Privacy: Theory meets Practice on the Map

Ashwin Machanavajjhala [†1], Daniel Kifer [†2], John Abowd [#3], Johannes Gehrke [†4], Lars Vilhuber [#5]

[†]*Department of Computer Science, Cornell University, U.S.A.*
[#]*Department of Labor Economics, Cornell University, U.S.A.*

[1]mvnak@cs.cornell.edu [2]dkifer@cs.cornell.edu [3]john.abowd@cornell.edu
[4]johannes@cs.cornell.edu [5]lars.vilhuber@cornell.edu

***Abstract*— In this paper, we propose the first formal privacy analysis of a data anonymization process known as the synthetic data generation, a technique becoming popular in the statistics community. The target application for this work is a mapping program that shows the commuting patterns of the population of the United States. The source data for this application were collected by the U.S. Census Bureau, but due to privacy constraints, they cannot be used directly by the mapping program. Instead, we generate synthetic data that statistically mimic the original data while providing privacy guarantees. We use these synthetic data as a surrogate for the original data.**

We find that while some existing definitions of privacy are inapplicable to our target application, others are too conservative and render the synthetic data useless since they guard against privacy breaches that are very unlikely. Moreover, the data in our target application is sparse, and none of the existing solutions are tailored to anonymize sparse data. In this paper, we propose solutions to address the above issues.

I. Introduction

In this paper, we study a real-world application of a privacy preserving technology known as synthetic data generation. We present the first formal privacy guarantees (to the best of our knowledge) for this application. This paper chronicles the challenges we faced in this endeavour. The target application is based on data developed by the U.S. Census Bureau's Longitudinal Employer-Household Dynamics Program (LEHD). By combining various Census datasets it is possible to construct a table Commute_Patterns with schema *(id,origin_block, destination_block)* where each row represents a worker. The attribute *id* is a random number serving as a key for the table, *origin_block* is the census block in which the worker lives, and *destination_block* is where the worker works. An origin block o corresponds to a destination block d if there is a tuple with *origin_block* o and *destination_block* d. The goal is to plot points on a map that represent commuting patterns for the U.S. population. For each destination block, we plot points on the map representing the corresponding origin blocks. There are roughly 8 million Census blocks so that the domain is very large and the data are very sparse.

Information about destination blocks has already been publicly released; i.e., the query `SELECT COUNT(*), destination_block FROM Commute_Patterns GROUP BY destination_block` is available to any adversary, thus *origin_block* is treated as the sensitive attribute. Due to privacy constraints and legal issues, unanonymized origin block data cannot be used as an input to such a mapping application. An anonymized version must be used instead.

The algorithm used to anonymize the data for the above mapping application is known as the synthetic data generation [1], which is becoming popular in the statistical disclosure limitationg community. The main idea behind synthetic data generation is to build a statistical model from the data and then to sample points from the model. These sampled points form the synthetic data, which is then released instead of the original data. While much research has focused on deriving the variance and confidence intervals for various estimators from synthetic data [2], [3], there has been little research on deriving formal guarantees of privacy for such an approach (an exception is [4]).

Much recent research has focused on deriving formal criteria for privacy. These include general notions of statistical closeness [5], variants of the notions of k-*anonymity* [6] and ℓ-*diversity* [7], [8], [9], [10], [11], (ρ_1, ρ_2)-*privacy* [12], and variants of *differential privacy* [13], [14]. However, we found that apart from the differential privacy criterion [13], none of the other privacy conditions applied to our scenario.

Picking an off-the-shelf synthetic data generation algorithm and tuning it to satisfy the differential privacy criterion was unsatisfactory for the following reasons. First, in order to satisfy the differential privacy criterion, the generated synthetic data contained little or no information about the original data. We show that this is because differential privacy guards against breaches of privacy that are very unlikely.

Next, no deterministic algorithm can satisfy differential privacy. Randomized algorithms can (albeit with a very small probability) return anonymized datasets that are totally unrepresentative of the input. This is a problem, especially, when we want to publish a single or only a few, anonymous versions of the whole data. We remedy these two issues by showing that a revised probabilistic version of differential privacy yields a practical privacy guarantee for synthetic data.

Finally, the data in our application are very sparse – there are roughly 8 million blocks on the U.S. map, and only about a few tens or hundreds of workers commuting to each destination. Most previous work deals with data where the number of individuals is typically larger than the size of the sensitive attribute domain. We identify this important open research problem and propose our first solutions for solving the sparsity issue by modifying the synthetic data generation

algorithm.

We present the derivation of our techniques as a case study of anonymizing data for the novel mapping application. In this spirit, we discuss the initial selection of an off-the-shelf privacy definition in Section II and an off-the-shelf anonymization algorithm in Section III based on the requirements of the mapping application. Since these initial choices turn out to be unsatisfactory, we iteratively refine them to preserve their strengths while removing their weaknesses: in Section IV we show that some of our problems are the results of extremely unlikely events, which leads us to a probabilistic version of differential privacy; using the new privacy definition, in Section V we revise our algorithm to improve data quality and minimize the negative effects of the sparse data and large domain in the mapping application. In Section VI we present experiments and discuss the analytic validity of the resulting synthetic data. We overview related work in Section VII and present conclusions in Section VIII.

To summarize, our contributions are as follows: we provide the first formal privacy analysis for a synthetic data generation method (to the best of our knowledge); we present a case-study of applying state-of-the-art research in privacy to real applications; we identify additional challenges for the privacy research community (such as handling large domains), and we propose initial solutions for these challenges.

II. Starting Point: Privacy Definition

To help select an initial privacy definition, we use the following guideline: the privacy definition should give us theoretical guarantees about privacy. There are three privacy definitions that have theoretical guarantees and which we felt are potentially applicable: ℓ-*diversity* [7], (d, γ)-privacy [14], and *differential privacy* [13].

In its most basic form, ℓ-diversity requires that for each destination block, the ℓ origin blocks with the most number of workers with jobs in this destination block, have roughly equal number of workers residing in them (see [7] for technical details and variations). Although ℓ-diversity can protect against adversaries with background knowledge, it does not always guarantee privacy when there is a semantic relationship between distinct sensitive values (here the semantic relationship is physical proximity). That is, ℓ-diversity does not offer theoretical guarantees against an adversary who has information such as "Bob probably lives near Newark and works near New York City."

(d, γ)-Privacy is a probabilistic privacy definition in which an adversary believes in some prior probability $P(t)$ of a tuple t appearing in the data. After seeing the anonymized data D, the adversary forms a posterior belief $P(t|D)$. (d, γ)-Privacy is only designed to protect against adversaries that are d-*independent*: an adversary is d-independent if for all tuples t are considered a priori independent and, the prior belief $P(t)$ satisfies the conditions $P(t) = 1$ or $P(t) \leq d$. For all such adversaries, the privacy definition requires that $P(t|D) \leq \gamma$ and $P(t|D)/P(t) \geq d/\gamma$. This privacy definition does not apply to our scenario for a couple of reasons. First, this privacy definition does not apply for adversaries that believe that $P(t) = d + \epsilon$ (no matter how small $\epsilon > 0$ is) for some t even though in those cases we would also like to have some guarantee about privacy. Second, tuple-independence is a very strong assumption that is not compatible with our application. In order to be d-independent, an adversary has to consider the facts "Worker #1234 commutes to Block 12 from Block 34" and "Worker #1234 commutes to Block 12 from Block 35" independent. This is not true in our application since, the two events described above are mutually exclusive.

