and even when it does, the intruder cannot be sure that the outcome found is the correct one.

The *Drop q Rule* works as follows. A user-defined universe that passes all of the previous rules has q records removed at random. To do this, the MAS will first draw a random value of $Q_v = q_1 \in \{2,\ldots,k\}$ from a discrete uniform distribution with probability mass function $\mathrm{P}(Q_v = q_1) = \frac{1}{k-1}$. Then, given $Q_v = q_1$, the MAS will subsample the universe $\mathrm{U}(n)$ by removing q_1 records at random from $\mathrm{U}(n)$ to yield a new subsampled universe $\mathrm{U}(n-q_1)$.

Within the MAS, all statistical analyses are performed on the subsampled universe $\mathrm{U}(n-q_1)$ and not on the original universe $\mathrm{U}(n)$. Each unique universe $\mathrm{U}(n)$ that is defined on the MAS will be subsampled independently according to the *Drop q Rule*. To prevent an "averaging of results" attack, the MAS will produce only one subsampled universe $\mathrm{U}(n-q_1)$ for each unique universe $\mathrm{U}(n)$, with this unique subsample persisting for the lifetime of the system. That is, all users who select a specific universe $\mathrm{U}(n)$ will have all analyses performed on exactly the same subsampled universe $\mathrm{U}(n-q_1)$. To avoid obvious storage issues, the MAS accomplishes consistent subsampling of universes by using the same random seed to perform the subsampling every time a given universe comes up. To receive the full disclosure protection offered by the *Drop q Rule*, it is necessary that the seed, while constant for a given universe, differs across universes, and this can be implemented by having the seed be a function of the set of units in the universe.

The discrete uniform distribution is ideal for this purpose because of all distributions on $\{2,\ldots,k\}$, it minimizes the probability that for two similar universes, the number of observations dropped will be the same for both universes, which is a necessary condition for an apparent disclosure to be made on a single observation.

Because each value used in the *Drop q Rule* is drawn from a discrete uniform distribution, a data intruder attempting the difference attack $\mathrm{T}_n - \mathrm{T}_{n-1} = \mathrm{T}_1$ may find results inconsistent with forming two universes where $\mathrm{U}(n-1) \subset \mathrm{U}(n)$, as shown in *Figure 2*. The values of x_{ij} are the random numbers giving the number of observations dropped from each cell of $\mathrm{U}(n)$ in forming $\mathrm{U}(n-q_1)$. Similarly, the values of y_{ij} are the number of observations dropped from each cell of $\mathrm{U}(n-1)$ in forming $\mathrm{U}(n-1-q_2)$ respectively. Hence:

$$\sum_i \sum_j x_{ij} = q_1, 0 \leq x_{ij} \leq q_1$$

$$\sum_i \sum_j y_{ij} = q_2, 0 \leq y_{ij} \leq q_2$$

Here, *i* and *j* index the rows and columns, respectively, of the contingency table, with the obvious generalizations involving higher order multiple sums for higher-dimensional data. The resulting table $\mathrm{T}_?$ *may* yield a successful disclosure of *gender* $= G_1$ (male) AND *employment status*$= ES_2$ (unemployed) for the one unique observation contained in U(1), but it is much more likely to supply nonsense to the intruder. Coupled with the difficulty of finding candidate differencing attack universes, data intruders will find their time better spent elsewhere. Section 4 contains a brief overview of the effectiveness of the *Drop q Rule* against differencing attack disclosures. The rule is a crucial part of our disclosure prevention strategy. The contracted work described by Roehrig et al. (2008) found several instances in which a prototype version of the MAS lacking this rule was susceptible to differencing attacks, not just in theory but also in practice. However, their approach was to run a large number of tabulation queries and search for universes that were almost the same. This method could be partly deterred by slowing down the system, requiring a wait time between each user query.

$$\begin{array}{r|cc} \mathrm{T}_{n-q_1} & ES_1 & ES_2 \\ \hline G_1 & n_{1,1}-x_{1,1} & n_{1,2}-x_{1,2} \\ G_2 & n_{2,1}-x_{2,1} & n_{2,2}-x_{2,2} \end{array} - \begin{array}{r|cc} \mathrm{T}_{n-1-q_2} & ES_1 & ES_2 \\ \hline G_1 & n_{1,1}-y_{1,1} & n_{1,2}-1-y_{1,2} \\ G_2 & n_{2,1}-y_{2,1} & n_{2,2}-y_{2,2} \end{array}$$

$$= \begin{array}{r|cc} \mathrm{T}_? & ES_1 & ES_2 \\ \hline G_1 & y_{1,1}-x_{1,1} & 1+y_{1,2}-x_{1,2} \\ G_2 & y_{2,1}-x_{2,1} & y_{2,2}-x_{2,2} \end{array}$$

Figure 2: *Differencing Attack Thwarted by the* Drop q Rule.

