Mach Learn
DOI 10.1007/s10994-013-5396-x

Differential privacy based on importance weighting

Zhanglong Ji · Charles Elkan

Received: 17 February 2013 / Accepted: 11 June 2013

Abstract This paper analyzes a novel method for publishing data while still protecting privacy. The method is based on computing weights that make an existing dataset, for which there are no confidentiality issues, analogous to the dataset that must be kept private. The existing dataset may be genuine but public already, or it may be synthetic. The weights are importance sampling weights, but to protect privacy, they are regularized and have noise added. The weights allow statistical queries to be answered approximately while provably guaranteeing differential privacy. We derive an expression for the asymptotic variance of the approximate answers. Experiments show that the new mechanism performs well even when the privacy budget is small, and when the public and private datasets are drawn from different populations.

Keywords Privacy · Differential privacy · Importance weighting

1 Introduction

Suppose that a hospital possesses a dataset concerning patients, their diseases, their treatments, and the outcomes of treatments. The hospital faces a fundamental conflict. On the one hand, to protect the privacy of the patients, the hospital wants to keep the dataset secret. On the other hand, to allow science to progress, the hospital wants to make the dataset public. This conflict is the issue addressed by research on privacy-preserving data mining. How can a data owner simultaneously both publish a dataset and conceal it?

We analyze here a new approach to resolving the fundamental tension between publishing and concealing data. The new approach is based on a mathematical technique called importance weighting that has proved to be valuable in several other areas of research (Hastings 1970). The essential idea is as follows. Let D be the set of records that the owner must

Editors: Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip Železný.

Z. Ji · C. Elkan (✉)
Department of Computer Science and Engineering 0404, University of California, San Diego, USA
e-mail: elkan@ucsd.edu

Published online: 28 June 2013

keep confidential. Let E be a different set of records from a similar domain, and suppose that E is already public. The owner should compute and publish a weight $w(x)$ for each record x in E. Given x in E, its weight is large if x is similar to the records in D while its weight is small otherwise. Data mining on E using the weights will then be approximately equivalent to data mining on D. The owner uses D privately to compute the weights, but never reveals D.

The approach outlined above was suggested originally in a workshop paper (Elkan 2010). This paper proves that the approach does achieve differential privacy, analyzes the variance of answers to queries provided by the approach, and shows experimentally that the approach provides useful accuracy, while still protecting privacy.

2 Framework and related research

A query is a question that people ask about a dataset. For example, if the dataset is a collection of health records, queries can be "how many people in the dataset have disease A?" and "how many people have both disease A and disease B?" In general, let Q be a set of queries. We denote the true answers to all queries in Q based on the dataset D as $Q(D)$. There is a kind of simple and common query called a counting query. These queries are about how many samples in the dataset meet certain conditions. The two example queries above are in this category.

If two datasets D_1 and D_2 differ on at most one entry, then we call them neighbors.[1] Since neighbors are different, the answers to queries on them may also differ. The largest change in the true answers, by some norm $|.|$ for all neighbor sets D_1 and D_2, is called the *sensitivity* of Q:

$$S_Q = \max_{D_1, D_2} \big|Q(D_1) - Q(D_2)\big|.$$

The maximization ranges over all neighbor sets D_1 and D_2. The $|.|$ can be any norm in the space that $Q(D)$ is from, but usually the L_1 or L_2 norm is used.

A (random) mechanism is a randomized algorithm whose input is a dataset and whose output is in a certain answer space. The notion of differential privacy captures how well a mechanism preserves privacy. The mechanism $\mathscr{K}$ is defined to have ϵ-differential privacy (Dwork 2006) if for all neighbor sets D_1 and D_2 and all subsets S of the answer space, the probability inequality

$$P\big(\mathscr{K}(D_1) \in S\big) \leq e^{\epsilon} P\big(\mathscr{K}(D_2) \in S\big)$$

holds. Note that e^{ϵ} equals $1 + \epsilon$ approximately when ϵ is small. In applications, the output $\mathscr{K}(D)$ often depends not only on $\mathscr{K}$ and D but also on a query set Q. A mechanism $\mathscr{K}$ is not required to be able to answer all queries. Given a set of queries Q which the mechanism $\mathscr{K}$ can answer, $\mathscr{K}_Q$ denotes the random answer to Q, which is a mapping from datasets to a random variable over the answer space.

In the definition of differential privacy, the smaller that ϵ is, the more that neighboring datasets lead to similar output probabilities, even though the datasets themselves are different. Therefore, when ϵ is smaller, less information is leaked and privacy is protected better.

[1] There are two different understandings of "differ on at most one entry." Some researchers consider deletion or addition of an entry (Hay et al. 2010; Mohammed et al. 2011), while others consider only replacement (Chaudhuri et al. 2011; Li et al. 2011). The two interpretations are both reasonable. We use the former because it is broader.

Since ϵ determines how accurately we can answer queries, it is called the privacy budget. A smaller budget corresponds to stronger privacy. Intuitively, to ensure stronger privacy, one way or another more noise must be introduced.

A simple but useful mechanism, which applies to queries having bounded sensitivity, is to add random noise as follows to their answers. Given a query set Q with sensitivity S, the mechanism outputs the answer vector $\mathscr{K}_Q(D) = Q(D) + \delta$ where $Q(D)$ is the true answer vector and the noise δ is a vector of real values, with probability density $p(\delta) \propto \exp(-|\delta|\epsilon/S)$. The function $|.|$ here is the same norm as in the definition of S. This mechanism is ϵ-differentially private by Theorem 2 of Dwork et al. (2006). Specifically, when $|.|$ is L_1 norm, the noise added to each dimension is i.i.d. and follows the Laplace distribution $\text{Lap}(S/\epsilon)$ whose density is $p(x; S/\epsilon) = \frac{\epsilon}{2S} e^{-|x|\epsilon/S}$. The bigger the sensitivity S, or the smaller the privacy budget ϵ, the bigger the added noise x on average.

Many differentially private mechanisms have been proposed. Some of them answer unrestricted queries without publishing data (Smith 2008; McSherry and Mironov 2009; Li et al. 2010; McSherry and Mahajan 2010; Rastogi and Nath 2010). The data owner gets queries that are issued by outsiders, and then returns noisy answers directly. These mechanisms share two drawbacks. First, if data owners answer queries independently then they must divide the total privacy budget between the queries. Each query will be answered with privacy budget smaller than ϵ, and hence greater noise. There has been some work taking constraints among the queries into consideration (Hay et al. 2010), but such constraints are not always known. Second, after all the privacy budget is spent, no more questions can be answered. Even if we only spend part of the privacy budget now, we can never release information with the full privacy budget later.

The two drawbacks have motivated researchers to devise data-publishing mechanisms that release a synthetic or modified dataset. If a new dataset that statistically approximates the original one is published, then all questions can be answered, albeit not exactly. If the mechanism that creates the new dataset achieves differential privacy, then all queries can get exact answers from the new dataset without the need to add further noise.

A straightforward data-publishing mechanism simply releases a version of the private dataset with noise added. The maximum L_1 norm of changes among two samples is computed, this is regarded as the sensitivity of the dataset, and i.i.d. Laplacian noise is added to each entry in the dataset. This method, which can be called Laplace perturbation, adds too much noise to be useful in practice; for details see Sect. 5.1.

Some methods publish data after analyzing a pre-determined set of given queries (Blum et al. 2008; Hardt et al. 2012; Hardt and Rothblum 2010). If there is a fixed query set Q, these mechanisms can publish a differentially private dataset that depends on Q, and they can make sure that the published dataset can answer queries in Q accurately with high probability. However if queries outside Q are asked, there is no guarantee that these queries can have accurate answers. Thus these methods are appropriate when the data owner has advance knowledge about what queries may be asked, but they do not provide a useful guarantee without advance knowledge, or when the owner wants to allow the freedom to ask any query after data publication.

There are other data-publishing mechanisms that are query-independent. Some of these methods cluster the whole dataset into several groups according to similarity or entropy (this step either involves randomness in order not to destroy privacy, or is data-independent), add noise to the counts of samples in each group, and publish the noisy counts (Xiao et al. 2010; Mohammed et al. 2011; Ding et al. 2011). These methods also have drawbacks. Partitioning typically clusters samples with different values of a variable into the same group, which loses information. A representative method is given in Mohammed et al. (2011), which publishes set-valued variables that may hide all information concerning some variables. Other

researchers make assumptions such as sparsity concerning the dataset, and use these assumptions to improve performance (Li et al. 2011).