Differential privacy is a privacy definition that can be motivated in several ways [13]. If an adversary knows complete information about all individuals in the data except one, the output of the anonymization algorithm should not give the adversary too much additional information about the remaining individual. Alternatively, if one individual is considering lying about their data to a data collector (such as the U.S. Census Bureau), the result of the anonymization algorithm will not be very different if the individual lied or not. Formally,

Definition 1 (ϵ-Differential Privacy): Let $\mathcal{A}$ be a randomized algorithm, let $\mathcal{S}$ be the set of possible outputs of the algorithm, and let $\epsilon > 1$. The algorithm $\mathcal{A}$ satisfies ϵ-*differential privacy* if for all pairs of data sets (D_1, D_2) that differ in exactly one row,

$$\forall S \in \mathcal{S}, \quad \left| \ln \frac{P(\mathcal{A}(D_1) = S)}{P(\mathcal{A}(D_2) = S)} \right| \leq \epsilon$$

Differential privacy has the benefits that we do not have to assume that tuples are independent or that an adversary has a prior belief encapsulated by a probability distribution. However, if the adversary is Bayesian, differential privacy guarantees that if the adversary has complete information about all individuals but one and believes in a prior probability P for the attributes of the remaining individual, then after seeing the anonymized data D, $|\ln P(t|D)/P(t)| < \epsilon$ for all tuples t [15].

III. Starting Point: Anonymization Algorithm

The original data can be viewed as a histogram where each combination of *origin_block* and *destination_block* is a histogram bucket. Histograms can be anonymized by modifying the counts in each bucket (for example, by adding random noise). Both [15] and [14] provide randomized anonymization algorithms for histograms that satisfy ϵ-differential privacy. One method is to add an independent Laplace random variable to each bucket of the histogram [15]. Another is to extract a Bernoulli subsample from the data and then to add independent Binomial random variables to each histogram bucket [14]. Intuitively, both methods mix the original data with a dataset that is generated from independently and identically distributed noise.

Instead of using one of these approaches, we use synthetic data generation [1] for protecting privacy. Prior to this work, synthetic data methods did not have formal privacy guarantees, despite there being significant work on performing statistical analyses of and drawing inferences from synthetic data [2], [3] based on Rubin's multiple imputation framework [16]. Thus,

our goal is to complement the research on utility for synthetic data by providing results about its privacy.

The main idea behind synthetic data generation is to build a statistical model from the data and then to sample points from the model. These sampled points form the synthetic data, which is then released instead of the original data. The motivation behind such statistical modeling is that inferences made on the synthetic data should be similar to inferences that would have been made on the real data.

Privacy comes from the fact that noise is added to the data from two sources: the bias that comes from the creation of the model, and the noise due to random sampling from the model. Note that the process of learning in the context of a model for synthetic data differs significantly from the normal application of machine learning. In machine learning, it is imperative not to overfit the data. For synthetic data, we want to overfit as much as possible (subject to privacy constraints), so that the synthetic data contain many of the characteristics of the original data.

For this application, we will use the multinomial model with dirichlet prior [17] as the initial mechanism for generating synthetic data. We describe the model below, starting with some necessary definitions:

Definition 2 (Multinomial distribution): Let $\overrightarrow{p} = (p_1, \ldots, p_k)$ be a vector of non-negative values such that $p_1 + \cdots + p_k = 1$. A multinomial distribution of size m with parameters $(p_1, \ldots, p_k)$, $\mathcal{M}(\overrightarrow{p}, m)$, is a probability distribution over k-dimensional vectors with non-negative integer coordinates that sum up to m, with

$$P(m_1, \ldots, m_k) = \frac{m!}{\prod_{i=1}^{k} m_i!} \prod_{i=1}^{k} p_i^{m_i}$$

Definition 3 (Dirichlet distribution): Let $\overrightarrow{\alpha} = (\alpha_1, \ldots, \alpha_k)$ be a vector of positive values and let $|\alpha| = \alpha_1 + \cdots + \alpha_k$. The *dirichlet distribution* with parameters $(\alpha_1, \ldots, \alpha_k)$, $\mathcal{D}(\overrightarrow{\alpha})$, is a probability distribution over all k-dimensional vectors with non-negative coordinates that sum up to 1, with density[1]

$$f(p_1, \ldots, p_k | \overrightarrow{\alpha}) = \frac{\Gamma(|\alpha|)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} p_i^{\alpha_i - 1}$$

The vector $\overrightarrow{\alpha} = (\alpha_1, \ldots, \alpha_k)$ is known as the *prior sample*, $|\alpha|$ is the *prior sample size*, and the vector $(\alpha_1/|\alpha|, \ldots, \alpha_k/|\alpha|)$ is the *shape* of the prior.

Multinomial sampling with a dirichlet prior $\mathcal{D}(\overrightarrow{\alpha})$ proceeds as follows:

1) Draw a vector $\overrightarrow{p} = (p_1, \ldots, p_k)$ from the $\mathcal{D}(\overrightarrow{\alpha})$ distribution.
2) Interpret $\overrightarrow{p}$ as a vector of multinomial probabilities and draw a sample of size n from the multinomial distribution $\mathcal{M}(\overrightarrow{p}, n,)$.

[1]where $\Gamma(t)$ is the gamma function defined as $\int_0^\infty x^{t-1} e^{-x} dx$

Algorithm 1 Synthesize

for all destination blocks d **do**
  Let $\overrightarrow{n(d)}$ be the histogram of origin blocks
  Choose prior sample $\overrightarrow{\alpha(d)}$ with $|\alpha(d)| = O(n(d))$
  Choose output sample size $m = O(n(d))$
  Sample m points from a multinomial distribution with prior $\mathcal{D}((n(d)_1 + \alpha_1, \ldots, n(d)_k + \alpha_k))$
end for

It is well known that if $(n_1, \ldots, n_k)$ was drawn using multinomial sampling with a dirichlet prior, then the posterior distribution $P(\overrightarrow{p}|(n_1, \ldots, n_k))$ is the dirichlet distribution $\mathcal{D}((\alpha_1 + n_1, \alpha_2 + n_2, \ldots, \alpha_k + n_k))$. This can be interpreted informally as first having no information, observing the sample data $(\alpha_1, \ldots, \alpha_k)$ (note the α_i do not have to be integers), and updating the prior to $\mathcal{D}((\alpha_1, \ldots, \alpha_k))$, then observing the new data $(n_1, \ldots, n_k)$, and updating the prior once again to $\mathcal{D}((\alpha_1 + n_1, \ldots, \alpha_k + n_k))$. Thus if we have a $\mathcal{D}((\alpha_1, \ldots, \alpha_k))$ prior, we are acting as if $(\alpha_1, \ldots, \alpha_k)$ was a piece of data we had already seen. For this reason $(\alpha_1, \ldots, \alpha_k)$ is called the prior sample.