The *Drop q Rule* is a generalization of the previously used *Drop 1 Rule* and *Drop 2 Rule*, where a small and fixed number of observations were removed before analysis. These rules led to tables that were susceptible to differencing attacks. One notable vulnerability could be exploited by starting, as usual, with two universes U(n) and U($n - 1$), identical with the exception of one unit, with the intention of performing a differencing attack. For example, an intruder might know that a certain geographical region contains exactly one Korean War veteran. The intruder could then consider the universe of all people in that region, as compared to the universe of all non-Korean War veterans in the region. However, instead of requesting a tabulation of these two universes, the intruder may augment each universe by adding to it the full population of a non-overlapping geographical region of size $N >> n$, such as a large state that does not contain the original region. Then a three-way tabulation could be done of veteran status versus state versus the variable that the intruder wishes to disclose for the augmented universes U($n+N$) and U($n-1+N$). In the case of the *Drop 2 Rule*, it is overwhelmingly likely that all four of the dropped observations will be in the large region of size N, thus leaving the portions of the provided tables representing the original region of interest unmodified. We are currently examining other disclosure rules to prevent this sort of "padding" attack.

A differencing attack leads to a correct inference when the difference between the two matrices represented by the modified tables contains a 1 in the correct cell and 0s in all other cells. In most cases, when the *Drop q Rule* is used, there are cells with both positive and negative numbers, and no inference can be reached by the intruder. It is also possible to obtain an apparent—but incorrect—inference, which occurs when the difference is a table with a 1 in one cell and 0s in all of the others, but the 1 is not in the correct cell.

3.1.3. Cutpoint Methods

The cutpoints used in universe formation in the MAS are generated by a separate program. Various methods exist in the program, and each provides a different set of cutpoints, as influenced by the empirical distribution of a variable. The methods implemented are fixed width, minimum width, increasing width, and partitioned binning. Cutpoints for each variable in the dataset can use a different strategy, but the final cutpoints for a given variable are generated only once, after choosing an appropriate strategy. What follows is a basic description of each strategy.

Fixed width binning ensures that all bins have the same width. This is implemented as finding a constant ω_{FW}, such as 10, so that the distance from the minimum value to the maximum value of each bin will be ω_{FW}. Because bin widths are constant, the number of observations in each bin will vary, causing some bins to be sparsely populated while others are dense. The fixed width is chosen to be the minimum value ω_{FW} such that all bins contain at least β_{FW} observations, for some pre-determined value β_{FW}. This can make ω_{FW} large, so that the resolution across dense areas of the data is too crude.

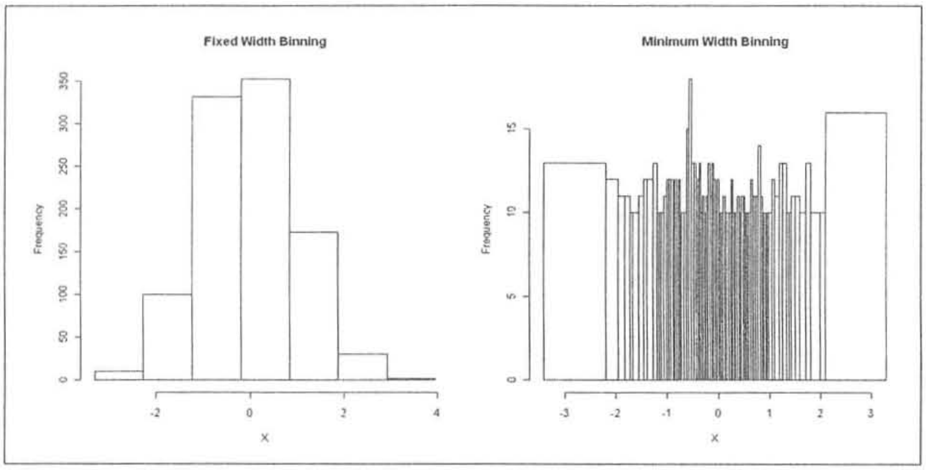

Figure 3: Fixed and Minimum Width Binning on 1,000 N(0,1) random samples.

In data following a Gaussian distribution, the bin width will be determined by the tails and the center bins will be quite dense.

Minimum width binning uses a value β_{MW} and creates bins such that each has as close to β_{MW} observations as possible. Identical realizations of the variable will not be split across multiple bins. For example, considering a numerical variable X with support $\mathbb{N}$, all observations with $X = 5$ will belong to the same bin regardless of the number of observations with $X = 5$. This approach tends to generate bins of smaller width than other approaches, since it allows for finer resolution in dense areas of the data but allows the bins to be much wider when covering sparse data in order to include at least β_{MW} observations.