Here, we describe a new data-publishing mechanism based on importance weighting that makes no assumptions concerning the private dataset, but still achieves differential privacy. Although there has been previous work that uses weighting to publish data with differential privacy (Hardt et al. 2012; Hardt and Rothblum 2010), it only provides guarantees for predetermined queries.

3 Importance weighting mechanism

Though counting queries are most common in the literature, queries may come in other forms. If someone wants to learn a model from the dataset, s/he may ask what the gradient vector or Hessian matrix of a loss function is. If s/he wants to study causation among variables in the dataset, s/he may ask what the values of correlation coefficients are. Generally, we suppose that the user wants to know the expectation of some function $b(x)$ over the distribution $p_D(\cdot)$ from which the private dataset D is drawn. That is, the goal is to know $E_D[b(x)] = E_{x\sim p_D(\cdot)}[b(x)]$. The function $b(x)$ is not limited to be an indicator function, as it is for counting queries. Note that E_D is an expectation over p_D, as opposed to over an empirical distribution defined by a specific dataset D.

Suppose that there exists another dataset E that is already public, whose samples are random from the distribution $p_E(\cdot)$. Since the samples in D have privacy concerns but those in E do not, we want to use E to help estimate $E_D[b(x)]$. Because D and E in general arise from different distributions, it is not reasonable to simply compute the average of $b(x)$ over E. Importance weighting varies the weights of the samples in E in order to improve accuracy. Let the cardinalities of E and D be N_E and N_D. The goal is to find a weight $w(x)$ for each x in E such that for any function $b(x)$ the following equation is approximately satisfied:

$$E_D\big[b(x)\big] = \frac{1}{N_E}\sum_{x\in E} b(x)w(x). \tag{1}$$

If E is already public and the owner of D publishes the weights $w(x)$ in a way that guarantees differential privacy, then outsiders can estimate $E_D[b(x)]$ without access to D, for any $b(x)$, without violating privacy, by computing $\frac{1}{N_E}\sum_{x\in E} b(x)w(x)$.

In general, no $w(x)$ can make (1) be satisfied exactly for all possible $b(x)$ when the dataset E is finite. So, we explain here a differentially private mechanism $\mathscr{K}$ based on logistic regression that yields weights that make the equation hold approximately. The output of the mechanism is the set of weights, that is $\mathscr{K}(D) = \{w(x) : x \in E\}$.

The so-called importance sampling identity is the equation

$$E_D\big[b(x)\big] = E_E\left[b(x)\frac{p_D(x)}{p_E(x)}\right].$$

To be valid, the support of the distribution p_E must contain the support of p_D, that is if $p_D(x) > 0$ then $p_E(x) > 0$ must be true also. Equation (1) and the identity make $p_D(x)/p_E(x)$ a natural choice for $w(x)$.

For a sample x, its importance weight $w(x)$ is the ratio of the probability density of x according to the two different distributions p_D and p_E. Both these distributions are in general high-dimensional densities, where the dimensionality is the length of the x vectors. Estimating high-dimensional densities is difficult at best, and often infeasible (Scott 1992).

Algorithm 1 Importance weighting mechanism

Require: Private dataset D, public dataset E, privacy budget ϵ, regularization strength λ. Each sample x in D has d components that are in $[0, 1]$.

Ensure: Weight $w(x)$ for each x in E.

1: Regularized logistic regression: Obtain β^* by solving

$$\beta^* = \arg\min_{\beta} -\frac{1}{N_E}\sum_{x\in E}\log p(x \in E|x \in D \cup E) - \frac{1}{N_D}\sum_{x\in D}\log p(x \in D|x \in D \cup E) + \frac{\lambda}{2}\|\beta\|^2$$

where $p(x \in D|x \in D \cup E) = 1 - p(x \in E|x \in D \cup E) = 1/(1 + \exp(-\beta^T x))$.

2: Add high dimensional Laplace noise to β^* to get the final perturbed β:

$$\beta = \beta^* + \delta \quad \text{where } p(\delta) \propto \exp\left(-\frac{\epsilon\|\delta\|_2 N_D\lambda}{\sqrt{d}}\right).$$

3: Output $w(x) = (N_E/Z)e^{\beta^T x}$ for each x in E, where $Z = \sum_{x\in E} e^{\beta^T x}$.

Fortunately, one can estimate the ratio $w(x)$ indirectly, without estimating p_D and p_E explicitly. Consider an equally balanced mixture of the distributions p_D and p_E, and suppose that samples from p_D are extended with the label $s = 1$ while those from p_E are extended with the label $s = 0$. A similar idea was used previously by Smith and Elkan (2004) and Elkan and Noto (2008). Then,

$$p(s = 1|x) = \frac{p(x|s = 1)p(s = 1)}{p(x)} = \frac{p_D(x)(1/2)}{p(x)}$$

by Bayes' rule. Therefore,

$$p(s = 1|x) = \frac{p_D(x)(1/2)}{p_D(x)(1/2) + p_E(x)(1/2)} = \frac{1}{1 + p_E(x)/p_D(x)}.$$

We can derive

$$w(x) = \frac{p_D(x)}{p_E(x)} = \frac{1}{1/p(s = 1|x) - 1}.$$

This equation lets us write each weight $w(x)$ as a deterministic transformation of $p(s = 1|x)$. The equation is correct as a statement of probability theory. Its practical usefulness depends on having a good model for $p(s = 1|x)$.

Concretely, we treat the datasets D and E as training sets for two classes $s = 1$ and $s = 0$. The logistic regression model

$$p(s = 1|x) = p(x \in D|x \in D \cup E) = \frac{1}{1 + e^{-\beta^T x}}$$

which yields $w(x) = e^{\beta^T x}$ is an obvious choice. However, it cannot ensure differential privacy directly, because there is no bound on the sensitivity of the logistic regression parameters β when D changes by one sample. If we use a strongly convex penalty function (definition follows), such as the sum of squared components of β in Step 1 of Algorithm 1, and if each sample x in D is a vector of length d with components that are in the range $[0, 1]$, then the following theorem says that ϵ-differential privacy is achieved. The proof is

in the appendix. The parameter of the Laplace distribution in Algorithm 1 has denominator $\sqrt{d}$ because that is the maximum norm of any x. In general, $\sqrt{d}$ can be replaced by the upper bound over D of the L_2 norm of samples.

Theorem 1 *The random mechanism of Algorithm 1 is ϵ-differentially private.*

A common issue with importance weighting is that a few samples may have large weights, and these increase the variance of estimates based on the weights. There are various proposals using techniques such as softmax to make weights more uniform. Let τ be a constant. When $0 \leq \tau < 1$, the modified weights $w'(x) \propto w(x)^\tau \propto \exp(\tau\beta^T x)$ are less extreme. This is equivalent to replacing β by $\tau\beta$. Since softmax makes the norm of β smaller, its effect is similar to that of a larger penalty coefficient λ in Algorithm 1. We can use a larger λ to reduce the impact of individual samples in E on estimates, and introducing a separate constant τ is not necessary.[2]

As the strength of regularization λ increases, the learned coefficients β^* in Algorithm 1 tend towards zero, and the weights $w(x)$ tend towards one. This implies that estimates computed using (1) increase in bias and tend towards the corresponding mean computed on the public dataset E. This property is evident in the statement of Theorem 2 below and in the experimental results (Fig. 1). In practice, solving the regularized optimization problem in Step 1 of the algorithm is computationally straightforward and fast regardless of the magnitude of λ.

Algorithm 1 adds noise to the coefficients β^* in order to protect privacy. An alternative approach to guarantee privacy with logistic regression is to perturb the objective function used for training (Chaudhuri et al. 2011). Although we do not have theoretical results showing how well this alternative approach works, experiments indicate that its performance is similar to that of Algorithm 1.

4 Analysis

For a query function $b(x)$, the estimate of its true expectation $E_D[b(x)]$ obtained via the differentially private importance weighting mechanism is

$$\frac{1}{N_E}\sum_{x\in E} b(x)w(x).$$

Here we analyze the variance of this estimate. We assume that the public dataset E is fixed, so the variance of the estimate comes from the randomness of the dataset D and from the

[2] In standard regularized logistic regression, the loss function that is minimized is

$$-\frac{1}{N_E+N_D}\left[\sum_{x\in E}\log p(x\in E)+\sum_{x\in D}\log p(x\in D)\right]+\frac{\lambda}{2}\|\beta\|^2.$$

Instead, we use the balanced loss function

$$-\frac{1}{N_E}\sum_{x\in E}\log p(x\in E)-\frac{1}{N_D}\sum_{x\in D}\log p(x\in D)+\frac{\lambda}{2}\|\beta\|^2$$

which gives the log likelihoods for examples from D and E equal mass. In our scenarios, the samples in E are fixed, while the samples in D are random. With the usual form of logistic regression, the asymptotic convergence, in Step 1 of Algorithm 1, of β^* to the true parameter vector is not guaranteed.

