Differentially Private Projected Histograms: Construction and Use for Prediction

Staal A. Vinterbo*

Division of Biomedical Informatics, University of California San Diego
San Diego, CA, USA
sav@ucsd.edu

Abstract. Privacy concerns are among the major barriers to efficient secondary use of information and data on humans. Differential privacy is a relatively recent measure that has received much attention in machine learning as it quantifies individual risk using a strong cryptographically motivated notion of privacy. At the core of differential privacy lies the concept of information dissemination through a randomized process. One way of adding the needed randomness to any process is to pre-randomize the input. This can yield lower quality results than other more specialized approaches, but can be an attractive alternative when *i*. there does not exist a specialized differentially private alternative, or when *ii*. multiple processes applied in parallel can use the same pre-randomized input.

A simple way to do input randomization is to compute perturbed histograms, which essentially are noisy multiset membership functions. Unfortunately, computation of perturbed histograms is only efficient when the data stems from a low-dimensional discrete space. The restriction to discrete spaces can be mitigated by discretization; Lei presented in 2011 an analysis of discretization in the context of M-estimators. Here we address the restriction regarding the dimensionality of the data. In particular we present a differentially private approximation algorithm for selecting features that preserve conditional frequency densities, and use this to project data prior to computing differentially private histograms. The resulting projected histograms can be used as machine learning input and include the necessary randomness for differential privacy. We empirically validate the use of differentially private projected histograms for learning binary and multinomial logistic regression models using four real world data sets.

1 Introduction

Concerns regarding individual privacy are among the major barriers to efficient secondary use of information and data on humans. One of the main difficulties associated with implementation of privacy preserving measures is the quantification of incurred risk to the individual. Differential privacy [5] is a relatively recent measure that has received much attention in the machine learning community

* This work was supported by NIH grants R01 LM07273 and U54 HL108460. Thanks go to the anonymous reviewers for their comments.

P. Flach et al. (Eds.): ECML PKDD 2012, Part II, LNCS 7524, pp. 19–34, 2012.

as it quantifies individual risk using a strong and cryptographically motivated notion of privacy. For an overview of contributions to the growing number of differentially private methods and theory, we defer to [4,6]. Central to differential privacy is the notion of information dissemination through a randomized process. For a given process that accesses private data and produces information, randomization can be introduced in several places: input perturbation before the data flows into the process, during the computation such as in objective function perturbation [3], and after the process has computed the results but before they are presented [5]. For a given process, analyzing and modifying it so that it yields differential privacy at a given level, and providing a quantification of the inevitable loss in result quality, can be difficult and require sophisticated analysis. These analyses sometimes impose restrictions, as in the type of regularizer in Chaudhuri et al.'s empirical risk minimization [3], and sometimes particular processes cannot be made differentially private in a useful manner, for example computing all k-way marginals [20].

As we will see, achieving a specified level of privacy by perturbing the input to an unconstrained process can yield lower quality results than applying a specialized differentially private version at a fixed privacy level. Nevertheless, this approach to achieving differential privacy can be attractive when *i.* there does not exist a specialized and tailored differentially private alternative, or when *ii.* multiple applications of differentially private processes can be replaced by one or more non-private processes that use a single perturbed instance of the input; the reason is that differential privacy risk for parallel processes composes additively in general. Furthermore, input perturbation can be seen as a noisy information release; having access to information an analysis is based on allows reproduction of results, a very important principle in science and research.

A simple way of producing randomized input from data is to compute a perturbed histogram [5], which essentially is a noisy multiset membership function for the collection of points constituting the data. In order to achieve differential privacy in a histogram by perturbation, noise is added not only to non-zero multiset memberships, but also to originally zero-valued memberships. Consequently, the support of a perturbed histogram extends to virtually all possible elements. This, and the fact that the goal of histogram perturbation includes making originally zero-valued entries indistinguishable from unit-valued entries, perturbed histograms are in practice only applicable to low dimensional and discrete spaces where data is "concentrated", i.e., when there are many duplicates.

The restriction to discrete data can be mitigated by discretization; Lei [15] analyzes discretized histograms in the context of M-estimators and shows consistency results when the discretization bandwidth is chosen carefully. In the following we address the restriction to low dimensionality in the context of learning classifiers. In particular we *a.* present a differentially private approximation algorithm for projecting data onto k features that maximally preserve conditional frequency densities, *b.* show how we can combine projection with thresholded histogram computation to produce histograms that can be used for learning probabilistic classifiers, and *c.* empirically show the viability of building binary

and multinomial logistic regression classifiers on differentially private projected histograms. We also show how to compute thresholded histograms in a way that exhibits optimal asymptotic time complexity for data from bounded size spaces.

Other Related Work. Jagadish et al. [13] study binning for histograms in a non-private setting and present optimality results for broad classes of query types. Mohammed et al. [17] and Xiao et al. [25] approach the coarsening of information granularity for differentially private histograms by clustering attribute codomains in a hierarchical manner much like in construction of classification trees. Hay et al. [12] explore tailoring of histograms for answering multiple queries for which we have consistency constraints, for example given as subset sums of query results. Similarly, Xu et al. [26] address the clustering of attribute codomains for a given sequence of counts from a data base, using among others ideas from [13]. The approach presented here is complementary to attribute codomain clustering methods in that the principal method for achieving courser information granularity is instead projection onto a subset of dimensions, consequently not requiring any binning or clustering for categorical attributes. Barak et al. [1] investigate differentially private k-way contingency tables. The projection algorithm in this work can be used as an approximation algorithm for finding a k-way contingency table that reflects a partition of the data with maximum discrimination among elements, and in this sense is related to rough set [18] "reducts". This suggests that projected histograms as computed in this work can also contribute to differentially private rough set theory.

2 Methods

Let $[n] = \{1, 2, \ldots, n\}$, and let $V_i \subseteq U$ for $i \in [d]$ be $d > 0$ finite subsets of some set U. Also let $|V_i| \geq 2$ for all i. We then define $V = V_1 \times V_2 \times \cdots \times V_d$ to be a d-dimensional space of size $m = \prod_{i \in [d]} |V_i|$. A *data set* $\mathcal{D}$ is a collection or multiset of points (records) from V and can be represented by a function $h_{\mathcal{D}} : V \to \mathbb{N}$ counting the number of occurrences of a point in $\mathcal{D}$. A *histogram* is a function $h : V \to \mathbb{R}$, and as such can be seen as a generalization of a multiset. The definition of privacy we use is Differential Privacy [7] and can be stated as follows.