Increasing width binning may be viewed as a compromise between fixed and minimum width binning. Increasing width binning starts with a fixed bin width, ω_{IW}, which gradually increases as the value of the variable increases. This corrects the problem in fixed width binning of bins tending to be large, while also allowing for a consistent bin width, which one does not get in minimum width binning. Considering income data, ω_{IW} might equal $25,000$ at $X = 0$, but when the cutpoint reaches $X = 100,000$, ω_{IW} may jump to 150,000 as a way to deal with sparser data in the tails. For sufficiently large X, we obtain a value of $\omega_{IW} = \infty$ once the number of remaining observations approaches some value $\alpha < 2\beta_{IW}$.

The previous binning methods are all referred to as bottom-up methods since they begin with some width value and starting point in the data and build bins from there. Alternatively, *partitioned binning* is a top-down binning strategy in that it uses the data as a whole in creating bins. Partitioned binning begins by sorting the data and then splits the set into two subsets containing approximately the same number of observations. These two subsets are themselves each split into two smaller subsets in the same fashion.

Table 3: *Bins created on the dataset* $\{1,1,2,2,4,4,5,6\}$.

Method	Bin 1	Bin 2	Bin 3	Bin 4
Fixed W.	1-2	3-4	5-6	NA
Min. W.	1-1	2-2	4-4	5-6
Inc. W.	1-2	3-6	NA	NA
Partitioned	1-1	2-2	4-4	5-6

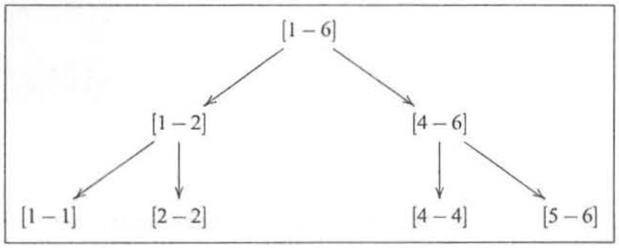

Figure 4: *Partitioned Binning on dataset* $\{1,1,2,2,4,4,5,6\}$.

This process continues as long as there are at least β_{PW} observations in each bin. The final result is a binary tree of bins of unequal width.

As a quick example of how each method performs on the same data, consider a dataset 1,1,2,2,4,4,5,6. *Table 3* shows the cutpoints, or boundaries, for each bin that the different algorithms will create. Assume that the minimum number of elements in each bin is $\beta_{MIN} = 2$.

The binary tree for the partitioned binning is shown in *Figure 4*. A user may choose pieces for the universe using any node shown in the diagram.

Each approach has its own strengths and weaknesses, so which performs best on a given variable depends both on the variable's support and distribution and on the properties desired by the user. However, none of the methods considers the underlying distribution of a variable in building the bins, so there is a necessity to analyze the performance of a chosen method. Consider how each would perform on a Gaussian distribution. Fixed width binning may not provide the resolution desired around the mean, and increasing width binning is primarily useful when the probability density function of the variable in question is decreasing over most of the range of the variable. Partitioned and minimum width binning will produce similar results, but the cutpoints in the minimum width and partitioned approaches may provide binning so fine that the exact values for some records are at risk.

3.2. Confidentiality Rules for Regression Models

The MAS implements a series of confidentiality rules for regression models, in addition to the universe restrictions already mentioned. For example, users may only select up to 20 independent variables for any single regression equation. Users are allowed to transform numerical variables only, and they must select their transformations from a pre-approved list. This prevents the user from performing transformations that deliberately overemphasize individual observations such as outliers. Currently, the allowable transformations are square, square root and natural logarithm.

Any fully interacted regression model that contains only dummy variables as predictors poses a significant potential disclosure risk, as described in Reznek (2003) and Reznek and Riggs (2004). Therefore, users are allowed to include only two-way and three-way interaction terms within any specified regression model, and no fully interacted models are allowed. Furthermore, a two-way interaction is allowed only if both of the interacted variables appear by themselves in the model, and a three-way interaction is allowed only if all three variables appear uninteracted in the model *and* each of the three associated two-way interactions appears. However, interactions do not count against the 20-variable limit (so that, for example, if a model includes two predictor variables and their interaction, this is considered two variables, not three, for the purpose of the limit). Categorical predictor variables are included in the model through the use of dummy variables for all categories except one reference category. The MAS uses the most common category as the reference category. In addition, each predictor dummy variable must represent a category containing a certain minimum number of observations; if this minimum is not met, the dummy variable is omitted from the model. In effect, this means that very sparse categories are absorbed into the reference category level. The minimum allowable number of observations in a category is not given here since it is Census confidential.