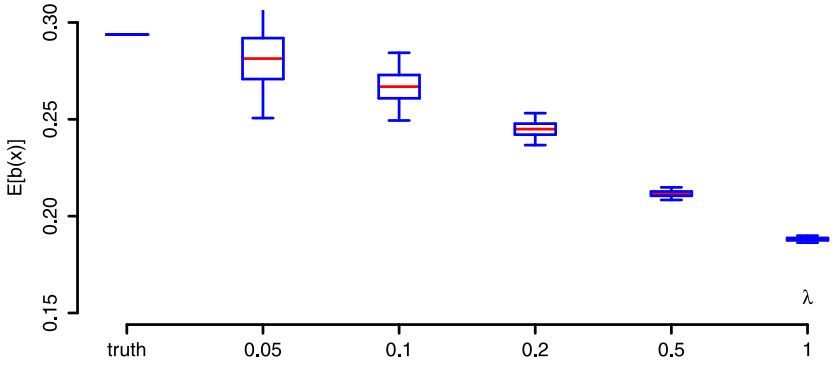

Fig. 1 Performance of the importance weighting method as the strength of regularization λ varies. The query is $b(x) = I(\text{income} > \$50K)$ and the privacy budget is $\epsilon = 0.1$. The *vertical axis* shows the estimated answer to the query, while the *horizontal axis* shows values of λ. "Truth" indicates the correct answer. *Box plots* show variation over 100 randomly created private datasets D. Note that answers based directly on D are unbiased and have standard deviation less than 0.003

noise in Step 2 of Algorithm 1. Note that even in the absence of privacy concerns, there is variance in any estimate of $E_D[b(x)]$ due to randomness in D.

The weights are based on the logistic regression parametric model that $p_D(x)/p_E(x) = \exp(\beta^T x)$ for some β. The difference between the estimate and the true value may not converge to zero when this parametric assumption is not true, that is when logistic regression is not well-specified. However, we can give an upper bound on the variance of the estimate that converges to zero asymptotically, that is as the cardinality of D tends to infinity, regardless of whether logistic regression is well-specified.

Theorem 2 *The total variance* $\text{Var}[\frac{1}{N_E}\sum_{x\in E} b(x)w(x)]$ *is asymptotically less than*

$$\alpha^T\left(\frac{d}{N_D\lambda^2}I + \frac{d(d+1)}{(N_D\lambda\epsilon)^2}I\right)\alpha$$

where d *is the dimensionality of data points* x, I *is the identity matrix, and*

$$\alpha = \frac{\sum_{x_i,x_j\in E} e^{\beta_0^T(x_i+x_j)}(b(x_i)-b(x_j))(x_i-x_j)}{\sum_{x_i,x_j\in E} e^{\beta_0^T(x_i+x_j)}}.$$

Proof See Appendix B. The vector β_0 minimizes the loss function of logistic regression on E and the distribution p_D. Details are in the appendix. □

Theorem 2 provides a strict inequality. We write Var[] and not $\text{Var}_D[]$ because the variance includes not only randomness from D, but also randomness from the noise in Step 2 of Algorithm 1. The factor α comes from the derivative with respect to β of the estimate $\frac{1}{N_E}\sum_{x\in E} b(x)w(x)$.

A large N_D ensures a decrease of the variance of β^* and of estimates, because more samples have less noise on average, and also because the noise needed for privacy is less due to smaller sensitivity of β^*. The rate of decrease $1/N_D$ is of the same order as for the variance of direct estimates $\frac{1}{N_D}\sum_{x\in D} b(x)$, which of course is $\frac{1}{N_D}\text{Var}_D[b(x)]$. Thus differential privacy can be achieved without slowing the convergence of estimates compared to the absence of privacy, that is using the dataset D directly.

A large λ can reduce the Laplacian noise significantly, but if it is too large, then the bias in estimates can be large. A large privacy budget ϵ helps reduce the Laplacian noise, and hence reduces the variance of estimates. However, ϵ may be specified by policy, and making it larger will harm privacy. Moreover, if $N_D\epsilon^2 \gg d$, then the first term dominates and a smaller ϵ cannot help reduce the variance.

When the number of dimensions d increases, the variance gets larger for two reasons. First, the L_2 sensitivity of β^* increases. Second, the curse of dimensionality worsens the situation: if $p(\delta) \propto \exp(-\|\delta\|_2)$ with $\delta \in R^d$ then $E[\|\delta\|_2]$ increases linearly with d. For details see Appendix B.

The factor α is the most complicated among the factors that determine the variance of estimates. It is not controllable, because the function $b(x)$ and the public dataset E must be taken as fixed. However, the expression for α reveals which $b(x)$ can be estimated with smaller variance: if the values of $b(x)$ in E are close to each other, especially on the samples for which $w(x)$ is large, then α can be small.

Theorem 2 is useful not only for bounding the variance, but also for bounding the total error under some conditions. Specifically, suppose that logistic regression is well-specified and regularization is weak, meaning that λ is small and β_0 exists such that $p_D(x)/p_E(x) = \exp(\beta_0^T x)$. The existence of β_0 means that (1) holds for any $b(x)$ highly accurately with $w(x) = \exp(\beta_0^T x)$. Small λ means that β^* is close to β_0 given large N_D, and that the β^* and β vectors are approximately unbiased. Hence, the estimate is approximately unbiased.

The argument about asymptotic unbiasedness is formalized in the appendix in Theorem 3. Combining Theorems 2 and 3, variance and bias are both small, and hence total error is small, when the four following conditions hold: (i) there exists β such that $\frac{p_D(x)}{p_E(x)} \propto \exp(\beta^T x)$, (ii) the regularization strength λ is small so that β_0 is close to β and thus the bias is small, (iii) the number of samples in D is large so that the estimate has small variance, and (iv) the number of samples in E is large so that the weighted sum over E converges to $E_E[b(x)\frac{p_D(x)}{p_E(x)}]$

5 Design of experiments

Here we investigate empirically the usefulness of the importance weighting method. We see how parameter values (the strength of regularization λ and the privacy budget ϵ) affect the accuracy of estimates obtained using the method, and how the method behaves with different target functions, that is queries.

The dataset we use is derived from the "adult" dataset in the UC Irvine repository (Frank and Asuncion 2010). The original dataset contains more than 40,000 records, each corresponding to a person. Each record has 15 features: sex, education level, race, national origin, job, etc. The first 14 features are often used to predict the last one, which is whether a person earns more than $50,000 per year. We use a processed version which has 63 binary variables obtained from 12 original features, taken from the R package named "arules" (Hahsler et al. 2011). In general, preprocessing a dataset is a computation that must be taken into account in a privacy analysis, but here we assume that the private dataset is the preprocessed one as opposed to the original one. The preprocessing was done by other researchers for reasons unrelated to privacy, so the dataset was not created to favor any particular approach to privacy preservation.

Our approach needs a public dataset E. There is a test set that has the same schema as the original "adult" set, but it is from the same distribution, so we expect all weights to be approximately $1/N_E$, which is uninteresting (but does not violate privacy). To simulate the

general situation where the public dataset is not from the same distribution as the private one, we split records from the pre-processed dataset by the feature sex. We place 90 % of males and 10 % of females in D, and the rest in E. The cardinalities of D and E are about 21,000 and 12,000 respectively. We then remove the feature sex, because in typical applications there will not be any single feature that makes learning the weights $w(x)$ easy. Splitting based on sex simulates, in an extreme way, situations where, for example, the public dataset consists of information from volunteers, while the private dataset consists of information from non-volunteers, who are quite different statistically from volunteers.

The experiments use $\lambda = 0.1$ and $\epsilon = 0.1$ as default values. This value for the privacy budget ϵ is commonly used in research. We choose $\lambda = 0.1$ as a baseline because it is a good choice for training a conventional logistic regression classifier on the preprocessed "adult" dataset. We vary λ and ϵ to see how they affect the accuracy of estimates obtained using the importance weighting method. For each pair of λ and ϵ, we use bootstrap sampling to create randomness in the private dataset D: each time N_D samples are drawn from D with replacement to form a new private dataset D', and this D' is used with the importance weighting method to get an estimate. The results of 100 estimates from 100 experiments are shown. Note that records in D' are regarded as independent. Even if bootstrap sampling makes two records be copies of the same record in D, only one of the copies may change in the definition of differential privacy.