Definition 1. *A randomized algorithm A is ϵ-differentially private if for any measurable set of outputs $\mathbf{S}$,*

$$P\left(A(\mathcal{D}) \in \mathbf{S}\right) \leq e^{\epsilon} P\left(A(\mathcal{D}') \in \mathbf{S}\right)$$

where $\mathcal{D}, \mathcal{D}'$ are any two databases of n records that share $n-1$ records in common. The probability is taken over the randomness in A.

The value ϵ is the quantification of individual privacy risk. An important probability distribution in the context of Differential Privacy is the Laplace distribution $\mathrm{Lap}(\mu, \lambda)$ with density

$$f_{\mathrm{Lap}}(r|\mu, \lambda) = \frac{1}{2\lambda} \exp\left(-\frac{|r - \mu|}{\lambda}\right). \tag{1}$$

The mean of this distribution is μ and the variance is $2\lambda^2$. Let $L \sim \text{Lap}(0, 2/\epsilon)$. Then

$$P(L > x) = \begin{cases} \frac{1}{2}\exp\left(-\frac{\epsilon x}{2}\right) & \text{if } x > 0 \\ 1 - \frac{1}{2}\exp\left(\frac{\epsilon x}{2}\right) & \text{otherwise.} \end{cases} \tag{2}$$

In the following we defer proofs to appendix A.

2.1 Differentially Private Histograms

We can create a perturbed version of $h_{\mathcal{D}}$ by adding a perturbation $r(\mathbf{x})$ distributed according to the Laplace distribution, and subsequently threshold by a data-value independent τ to suppress small as well as negative entries. Thresholding by τ has practical implications: the multi-set analogy breaks down with negative entries, learning predictive models from histograms with negative values becomes non-standard, and both the computational effort and space needs associated with thresholded histograms is in practice reduced. Lei [15] proposes $\tau = A \log n/\epsilon$ on the intuition that the maximal noise will be $O(\log n/\epsilon)$. We follow Lei in this definition of τ. Formally,

$$\tilde{h}_{\mathcal{D},\epsilon}(\mathbf{x}) = h_{\mathcal{D}}(\mathbf{x}) + r(\mathbf{x}) \tag{3}$$

$$\tilde{h}_{\tau,\epsilon,\mathcal{D}}(\mathbf{x}) = \begin{cases} \tilde{h}_{\mathcal{D},\epsilon}(\mathbf{x}) & \text{if } \tilde{h}_{\mathcal{D},\epsilon}(\mathbf{x}) > \tau \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

As we do not consider the size of $\mathcal{D}$ private, the subsequent proposition follows directly from results in [5].

Proposition 1. *If τ is only dependent on n and $r(\mathbf{x})$ is distributed according to $\mathsf{Lap}(0, 2/\epsilon)$, then (3) and (4) yield ϵ-differential privacy.*

Since $\tilde{h}_{\tau,\epsilon,\mathcal{D}}$ is constructed in a privacy preserving manner, the data set $\tilde{\mathcal{D}}$ obtained by $h_{\tilde{\mathcal{D}}}(\mathbf{x}) = \lceil \tilde{h}_{\tau,\epsilon,\mathcal{D}}(\mathbf{x}) - 0.5 \rceil$ is privacy preserving.

We have that the value $\tilde{h}_{\tau,\epsilon,\mathcal{D}}(\mathbf{x})$ is distributed according to $\text{Lap}(h_{\mathcal{D}}(\mathbf{x}), 2/\epsilon)$ if $r(\mathbf{x})$ is distributed according to $\text{Lap}(0, 2/\epsilon)$. Consequently, $P(\tilde{h}_{\mathcal{D},\epsilon}(\mathbf{x}) > \tau) = P(L > \tau - h_{\mathcal{D}}(\mathbf{x}))$, where $L \sim \text{Lap}(0, 2/\epsilon)$. Applying (2), the probability of a point $\mathbf{x}$ in V making it into the support S of $\tilde{h}_{\tau,\epsilon,\mathcal{D}}$ is

$$P(\mathbf{x} \in S) = P(L > \tau - h_{\mathcal{D}}(\mathbf{x})) = \begin{cases} \frac{1}{2}\left(\frac{\exp\left(\frac{\epsilon h_{\mathcal{D}}(\mathbf{x})}{2}\right)}{n^{A/2}}\right) & \text{if } \tau > h_{\mathcal{D}}(\mathbf{x}), \\ 1 - \frac{1}{2}\left(\frac{\exp\left(\frac{\epsilon h_{\mathcal{D}}(\mathbf{x})}{2}\right)}{n^{A/2}}\right)^{-1} & \text{otherwise.} \end{cases} \tag{5}$$

If we now let $Y_i \sim \text{Bernoulli}(p_i)$, where $p_i = P(\tilde{h}_{\mathcal{D},\epsilon}(\mathbf{x}_i) > \tau)$ for $\mathbf{x}_i \in V$, we get that the expected size of S is

$$E[|S|] = E[\sum_i Y_i] = \sum_{j=0}^{\infty} |h_{\mathcal{D}}^{-1}(j)| P(L > \tau - j). \tag{6}$$

Let $n' = |\textsc{Supp}(h_{\mathcal{D}})| = \sum_{j=1}^{\infty} |h_{\mathcal{D}}^{-1}(j)|$. If we assume that $\tau = A\log(n)/\epsilon > 1$, then by (5) and (6)

$$n' + \frac{m-n'}{2n^{A/2}} \geq E[|S|] \geq \frac{n'\exp(\frac{\epsilon}{2})}{2n^{A/2}} + \frac{m-n'}{2n^{A/2}} = \frac{\left(\exp(\frac{\epsilon}{2})-1\right)\, n' + m}{2n^{A/2}}. \quad (7)$$

In (7) we see that $E[|S|]$ grows linearly in m and consequently exponentially in d.