Prior to passing any regression output back to the user, the MAS also checks that R^2 is not too close to 1. If R^2 is too close to 1, then the MAS will suppress the output of the regression analysis, as releasing the results of the regression would allow estimation of the response variable with a high degree of accuracy if the values of the predictor variables for any unit were known. It may also be the case that the regression does not have an unreasonably high R^2, but that there is a subset of units for whom the response variable can be predicted unusually well given the predictor variables. Regressions with this feature may also be suppressed. The system may also suppress instances where an interaction term leads to a sparse combination of categories, as this may be a disclosure risk. If all of these requirements are satisfied, then the MAS will pass the estimated regression coefficients and the Analysis of Variance (or Deviance) table to the user without restrictions (except for the absorption of categories mentioned above). If the requirements are not satisfied, the system may attempt to absorb additional categories of any categorical predictors into the reference category, as this may result in a regression whose output is allowed to be released.

Sparks et al. (2008) propose some other confidentiality rules for regression, such as using robust regression to lessen the influence of outliers, although at the moment we still plan to use ordinary least squares regression when the response variable is numerical.

3.2.1. Synthetic Residual Plots

To determine whether the regression adequately describes the data, diagnostics such as residual plots are necessary. Actual residual values pose a potential disclosure risk, since a data intruder can obtain the values of the dependent variable by simply adding the residuals to the fitted values obtained from the regression model. Therefore, the MAS does not pass the actual residual values back to the user. To help data users assess the fit of their ordinary least squares regression models, diagnostic plots are based on synthetic residuals and synthetic real values. These plots are designed to mimic the actual patterns seen in the scatter plots of the real residuals versus the real fitted values, or of the real residuals versus the values of the individual variables.

The first step in creating synthetic residual plots is to create the synthetic dataset in such a way that the synthetic data mimic the actual data. Using the notation of Reiter (2003), let $\boldsymbol{x}_p$ be a variable in the collected dataset, for $p = 1, \ldots, d$. In the synthetic dataset, $\boldsymbol{x}_p^s$ corresponds to the original $\boldsymbol{x}_p$ variable, with the superscript s indicating the use of a synthetic dataset. There are various methods to generate $\boldsymbol{x}_p^s$, but this discussion will follow the method described in Reiter (2003), both for creating synthetic data and for creating synthetic residuals, and our exposition and notation here mostly follow his.

For categorical variables $\boldsymbol{x}_p$, $\boldsymbol{x}_p^s$ are generated from bootstrap sampling the collected data. If some categories are sparsely populated, there is the potential for averaging the synthetic residual values at the sparse category to disclose real residuals, but otherwise this part of the algorithm poses negligible disclosure risk. One possible approach to this problem is to suppress residuals for categories that are sufficiently sparse. For continuous variables $\boldsymbol{x}_p$, the distribution of the variable is approximated non-parametrically using a kernel density estimator, and then inverse-cdf sampling is used to generate $\boldsymbol{x}_p^s$ from the approximate distribution. When Reiter's method is used, there is no one-to-one correspondence between real observations and synthetic observations, so there need not be any particular relationship between the size of the actual dataset and the size of the synthetic sample. This feature helps to protect outliers, as an outlier in the original data may not appear in the synthetic plot or may appear more than once. In the case of categorical predictor variables, we let the synthetic sample size equal the actual sample size, while in the case of numerical predictor variables, we let the synthetic sample size be the minimum of 5,000 and the actual sample size. This is because when making the synthetic and actual sample sizes equal in the numerical case, we found that the system was slow when dealing with large datasets, and that the vast majority of the time that the analysis took was spent on creating the synthetic residual plots for numerical variables.

A shortcoming of the method for creating synthetic continuous predictors is that the kernel density estimator is not able to identify a probability mass at a single point, but rather will assume that the probability density function should be high in the neighborhood of that point. This should not invalidate the method, but it will affect the distribution along the x-axis for a predictor variable such as income, for whom many people have a true value of 0.

It should be noted that both of these methods for creating the synthetic data work with one variable at a time, i.e., $\boldsymbol{x}_p^s$ are drawn marginally, not jointly, and thus no valid analysis can be performed based on the joint distribution of the synthetic variables. This is not currently a major concern, as it is not our intention to release synthetic data through the MAS. However, this does impose a limitation on the range of diagnostics that we can make available in the future based on synthetic variables generated using this method.