5.1 Alternative mechanisms

Some non-data-publishing mechanisms can answer individual queries more accurately than the importance weighting method. In particular, the sensitivity of a count query is 1, so the Laplace mechanism can answer these queries, including the query $b(x) = I(\text{income} > \$50K)$ used later, directly with high accuracy on the "adult" dataset. For example, with $\epsilon = 0.1$ and $|D| = 21,000$ as above, the answer is unbiased, with standard deviation approximately $10\sqrt{2}/21,000 \simeq 0.0007$. However, non-data-publishing mechanisms must consume some of the available privacy budget for each query, leaving a smaller privacy budget for future queries. The point of this paper, in contrast, is to provide a once-and-for-all method of publishing data, after which an unlimited number and range of queries can be answered without consuming any further privacy budget. Therefore, we compare experimentally only to other data publishing mechanisms.

Section 2 describes the alternative data-publishing mechanisms of which we are aware. On the one hand, for the methods that require a predetermined query set Q. it is hard to find a reasonable choice for this set Q. It is too restrictive to make Q simply equal the specific test queries used below. On the other hand, most existing query-independent data-publishing mechanisms either eliminate many features or feature values, or place restrictions on the dataset, so they are not useful for this dataset.

The Laplace perturbation data-publishing mechanism adds noise to each feature in each sample in the dataset. This method is query-independent and does not eliminate any features. However, unfortunately, so much noise must be added that answers to queries are not useful. With 63 binary features obtained from 12 original categorical features, the L_1 sensitivity of the private dataset (viewed as a query) is at least 24. Given the privacy budget 0.1, noise from Lap(240) must be added to each binary feature value in D. Suppose that we want to estimate the average value of a feature, a number between 0 and 1. The average of these noisy values is an unbiased estimate, but the standard deviation of the noisy average can be as large as $\sqrt{2 \cdot 240^2/21,000} \simeq 2.34$. This standard deviation is too large for the Laplace publishing mechanism to be practical.

In an alternative use of the Laplace publishing mechanism, the noisy features are trimmed to $[0, 1]$. In this case the variance can be small, about $0.25/\sqrt{21,000} \simeq 0.002$. However, trimming causes large bias. When the noise-free true answer is 1, the expectation of the answer based on trimmed noisy values is 0.501. Similarly, when the true answer is 0, the expectation is 0.499. In both cases, the bias is 0.499.

In summary, for the first experiment we are not aware of an alternative method with which comparison would be appropriate. In the second experiment, we do compare the importance weighting method to the non-data-publishing method of Chaudhuri et al. (2011).

5.2 Queries and measures of success

The queries used in the two experiments are as follows. The first is a typical count query, namely the function $b(x) = I(\text{income} > \$50K)$. The second is a sequence of complex queries: all functions of the training data computed by the LIBLINEAR software while training a linear SVM. We investigate this because outsiders will often want to use the dataset E and the published weights to learn a model that applies to the private dataset D, or to learn relationships between features within it. Linear SVMs are one of the most popular modeling methods. The outcome of SVM training depends on the gradients of the loss function, so training an SVM is equivalent to getting answers to queries concerning these. To evaluate success, we compare the SVM parameters β^D and β^E learned directly from D versus from the weighted E. As is standard, the linear SVM is trained to predict income $> \$50K$ from the other features.

For the count query, we plot the true empirical average on D and the estimates obtained using the importance weighting mechanism. To show the distribution of estimates, we plot the 95 % confidence interval and quantiles at $1/4$, $1/2$ and $3/4$. For the SVM, we plot the distribution of the Euclidean distance between the weight vectors β^D and β^E. We do not compare the prediction errors because the weight vectors are more informative, and because the relationship between prediction error and the gradient queries is not as close as the relationship between the parameters and the queries. Since the parameter corresponding to an unpredictive feature is close to 0, absolute Euclidean distance is more informative than relative distance $\sum_i (\beta_i^D - \beta_i^E)/\beta_i^D$ where i ranges over the components of β^D and β^E.

We compare SVM learning results with results from the method of Chaudhuri et al. (2011), which outputs differentially private SVM parameters directly. Note that this comparison method is more specialized than the importance weighting method, which is general for all queries and all learning algorithms, linear and nonlinear.

6 Results of experiments

The unweighted average of $b(x) = I(\text{income} > \$50K)$ on E is around 0.15, which is far from the true value of $E_D[b(x)]$, which is approximately 0.3. However, in most of the experiments below, the estimates from the importance weighting method are close to 0.3. This shows that the method is successful on a typical query, for a real-world dataset of limited size and a realistic privacy budget.

Figure 1 shows that the variance decreases as λ gets larger, while the bias increases and the estimate tends towards $E_E[b(x)] = 0.15$. This happens because when regularization becomes stronger, the β^* from the logistic regression is closer to the zero vector, and all the weights are closer to 1. Then $E_E[b(x)w(x)]$ tends to $E_E[b(x)]$. Note that privacy is guaranteed by setting $\epsilon = 0.1$ regardless of λ.

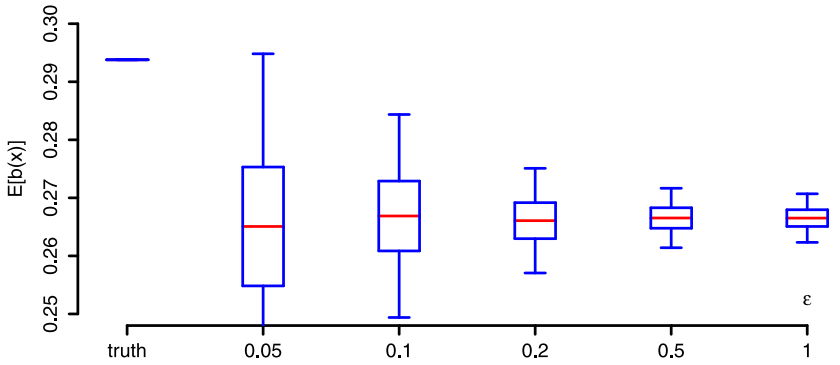

Fig. 2 Performance of the importance weighting method as the privacy budget ϵ varies. The query is $b(x) = I(\text{income} > \$50K)$ and $\lambda = 0.1$. The *vertical axis* shows the estimated answer to the query (note the magnified scale compared to Fig. 1), while the *horizontal axis* shows values of ϵ. "Truth" indicates the correct answer. *Box plots* show variation over 100 randomly created private datasets D

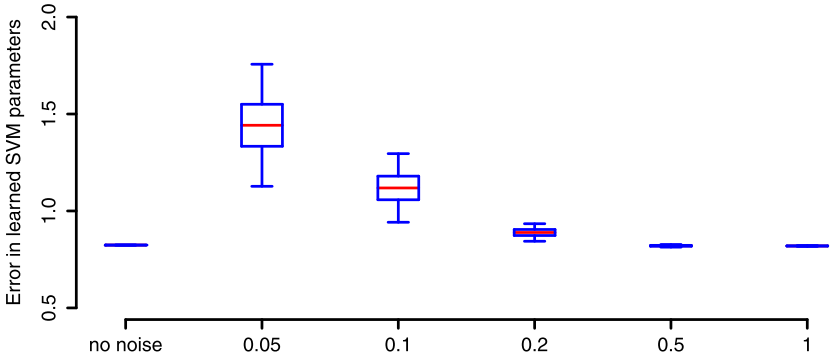

Fig. 3 Euclidean distance (*vertical axis*) between linear SVM parameter vectors learned from D and from E, with $\lambda = 0.1$ and regularization strength $\Lambda = 0.1$ for the SVM. The *horizontal axis* shows various values of the privacy budget ϵ. The "No Noise" result is the distance with bootstrapping but without privacy-protecting noise added. *Box plots* show variation over 100 random versions of D

Figure 2 shows that changing ϵ has a large effect on the variance of the estimate, but little effect on its mean. This means that a smaller privacy budget causes greater noise in estimates, but does not make these estimates more biased. This behavior is the best that we can hope for from any method that preserves privacy.

Figure 3 shows the Euclidean distance between the parameters of the SVM model trained on D and the parameters of the model trained on E using weights. The norm of the parameters learned from D is 7.17, so distances around 1 indicate successful SVM training. As expected, the variance and bias both become smaller when the privacy requirement is less strict, that is when ϵ is larger. Regardless of how relaxed the privacy requirement is, distances remain above 0.8. Increasing ϵ cannot reduce the distance to zero mainly because $p_D(x)/p_E(x) \propto \exp(\beta^T x)$ is not satisfied exactly. With a better-specified model for the importance weights, the proposed method would perform even better.

We also compare our result with that of the differentially private SVM derived by Chaudhuri et al. (2011). We use the first algorithm of that paper, which adds noise to the true SVM coefficients. Fortunately, the scale of noise in the algorithm can be computed explicitly.