Efficient Construction and Representation. Since V of size m is a product of d finite sets V_i of sizes v_i, we can implement a bijection enc between V and $[m]$ using a multi-base positional number system that maps an element in $O(d)$ time. In our application, the assumption that d and v_i's are small enough so that m can be bounded to fit into a machine word (of typically 64 bits) is mild. Under this assumption, we now show how construct $\tilde{h}_{\tau,\epsilon,\mathcal{D}}$ in expected time that is linear in the size of $\mathcal{D}$ and the output histogram. This is asymptotically optimal since any algorithm computing a histogram from data must input the data and output the histogram. The bounded integer encoding lets us use radix sort to sort a list of integers in linear time, forming the basis of linear time set related operations. Furthermore, given two small constants c_1 and c_2 (≤ 2) and a size z set of keys Z from $[m]$, we can in $O(z)$ time compute a perfect hash function $f_z : Z \to [\lceil c_1 z \rceil]$ that require $O(c_2 z)$ bits of space and has constant evaluation time [2]. Given this result, we can construct a constant access time and $O(z)$ size "hash map" data structure for $Z \subseteq [m]$ in $O(z)$ time. Let $\mathcal{D}'$ be the encoded version of $\mathcal{D}$ created in $O(nd)$ time, and let S' be the set of n' unique elements in $\mathcal{D}'$, which we can extract from $\mathcal{D}'$ in $O(n)$ time using radix sort. Given S', and the hash map data structure we construct the restriction of $\tilde{h}_{\tau,\epsilon,\mathcal{D}'}$ to S' in $O(n)$ time. Naive computation of the remainder of $\tilde{h}_{\tau,\epsilon,\mathcal{D}'}$ would require enumerating $[m] - S'$. We can avoid this explicit enumeration by first determining how many elements from $[m] - S'$ will enter the histogram. This number is the number $\hat{q}$ of successes in $m - n'$ Bernoulli trials with equal success probability $p = P(L > \tau)$. Then we uniformly sample $\hat{q}$ points S'' from $[m] - S'$ without replacement, and assign each a value v in (τ, ∞) chosen with probability proportional to $f_{\text{Lap}}(v|0, 2/\epsilon)$, forming the second part of $\tilde{h}_{\tau,\epsilon,\mathcal{D}'}$. The efficiency of this scheme depends on the efficiency of sampling the distinct points from $[m] - S'$. Using an efficient method [24] for sequential random sampling we sample $\hat{q} + n'$ elements Q without replacement from $[m]$ in expected $O(\hat{q} + n')$ time. We then compute $Q' = Q - S'$ in $O(\max(\hat{q}, n'))$ time using radix sort, and finally sample $\hat{q}$ elements S'' from Q' without replacement in expected $O(\hat{q})$ time using the efficient method. Noting that using hash maps to store the two parts of $\tilde{h}_{\tau,\epsilon,\mathcal{D}'}$, we can merge them in time linear in the sum of their sizes which is $O(n')$ and $O(\hat{q})$, respectively. Consequently, the total time to construct $\tilde{h}_{\tau,\epsilon,\mathcal{D}'}$ is expected $O(nd) + O(\hat{q} + n')$. We can now let $\tilde{h}_{\tau,\epsilon,\mathcal{D}}(x) = \tilde{h}_{\tau,\epsilon,\mathcal{D}'}(\text{enc}(x))$.

Numeric Data. Numeric data is in general too fine-grained to be suitable for direct applications of histogram-based methods, and discretization is required. There is a trade-off between the granularity of the discretization, dictating the

preservation of properties of the original data, and suitability for use in the differentially private histogram sampling approach. Lei [15] studies this trade-off in the context of M-estimators, and shows consistency results for differentially private histograms with non-negative counts (i.e., 0-thresholded) when predictors in [0,1] are discretized with bandwidth $\mathrm{bw}(n,d) = \left(\frac{\log(n)}{n}\right)^{1/(d+1)}$. Based on this, we assume that numeric predictors are bounded. This means that they can scaled into [0,1] in a data-independent manner, and subsequently be discretized into $\lceil 1/\mathrm{bw}(n, d-1) - 0.5 \rceil$ interval bins. In order to "reverse" the discretization after histogram sampling, we replace each bin identifier with the bin interval midpoint.

2.2 Differentially Private Discernibility

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a set of n points x_i from some set $\mathcal{U}$, and let $A = \{a_1, a_2, \ldots, a_{d-1}\}$ and $f = a_d$ be d functions $a_i : \mathcal{U} \to V_i$. Also, let $=$ be an equivalence relation on V (and V_i). In this formulation, we can say that point $\mathbf{x}_i$ in $\mathcal{D}$ is simply $(a_1(x_i), \ldots, a_{d-1}(x_i), f(x_i))$.

From a classification perspective, we can think of $\mathcal{D}$ as a lookup table: given $A(x) = (a_1(x_i), \ldots, a_{d-1}(x_i))$ look up the conditional relative frequency densities of labels V_d in $\mathcal{D}$ (the distribution of labels associated with elements in $\mathcal{D}$ matching $A(x)$). This approach will fail for a fraction $(1 - \frac{n}{|V|}) \geq (1 - \frac{n}{2^{d-1}})$ of possible points we could want to classify, which becomes a problem if n is much smaller than $|V|$. For fixed n this fraction can be improved by decreasing d, the number of dimensions used. In the following, we present an approach to doing this in a differentially private manner.

We say that a function g *discerns* a pair $(x, y) \in \mathcal{U}^2$ if $g(x) \neq g(y)$, furthermore, for $X \subseteq \mathcal{U}$

$$\begin{aligned} \pi_X(g) &= \{(x, y) \in X^2 | g(x) \neq g(y)\} \\ \pi_X(S) &= \cup_{a \in S} \pi_X(a), \text{ for } S \subseteq A. \end{aligned} \tag{8}$$

For any set $S \subseteq A$ we have that $E_X(S) = X^2 - \pi_X(S)$ is an equivalence relation on X. We denote the equivalence class containing $x \in X$ as $E_X(S)(x)$. We can now express the conditional relative frequency density of $f(x) = v$ for any $x \in \mathcal{U}$ and $v \in V_f$ given X and S as

$$\hat{p}(f(x) = v | S, X) = \begin{cases} \frac{|E_X(S)(x) \cap f^{-1}(v)|}{|E_X(S)(x)|} & \text{if } |E_X(S)(x)| > 0, \\ \frac{|f^{-1}(v) \cap X|}{|X|} & \text{otherwise.} \end{cases}$$

The above uses the unconditional relative frequency density if we have not seen the covariate pattern $S(x)$ before.

Proposition 2. *Let $S \subseteq A$. Then,*

$$\pi_X(S) \cap \pi_X(f) \supseteq \pi_X(A) \cap \pi_X(f) \implies \hat{p}(f(x) = v | S, X) = \hat{p}(f(x) = v | A, X).$$

for any $x \in X$ and any $v \in V_f$.