The next step is to generate the standardized synthetic residuals $\boldsymbol{t}_p^s$ so that the relationship between $\boldsymbol{t}_p^s$ and $\boldsymbol{x}_p^s$ at any point x_{kp}^s in $\boldsymbol{x}_p^s$ is consistent with the relationship between $\boldsymbol{t}$ and $\boldsymbol{x}_p$ around point x_{kp}^s. To accomplish this, we must make a different set of synthetic residuals for each predictor variable. Note that x_{kp}^s, if numerical, will not necessarily be a value observed in continuous real data, but may be drawn with the inverse-cdf method.

For each variable, the goal is to give the user something akin to a plot of the standardized residuals of the full (possibly multiple) regression model versus the value of $\boldsymbol{x}_p$. For a variable p and an index k, define

$$t_{kp}^s = b_{kp} + v_{kp} + n_{kp}$$

The first term gives the expected value of the standardized residual for any given value of p; the second accounts for the variation of the actual standardized residuals around their expected values (which may change depending on the value of x_{kp} if heteroscedasticity is present); and the third adds noise to further prevent disclosure.

To calculate the first term b_{kp}, a generalized additive model (GAM) is built for $\boldsymbol{t}$ and $\boldsymbol{x}_p$. The value b_{kp} equals the value of the GAM curve at the point x_{kp}^s and is used to fit the values t_{kp}^s to the general relationship of $\boldsymbol{t}$ and $\boldsymbol{x}_p$, ignoring for the moment the variation of $\boldsymbol{t}$ around its local mean. Note that $\boldsymbol{t}_p^s$ will differ for every regression a user requests, and that it is important that the GAM not be overfit. In extreme cases, an overfit GAM can create some of the same disclosure risks as releasing a regression with a high R^2. There may be some difficulty in avoiding such an overfit in an automated setting. For categorical variables, a GAM cannot be fit, and we set $b_{kp} = 0$ because whenever a regression including a categorical variable is performed, the mean residual among observations with any particular level of that categorical variable is 0.

Next, t_{kp}^s is shifted off the curve b_{kp} by v_{kp}, which represents the amount by which the points in the real data around x_{kp}^s deviate from the curve. For the case where $\boldsymbol{x}_p$ is numerical, we consider the real data standardized residual t_j, where

$$j = \arg\min_i |x_{kp}^s - x_{ip}|$$

is the index of the unit in $\boldsymbol{x}_p$ whose value is closest to x_{kp}^s. Ties can be broken by selecting randomly from all tied choices. Having found j, we compute $v_{kp} = t_j - b_{jp}$ where b_{jp} is the value obtained from the GAM at x_{jp}. If $\boldsymbol{x}_p$ is categorical, j is the index of a randomly selected observation in the real data such that $x_{jp} = x_{kp}^s$, so we set $v_{kp} = t_j$, since $b_{jp} = 0$.

Finally, a noise term $n_{kp} \sim N(0,\sigma)$ is added to t^s_{kp} where, for each regression, σ should remain constant so that there is not artificial heteroscedasticity in the synthetic residuals. The same random seed should be used for all regressions using the same dependent variable; if this were not done, there would be the possibility of running the same or similar models a number of times and averaging the different results, creating a disclosure risk. Careful selection of σ is important, as a value that is too small may not provide enough protection against disclosure, while a value that is too large may cause patterns that are of interest to a legitimate user to be dwarfed by random variation.

When all steps are complete, the system creates a scatterplot of the synthetic residuals versus each numerical synthetic predictor variable, as well as a scatterplot of the synthetic residuals against the fitted value, with a kernel smoother used to show the general shape of the latter curve. To protect outliers, the scatterplot requires all synthetic standardized residuals to be in the interval [-4,4], with values that would otherwise be outside this range truncated appropriately.

Since categorical predictors do not lend themselves to scatterplots, the residual plots for categorical variables are replaced by side-by-side boxplots. Sparks et al. (2008) propose that numerical predictor variables be binned in a cutpoint-like fashion, and that the bins be used to create categories for side-by-side boxplots, which can be returned to the user instead of scatterplots, with Winsorization being performed to protect outliers. Since this binning lowers the resolution with which we can see the variable along the x-axis, Sparks et al. (2008) use it as a substitute for synthetic data.

We are beginning to implement regression diagnostics for logistic regressions in the manner described in Reiter and Kohnen (2005).

4. Evaluation: Effectiveness of the *Drop q Rule*

What follows is a generalization of some results in Lucero et al. (2009a), although that paper considered an earlier, less secure version of the *Drop q Rule* in which *q* was a fixed value chosen in advance. We present only a brief overview of this evaluation here; full details are in Lucero (2010b). Given a pair of similar universes, U(n) and U($n-1$), differing by only one unique observation, with n large, we consider the effectiveness of the *Drop q Rule* in preventing contingency table differencing attack disclosures of the form $\mathrm{T}_1 = \mathrm{T}_{n-q_1} - \mathrm{T}_{n-1-q_2}$, as was shown in *Figure 2*.