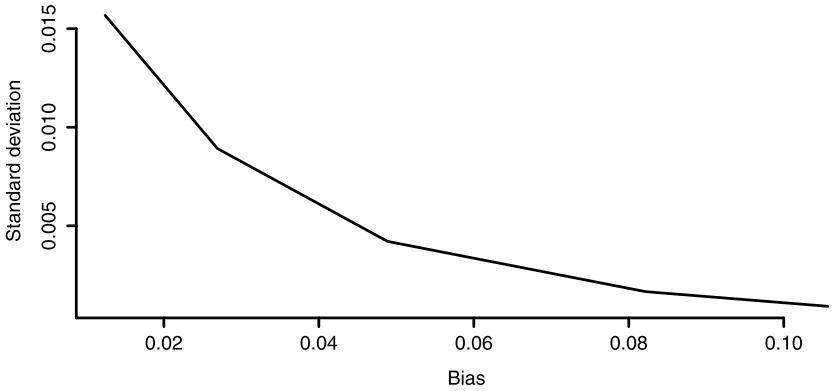

Fig. 4 Trade-off between bias (*horizontal axis*) and standard deviation (*vertical axis*) when the strength of regularization λ varies, for the query $b(x) = I(\text{income} > \$50K)$ and with privacy budget $\epsilon = 0.1$

The sensitivity stated in the paper is $2/n\Lambda$, under the assumption that $\|x\|_2 \leq 1$, where n is the cardinality of the training set and Λ is the regularization strength of the SVM. Because $\|x\|_2 \leq \sqrt{d}$ for the "adult" dataset, the sensitivity for it is $2\sqrt{d}/N_D\Lambda\epsilon$ and the density function of noise b in the algorithm is $v(b) \propto \exp(-\frac{N_D\Lambda\epsilon}{2\sqrt{d}}\|b\|_2)$.

The distribution of noise is symmetric around zero and $b \in \mathbb{R}^{+d} = [0, +\infty]^d$, so

$$
\begin{aligned}
E\left[\|b\|_2^2\right] &= \int_{\mathbb{R}^{+d}} v(b)\|b\|_2^2 db \\
&= \frac{\int_{\mathbb{R}^{+d}} \exp(-\frac{N_D\Lambda\epsilon\|b\|_2}{2\sqrt{d}})\|b\|_2^2 db}{\int_{\mathbb{R}^{+d}} \exp(-\frac{N_D\Lambda\epsilon\|b\|_2}{2\sqrt{d}})db} \\
&= \frac{4d}{N_D^2\Lambda^2\epsilon^2}\frac{\int_{\mathbb{R}^{+d}} \exp(-\|s\|_2)\|s\|_2^2 ds}{\int_{\mathbb{R}^{+d}} \exp(-\|s\|_2)ds} \\
&= \frac{4d}{N_D^2\Lambda^2\epsilon^2}\frac{\int_{\mathbb{R}^{+}} t^2\exp(-t)d(t^d)}{\int_{\mathbb{R}^{+}} \exp(-t)d(t^d)} \\
&= \frac{4d}{N_D^2\Lambda^2\epsilon^2}\frac{\int_{\mathbb{R}^{+}} t^{d+1}\exp(-t)dt}{\int_{\mathbb{R}^{+}} t^{d-1}\exp(-t)dt} \\
&= \frac{4d}{N_D^2\Lambda^2\epsilon^2}\frac{\Gamma(d+2)}{\Gamma(d)} = \frac{4d^2(d+1)}{N_D^2\Lambda^2\epsilon^2}.
\end{aligned}
$$

Thus the expected L_2 norm of the noise is $\frac{2d\sqrt{d+1}}{N_D\Lambda\epsilon} \simeq 4.8$ given dimensionality $d = 63$. The importance weighting method has smaller error, less than 1.5.

Another experimental question is the effect of λ on the accuracy of estimates. We know theoretically that larger λ brings smaller standard deviation and larger bias, and vice versa. Figure 4 shows this trade-off between bias and standard deviation.

Last but not least, we would like to know how the importance weighting mechanism performs in extreme cases. One such case occurs when the public dataset and the private dataset are the same. Another extreme case is when the public dataset is uniformly drawn from the sample space. Results for these cases are shown in Figs. 5 and 6. As before, $\epsilon = 0.1$ and $\lambda = 0.1$, and the same two queries from before are used, so previous experimental results are shown. Not surprisingly, for both queries the best performance is when E is identical

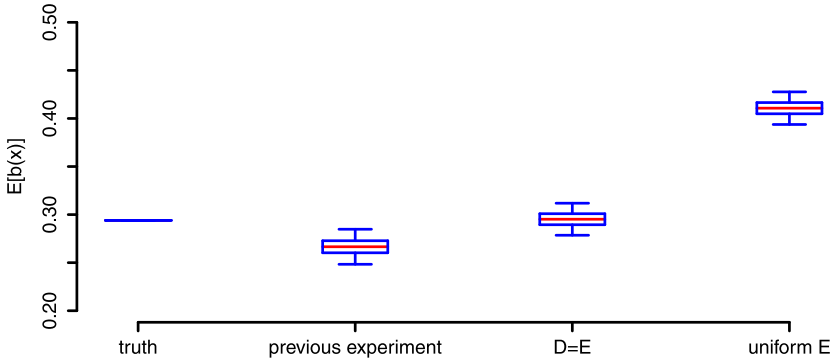

Fig. 5 Performance of the importance weighting mechanism in extreme cases, for the query $b(x) = I(\text{income} > \$50K)$. The closer an output is to the truth, the better

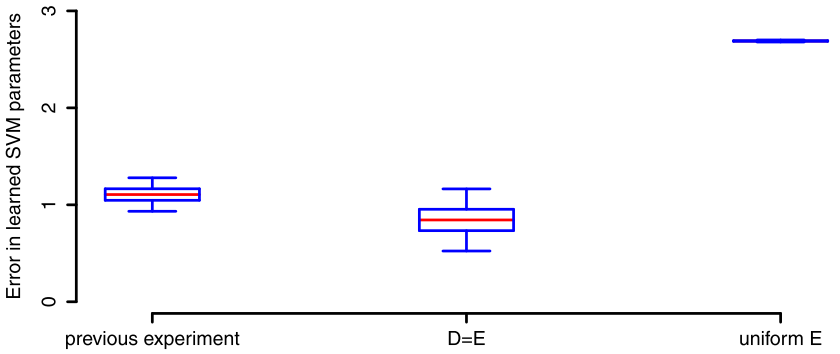

Fig. 6 Performance of the importance weighting mechanism in extreme cases, for training an SVM. The smaller the distance is, the better. The norm of the true SVM parameter vector is about 7, so the mechanism provides useful information in all three cases

to D. Performance with the skewed E used previously is not much worse. Performance with the uniformly drawn E is worst, but in particular the trained SVM classifier (Fig. 6) is still useful.

7 Discussion

The experimental results in Sect. 6 show that the differential privacy mechanism proposed in this paper is useful in practice, both for answering individual queries and for training supervised learning models. The theoretical results in Sect. 4 show that if the private dataset is large, then privacy can be preserved while still allowing queries to be answered with variance asymptotically similar to the variance that stems from the private dataset itself being random.

Naturally, variations on the importance weighting approach are possible. One idea is to draw a new dataset from E using the computed weights, instead of publishing the weights. However, this will increase the variance of estimates without changing their expectation. Thus publishing the weights explicitly is preferable. Algorithm 1 ensures that the weights are limited in magnitude and have enough noise to protect privacy.

The regularized logistic regression approach of Algorithm 1 is not the only possible way to obtain privacy-preserving importance weights. As mentioned earlier, the approach to privacy-preserving logistic regression of Chaudhuri et al. (2011) could be applied also. Other methods of estimating well-calibrated conditional probabilities (Zadrozny and Elkan 2001; Kanamori et al. 2009; Menon et al. 2012) can be used also, if modified to guarantee differential privacy.

The theory of importance weighting says that the closer the two distributions p_D and p_E are, the better the estimates based on E are. Thus, not surprisingly, the more similar the distribution of E is to that of D, the better. However, the experiments above use sets D and E with quite different distributions, and results are still good. Specifically, the set D is 90 % male, while the set E is 90 % female.

An obvious issue is where the public dataset E can come from. This question has no universal answer, but it does have several possible answers. First, E may be synthetic. The experiments section shows that even if E is uniformly drawn from the sample space, the importance mechanism can still provide useful output. Second, E may be the result of a previous breach of privacy. Any such event is regrettable, but if it does happen, using E as suggested above does not worsen the breach. Third, E may be a subset of examples from the original dataset for which privacy is not a concern. In a medical scenario, E may contain the records of volunteers who have agreed to let their data be used for scientific benefit. In the U.S., laws on the privacy of health information are less restrictive when a patient is deceased, and such records have already been released for research by some hospitals.