Now we can describe the initial algorithm for generating synthetic data. Let k be the number of Census blocks. We number the blocks from 1 to k and for each destination block d we form a k-dimensional vector $\overrightarrow{n(d)} = (n(d)_1, \ldots, n(d)_k)$ where $n(d)_i$ is the number of people whose origin block is the Census block i and who commute to destination block d. For each destination block d we choose a prior sample $\overrightarrow{\alpha(d)}$ with $|\alpha(d)| = O(n(d))$ (the choice of $\overrightarrow{\alpha(d)}$ is discussed in Sections IV and V), condition on the $\overrightarrow{n(d)}$ to get a dirichlet $\mathcal{D}((n(d)_1 + \alpha(d)_1, \ldots, n(d)_k + \alpha(d)_k))$ prior and then use multinomial sampling with this prior to create a vector $\overrightarrow{m} = (m_1, \ldots, m_k)$ of $|m| = O(n(d))$ synthetic people such that m_i synthetic people commute to destination block d from origin block i. This procedure is described in Algorithm 1.

Note that when the destination block d is clear from context, we will drop the notational dependency on d and abbreviate $\overrightarrow{n(d)}$ and $\overrightarrow{\alpha(d)}$ as $\overrightarrow{n} = (n_1, \ldots, n_k)$ and $\overrightarrow{\alpha} = (\alpha_1, \ldots, \alpha_k)$, respectively.

IV. Revising the privacy definition

In this Section we evaluate our initial choice of privacy definition on the initial choice of anonymization algorithm. Contrary to intuition, it will turn out that these initial choices do not give good results for privacy. By analyzing the problem, we will show where intuition fails, and then we will revise the privacy definition to account for the discrepancy.

A. Analysis of Privacy

First, we start with an explanation behind the intuition that synthetic data should preserve privacy. Since the anonymized data is synthesized, it consists of synthetic people. Thus, linking real-world individuals to synthetic individuals (as in [6]) does not make sense. The prior sample $\overrightarrow{\alpha(d)}$ controls the

amount of noise we add to the data generator. The larger the components of $\overrightarrow{\alpha(d)}$, the more noise is added. Now, we could have achieved ϵ-differential privacy by adding i.i.d. Laplace random variables with density $\frac{\epsilon}{4}\exp(-\epsilon x/2)$ and variance $8/\epsilon^2$ to the counts in each origin block [15]. For common values of ϵ (i.e., $\epsilon > 1$) this is a relatively small amount of noise per origin block. So intuitively, we also shouldn't need to add too much noise (in terms of a large prior sample) using the synthetic data generator.

This turns out not to be true. Let $\overrightarrow{\alpha(d)}$ be the prior sample for a destination block d (note that the prior sample does not depend on the data). Differential privacy requires us to consider adversaries who have complete information about all but one of the individuals in the data. Now, a histogram of destination blocks has already been published, so our adversary can use this information to determine the destination block d of the remaining individual. Thus we need to determine how well the individual's origin block is hidden from the adversary, and only need to look at the synthetic data generated for destination block d. There are n individuals with that destination block, and we will generate m synthetic individuals for that destination block. To determine if the synthetic data generation satisfies ϵ-differential privacy (Definition 1) we have to find the maximum of

$$\frac{P((m_1,\ldots,m_k) \mid (n_1,\ldots,n_k), \overrightarrow{\alpha})}{P((m_1,\ldots,m_k) \mid (n'_1,\ldots,n'_k), \overrightarrow{\alpha})} \tag{1}$$

over all non-negative integer vectors $(m_1,\ldots,m_k)$, $(n_1,\ldots,n_k)$, $(n'_1,\ldots,n'_k)$ with $\sum m_i = m$, $\sum n_i = \sum n'_i = n$, $n_i = n'_i + 1$ for some i, $n_j + 1 = n'_j$ for some $j \neq i$, and $n_h = n'_h$ for all $h \notin \{i,j\}$ (thus the difference between $\overrightarrow{n}$ and $\overrightarrow{n'}$ is the origin block of the unknown individual). If this maximum is less than $\exp(\epsilon)$, the the synthetic data generator satisfies ϵ-differential privacy.

Theorem 4.1: The maximum of Equation 1 is $\frac{m+\min_i(\alpha_i)}{\min_i(\alpha_i)}$ and this is at most $\exp(\epsilon)$ if and only if each $\alpha_i \geq \frac{m}{\exp(\epsilon)-1}$.

The effect of Theorem 4.1 can be illustrated with the following example. Suppose there are one million people in destination block d and we choose to synthesize one million data points (i.e. we set $m = 1,000,000$). By setting $\epsilon = 7$ we are effectively requiring that the ratio in Equation 1 is at most $\exp(\epsilon) \approx 1096$. To achieve this privacy definition, we need to set $\alpha_i \geq 914$ for each i. In other words, for destination d, our prior sample $\overrightarrow{\alpha}$ has to have at least 914 people in each origin block, which in many cases will be more than n_i (the actual number of people in that origin block that commute to destination block d).

We can analyze where the intuition went wrong by examining the worst case in Theorem 4.1. The adversary has complete information about $n-1$ individuals and is trying to determine the origin block of the remaining individual. Suppose that α_i achieves its minimum value in origin block i, that the adversary does not know of any individual in origin block i, and that all m synthetic data points occur in origin block i. In this case, Equation 1 is maximized and can be interpreted as the likelihood that the remaining individual came from origin block i divided by the likelihood that the remaining individual did not come from origin block i. Note that this scenario leaks the most amount of information about the remaining individual and it is also a scenario where the synthetic data is completely unrepresentative of the original data – in the original data at most one person could have come from origin block i, but in the synthetic data all of the synthetic people came from origin block i. The probability of such an unrepresentative sample is at most $\frac{\Gamma(n+|\alpha|)\Gamma(m+\alpha_i+1)}{\Gamma(1+\alpha_i)\Gamma(m+n+|\alpha|)}$. As an example, let $m = n = 20$ and $\epsilon = 7$, and let all the α_j have their minimum values of $m/(\exp(7)-1) \approx 0.018$. If there are even just $k = 2$ origin blocks then the probability of that event is approximately $e^{-24.9}$. Thus the worst case is an extremely unlikely event.

B. Bounding the worst case

In Section IV-A we saw that the worst-case privacy breaches are in outputs of the synthetic data generator that are extremely unlikely to occur. We say that a function $f(x)$ is *negligible* if $f(x)/x^{-k} \to 0$ as $x \to \infty$ for all $k > 0$. An output $\overrightarrow{m(d)}$ of the synthetic data generator is negligible if $P(\overrightarrow{m(d)} \mid \overrightarrow{n(d)}, \overrightarrow{\alpha(d)})$ is negligible in $|n(d)|$ (i.e. $P(\overrightarrow{m(d)} \mid \overrightarrow{n(d)}, \overrightarrow{\alpha(d)})/|n(d)|^{-k} \to 0$ for all $k > 0$).

Because these outputs are so unlikely, we would like to exclude them from the analysis of privacy. This leads us to the following privacy definition [18] which is a relaxation of differential privacy.