For this section, we will consider a contingency table giving the values of two categorical variables, with the same setup as described in Section 3.1. To make the notation somewhat less unwieldy, we denote the size of each cell in the contingency table using a single subscript, as shown in *Figure 5*, instead of the double subscript used previously. In the simplest case, the contingency table is 2 × 2 (two categories for each of two variables), but it could conceivably be larger—including either more categories for a particular variable or more variables, which would lead to more dimensions and would require a more elaborate graphical representation.

T_n	ES_1	ES_2
G_1	n_1	n_2
G_2	n_3	n_4

Figure 5: *Illustration of notation used in Section 4.*

We also let $\mathbf{\Pi} = (\Pi_1, \ldots, \Pi_4)$ denote the proportions of observations within each of the cells of T_n and let $\mathbf{\Psi} = (\Psi_1, \ldots, \Psi_4)$ denote the proportions within T_{n-1}. If n is large, then $\mathbf{\Pi} \approx \mathbf{\Psi}$. Furthermore, let $\boldsymbol{X}$ denote the vector giving the number of observations removed from each of the four cells when q_1 observations are dropped from T_n to produce T_{n-q_1}, and let $\boldsymbol{Y}$ denote the vector giving the number of observations removed from each of the four cells when q_2 observations are dropped from T_{n-1} to produce T_{n-1-q_2}. A correct disclosure will occur if and only if $\boldsymbol{X} = \boldsymbol{Y}$, and this may occur only when $q_1 = q_2$.

Since sampling with replacement is very similar to sampling without replacement when n is large, we can say that for a given q_1 and q_2, $\boldsymbol{X}$ is approximately a multinomial random variable with size q_1 and probabilities given by $\mathbf{\Pi}$, and $\boldsymbol{Y}$ is approximately a multinomial random variable with size q_2 and probabilities given by $\mathbf{\Psi}$. Substituting $\mathbf{\Pi}$ for $\mathbf{\Psi}$ and performing some other manipulations gives a formula for the approximate probability of disclosure for a given number of cells J, maximum number of cells dropped k and vector $\mathbf{\Pi} = (\Pi_1, \ldots, \Pi_J)$:

$$\xi_{J,k}(\Pi_1,\ldots,\Pi_J) = \sum_{q_1=2}^{k} \sum_{x_1,\ldots,x_J \geq 0}^{x_1+\ldots+x_J=q_1} \left(\frac{1}{k-1}\right)^2 \left(\frac{q_1!}{x_1!\cdot\ldots\cdot x_J!}\right)^2 \Pi_1^{2x_1}\cdot\ldots\cdot\Pi_J^{2x_J} \qquad (5)$$

This formula has a total of $\binom{J+k}{J} - (J+1)$ summands within a rather involved summation, which makes it cumbersome, but it may be useful in assessing the risk involved with releasing a given table with a given value of k. Further research may focus on finding simpler approximations for the value in this sum.

A large number of differencing attacks were simulated, as described in Lucero (2010b), for a pair of tables, differing by one observation, with $n = 978$ and $k \in \{3,4,5,6,7\}$. The data were from the Current Population Survey March 2000 Demographic Supplement. The simulation led to the conclusion that the summation in (5) generally agrees with the empirical probability of a disclosure to two decimal places for this sample size.

It may also be desirable to find bounds on the summation in (5) in the case in which $\mathbf{\Pi}$ is not known. This would be useful, for example, if we were looking at the same table, but for a number of different universes. The derivation of bounds makes use of the fact that the function in (5) is a Schur-convex function of $\mathbf{\Pi}$; for more on Schur-convex functions, see Marshall and Olkin (1979) or Lucero (2010b). The Schur-convexity allows us to identify the most extreme cases, and leads to the following bounds:

$$\left(\frac{1}{k-1}\right)^2 \sum_{q_1=2}^{k} \sum_{x_1,\ldots,x_J \geq 0}^{x_1+\ldots+x_J=q_1} \left(\frac{q_1!}{x_1! \cdot \ldots \cdot x_J!}\right)^2 \left(\frac{1}{J}\right)^{2q_1} \leq \xi_{J,k}(\Pi_1,\ldots,\Pi_J) \leq \frac{1}{k-1} \quad (6)$$

The righthand portion of inequality (6) says that the probability of an accurate disclosure is at most the probability that the same value of q will be chosen for each of the two tables. The lefthand portion gives a best case for the probability of disclosure, upon which we cannot improve without modifying which cells are in the table or changing k (with the proviso that all probabilities are approximate). In particular, the best case is that all cells of the table include exactly the same proportion of the population, i.e. that $\Pi = \left(\frac{1}{J},\ldots,\frac{1}{J}\right)$.