Another issue is how to define E if more than one public dataset is available. If we know which public dataset was sampled from a distribution most similar to that of the private dataset D, then is natural to select that dataset as E. Otherwise, in particular if all the public datasets follow the same distribution or if their distributions are unknown, then it is natural to take their union as E. However, if the public datasets follow varying distributions, then logistic regression is likely to be mis-specified for representing the contrast between D and the union of the public datasets, so it can be preferable to select just one of these datasets, for example the one with highest cardinality.

The schemas of D and E may be different. In this case, only the features that appear in both datasets can be used. However, if prior knowledge is available, disparate features can be used after pre-processing. For example, D may include patients' diseases, while E records patients' medications. If a probabilistic model relating diseases and medications is known, and this model is independent of the datasets D and E, then the two features can still contribute to the ratio of probability densities.

The usefulness of the method proposed in this paper is not restricted to medical domains. For example, consider a social network such as Facebook or Linkedin, and an advertiser such as Toyota. Let the profiles of all users be the dataset D. For privacy reasons, the network cannot give the advertiser direct access to D. However, suppose that some users have opted-in to allowing the advertiser access to their profiles. The profiles of these users can be the dataset E. The social network can compute privacy-protecting weights that make the dataset E reflect the entire population D, and let the advertiser use these weights. Note that both in medical and other domains, an advantage of the importance weighting method is that all analysis is performed on genuine data, that is on the records of E. In contrast, other data-publishing methods require analyses to be done on synthetic or perturbed data.

Acknowledgements Zhanglong Ji was funded in part by NIH grants UH2HL108785, U54HL108460, and UL1TR0001000. Charles Elkan was funded in part by NIH grant GM077402-05A1. The authors are grateful to the anonymous reviewers and to Kamalika Chaudhuri for comments that helped to improve the paper notably.

Appendix A: Proof of differential privacy

With a strongly convex loss function (definition follows), such as the sum of squares of β in Step 1 of Algorithm 1, and if each sample x in D has d components that are in the interval $[0, 1]$ then Algorithm 1 achieves differential privacy. In the following, $\|.\|$ always means L_2 norm.

Definition The function f is λ-strongly convex if and only if for every $x_1 < x_2$ and all $0 \le \alpha \le 1$

$$f\big(\alpha x_1 + (1-\alpha)x_2\big) \le \alpha f(x_1) + (1-\alpha) f(x_2) - \frac{\lambda}{2}\alpha(1-\alpha)(x_1 - x_2)^2.$$

Lemma 1 *If $G(x)$ and $G(x) + g(x)$ are λ-strongly convex, continuous, and differentiable at all points, and the norm of the first derivative of $g(x)$ is at most c, then the points that minimize $G(x)$ and $G(x) + g(x)$ differ by at most c/λ.*

Proof This is Lemma 7 of Chaudhuri et al. (2011). □

Lemma 2 *Let the dimension of each training example be d, let each example component be in $[0, 1]$, and let the logistic regression parameters based on D_1 and D_2 be β_1^* and β_2^*. Then $\|\beta_1^* - \beta_2^*\|$ is bounded by $\sqrt{d}/N_D\lambda$ where $N_D = \max\{\#D_1, \#D_2\}$.*

Proof For deletion or addition, suppose $D_2 = D_1 \backslash \{x_0\}$ and $N_D = \#D_1$. Then the regularized loss functions for training on D_1 and D_2 are

$$G_1(\beta) = \frac{1}{N_E}\sum_{x\in E}\log\big(1+\exp\big(\beta^T x\big)\big) + \frac{1}{N_D}\sum_{x\in D_1}\log\big(1+\exp\big(-\beta^T x\big)\big) + \frac{\lambda}{2}\|\beta\|^2$$

$$G_2(\beta) = \frac{1}{N_E}\sum_{x\in E}\log\big(1+\exp\big(\beta^T x\big)\big) + \frac{1}{N_D-1}\sum_{x\in D_2}\log\big(1+\exp\big(-\beta^T x\big)\big) + \frac{\lambda}{2}\|\beta\|^2.$$

Define $g_1(\beta)$ and $g_2(\beta)$ as

$$g_1(\beta) = \frac{1}{N_D(N_D-1)}\sum_{x\in D_2}\log\frac{1}{1+\exp(-\beta^T x)}$$

$$g_2(\beta) = \frac{1}{N_D}\log\frac{1}{1+\exp(-\beta^T x_0)}.$$

The difference between G_1 and G_2 is

$$g(\beta) = G_1(\beta) - G_2(\beta) = g_2(\beta) - g_1(\beta).$$

Because the unregularized loss function in logistic regression is convex, $G_1(\beta)$ and $G_2(\beta)$ are both λ-strongly convex. In addition, because each partial derivative of the loss function is in $(0, 1)$, all components of $g_1'(\beta)$ and $g_2'(\beta)$ are in $[0, 1/N_D]$, and so are the absolute values of components of $g'(\beta) = g_1'(\beta) - g_2'(\beta)$. Therefore $\|g'(\beta)\| \le \sqrt{d}/N_D$, as there are at most d components. Then according to Lemma 1, $\|\beta_1^* - \beta_2^*\|$ is bounded by $\sqrt{d}/N_D\lambda$.

For replacement, suppose $D_2 = D_1 \backslash \{x_1\} \cup \{x_2\}$ and $\#D_1 = \#D_2 = N_D$. Now $G_1(\beta)$ is the same as above but

$$G_2(\beta) = \frac{1}{N_E}\sum_{x\in E}\log\big(1+\exp\big(\beta^T x\big)\big) + \frac{1}{N_D}\sum_{x\in D_2}\log\big(1+\exp\big(-\beta^T x\big)\big) + \frac{\lambda}{2}\|\beta\|^2$$

so

$$g(\beta) = G_1(\beta) - G_2(\beta) = \frac{1}{N_D}\log\big(1+\exp\big(-\beta^T x_1\big)\big) - \frac{1}{N_D}\log\big(1+\exp\big(-\beta^T x_2\big)\big).$$

So again $\|g'(\beta)\| \le \sqrt{d}/N_D$. Thus $\|\beta_1^* - \beta_2^*\|$ is bounded by $\sqrt{d}/N_D\lambda$. Therefore, $\|\beta_1^* - \beta_2^*\| \le \sqrt{d}/N_D\lambda$ always holds. End of proof of Lemma 2. □

Lemma 3 *The Laplacian noise mechanism yielding β in Step 2 of Algorithm 1 is ϵ-differentially private.*

Proof From Lemma 2 and Proposition 1 of Dwork et al. (2006), this mechanism is ϵ-differentially private. □

Theorem 1 *The mechanism $\mathscr{K}$ specified in Algorithm 1 is ϵ-differentially private.*

Proof Lemma 3 says that the mechanism $\mathscr{K}_2$ in Step 2 is ϵ-differentially private. That is, for all $S_\beta \subset \text{Range}(\mathscr{K}_2)$ and neighboring datasets D_1 and D_2

$$P\big(\mathscr{K}_2(D_1) \in S_\beta\big) \le e^\epsilon P\big(\mathscr{K}_2(D_2) \in S_\beta\big).$$

Furthermore, for all $S_w \subset \text{Range}(\mathscr{K})$, there is a $S_\beta = \{\beta | w(x) \propto e^{\beta^T x} \in S_w\} \subset \text{Range}(\mathscr{K}_2)$ such that

$$P\big(w_1(x) \in S_w\big) = P\big(\mathscr{K}(D_1) \in S_w\big) = P\big(\mathscr{K}_2(D_1) \in S_\beta\big)$$
$$P\big(w_2(x) \in S_w\big) = P\big(\mathscr{K}(D_2) \in S_w\big) = P\big(\mathscr{K}_2(D_2) \in S_\beta\big).$$

To summarize,

$$\begin{aligned} P\big(w_1(x) \in S_w\big) &= P\big(\mathscr{K}_2(D_1) \in S_\beta\big) \\ &\le e^\epsilon P\big(\mathscr{K}_2(D_2) \in S_\beta\big) = e^\epsilon P\big(w_2(x) \in S_w\big). \end{aligned}$$

So $\mathscr{K}$ is ϵ-differentially private. End of proof of Theorem 1. □

Appendix B: Variance of estimates

In the following proofs, for square matrices A and B the expression $A \le B$ means $a^T Aa \le a^T Ba$ for all vectors a. The vector x has length d and each of its components is in the range $[0, 1]$.