Definition 4 (Indistinguishability): Let $\epsilon > 0$. Let $\mathcal{A}$ be a randomized algorithm and let $\mathcal{S}$ be the space of outputs of $\mathcal{A}$. Algorithm $\mathcal{A}$ satisfies (ϵ,δ)-indistinguishability if for all $T \subset \mathcal{S}$ and all tables X_1 and X_2 differing in one row,

$$P(\mathcal{A}(X_1) \in T) \leq e^{\epsilon} P(\mathcal{A}(X_2) \in T) + \delta(|X_2|)$$

where δ is a negligible function.

Now, for a given destination block d, $|m(d)| = O(|n(d)|)$ by definition. The number of synthetic data sets for destination block d is the number of ways to assign $|m(d)|$ people to k origin blocks and is equal to $\binom{|m(d)|+k-1}{k-1}$, which is a polynomial in $|n(d)|$, and thus not all outputs are negligible.

We can characterize outputs that have a negligible probability in terms of entropy by using Theorem 4.2.

Theorem 4.2: If for all destination blocks d we have $\alpha(d)_i \in O(n)$ and $\alpha(d)_i \in \omega \log n$ for all i then Algorithm 1 satisfies (ϵ,δ)-indistinguishability for $\epsilon \geq \ln(\Xi + 1/\min(\alpha(d)_i))$ (where $\Xi > 2$) and $n \geq m$.

Now, (ϵ,δ)-indistinguishability is an asymptotic privacy guarantee and so requires each destination block to have a large number of people commuting to it. Since our application contains many destination blocks with a small amount of commuters, this asymptotic privacy definition would not provide usable guarantees. For this reason we developed a different relaxation of differential privacy. First, we need to identify which outputs are "bad" in the sense that they leak too much information.

Definition 5 (Disclosure Set): Let D be a table and $\mathcal{D}$ be the set of tables that differ from D in at most one row. Let $\mathcal{A}$

be a randomized algorithm and $\mathcal{S}$ be the space of outputs of $\mathcal{A}$. The *disclosure set* of D, denoted by $\mathrm{Disc}(D,\epsilon)$ is $\{S \in \mathcal{S} \mid \exists X_1, X_2 \in \mathcal{D}, |X_1 \setminus X_2| = 1 \wedge |\ln \frac{P(\mathcal{A}(X_1)=S)}{P(\mathcal{A}(X_2)=S)}| > \epsilon\}$

Intuitively, the disclosure set for D is constructed as follows. For each tuple $x \in D$, let D^{-x} be the table D with tuple x removed. Treat D^{-x} as the adversary's background knowledge, so that the adversary is trying to guess the origin block for x. Now, if our data generator creates the synthetic data $\overrightarrow{m}$, then there are two likelihoods we are interested: the maximum likelihood of $\overrightarrow{m}$ over all possible origin blocks for x, the minimum likelihood of $\overrightarrow{m}$ over all possible origin blocks for x. If the ratio of the two likelihoods is greater than e^ϵ then $\overrightarrow{m}$ is an output that leaks too much information for some adversary. Consequently $\overrightarrow{m}$ is in the disclosure set for D, and all $\overrightarrow{m}$ in the disclosure set for D arise in this way.

Thus, to preserve privacy with high probability, we want the disclosure set to have a low probability, and so we arrive at the following privacy definition.

Definition 6 (Probabilistic Differential Privacy (pdp)): Let $\mathcal{A}$ be a randomized algorithm and let $\mathcal{S}$ be the set of all outputs of $\mathcal{A}$. Let $\epsilon > 0$ and $0 < \delta < 1$ be constants. We say that $\mathcal{A}$ satisfies (ϵ,δ)*-probabilistic differential privacy* (or, (ϵ,δ)-pdp) if for all tables D, $P(\mathcal{A}(D) \in \mathrm{Disc}(D,\epsilon)) \leq \delta$.

Note that in our application, a histogram of destination blocks has already been made publicly available. Thus the assumption that the adversary contains full information about all but one individuals in the data implies that the adversary knows the destination block of the remaining individual. Only the identity of the origin block is unknown to the adversary. For this reason we can partition the origin/destination dataset by destination block and treat each partition as a disjoint dataset. Hence, a dataset satisfies (ϵ,δ)-probabilistic differential privacy if each partition of the dataset satisfies (ϵ,δ)-probabilistic differential privacy. Moreover, in such datasets, for an adversary with complete information about all but one individuals in the dataset, the probability that the adversary gains significant information about the remaining individual is at most δ.

To state the privacy guarantees for Algorithm 1 in terms of probabilistic differential privacy, we will need the following definition:

Definition 7: Given constants $n, m, \alpha_1, \alpha_2, c$, define the function $f(x) = \min(m, c \cdot (\alpha_1 + [x-1]^+))$ (where, $[y]^+ = \max(y, 0)$). Then the *reference 0-sample*, denoted by $\rho(n, m, \alpha_1, \alpha_2, c)$ is the quantity:

$$\max_x \frac{\frac{\Gamma(m+1)}{\Gamma(f(x)+1)\Gamma(m-f(x)+1)} \frac{\Gamma(n+\alpha_1+\alpha_2)}{\Gamma(x+\alpha_1)\Gamma(n-x+\alpha_2)}}{\frac{\Gamma(m+n+\alpha_1+\alpha_2)}{\Gamma(x+f(x)+\alpha_1)\Gamma(n-x+m-f(x)+\alpha_2)}}$$

where the max is taken over all integers x in the range $[0, n]$.

First we need to identify the disclosure set:

Theorem 4.3: For the commuting patterns dataset, a synthetic dataset is *not* in the disclosure set $\mathrm{Disc}(D,\epsilon)$ if for for every destination block d and every origin block i, $m(d)_i \leq (e^\epsilon - 1)(n(d)_i + \alpha(d)_i - 1)$ when $n(d)_i \geq 1$ and $m(d)_i \leq (e^\epsilon - 1)(n(d)_i + \alpha(d)_i)$ when $n(d)_i = 0$

Next we will show that the probability of the disclosure set is bounded by $\rho[n(d), m(d), \min(\alpha(d)_1), |\alpha(d)| - \min(\alpha(d)_1), e^\epsilon - 1] \cdot 2ke^\epsilon/(e^\epsilon - 2)$:

Theorem 4.4: Let D be a data set, let $\epsilon > \ln 3$ and let $m(d) = n(d)$ for each destination block d. Let there be a total of k Census blocks. Algorithm 1 satisfies (ϵ,δ)-probabilistic differential privacy if for each destination block d, the reference 0-sample $\rho(n(d), m(d), \min(\alpha(d)_1), |\alpha(d)| - \min(\alpha(d)_1), e^\epsilon - 1)$ is at most $\delta(e^\epsilon - 2)/(2ke^\epsilon)$.