5. Other Approaches

Since the idea of a remote access system has been in existence for several years, a number of approaches have been proposed that differ from ours to varying degrees, and we survey some of them here.

Schouten and Cigrang (2003) present a variant of the idea of a remote access system, which allows outstanding versatility, but is also difficult to create and expensive and laborious to maintain. Their proposed system allows users to submit queries by email, written in any of several statistical programming languages. If a query is approved, the user receives the results by email. Before the analysis is performed, an automated system determines the legitimacy of the request, with particularly difficult cases handled manually. As with the MAS, certain types of output are allowed and certain types are not, but since the code is user-generated, rather than generated by the system behind the scenes, it is challenging to identify all unallowable queries. This is especially true because, as the authors emphasize, the validity of a query may depend on information already released as a result of previous successful queries. The authors write, "Computers are simply not fast enough and the construction of a system that fully evaluates the risk of disclosure may be too costly and complex and therefore not feasible." Thus, in a system like this, it may be necessary to perform some disclosure avoidance analysis on a query after the result of the query has already been returned. This is not ideal, as a query that is a disclosure threat might not be identified until its output has already been provided. However, such a method could be effective if the users are from large institutions and have signed a contract describing their research and pledging to uphold confidentiality. In this case, the fear of a user or institution's jeopardizing its future access to the data may serve as a sufficient deterrent to its deliberately submitting an invalid query. In this type of system, a username and password would be necessary so that individual users' actions could be properly tracked.

Sparks et al. (2008) propose a system—Privacy-Preserving Analytics®—that performs a number of methods for disclosure avoidance, including keeping track of the regression models a user requests and ensuring that only a limited (although large) number are run for each possible response variable. They also ensure that a user does not make too many closely related requests.

Gomatam et al. (2005) make a distinction between *static servers* and *dynamic servers*. A static server has a pre-determined set of queries to which it will provide an answer. A dynamic server receives a query and makes a decision on whether to provide an answer. A dynamic server—such as the one described in Schouten and Cigrang (2003)—would keep a running record of all previously answered queries, and whenever a new query was submitted, it would be compared against the list to determine whether providing an answer would lead to a disclosure risk when the new answer was combined with previously provided answers. A dynamic server has the highly undesirable property that the order in which queries are submitted by the collective group of users plays a large role in determining which queries are answered, and that eventually the server reaches a point where no new queries can be answered. Since queries are answered or rejected as they are received, the set of queries that are ultimately answered is not the result of a careful assessment of which analyses would provide the most utility to legitimate researchers while keeping disclosure risk at an acceptable level. Gomatam et al. (2005) write that "[w]hether dynamic servers are possible remains an open question." The MAS is at its heart a static server, since it operates under a set of rules that do not depend on previous queries. However, it operates in a dynamic fashion, since the rules are checked for each new query that is submitted, rather than comparing it to a pre-computed list, as creating such a list would be prohibitive. In a way, the MAS does not fit into the framework of Gomatam et al. (2005), as it sometimes will provide regression output that is less detailed than the user might have liked instead of refusing output altogether.

Another approach to protecting privacy from a query-accepting statistical database is to suppress from any tables any cells that are deemed a disclosure risk, either directly or indirectly. Adam and Worthmann (1989) discuss this possibility and note that in certain systems, cell suppression is not a feasible solution to the disclosure problem.

6. Future Work

The MAS will continue to be developed within DataFERRETT. We will soon be testing the software itself and the confidentiality rules within the MAS beta prototype to ensure that they properly uphold disclosure avoidance standards. We will draft a set of confidentiality rules for cross-tabulations, and add different types of statistical analyses within the system. We will explore other types of differencing attack disclosures, and investigate ways to prevent such differencing attacks. Also of potential interest is doing more theoretical explorations to evaluate disclosure risk. For example, it would be of interest to determine the probability of a correct disclosure given that there is an apparent disclosure resulting from a differencing attack. If this number were small enough, it could lead to a higher level of protection for the system, as an intruder would not be able to be highly confident of the correctness of an apparent disclosure.

References

N. Adam and J. Worthmann. Security-control methods for statistical databases: a comparative study. *ACM Computing Surveys (CSUR)*, 21(4):515–556, 1989. ISSN 0360-0300.

Z. Ben-Haim and T. Dvorkind. Majorization and applications to optimization. Technion Institute, 2004.

M. Chaudhry. Overview of the Microdata Analysis System. Statistical Research Division internal report, U.S. Census Bureau, 2007.

G. Duncan, S. Keller-McNulty, and S. Stokes. Disclosure risk vs. data utility: The R-U confidentiality map. In *Chance*. Citeseer, 2001.