Lemma 4 *For any vector β that has the same length as x*

$$\text{Var}_D\left[\frac{x}{1+\exp(\beta^T x)}\right] \le E_D\big[xx^T\big] \le dI.$$

Proof For the first inequality, since $\text{Var}[y] = E[yy^T] - E[y]E[y^T]$, it is always true that $\text{Var}[y] \le E[yy^T]$. Therefore we just need to prove that

$$E_D\left[\frac{xx^T}{(1+\exp(\beta^T x))^2}\right] \le E_D\big[xx^T\big]$$

As $\exp(\beta^T x)$ is always larger than 0, $\frac{xx^T}{(1+\exp(\beta^T x))^2} \le xx^T$ always holds, thus this is true.

For the second inequality, since for all vectors a,

$$\begin{aligned} a^T E_D\big[xx^T\big]a &= E_D\big[a^T xx^T a\big] = E_D\big[\|a^T x\|^2\big] \\ &\le E\big[\|a\|^2\|x\|^2\big] = \|a\|^2 E\big[\|x\|^2\big] \\ &\le d\|a\|^2 = a^T(dI)a \end{aligned}$$

it follows that $E_D[xx^T] \le dI$. End of proof of Lemma 4. □

For the next two lemmas, let

$$g(\beta) = \frac{1}{N_E}\sum_{x\in E}\log\big(1+\exp\big(\beta^T x\big)\big) - \frac{\lambda}{2}\|\beta\|^2$$

and let the vector β_0 optimize the loss function of logistic regression on fixed E and the true distribution of D:

$$\beta_0 = \arg\max_{\beta} g(\beta) + E_D\big[\log\big(1+\exp\big(-\beta^T x\big)\big)\big].$$

Lemma 5 *Let E be fixed and let D be random. The variance of the output parameters β^* of the regularized logistic regression is asymptotically*

$$\begin{aligned} &\frac{1}{N_D}\left(g''(\beta_0) + E_D\left[\frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}\right]\right)^{-1} \\ &\quad\times \mathrm{Var}_D\left[\frac{x}{1+\exp(\beta_0^T x)}\right]\left(g''(\beta_0) + E_D\left[\frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}\right]\right)^{-1} \end{aligned}$$

where g'' is the second derivative of g.

Proof Note that all three factors in the variance of β^* are matrices, and that the first and third factors are the same. Since only the set D is random, $g(\beta)$ is a deterministic function of β. The solution β^* is

$$\beta^* = \arg\max_{\beta} g(\beta) + \frac{1}{N_D}\sum_{x\in D}\log\big(1+\exp\big(-\beta^T x\big)\big).$$

As D is drawn from an underlying distribution, β^* is a random variable.

When N_D is large, β^* is close to β_0 with high probability. Furthermore, all the functions here are infinitely differentiable. Thus we can use a Taylor expansion to express the target function using its first and second derivatives at β_0:

$$\begin{aligned} \beta^* = \arg\max_{\beta}\ & g(\beta_0) + (\beta-\beta_0)^T g'(\beta_0) + \frac{1}{2}(\beta-\beta_0)^T g''(\beta_0)(\beta-\beta_0) \\ &+ \frac{1}{N_D}\sum_{x\in D}\Bigg[\log\big(1+\exp\big(-\beta_0^T x\big)\big) - (\beta-\beta_0)^T\frac{x}{1+\exp(\beta_0^T x)} \\ &+ (\beta-\beta_0)^T\frac{\exp(\beta_0^T x)xx^T}{2(1+\exp(\beta_0^T x))^2}(\beta-\beta_0)\Bigg] + o\big((\beta-\beta_0)^T(\beta-\beta_0)\big). \end{aligned}$$

The maximization is an unconstrained optimization problem, so the first derivative of this expression is zero at the maximum point:

$$0 = g'(\beta_0) + g''(\beta_0)\big(\beta^* - \beta_0\big) + \frac{1}{N_D}\sum_{x\in D}\left[-\frac{x}{1+\exp(\beta_0^T x)} + \frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}\big(\beta^* - \beta_0\big)\right] + o\big(\beta^* - \beta_0\big).$$

Omitting the asymptotically negligible term yields

$$\beta^* - \beta_0 = -\left[g''(\beta_0) + \frac{1}{N_D}\sum_{x\in D}\frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}\right]^{-1} \times \left[g'(\beta_0) - \frac{1}{N_D}\sum_{x\in D}\frac{x}{1+\exp(\beta_0^T x)}\right].$$

The law of large numbers ensures that the expression inside the matrix inverse converges to $g''(\beta_0) + E_D[\exp(\beta_0^T x)xx^T/(1+\exp(\beta_0^T x))^2]$ as N_D increases.

Also, because β_0 minimizes $g(\beta) + E_D \log(1+\exp(-\beta^T x))$, and this minimization is unconstrained, $0 = g'(\beta_0) - E_D\frac{x}{1+\exp(\beta_0^T x)}$. Therefore according to the central limit theorem, the second factor

$$g'(\beta_0) - \frac{1}{N_D}\sum_{x\in D}\frac{x}{1+\exp(\beta_0^T x)} \sim N\left(0, \frac{1}{N_D}\operatorname{Var}\left[\frac{x}{1+\exp(\beta_0^T x)}\right]\right)$$

asymptotically. Finally, the asymptotic variance of β^* is

$$\begin{aligned}\operatorname{Var}\big[\beta^*\big] &= \operatorname{Var}\big[\beta^* - \beta_0\big] \\ &= \frac{1}{N_D}\left(g''(\beta_0) + E_D\left[\frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}\right]\right)^{-1} \\ &\quad \times \operatorname{Var}_D\left[\frac{x}{1+\exp(\beta_0^T x)}\right]\left(g''(\beta_0) + E_D\left[\frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}\right]\right)^{-1}.\end{aligned}$$

End of proof of Lemma 5. □

The previous lemma gives an exact asymptotic expression for $\operatorname{Var}[\beta^*]$ when the cardinality of D tends to infinity. However, β_0 in the expression is unknown. The following lemma gives an upper bound for the variance that depends only on the underlying distribution of D and on λ.

Lemma 6 *Let E be fixed and let D be random. The variance of the output parameters β^* of the regularized logistic regression is asymptotically less than $\frac{dI}{N_D\lambda^2}$.*

Proof Because $g(\beta)$ is the sum of a convex function and $\frac{\lambda}{2}\|\beta\|^2$, its second derivative is larger than λ. Also, $E_D[\frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}] \geq 0$. Therefore

$$\begin{aligned}\operatorname{Var}\big[\beta^*\big] &= \frac{1}{N_D}\left(g''(\beta_0) + E_D\left[\frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}\right]\right)^{-1} \\ &\quad \times \operatorname{Var}_D\left[\frac{x}{1+\exp(\beta_0^T x)}\right]\left(g''(\beta_0) + E_D\left[\frac{\exp(\beta_0^T x)xx^T}{(1+\exp(\beta_0^T x))^2}\right]\right)^{-1} \\ &< \frac{1}{N_D}\lambda^{-1}\operatorname{Var}_D\left[\frac{x}{1+\exp(\beta_0^T x)}\right]\lambda^{-1}\end{aligned}$$

$$= \frac{1}{N_D \lambda^2} \mathrm{Var}_D \left[\frac{x}{1 + \exp(\beta_0^T x)} \right] \leq \frac{dI}{N_D \lambda^2}$$

using Lemma 4 for the last inequality. End of proof of Lemma 6. □

The following lemma takes into account not just randomness from D, but also randomness from the noise added to protect privacy in Step 2 of Algorithm 1. Here, I is the identity matrix.

Lemma 7 *The total variance of β is asymptotically less than*

$$\frac{dI}{N_D \lambda^2} + \frac{d}{d+1} \frac{1}{(N_D \lambda \epsilon)^2} I.$$

Proof The noise $\delta = \{\delta_1, \ldots, \delta_d\}$ added to β^* is independent of β^*, so the variance of β is the variance of β^* plus the variance of the noise:

$$\mathrm{Var}[\beta] = \mathrm{Var}\left[\beta^*\right] + \mathrm{Var}[\delta].$$

The probability density of δ is $p(\delta) \propto \exp(-\delta/\gamma)$ where $\gamma = S/\epsilon = \sqrt{d}/N_D \lambda \epsilon$.