Notice that in probabilistic differential privacy, δ is a constant while in (ϵ,δ)-indistinguishability it is a negligible function. To achieve (ϵ,δ)-indistinguishability, it was sufficient for each α_i to grow faster than $\ln n$. We can show for probabilistic differential privacy that it is enough to have $\alpha_i \in O(f)$ where f is any function that grows faster than $\sqrt{n \ln n}$, although we have found that the α_i can be very small. Thus, the amount of noise (the prior sample) is very small compared to the data and its influence the output reduces as the amount of real data grows. Thus with probabilistic differential privacy we satisfy the intuition that synthetic data generation (Algorithm 1) can achieve privacy with little noise (as discussed at the beginning of Section IV-A).

V. Revising the algorithm

In this section we discuss several problems with the utility of Algorithm 1 and refine Algorithm 1 to make it produce more useful synthetic data. The first problem is that the resulting data may be very unrepresentative of the original data (and therefore useless). For example, it is possible (albeit with very small probability) that in the synthetic data, all the workers commuting to New York city come from Boston (or worse, from San Francisco) even though this is not the case in the original data. The second problem is that with a large domain, the total amount of noise required to guarantee privacy may swamp most of the signal. These issues are not unique to the Algorithm 1 and our mapping application, but also to existing techniques (like [14] and [13]). We show how to ameliorate these effects by employing the probabilistic differential privacy as the privacy criterion. In Section V-A we will discuss when to throw away unrepresentative synthetic data, and in Section V-B we will discuss how to effectively shrink the size of the domain.

A. Accept/Reject

Randomized synthetic data generation algorithms produce unrepresentative outputs with small probability. Although rare, these events cause problems for data analysis. A simple technique to avoid unrepresentative outputs, which we call the *accept/reject* method, is to choose a "representativeness" metric and rerun the algorithm until we get an output which is representative of the input. If the algorithm generates an unrepresentative output with a probability at most p, then the expected number of steps before the accept/reject algorithm halts is at most $\frac{1}{1-p}$. However, special care must be taken to avoid privacy breaches.

Lemma 5.1: Let $\mathrm{Good}(X)$ be the set of outputs that are representative of X. If there exists X_1 and X_2 that differ in only one entry and $\mathrm{Good}(X_1) \neq \mathrm{Good}(X_2)$, then for every $\epsilon > 1$, Algorithm 1 combined with the accept/reject method does not satisfy ϵ-differential privacy.

We illustrate the above lemma with an example. Intuitively, the number of people generated in origin block i by the synthetic data generation process should be proportional to $n(d)_i + \alpha(d)_i$. Hence, consider the following γ-*representativeness* metric for a dataset X:

$$\gamma\text{-}Rep(X) = \{S | \forall i, \forall d, m(d)_i \leq \gamma(\alpha(d)_i + [n(d)_i - 1]^+)\}$$

where $[y]^+$ is the non-negative part of y (i.e. $[y]^+ = \max(y, 0)$). In fact, we showed in Theorem 4.3 that any output S which is in γ-Rep(X) is not in Disc$(X, \ln(\gamma + 1))$. Hence, one may expect that the synthetic data generator coupled with the accept/reject method (denoted by $\mathcal{A}^{a/r}$) guarantees ϵ-differential privacy, for $\epsilon = \ln(\gamma + 1)$. On the contrary, however, this algorithm violates differential privacy because the probability $P(\mathcal{A}^{a/r}(X) = S)$ is no longer the same as $P(\mathcal{A}(X) = S)$. Consider, a synthetic dataset S with $m(d)_i = \gamma(\alpha(d)_i + [n(d)_i - 1]^+)$, $\gamma > 1$. Let X' be a table that differs from X in one entry such that $n'(d)_i = n(d)_i - 1$. Clearly, $S \notin \gamma\text{-}Rep(X')$. Hence,

$$\frac{P(\mathcal{A}^{a/r}(X) = S)}{P(\mathcal{A}^{a/r}(X') = S)} = \infty$$

Despite this result, the accept/reject method is compatible with the (ϵ, δ)-probabilistic differential privacy approach. Again, let p denote the probability that the synthetic data generator outputs a synthetic dataset S that is not representative of the data ($S \notin \gamma\text{-}Rep(X)$). We show, in Lemma 5.2, that the accept/reject algorithm guarantees privacy with probability at least $(1 - p)$ when it only rejects synthetic datasets that *does not* belong to the set γ-*Good*, defined as follows:

$$\gamma\text{-}Good(X) = \bigcup_{X' \,:\, \substack{|X \setminus X'| \leq 1, \\ |X' \setminus X| \leq 1}} \gamma\text{-}Rep(X')$$

Before we prove Lemma 5.2, we remark on the implications for utility when we accept data from $\gamma\text{-}Good(X)$. We can easily show that the counts in a good S satisfy the following condition

$$S \in \gamma\text{-}Good(X) \;\Rightarrow\; \forall i, \forall d, m(d)_i \leq \gamma(n(d)_i + 1 + \alpha(d)_i)$$

Lemma 5.2: Let $\mathcal{A}^{a/r}$ denote the synthetic data generator coupled with an accept/reject method which discards $S \notin \gamma\text{-}Good(X)$. If p is the maximum probability over all X that the synthetic data is not in $\gamma\text{-}Rep(X)$, then $\mathcal{A}^{a/r}$ guarantees (ϵ, p)-probabilistic differential privacy, where $\epsilon = \ln \frac{(\gamma+1)}{1-p}$.

For the accept/reject method, there are theorems analogous to Theorems 4.3 and 4.4 which can be used to select the α_i (details are omitted due to space constraints).

B. Shrinking the Domain

For a randomized algorithm to satisfy differential privacy, (ϵ, δ)-indistinguishability, or probabilistic differential privacy, it must add noise to every origin block so that for every origin block/destination block pair, there is a chance that the synthetic data will contain a synthetic individual that commutes from that origin block to that destination block. On the contrary, if noise is not added to some origin block, privacy can be breached as follows. Consider all the workers with jobs in a destination d (say, in New York). Suppose the adversary knows the origins of all the workers except one, and suppose the adversary knows that the last worker commutes from either o_1 (in Albany) or o_2 (in Schenectady), and no other worker commutes from o_1 or o_2. Now, if noise is added to o_1, but not to o_2, and the output synthetic data contains at least one worker commuting from o_2, then it is clear that the last worker comes from o_2 and not from o_1, thus breaching his privacy.

For Algorithm 1 to maintain privacy guarantees, for each destination block d, it needs to set a value $c(d)$ so that $\alpha(d)_i \geq c(d)$ for all origin blocks i. Since the data is sparse (i.e. the origin blocks for a destination block d are usually the surrounding census blocks and also major metropolitan areas, rather than the whole United States), most of this noise is concentrated in areas where there are no commuters to destination block d. In fact, the amount of noise is the sum of the $\alpha(d)_i$ for all blocks i that have no commuters to d. Thus the data generator may generate many strange fictitious commuting patterns.

One way to tackle this problem is to reduce the size of the origin block domain and therefore reduce the number of blocks with no commuters to d, thus reducing the sum of $\alpha(d)_i$ for such blocks. In this paper we propose two ways of handling this: coarsening the domain and randomly choosing which parts of the domain will contain noise.