I. Fellegi. On the question of statistical confidentiality. *Journal of the American Statistical Association*, 67(337):7–18, 1972. ISSN 0162-1459.

I. Fellegi, S. Goldberg, and S. Abraham. *Some Aspects of the Impact of the Computer on Official Statistics*. Dominion Bureau of Statistics, 1969.

S. Gomatam, A. Karr, J. Reiter, and A. Sanil. Data dissemination and disclosure limitation in a world without microdata: A risk-utility framework for remote access analysis servers. *Statistical Science*, 20(2):163–177, 2005. ISSN 0883-4237.

S. Kaufman, M. Seastrom, and S. Roey. Do disclosure controls to protect confidentiality degrade the quality of the data? In *American Statistical Association, Proceedings of the Section on Survey Research Methods*, pages 1218–1225, 2005.

S. Keller-McNulty and E. Unger. A database system prototype for remote access to information based on confidential data. *Journal of Official Statistics*, 14:347–360, 1998. ISSN 0282-423X.

J. Lucero. Confidentiality rules for universe formation and geographies for the Microdata Analysis System. Statistical Research Division Confidential Research Report CCRR–2009/01, U.S. Census Bureau, 2009.

J. Lucero. Confidentiality rule specifications for performing regression analysis on the Microdata Analysis System. Statistical Research Division Confidential Research Report, U.S. Census Bureau, 2010a.

J. Lucero. Evaluation of the effectiveness of the Drop q Rule against differencing attack disclosures. Statistical Research Division Confidential Research Report CCRR-2010/03, U.S. Census Bureau, 2010b.

J. Lucero, L. Singh, and L. Zayatz. Recent work on the Microdata Analysis System at the Census Bureau. Statistical Research Division Confidential Research Report CCRR–2009/09, U.S. Census Bureau, 2009a.

J. Lucero, L. Zayatz, and L. Singh. The current state of the Microdata Analysis System at the Census Bureau. In *Proceedings of the American Statistical Association, Government Statistics Section*, 2009b.

A. Marshall and I. Olkin. *Inequalities: Theory of Majorization and Its Applications*, volume 143. Academic Press New York, 1979.

C. O'Keefe and N. Good. Regression output from a remote analysis server. *Data & Knowledge Engineering*, 68(11):1175–1186, 2009. ISSN 0169-023X.

J. Reiter. Model diagnostics for remote access regression servers. *Statistics and Computing*, 13(4):371–380, 2003. ISSN 0960-3174.

J. Reiter and C. Kohnen. Categorical data regression diagnostics for remote access servers. *Journal of Statistical Computation and Simulation*, 75(11):889–903, 2005. ISSN 0094-9655.

A. Reznek. Disclosure risks in cross-section regression models. In *Proceedings of the Section on Government Statistics, JSM*, 2003.

A. Reznek and T. Riggs. Disclosure risks in regression models: Some further results. In *Proceedings of the Section on Government Statistics, JSM*, 2004.

S. Roehrig, S. Bayyana, S. Ganapatiraju, and S. Santhanam. Final report to the Bureau of the Census for the project "Auditing the Census Bureau's confidentiality preserving model server". Contracted report for the U.S. Census Bureau, 2008.

S. Rowland and L. Zayatz. Automating access with confidentiality protection: The American FactFinder. In *Proceedings of the Section on Government Statistics*, 2001.

B. Schouten and M. Cigrang. Remote access systems for statistical analysis of microdata. *Statistics and Computing*, 13(4):381–389, 2003. ISSN 0960-3174.

R. Sparks, C. Carter, J. Donnelly, C. O'Keefe, J. Duncan, T. Keighley, and D. McAullay. Remote access methods for exploratory data analysis and statistical modelling: Privacy-Preserving Analytics®. *Computer Methods and Programs in Biomedicine*, 91(3):208–222, 2008. ISSN 0169-2607.

P. Steel. Design and development of the Census Bureau's Microdata Analysis System: Work in progress on a constrained regression server. Presentation at Federal Committee on Statistical Methodology Statistical Policy Seminar, November 2006.

P. Steel and A. Reznek. Issues in designing a confidentiality preserving model server. *Monographs of Official Statistics*, 9:29, 2005.

D. Weinberg, J. Abowd, P. Steel, L. Zayatz, and S. Rowland. Access methods for United States microdata. U.S. Census Bureau Center for Economic Studies paper CES-WP-07-25, August 2007. Paper for Institute for Employment Research Workshop on Data Access to Micro-Data, Nuremberg, Germany.

X. Zhang. Schur-convex functions and isoperimetric inequalities. *Proceedings of the American Mathematical Society*, 126(2):461–470, 1998. ISSN 0002-9939.