Proposition 2 characterizes subsets S of A that preserve conditional relative frequency densities. Note that finding a minimum size S such that $\pi_X(S) \cap \pi_X(f) \supseteq \pi_X(A) \cap \pi_X(f)$ is NP-hard, as it essentially is the problem of covering $\pi_X(f)$ by sets $\pi_X(a)$, $a \in S$. Motivated by Proposition 2, we formalize our attribute selection problem as finding $S \subseteq A$ with cardinality k, that maximizes $|\pi_X(S) \cap \pi_X(f)|$. We call this problem k-discernibility, and in the following present an algorithm for it. To start, note that we can partition any $R \subseteq X^2$ into n (possibly empty) sets $R(x_i) = \{y|(x_i, y) \in R\}$. Therefore, $|R| = \sum_i |R(x_i)|$. Let $\pi_X(S)(x) = \{y|(x, y) \in \pi_X(S)\}$. Then we have that $(x, y) \in \pi_X(S) \cap \pi_X(f) \iff y \in \pi_X(S)(x) \cap \pi_X(f)(x)$. Consequently,

$$F(S) = \frac{|\pi_X(S) \cap \pi_X(f)|}{n} = \sum_{i=1}^{n} F_i(S) \tag{9}$$

where

$$F_i(S) = \frac{|\pi_X(S)(x_i) \cap \pi_X(f)(x_i)|}{n}.$$

Lemma 1. *F is submodular and non-decreasing for any constant $n > 0$.*

Now consider the function PrivateKD in Algorithm 1, and let F be defined by (9) with n being the number of elements in the data base. If we had picked a such

Algorithm 1. The differentially private algorithm for the k-discernibility problem.

```
PrivateKD(A, F, k, ε')
S ← ∅
A' ← A
for i = 1 to k do
   Pick a from A' with P(a) ∝ exp(ε'(F(S ∪ {a}) − F(S)))
   S ← S ∪ {b}
   A' ← A' − {b}
end for
return S
```

that it maximized $F(S \cup \{a\}) - F(S)$ at each step, the resulting NonPrivateKD would have been a $(1 - 1/e)$ approximation algorithm for k-discernibility since F is submodular [8]. The formulation of k-discernibility in terms of F and PrivateKD allows us to essentially reuse Theorem 8.2 in Gupta et al. [10] as follows.

Theorem 1. PrivateKD *is $4k\epsilon'$-differentially private. Furthermore, if $m = d - 1$ then except with probability $O(1/poly(m))$,* PrivateKD *returns a solution not worse than $(1 - 1/e)\mathrm{OPT} - O(k \log m/\epsilon')$ where $\mathrm{OPT} = F(S^*)$ where S^* is an optimal solution.*

Corollary 1. *If we set* $\epsilon' = \epsilon/(4k)$ *then* PrivateKD *is* ϵ*-differentially private, and has expected solution quality* $(1 - 1/e)\mathrm{OPT} - O(k^2 \log m/\epsilon)$.

We note that $n(F(S \cup \{a\}) - F(S)) = |\pi_X(f) - \pi_X(S)| - |\pi_X(f) - \pi_X(S \cup \{a\})|$, and that $|\pi_X(f) - \pi_X(S)|$ can be computed in terms of the refinement of the partition of X in induced by S by f. We can exploit this to implement the above algorithm to run in $O(k(d-1)n)$ time by at each step keeping track of, and refining, the partition of X induced by the current solution S.

2.3 Projected Histograms

For data $\mathcal{D}$ with target attribute f, and parameters k, ϵ, and $\gamma \in (0, 1]$, we can construct projected histogram as follows. First we spend $\epsilon_p = (1 - \gamma)\epsilon$ of the risk on PrivateKD, finding a $k + 1$ size set of dimensions (including f) to project $\mathcal{D}$ onto. Subsequently we use the remaining risk $\epsilon_h = \gamma\epsilon$ to construct a differentially private histogram from the projected $\mathcal{D}$. We now discuss how to estimate k and γ if they are not given.

Choosing k. Ideally, we are looking for a minimal size k set $S \subseteq A$ such that

$$\frac{|\pi_{\mathcal{D}}(S) \cap \pi_{\mathcal{D}}(f)|}{|\pi_{\mathcal{D}}(f) \cap \pi_{\mathcal{D}}(A)|} \geq 1 - \sigma \tag{10}$$

for some $\sigma \geq 0$. The parameter σ can be seen as a noise suppressing parameter [22]. From a viewpoint of the empirical conditional probability, increasing σ yields a step towards the unconditional prior as equivalence classes are collapsed. Given a σ, we can approximate S for a data set $\mathcal{D}$ by using the NonPrivateKD to find a minimum size set S such that $F(S)/F(A) \geq 1 - \sigma$. Call this algoritm for finding a minimum size set of discerning attributes MDA. Given a non-private data set $\mathcal{D}'$ that is sufficiently close in structure to $\mathcal{D}$ we can now compute $k = |\mathrm{MDA}(f, \mathcal{D}', \sigma)|$. This approach allows the use of non-private data sets deemed appropriate, if they exist. Based on our experience with non-private use of projections [22,23], $\sigma \in [0.1, 0.01]$ is appropriate. We choose $\sigma = 0.1$.

If such a public and representative set is not available, we propose using a simulated artificial data set $\mathcal{D}'$ instead. Assuming that the size q of the range of f is not considered private (a reasonable assumption), we construct $\mathcal{D}'$ by sampling n i.i.d. points as follows. For a point $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ the label x_d is uniformly sampled from $[q]$, and each predictor $i < d$ is assumed distributed as $P(x_i = j | x_d) \sim N(0, 1)$, where $N(0, 1)$ is the standard normal distribution. After sampling, all predictors are scaled into [0,1], and discretized using a bandwidth $\mathrm{bw}(n, d-1)$. We then estimate k as $|\mathrm{MDA}(f, \mathcal{D}', 0.1)|$.