To coarsen the domain, we partition the origin blocks and merge together blocks within each partition. A partition must be chosen with care because it can leak information about the data. One way to do this is to cluster the data using a privacy-preserving clustering algorithm that is compatible with differential privacy, such as the k-means algorithm in the SULQ framework [19]. One difficulty with this approach lies in evaluating the quality of the clustering. Despite metrics that measure the quality of a clustering, numbers do not tell the whole story and an expert's subjective judgment is often necessary. The effect of the expert (the data publisher generating synthetic data) on privacy is difficult to quantify and so it lies outside the differential privacy framework.

For this reason, we suggest that the partition be selected from publicly available data. For our application this is possible because of a previous release of similar data. Thus for a given destination block d we can coarsen the domain of its origin blocks using this partition. To plot (on a map) the origin of a synthetic individual, we first select the partition the individual is commuting from (as in Algorithm 1) and then we choose a specific point inside this partition using a

Algorithm 2 Sample_Domain

Require: $\overrightarrow{n(d)}$, function $f_d : \{1, \ldots, k\} \rightarrow (0, 1]$
  Select $\overrightarrow{\alpha(d)}$ so that Theorem 4.4 is satisfied
  New_Domain= $\emptyset$
  for $i = 1..k$ **do**
    if $n(d)_i > 0$ **then**
      New_Domain=New_Domain $\bigcup\{i\}$
    else
      Let X be a binomial($f_d(i)$) random variable
      if $X == 1$ **then**
        New_Domain=New_Domain $\bigcup\{i\}$
      else
        $\alpha(d)_i = 0$
      end if
    end if
  end for
  return New_Domain

density function derived from external data (such as publicly available population density information).

Theorem 5.1: Let $\mathcal{A}$ be a randomized algorithm that satisfies ϵ-differential privacy, (ϵ, δ)-indistinguishability, or (ϵ, δ)-probabilistic differential privacy for the origin block/destination block application. If for each destination block d the domain of origin blocks is coarsened, then $\mathcal{A}$ satisfies the same privacy criterion with the same parameters.

Even after coarsening, the domain may still be large. We can trade off privacy versus the amount of noise added to regions with no commuters with the following approach. For a given destination block d, let f_d be a function that assigns to every origin block a number in the interval $(0, 1]$ (this function must not depend on the data). For each origin block i that does not appear in the data, we keep it in the domain with probability $f_d(i)$ and drop it from the domain with probability $1 - f_d(i)$. Those blocks that are dropped from the domain have their α_i set to 0 in Algorithm 1. Effectively, we are reducing the size of the domain by throwing out origin blocks when creating synthetic data for destination block d. This procedure is illustrated in Algorithm 2. Note that it is important to choose the vector $\overrightarrow{\alpha(d)}$ (in particular, to determine the minimum value of any $\alpha(d)_i$) before shrinking the domain (and for those i that do not belong to the domain, α_i is set to 0). Algorithm 1 is then run using this new domain and $\overrightarrow{\alpha(d)}$. We can quantify the loss in privacy due to applying Algorithm 2 for each destination block d followed by Algorithm 1 with the following theorem:

Theorem 5.2: Let $\mathcal{A}$ be a randomized algorithm for which the reference 0-sample satisfies the conditions in Theorem 4.4. Let $\mathcal{C}$ be the randomized algorithm that chooses the domain of origin blocks for each destination block d using Algorithm 2 and then applies $\mathcal{A}$ on this new domain. Then $\mathcal{C}$ satisfies (ϵ', δ)-probabilistic differential privacy, respectively, where $\epsilon' = \epsilon + \max_i \ln(1/f_d(i)) + \max_i \lceil \alpha(d)_i \rceil \ln 2$.

VI. Experiments

The goal of combining probabilistic differential privacy with synthetic data generation was to develop a system that can be used for practical distribution of products from statistical agencies. In order to assess our progress towards that goal, we applied Algorithms 1 and 2 to public-use data from the Census Bureau's OnTheMap (OTM) micro-data files (http://lehdmap2.did.census.gov/themap/). The versions of those files that are available for public use are themselves synthetic data. However, in our experimental evaluation we treat the Census Bureau's released file as if it were the ground truth. Thus we measure the privacy protection and analytical validity of our experimental synthetic data against the "gold standard" of the OTM data.

Although much privacy research has focused on answering range queries [20], [9], [14], [21], in contrast to this work, we decided to evaluate the quality of the data using a measure for which we did *not* explicitly tune our anonymization algorithm. We computed the average commute distance for each destination block and compared it to the ground truth from OnTheMap. Note that the domain covers a two-dimensional surface since the average commute distance is not a linear statistic.

For our data evaluation, we selected a subset of OTM Version 2 data such that all destination workplaces are in Minnesota, yielding 1,495,415 distinct origin/destination block pairs (O/D pairs) which contain a commuter traveling from the origin block to the destination block. Many of these O/D pairs are singletons (i.e., they had only one commuter).

The actual partition of the United States into Census blocks comes from the Census 2000 Summary File 1, which also contains information about the block characteristics and population counts. 8,206,261 blocks cover the 50 United States and the District of Columbia. We also used data from the previously published Census Transportation Planning Package (CTPP), which contains data similar to the O/D pairs from OnTheMap, based on Census 2000. We used CTPP to reduce the size of the domain for the origin blocks.

A preliminary data analysis using the CTPP data revealed that in Minnesota, 50% of all O/D work commutes were less than 6.23 miles, 75% less than 13.1 miles, and 90 percent less than 21.54 miles. Commutes longer than 200 miles were beyond the 99^{th} percentile and so we removed from the domain all origin blocks that were over 200 miles from their destination (and we also suppressed such O/D pairs in the data). As a result of this pruning, the maximum size of the domain of origin blocks in the experiments is 233,726.

We selected the minimum $\alpha(d)_i$ values using Theorem 4.4 with $\epsilon = 4.6$ and $\delta = 0.00001$. Recall that ϵ represents the maximum allowed disclosure due to the generation of synthetic data, and δ represented the maximum probability of a breach of ϵ-privacy. We also used Algorithm 2 to shrink the domain and the probability function f_d that is used in Algorithm 2 was created based on the CTPP data. The probability function f_d was roughly proportional to the

histogram of commute distances as determined by the CTPP data. The maximum of the $\lceil \alpha(d)_i \rceil$ was 1 and the minimum value of $f_d(i)$ was 0.0378. Thus the additional disclosure due to shrinking the domain was $\ln(1/0.0378) + \ln 2 \leq 4$. The overall ϵ for the procedure was 8.6 and disclosure probability δ was $= 0.00001$.

The results presented in this section are those of 120 blocks that are selected uniformly at random, averaging 15 primary job holders per destination block. For each block we computed the length of the average commute to that block, and compared the corresponding average commute distances in the OnTheMap data to the synthetic data that we generated.

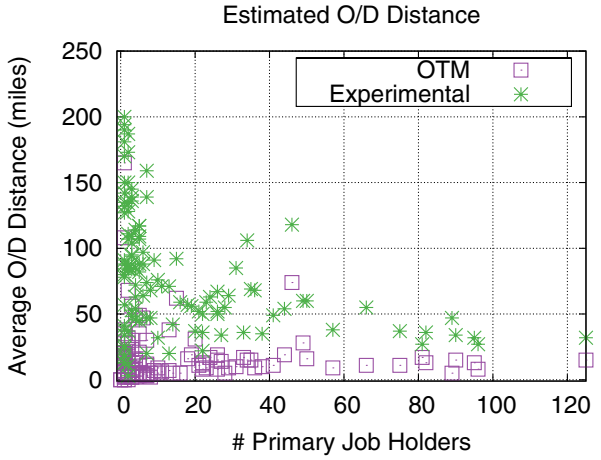

Fig. 1.