Because of independence and symmetry among the elements of δ, its covariance matrix A is cI for some scalar

$$\begin{aligned}
c &= \frac{1}{d} \sum_{i=1}^{d} A_{ii} = \frac{1}{d} \sum_{i=1}^{d} \mathrm{Var}[\delta_i] \\
&= \frac{1}{d} \sum_{i=1}^{d} E\left[\delta_i^2\right] = \frac{1}{d} E\left[\delta^T \delta\right] \\
&= \frac{1}{d} \frac{\int_0^{+\infty} r^2 \exp(-r/\gamma) r^{d-1} dr}{\int_0^{+\infty} \exp(-r/\gamma) r^{d-1} dr} \\
&= \frac{\gamma^2}{d} \frac{\int_0^{+\infty} t^2 \exp(-t) t^{d-1} dt}{\int_0^{+\infty} \exp(-t) t^{d-1} dt} \\
&= \frac{\gamma^2}{d} \frac{\Gamma(d+2)}{\Gamma(d)} \\
&= \frac{(d+1)d}{(N_D \lambda \epsilon)^2}.
\end{aligned}$$

This result, with Lemma 6, gives the bound on the total variance of β. End of proof of Lemma 7. □

At last, we are in a position to prove the theorem about the asymptotic variance of the estimate of the expectation of a query function $b(x)$.

Theorem 2 *The total variance of the estimate $\frac{1}{N_E} \sum_{x \in E} b(x) w(x)$ is asymptotically*

$$\begin{aligned}
\mathrm{Var}\left[\frac{1}{N_E} \sum_{x \in E} b(x) w(x) \right] &= \alpha^T \mathrm{Var}[\beta] \alpha \\
&< \alpha^T \left(dI/N_D \lambda^2 + d(d+1) I/(N_D \lambda \epsilon)^2 \right) \alpha
\end{aligned}$$

where

$$\alpha = \frac{\sum_{x_i,x_j\in E} e^{\beta_0^T(x_i+x_j)}(b(x_i)-b(x_j))(x_i-x_j)}{\sum_{x_i,x_j\in E} e^{\beta_0^T(x_i+x_j)}}.$$

Proof Using the definition of the weights $w(x)$, the variance is

$$\mathrm{Var}\left[\frac{1}{N_E}\sum_{x\in E} b(x)w(x)\right] = \mathrm{Var}\left[\sum_{x\in E} b(x)e^{\beta^T x}\Big/\sum_{x\in E} e^{\beta^T x}\right] = f(\beta).$$

Since E is fixed and $b(x)$ is given, the variance arises only from β. As β asymptotically converges to β_0, $f(\beta)$ satisfies the following equations asymptotically:

$$f(\beta) = f(\beta_0) + f'(\beta)(\beta-\beta_0)$$
$$\mathrm{Var}\big[f(\beta)\big] = \big(f'(\beta)\big)^T \mathrm{Var}[\beta] f'(\beta).$$

The derivative of $\sum_{x\in E} b(x)e^{\beta^T x}/\sum_{x\in E} e^{\beta^T x}$ is

$$\alpha = \frac{\sum_{x_i,x_j\in E} e^{\beta_0^T(x_i+x_j)}(b(x_i)-b(x_j))(x_i-x_j)}{\sum_{x_i,x_j\in E} e^{\beta_0^T(x_i+x_j)}}.$$

Hence the variance of the estimate is

$$\alpha^T \mathrm{Var}[\beta]\alpha < \alpha^T\big(dI/N_D\lambda^2 + d(d+1)I/(N_D\lambda\epsilon)^2\big)\alpha.$$

End of proof of Theorem 2. □

Theorem 3 *The bias of the estimate is asymptotically*

$$\sum_{x\in E} b(x)\frac{\exp(\beta_0^T x)}{\sum_{y\in E}\exp(\beta_0^T y)} - E_D\big[b(x)\big]$$

where β_0 minimizes the loss function of regularized logistic regression on E and p_D, as in Theorem 2.

Proof When the number of samples in D is large, the logistic regression parameter vector obtained in the first step of Algorithm 1 converges to β_0, and the noise added in the second step converges to 0. Therefore the vector β used to compute the weights also converges to β_0. Since the weights and the estimate are both continuous with respect to β, the estimate converges to

$$\sum_{x\in E} b(x)\frac{\exp(\beta_0^T x)}{\sum_{y\in E}\exp(\beta_0^T y)}.$$

The bias is the difference between the convergence point and the true expectation. End of proof of Theorem 3. □

References

Blum, A., Ligett, K., & Roth, A. (2008). A learning theory approach to non-interactive database privacy. In C. Dwork (Ed.), *STOC* (pp. 609–618). New York: ACM.

Chaudhuri, K., Monteleoni, C., & Sarwate, A. D. (2011). Differentially private empirical risk minimization. *Journal of Machine Learning Research*, *12*, 1069–1109.

Ding, B., Winslett, M., Han, J., & Li, Z. (2011). Differentially private data cubes: optimizing noise sources and consistency. In *SIGMOD conference* (pp. 217–228).

Dwork, C. (2006). Differential privacy. In M. Bugliesi, B. Preneel, V. Sassone, & I. Wegener (Eds.), *Lecture notes in computer science: Vol. 4052. ICALP (2)* (pp. 1–12). Berlin: Springer.

Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In S. Halevi & T. Rabin (Eds.), *Lecture notes in computer science* (Vol. 3876, pp. 265–284). Berlin: Springer.

Elkan, C. (2010). Preserving privacy in data mining via importance weighting. *Lecture notes in computer science:* In *Proceedings of the ECML/PKDD workshop on privacy and security issues in data mining and machine learning (PSDML)*. Berlin: Springer.

Elkan, C., & Noto, K. (2008). Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining*, Las Vegas, Nevada (pp. 213–220).

Frank, A., & Asuncion, A. (2010). UCI machine learning repository. http://archive.ics.uci.edu/ml.

Hahsler, M., Grün, B., & Hornik, K. (2011). *arules: Mining Association Rules and Frequent Itemsets*. http://CRAN.R-project.org/, R package version 1.0-7.

Hardt, M., & Rothblum, G. N. (2010). A multiplicative weights mechanism for privacy-preserving data analysis. In *FOCS* (pp. 61–70).

Hardt, M., Ligett, K., & McSherry, F. (2012). A simple and practical algorithm for differentially private data release. In *NIPS* (pp. 2348–2356). http://books.nips.cc/papers/files/nips25/NIPS2012_1143.pdf.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57*(1), 97–109.

Hay, M., Rastogi, V., Miklau, G., & Suciu, D. (2010). Boosting the accuracy of differentially private histograms through consistency. *Proceedings of the VLDB Endowment*, *3*(1), 1021–1032.

Kanamori, T., Hido, S., & Sugiyama, M. (2009). A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, *10*, 1391–1445.

Li, C., Hay, M., Rastogi, V., Miklau, G., & McGregor, A. (2010). Optimizing linear counting queries under differential privacy. In *PODS* (pp. 123–134).

Li, Y. D., Zhang, Z., Winslett, M., & Yang, Y. (2011). Compressive mechanism: utilizing sparse representation in differential privacy. In *Proceedings of the 10th annual ACM workshop on privacy in the electronic society* (pp. 177–182). New York: ACM.

McSherry, F., & Mahajan, R. (2010). Differentially-private network trace analysis. In *SIGCOMM* (pp. 123–134).

McSherry, F., & Mironov, I. (2009). Differentially private recommender systems: building privacy into the netflix prize contenders. In *KDD* (pp. 627–636).

Menon, A., Jiang, X., Vembu, S., Elkan, C., & Ohno-Machado, L. (2012). Predicting accurate probabilities with a ranking loss. In *Proceedings of the international conference on machine learning (ICML)*.

Mohammed, N., Chen, R., Fung, B. C. M., & Yu, P. S. (2011). Differentially private data release for data mining. In C. Apte, J. Ghosh, & P. Smyth (Eds.), *KDD* (pp. 493–501). New York: ACM.

Rastogi, V., & Nath, S. (2010). Differentially private aggregation of distributed time-series with transformation and encryption. In *SIGMOD conference* (pp. 735–746).

Scott, D. W. (1992). *Multivariate density estimation: theory, practice, and visualization*. New York: Wiley-Interscience.

Smith, A. (2008, preprint). Efficient, differentially private point estimators. arXiv:0809.4794.

Smith, A., & Elkan, C. (2004). A Bayesian network framework for reject inference. In *Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining (KDD)* (pp. 286–295).

Xiao, Y., Xiong, L., & Yuan, C. (2010). Differentially private data release through multidimensional partitioning. In W. Jonker & M. Petkovic (Eds.), *Secure data management, Springer, lecture notes in computer science* (Vol. 6358, pp. 150–168).

Zadrozny, B., & Elkan, C. (2001). Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. In *Proceedings of the 18th international conference on machine learning* (pp. 609–616). San Mateo: Morgan Kaufmann.