An alternative to the above is to apply the parameter selection method of Chaudhuri et al. [3]. In this approach, we decide on l possibilities of values for k, and partition $\mathcal{D}$ into $l + 1$ parts. Using PrivateKD with privacy ϵ_p, we then compute S_i from $\mathcal{D}_i$ using value k_i, and evaluate each S_i on $\mathcal{D}_{l+1}$, yielding a utility u_i based on $\frac{|\pi_{\mathcal{D}_{l+1}}(S_i) \cap \pi_{\mathcal{D}_{l+1}}(f)|}{|\pi_{\mathcal{D}_{l+1}}(f) \cap \pi_{\mathcal{D}_{l+1}}(A)|}$ and k_i. Finally, we use the exponential

mechanism to choose S_i with probability proportional to $\exp(\epsilon_p u_i)$. According to results by Chaudhuri et al. this procedure yields ϵ_p-differential privacy. As each S_i is computed only using a fraction $\frac{1}{l+1}$ of the available data, this approach is only suitable when n/l is large.

Choosing γ. The parameter γ distributes the privacy budget ϵ between the two tasks of projection and histogram sampling. Intuitively, projecting poorly yields bad model instances even when the modeling is done non-privately. On the other hand, a very diffuse histogram, even when it is on a good projection, will make learning good classifiers difficult. From the above, we know that the quality of projection for a fixed k is determined in part by the fraction of pairs l from $\pi_{\mathcal{D}}(f)$ that are not discerned due to privacy. The quality of the histogram sampling is determined by two probabilities: the probability of a point $\mathbf{x} \in \mathcal{D}$ making it into the support $S = \textsc{Supp}(\tilde{h}_{\tau,\epsilon,\mathcal{D}})$ of the sampled histogram, and the probability of a point $\mathbf{y} \notin \mathcal{D}$ making it in as well. However, applying $h_{\mathcal{D}}(\mathbf{y}) = 0$ to (5) we get that $P(\mathbf{y} \in S) = (2n^{A/2})^{-1}$ and consequently does not depend on ϵ. Hence we will concentrate on finding two values: ϵ_h that yield an acceptable $p = P(\mathbf{x} \in S)$, and ϵ_p that yields an acceptable value for l. Once these have been determined we compute γ as

$$\gamma = \frac{\epsilon_h}{\epsilon_h + \epsilon_p}. \tag{11}$$

Determining ϵ_p. We know from Corollary 1 of Theorem 1 that with a high probability, the solution quality our projection algorithm is $\mathrm{OPT}(1-\exp(-1)) + O(k^2 \log(d-1)/\epsilon)$. This quality is on the scale of F, this means that the number of pairs not discerned due to privacy is $O(nk^2 \log(d-1)/\epsilon)$. Note that the upper bound of the number of pairs f can discern given n elements is $\hat{\pi}(q) = \frac{n^2(q-1)}{2q}$. Using a tuning parameter $B > 0$, we use

$$l \leq \frac{nBk^2 \log(d-1)}{\epsilon_p \hat{\pi}(q)}$$

to relate the fraction of pairs l "lost" due to privacy at level ϵ_p. Consequently, we get

$$\epsilon_p \geq \frac{nBk^2 \log(d-1)}{l\hat{\pi}(q)}$$

for a fixed l. Note that l can be seen as having a similar role as the "overfitting" parameter σ in (10). As the algorithm incurs a loss in discerned pairs of at most $\mathrm{OPT}(1-\exp(1)^{-1})$ without privacy, we choose $l = 0.05$, half the value of σ used for selecting k in order to compensate for this to some degree. We also choose $B = 1/2$ in an ad-hoc manner.

Determining ϵ_h. For a point $\mathbf{x} \in \mathcal{D}$, we have that $P(\mathbf{x} \in S)$ is given by (5). If we let $z = h_{\mathcal{D}}(\mathbf{x})$, then if

$$\epsilon_h \geq \max \left(\frac{\log(n)\ A - 2\log\left(\frac{1}{2p}\right)}{z}, \frac{\log(n)\ A - 2\log(2-2p)}{z} \right)$$

we get $P(\mathbf{x} \in S) \geq p$. What remains is to determine which value z to use. In the absence of other information, we estimate a value for z as follows. Recall that the target is labelled with q labels. We assume that these are uniformly distributed, meaning that the expected number of points in $\mathcal{D}$ that are labelled with the same label is n/q. Furthermore, the projection algorithm aims at approximating the partition induced by the target f, ideally such that all points within an equivalence class induced by the predictors are a subset of an equivalence class induced by f. Also, we assume that predictors are independent given the target. This means that if $p_{\mathbf{x}}$ denotes the probability that a randomly and independently chosen point falls into the equivalence class of $\mathbf{x}$, we can estimate the size of the equivalence class containing $\mathbf{x}$ as $z_{\mathbf{x}} = p_{\mathbf{x}} n/q$. Finally, we assume that each of the $d-1$ predictors is produced from discretizing truncated normally distributed i.i.d. random variables into $1/b$ equally spaced bins after scaling the values to fall into [0,1]. As above, we compute the discretization bandwidth as $b = \mathrm{bw}(n, d-1)$, yielding $s = \lceil 1/b - 0.5 \rceil$ bins. Let p_i be the probability mass of bin i. Then, $p_{\mathbf{x}} = (\sum_{i=1}^{s} p_i^2)^k$ for k predictors. The p_i can be estimated by sampling N points from $N(0,1)$, scaling these into [0,1], partition into s bins c_i, and letting $\hat{p}_i = \frac{|c_i|}{N}$.

3 Experiments

As stated in Section 2.1, given a privacy preserving histogram $\tilde{h}_{\tau,\epsilon,\mathcal{D}}$, we can generate a privacy preserving data set $\tilde{\mathcal{D}}$ by letting $h_{\tilde{\mathcal{D}}}(\mathbf{x}) = \lceil \tilde{h}_{\tau,\epsilon,\mathcal{D}}(\mathbf{x}) - 0.5 \rceil$. We performed experiments to elicit the utility of projected histograms as a non-classifier specific platform to build differentially private classifiers on. We did this by comparing the performance of binary and multinomial logistic regression classifiers build on $\tilde{\mathcal{D}}$ versus built on the original $\mathcal{D}$, as well with classifiers based on a particular method for differentially private logistic regression trained on $\mathcal{D}$. We used the R environment environment for statistical computing [19] for all our experiments.