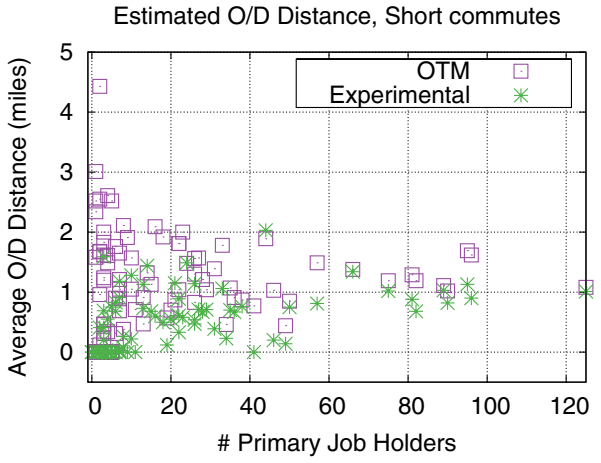

Fig. 2.

In experiment 1, our privacy algorithm added an additional 419 primary job holders to each destination with a $\min \alpha(d)_i$ of 0.01245. Figure 1 shows the relation between average commute distance as measured in OTM and in our experimental data. Each point in the figure corresponds to a particular destination block. The x-axis is the number of people whose primary job is in the destination block and the y-axis is the average commute distance to that destination block. It is clear that the experimental shape of f_d and values of $\min f_d$ and $\min \alpha_i$ admitted too many distant synthetic workers. The synthetic data overestimated the number of commuters with long commutes. This effect is strongest when the destination block has few workers and diminishes as the number of workers increases.

Figure 2 shows the same plot except that it is restricted to short commutes (i.e., those commutes that are shorter than the 6.23 miles which is the median commute distance in CTPP). Here the synthetic data better matches the ground truth. Note that the synthetic data slightly underestimates the commute distances (as a result of the fact that long commutes were overestimated, while the total number of commuters matched the ground truth). Again, the estimation error diminishes as the number of workers in a destination block increases.

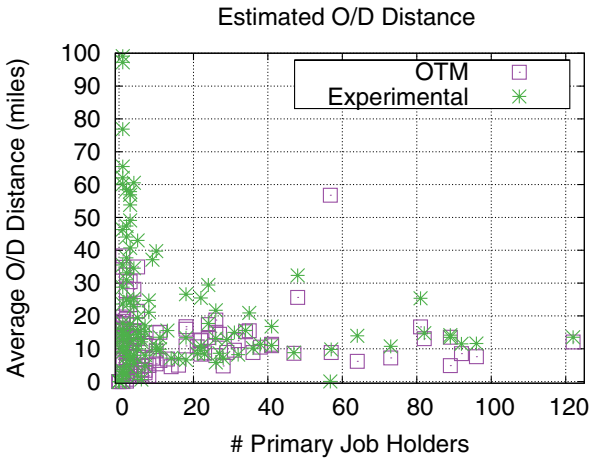

Fig. 3.

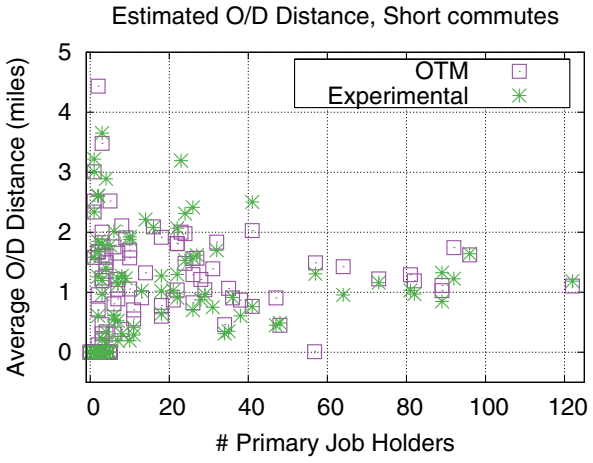

Fig. 4.

To further limit the effect of domain size on the estimation of commute distances, in our second experiment we restricted the domain of the origin blocks to be within 100 miles of

the destination, keeping ϵ and δ values unchanged. Commutes of more than 100 miles were still beyond the 99^{th} percentile for CTPP. This reduced the domain size to 120,690. The overall ϵ and δ values remained unchanged. In this case, $\min \alpha(d)_i = 0.039764$. Figures 3 and 4 show the results for all distances and for short commutes, respectively. The tighter restriction on the domain significantly enhanced data quality in the moderate commute distances, did not diminish quality in the short distances, and reduced the bias at all distances. We can see that the extremely long commutes are again overestimated, but the destination blocks with long average commutes have few workers, so accurate estimates are not expected. This suggests that long commutes should be modeled separately when creating synthetic data. We leave this as an interesting avenue for future work.

VII. Related Work

This work builds on a large amount of research on privacy in both the statistics and the computer science communities. Though the early privacy aware algorithms on statistical databases had sound intuition [22], most of them lacked formal privacy guarantees. These algorithms include generalizations [23], [24], [25], cell and tuple suppression [26], [25], random perturbation [22], [27], publishing marginals that satisfy a safety range [28], and data swapping [29] (where attributes are swapped between tuples so that certain marginal totals are preserved). Recent research in the statistics community has focused on generating partial and fully synthetic populations [1], [30], [3] that preserve some statistics of the original data using re-sampling and multiple imputation techniques [16]. In this paper we studied the privacy properties of one such technique and adapted it to meet formal privacy guarantees.

Many recent papers have considerably advanced the state of the art in defining privacy and giving formal guarantees for privacy. k-Anonymity [6] is a simple privacy criterion defined for techniques that use generalization and suppression to guard against linking attacks. Many algorithms have been proposed to ensure k-anonymity [31], [24], [23] . However, k-anonymity does not model sensitive information and attacker background knowledge. ℓ-Diversity [7] is a privacy criterion that guards against certain kinds of background knowledge that an adversary may use to infer sensitive information. It is also compatible with many k-anonymity algorithms. Subsequent work [8] characterized the worst case knowledge needed to break the anonymity of a data-set and extended the work to releases of multiple views of the data [32]. These privacy formulations assume a bound on the adversary's background knowledge. Many variations on these privacy conditions have also been proposed [33], [21].

(ρ_1, ρ_2)-privacy and γ-amplification [12] bound the point-wise distance between the adversary's prior belief in a property and the adversary's posterior belief in that property after seeing a randomized version of the data. This definition was applied in the context of association rule mining. Other approaches [20] ensure (ρ_1, ρ_2)-breaches only on some parts of the data. (d, γ)-privacy [14] is a probabilistic privacy definition for data publishing in which all tuples are considered independent and the privacy is guaranteed by bounding the prior $P(t)$ and the posterior $P(t|D)$ after seeing the published data D.