3.1 Setup

Classifiers. The classification tasks in our experiments are both binary and multinomial. For both we used three different classifier learning algorithms. In particular, for binary classification these are OLR – non-private logistic regression using the generalized linear modeling `glm` function of the `stats` [19] library, HLR – an OLR classifier trained on data created from a differentially private projected histogram, and PLR – logistic regression using the differentially private

empirical risk minimization [3]. For multinomial classification these are OMLR – non-private multinomial logistic regression using the `multinom` function of the `nnet` [21] package, HMLR – an OMLR classifier trained on data created from a differentially private projected histogram, and PMLR – a classifier constructed from learning an ϵ/q-differentially private PLR classifier for each of the q classes and applying them in parallel to obtain normalized probabilities for each class.

Data Sets. In the experiments we used four data sets. Two intended for binary prediction and two for multi-class prediction. For each of binary and multi-class, one data set is smaller and one is larger.

Iris – a 3 class data set of 4 predictor attributes on 150 cases. The data set describes 4 measurements of three different types of iris plants. The data set is available from the UCI Machine Learning Repository [9] as the "Iris Data Set".

Satellite – a 6-class data set of 36 predictor attributes on 6435 cases. The data set describes a 3x3 neighborhood of pixels in a Landsat satellite multi-spectrum digital image. The six classes are soil types. The data set is available from the UCI Machine Learning Repository [9] as the "Statlog (Landsat Satellite) Data Set".

Infarct – a 2 class data set of 44 predictor attributes on 1753 cases. The data set describes 44 measurements taken from patients presenting with chest pain at the emergency rooms of two hospitals, one in Sheffield, England, and one in Edinburgh, Scotland. The classification is according to whether the patients received a diagnosis of myocardial infarction or not. The data set is not publicly available, and has been used in [14].

Adult – a 2 class data set of 14 predictor variables on 45083 cases. The data set describes measurements on cases taken from the 1994 Census data base. The classification is whether or not a person has an annual income exceeding 50000 USD. The data set is available from the UCI Machine Learning Repository [9] as the "Adult Data Set".

Estimation of Performance. In order to be able to relate performances across binary and multinomial classification tasks, we assessed binary classification quality using the area under the receiver operating curve (AUC), and the generalization of the AUC given by Hand et al. [11] for the multinomial tasks.

Construction of a projected histogram has parameters ϵ – the differential privacy level, k – the number of sub-dimensions to project onto, and γ – which decides the distribution of ϵ among projection and histogram creation. Given ϵ, and the size of the input data, the algorithm can estimate the two latter parameters k and γ. We ran separate ten-fold cross-validation experiments stratified on class labels to *i.* compare classifier performances given a baseline privacy requirement $\epsilon = 1$ and using estimated k and γ, and *ii.* investigate the quality of the estimation applied to produce k and γ, as well as behavior of histogram based classifiers under varying privacy requirements ϵ. Each parameter variation was investigated in independent experiments where the other parameters were set to either to baseline ($\epsilon = 1$) or left for the algorithm to estimate (k and γ). The tuning parameters A and B were left fixed at $A = 1/2$ and $B = 1/2$.

3.2 Results

Figure 1 shows box-and-whisker plots of the cross-validation AUCs for the classifiers on all four data sets (upper row: multi-class, lower row: binary class, left

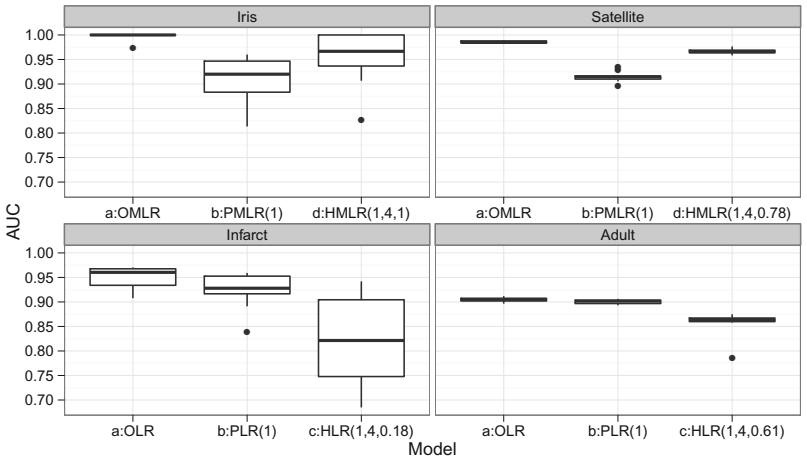

Fig. 1. Box-and-whisker plots of the 10-fold cross-validation AUC performance values for the classifiers on the four data sets. The quantiles in the plots are the standard 25%, 50%, and 75%. The whiskers denote the most extreme points within 1.5 times the respective interquartile ranges (i.e., the 1.5 "rule"), while outliers are denoted by dots.

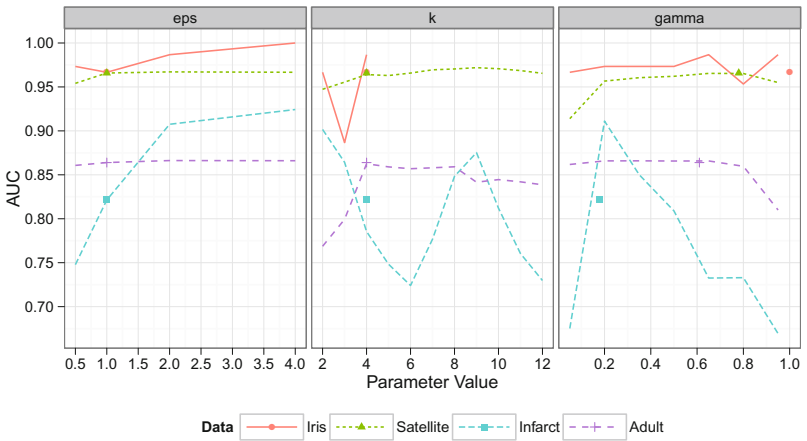

Fig. 2. The median cross-validation AUC for the histogram based method as we vary parameters ϵ, k, and γ on each of the four data sets. The dots represent the baseline/estimated value for the parameter and the AUC value taken from the ϵ series of cross-validation experiments.

column: smaller, right column: larger). Figure 2 shows the behavior of the median cross-validation AUC for the projected histogram method H(M)LR as we vary parameters ϵ, k, and γ.

4 Discussion

We presented a differentially private feature selection algorithm and used it to project data before computing differentially private histograms. We provided a method for distributing the allowed privacy risk between the two tasks.