Differential privacy [13] is a privacy criterion related to γ-amplification. It requires an anonymization algorithm to be fairly insensitive to perturbations in the data. Thus, if the algorithm were run on two datasets that differ in one entry, the corresponding probabilities of different outputs would be similar. This privacy condition, however, has mostly been applied in the context of output randomization [34], [15], [18]. These techniques require the knowledge of the exact set of queries that need to be answered by the database. However, this does not fit the exploratory nature of most data analysis. Recent approaches have tried to apply this framework to the release of anonymized contingency tables [35]. This technique adds random noise to the Fourier coefficients of the collection of tables and the post-processes them so that table entries are integral and non-negative. Random sampling [36] has also been analyzed in the differential privacy framework.

In the statistics community, a popular data anonymization technique is the creation of synthetic data. It was first proposed by [1]. Subsequently, many techniques have been proposed for generating synthetic data [30], [37], [38] and creating statistical inferences from them [2], [3]. On the other hand, there has been comparatively little research on the privacy of synthetic methods, with most work only focusing on the ability of an attacker to determine that an individual is in the data [4]. Hence, the results in this paper complements the work in the statistics literature.

VIII. Conclusions

In this paper we showed that, with a little work, state-of-the art ideas from the privacy research about privacy guarantees and statistical inference can be combined for a practical application. One remaining challenge is handling datasets with large domains because noise must be spread throughout the domain even if the data are sparse. This is necessary because if an outlier appears in the synthetic data, it may be more likely that a similar outlier was present in the real data and less likely that it was due to random noise. In our application we were able to use exogenous data and other techniques to help reduce the domain size. However, such data are not always available. Furthermore, even though we reduced the domain size, the data were still sparse and as a result, the addition of noise to all parts of the reduced domain created many outliers, so that the distribution of commute distances was reasonable only for the study of commutes that were not extremely long.

We believe that judicious suppression and separate modeling of outliers may be the key since we would not have to add noise to parts of the domain where outliers are expected. For future work, we are considering methods for incorporating outlier identification, suppression, and modeling to the privacy and utility guarantees for the mapping application.

References

[1] D. B. Rubin, "Discussion statistical disclosure limitation," *Journal of Official Statistics*, vol. 9, no. 2, 1993.

[2] T. Raghunathan, J. Reiter, and D. Rubin, "Multiple imputation for statistical disclosure limitation," *Journal of Official Statistics*, vol. 19, pp. 1–16, 2003.

[3] J. Reiter, "Inference for partially synthetic, public use microdata sets," *Survey Methodology*, vol. 29, no. 2, pp. 181–188, 2003.

[4] ——, "Estimating risks of identification disclosure for microdata," *Journal of the American Statistical Association*, vol. 100, pp. 1103 – 1113, 2005.

[5] G. T. Duncan and D. Lambert, "Disclosure-limited data dissemination," *Journal of the American Statistical Association*, vol. 81, no. 393, 1986.

[6] L. Sweeney, "k-anonymity: a model for protecting privacy," *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, vol. 10, no. 5, pp. 557–570, 2002.

[7] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "ℓ-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, 2007.

[8] D. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J. Halpern, "Worst case background knowledge for privacy preserving data publishing," in *ICDE*, 2007.

[9] X. Xiao and Y. Tao, "Anatomy: Simple and effective privacy preservation," in *VLDB*, 2006.

[10] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and ℓ-diversity," in *ICDE*, 2007.

[11] B. Chen, K. Lefevre, and R. Ramakrishnan, "Privacy skyline: Privacy with multidimensional adversarial knowledge," in *VLDB*, 2007.

[12] A. Evfimievski, J. Gehrke, and R. Srikant, "Limiting privacy breaches in privacy preserving data mining," in *PODS*, 2003.

[13] C. Dwork, "Differential privacy," in *ICALP*, 2006.

[14] V. Rastogi, D. Suciu, and S. Hong, "The boundary between privacy and utility in data publishing," University of Washington, Tech. Rep., 2007.

[15] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis." in *TCC*, 2006, pp. 265–284.

[16] D. B. Rubin, *Multiple Imputation for Nonresponse in Surveys*. Wiley-Interscience, 2004.

[17] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis*, 2nd ed. Chapman & Hall/CRC, 2003.

[18] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis," in *39th ACM Symposium on Theory of Computing (STOC)*, 2007.

[19] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical privacy: The sulq framework," in *PODS*, 2005.

[20] R. Agrawal, R. Srikant, and D. Thomas, "Privacy preserving olap," in *Proceedings of the 23th ACM SIGMOD Conference on Management of Data*, June 2004.

[21] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu, "Aggregate query answering on anonymized tables," in *ICDE*, 2007.

[22] N. R. Adam and J. C. Wortmann, "Security-control methods for statistical databases: A comparative study." *ACM Comput. Surv.*, vol. 21, no. 4, pp. 515–556, 1989.

[23] R. J. Bayardo and R. Agrawal, "Data privacy through optimal k-anonymization," in *ICDE*, 2005.

[24] K. LeFevre, D. DeWitt, and R. Ramakrishnan, "Incognito: Efficient fulldomain k-anonymity," in *SIGMOD*, 2005.

[25] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," CMU, SRI, Tech. Rep., 1998.

[26] L. H. Cox, "Suppression, methodology and statistical disclosure control," *Journal of the American Statistical Association*, vol. 75, 1980.

[27] R. Agrawal and R. Srikant, "Privacy preserving data mining," in *SIGMOD*, May 2000.

[28] A. Dobra, "Statistical tools for disclosure limitation in multiway contingency tables," Ph.D. dissertation, Carnegie Mellon University, 2002.

[29] T. Dalenius and S. Reiss, "Data swapping: A technique for disclosure control," *Journal of Statistical Planning and Inference*, vol. 6, pp. 73–85, 1982.

[30] J. M. Abowd and S. D. Woodcock, "Disclosure limitation in longitudinal linked data," *Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies*, pp. 215–277, 2001.

[31] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu, "Approximation algorithms for k-anonymity," *Journal of Privacy Technology (JOPT)*, 2005.

[32] D. Kifer and J. Gehrke, "Injecting utility into anonymized datasets," in *SIGMOD*, 2006.

[33] X. Xiao and Y. Tao, "Personalized privacy preservation," in *SIGMOD*, 2006.

[34] I. Dinur and K. Nissim, "Revealing information while preserving privacy," in *PODS*, 2003, pp. 202–210.

[35] B. Barak, K. Chaudhuri, C. Dwork, S. Kale, F. McSherry, and K. Talwar, "Privacy, accuracy and consistency too: A holistic solution to contingency table release," in *PODS*, 2007.

[36] K. Chaudhuri and N. Mishra, "When random sampling preserves privacy," in *CRYPTO*, 2006.

[37] J. Reiter, "Using cart to generate partially synthetic public use microdata," *Journal of Official Statistics*, pp. 441–462, 2005.

[38] A. B. Kennickell, "Multiple imputation and disclosure protection: The case of the 1995 survey of consumer finances," *Record Linkage Techniques*, pp. 248–267, 1997.