We experimented with logistic regression in order to investigate the classifier performance loss between using perturbed projected histograms for input perturbation over a specialized differentially private method. While loss is present, as can be seen in the bottom row of Figure 1, projected perturbed histograms offer utility as the performance still is arguably good.

To demonstrate the use of projected histograms when there are no specialized alternatives available, we used projected histograms to construct differentially private multinomial logistic regression models. We know of no differentially private and available algorithm for learning non-binary probabilistic classifiers; as the alternative for q classes, we used estimates from q differentially private binary logistic regression models (we also experimented with using parameters from $q-1$ models in a softmax function, but this is sensitive to base class selection and performed worse than using q binary model estimates directly). The models learned from perturbed projected histograms outperformed the private multi-model alternative as seen in the top row in Figure 1. This supports that considering using perturbed projected histograms for input perturbation over using multiple applications of specialized methods that each incur separate and additive risk is warranted.

A limitation of our approach is that the parameter k must be estimated. In the middle panel of Figure 2 we see that while in general $k = 4$ yields good performance, it is not optimal. In particular, for the smaller binary class data set, In fact, $k = 2$ would have increased the competitiveness of the binary logistic regression based on histograms significantly. However, the right panel in Figure 2 shows that for the estimated k, the distribution of the allowed privacy risk between projection and histogram computation is near optimal.

For both binary and multinomial regression we chose two data sets, one smaller and one larger. The leftmost panel in Figure 2 shows the expected improvement of histogram based models with increase in allowed privacy risk. This improvement is much stronger for smaller data. Our intuition for this is that for larger data sets, the reduction in performance due to privacy is dominated by the privacy independent loss in performance due to projection and discretization. We speculate that projected histograms might be even better suited for classification tasks in discrete spaces.

As the size of differentially private histograms grows exponentially in the dimensionality of the feature space, efficient representation and computation is of importance. For spaces of size bounded by 2^B where B is the machine

word length, we presented an efficient approach to both computation as well as representation of these histograms.

References

1. Barak, B., Chaudhuri, K., Dwork, C., Kale, S., McSherry, F., Talwar, K.: Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In: PODS. pp. 273–282 (2007)
2. Belazzougui, D., Botelho, F., Dietzfelbinger, M.: Hash, displace, and compress. In: Fiat, A., Sanders, P. (eds.) Algorithms - ESA 2009, Lecture Notes in Computer Science, vol. 5757, pp. 682–693. Springer Berlin / Heidelberg (2009)
3. Chaudhuri, K., Monteleoni, C., Sarwate, A.: Differentially private empirical risk minimization. JMLR 12, 1069–1109 (March 2011)
4. Dwork, C.: Differential privacy: A survey of results. Theory and Applications of Models of Computation pp. 1–19 (2008)
5. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: 3rd IACR TCC. pp. 486–503 (2006)
6. Dwork, C., Smith, A.: Differential privacy for statistics : What we know and what we want to learn. J. Privacy and Confidentiality 1(2), 135–154 (2008)
7. Dwork, C.: Differential privacy. In: M. Bugliesi, B. Preneel, V.S., Wegener, I. (eds.) ICALP (2). Lecture notes in computer science, vol. 4052, pp. 1–12. Springer Verlag (2006)
8. Fisher, M., Nemhauser, G., Wolsey, L.: An analysis of approximations for maximizing submodular set functions—ii. Polyhedral combinatorics pp. 73–87 (1978)
9. Frank, A., Asuncion, A.: UCI machine learning repository (2010)
10. Gupta, A., Ligett, K., McSherry, F., Roth, A., Talwar, K.: Differentially private combinatorial optimization. In: Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 1106–1125. Society for Industrial and Applied Mathematics (2010)
11. Hand, D.J., Till, R.J.: A simple generalisation of the area under the roc curve for multiple class classification problems. Machine Learning 45, 171–186 (2001), 10.1023/A:1010920819831
12. Hay, M., Rastogi, V., Miklau, G., Suciu, D.: Boosting the accuracy of differentially private histograms through consistency. Proceedings of the VLDB Endowment 3(1-2), 1021–1032 (2010)
13. Jagadish, H., Koudas, N., Muthukrishnan, S., Poosala, V., Sevcik, K., Suel, T.: Optimal histograms with quality guarantees. In: Proceedings of the International Conference on Very Large Data Bases. pp. 275–286. INSTITUTE OF ELECTRICAL & ELECTRONICS ENGINEERS (1998)
14. Kennedy, R.L., Burton, A.M., Fraser, H.S., McStay, L.N., Harrison, R.F.: Early diagnosis of acute myocardial infarction using clinical and electrocardiographic data at presentation: Derivation and evaluation of logistic regression models. European Heart Journal 17, 1181–1191 (1996)
15. Lei, J.: Differentially private m-estimators. In: NIPS. pp. 361–369 (2011)
16. McSherry, F., Talwar, K.: Mechanism design via differential privacy. In: FOCS. pp. 94–103 (2007)
17. Mohammed, N., Chen, R., Fung, B., Yu, P.: Differentially private data release for data mining. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 493–501. ACM (2011)

18. Pawlak, Z.: Rough Sets, Theoretical Aspects of Reasoning about Data, Series D: System Theory, Knowledge Engineering and Problem Solving, vol. 9. Kluwer Academic Publishers (1991)
19. R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2011), ISBN 3-900051-07-0
20. Ullman, J., Vadhan, S.: Pcps and the hardness of generating synthetic data. In: ECCC. vol. 17, p. 17 (2010)
21. Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S. Springer, New York, fourth edn. (2002), iSBN 0-387-95457-0
22. Vinterbo, S., Øhrn, A.: Minimal approximate hitting sets and rule templates. International Journal of Approximate Reasoning 25(2), 123–143 (2000)
23. Vinterbo, S.A., Kim, E.Y., Ohno-Machado, L.: Small, fuzzy and interpretable gene expression based classifiers. Bioinformatics 21(9), 1964–1970 (Jan 2005)
24. Vitter, J.S.: An efficient algorithm for sequential random sampling. ACM Trans. Math. Softw. 13(1), 58–67 (Mar 1987)
25. Xiao, Y., Xiong, L., Yuan, C.: Differentially private data release through multidimensional partitioning. Secure Data Management pp. 150–168 (2011)
26. Xu, J., Zhang, Z., Xiao, X., Yang, Y., Yu, G.: Differentially private histogram publication. In: Proceedings of the IEEE International Conference on Data Engineering (2012)

A Proofs

Proof of Proposition 2. For readability, let $E_B(x) = E_X(B)(x)$ for all $B \subseteq A$, and let $E_f(x) = E_X(f)(x)$. First, we note that $E_A(x) \subseteq E_S(x)$. If $E_S(x) \subseteq E_f(x)$ for any $S \subseteq A$ we have that the proposition holds as

$$E_S(x) \cap f^{-1}(v) = \begin{cases} E_S(x) & \text{if } f(x) = v, \text{ and} \\ \emptyset & \text{otherwise.} \end{cases}$$

Hence, in order to prove the proposition we need to show that for $S \subseteq A$

$$(\pi(S) \cap \pi(f) \supseteq \pi(A) \cap \pi(f) \wedge E_S(x) \not\subseteq E_f(x)) \implies E_S(x) = E_A(x). \quad (12)$$

We start by noting that for all $x \in X$ $\pi(S) \cap \pi(f) \supseteq \pi(A) \cap \pi(f) \implies E_s(x) \cup E_f(x) \subseteq E_A(x) \cup E_f(x)$, which in turn implies that $(E_S(x) - E_f(x)) \subseteq E_A(x)$. This means that as $E_S(x) \not\subseteq E_f(x)$ we can pick $y \in E_S(x) \cap E_A(x)$ such that $y \notin E_f(x)$. Assuming the negation of (12) (giving $E_S(x) \neq E_A(x)$) we can pick an additional point $z \in E_S(x) \cap E_f(x)$ such that $z \notin E_A(x)$. This means $(y, z) \notin \pi(S)$ and $(y, z) \in \pi(A) \cap \pi(f)$ which creates a contradiction, from which we conclude that (12) must hold. □

Proof of Lemma 1. (Sketch:) The requirement for F being submodular is that $F(S \cup S') + F(S \cap S') \leq F(S) + F(S')$ for any $S, S' \subseteq A$. For π defined by (8), we have that $A \subseteq B \implies \pi(A) \subseteq \pi(B)$, and $\pi(A \cup B) = \pi(A) \cup \pi(B))$. We then have that $\pi(A) \cap \pi(B) = (\pi(A - B) \cup \pi(A \cap B)) \cap (\pi(B - A) \cup \pi(B \cap A)) \supseteq \pi(A \cap B)$. Consequently, $m(S) = |\pi(S)|$ is submodular and non-decreasing, as is $F(S) = |\pi(S) \cap \pi(f)|/n$ for fixed f and n. □

Proof of Theorem 1. Consider any $X' \subseteq U$ such that $(X' \cup X) - (X' \cap X) = \{\mathbf{x}_i\}$. This means that X' and X only differ in the element $\mathbf{x}_i$. Changing $\mathbf{x}_i$ leads to change in $F_i^X(S) = |\pi(S)(\mathbf{x}_i) \cap \pi(f)(\mathbf{x}_i)|/n$ at most 1, while in $F_j^X(S)$ at most $1/(n-1)$ for $j \neq i$. This means that $F^X = \sum_i F_i^X$ yields $\Delta(F) \leq 2$. The probability $p(a)$ of selecting a from $A - S$ is

$$p(a) = \frac{e^{\epsilon'(F(S\cup\{a\})-F(S))}}{\sum_{a'\in A-S} e^{\epsilon'(F(S\cup\{a'\})-F(S))}} = \frac{e^{\epsilon' F(S\cup\{a\})}}{\sum_{a'\in A-S} e^{\epsilon' F(S\cup\{a'\})}} \tag{13}$$

As in the standard argument for privacy of the exponential mechanism, we note that a change of $\Delta(F)$ contributes at most $e^{\epsilon'\Delta(F)}$ to the numerator of (13), and at least $e^{-\epsilon'\Delta(F)}$ to the denominator, yielding $2\Delta(F)\epsilon'$-differential privacy. We are selecting k times, each with $4\epsilon'$-differential privacy, hence $4k\epsilon'$ total differential privacy.

Similarly to McSherry and Talwar's analysis of the exponential mechanism [16], let $s(a) = F(S \cup \{a\}) - F(S)$ for a fixed $S \subseteq A$, let $\mathrm{OPT} = \max_a s(a)$, and $S_t = \{a | \exp(\epsilon s(a)) > t\}$. Then we have that

$$\begin{aligned} P(\overline{S}_{(t/\epsilon')}) &\leq \frac{P(\overline{S}_{(\mathrm{OPT}-t/\epsilon')})}{P(S_{\mathrm{OPT}})} = \frac{\sum_{a\in\overline{S}_{(\mathrm{OPT}-t/\epsilon')}} \exp(\epsilon' s(a))}{\sum_{a\in S_{\mathrm{OPT}}} \exp(\epsilon' s(a))} \\ &\leq \frac{\sum_{a\in\overline{S}_{(\mathrm{OPT}-t/\epsilon')}} \exp(\epsilon'(\mathrm{OPT} - t/\epsilon'))}{\sum_{a\in S_{\mathrm{OPT}}} \exp(\epsilon'\mathrm{OPT})} = \frac{|\overline{S}_{(\mathrm{OPT}-t/\epsilon')}|}{|S_{\mathrm{OPT}}|}\exp(-t) \\ &\leq |A - S|\exp(-t) \leq m\exp(-t). \end{aligned} \tag{14}$$

If we let $t = \log(m) + t'$, we get that $P(\overline{S}_{(\mathrm{OPT}-(\log(m)+t')/\epsilon')}) \leq \exp(-t')$. If we now let $t' = a\log(m)$, we get that $P(\overline{S}_{(\mathrm{OPT}-((a+1)\log(m))/\epsilon')}) \leq \exp(-a\log(m)) = 1/m^a$, and $P(S_{(\mathrm{OPT}-((a+1)\log(m))/\epsilon')}) \geq 1 - 1/m^a$. This means that the probability that the algorithm chooses k times within $S_{(\mathrm{OPT}-((a+1)\log(m))/\epsilon')}$ is at least $1 - k/m^a$, i.e., we choose well except with a probability $O(k/m^a) = O(1/poly(m))$. When we choose well, since F is sub-modular, every choice adds at most a factor $(1 - 1/k)\mathrm{OPT}$ to the final solution quality gap while adding at most $(a+1)\log(m)/\epsilon'$ to the same. Consequently the gap is $\mathrm{OPT}(1 - 1/e) + O(k\log(m)/\epsilon')$. □