Journal of Privacy and Confidentiality (2012) **4**, Number 1, 65–100

Learning in a Large Function Space: Privacy-Preserving Mechanisms for SVM Learning

Benjamin I. P. Rubinstein*, Peter L. Bartlett†, Ling Huang‡, and Nina Taft§

Abstract. The ubiquitous need for analyzing privacy-sensitive information—including health records, personal communications, product ratings, and social network data—is driving significant interest in privacy-preserving data analysis across several research communities. This paper explores the release of Support Vector Machine (SVM) classifiers while preserving the privacy of training data. The SVM is a popular machine learning method that maps data to a high-dimensional feature space before learning a linear decision boundary. We present efficient mechanisms for finite-dimensional feature mappings and for (potentially infinite-dimensional) mappings with translation-invariant kernels. In the latter case, our mechanism borrows a technique from large-scale learning to learn in a finite-dimensional feature space whose inner-product uniformly approximates the desired feature space inner-product (the desired kernel) with high probability. Differential privacy is established using algorithmic stability, a property used in learning theory to bound generalization error. Utility—when the private classifier is pointwise close to the non-private classifier with high probability—is proven using smoothness of regularized empirical risk minimization with respect to small perturbations to the feature mapping. Finally we conclude with lower bounds on the differential privacy of any mechanism approximating the SVM.

1 Introduction

The goal of a well-designed statistical database is to provide aggregate information about a database's entries while maintaining individual entries' privacy. These two goals of *utility* and *privacy* are inherently discordant. For a mechanism to be useful, its responses must closely resemble some target statistic of the database's entries. However, to protect privacy, it is often necessary for the mechanism's response distribution to be 'smoothed out', i.e., the mechanism must be randomized to reduce the individual entries' influence on this distribution. A key interest of the theory, learning, and statistical database communities is to understand when the goals of utility and privacy can be efficiently achieved simultaneously (Dinur and Nissim, 2003; Barak et al., 2007; Blum et al., 2008; Chaudhuri and Monteleoni, 2009; Kasiviswanathan et al., 2008; Hardt and Talwar, 2010; Beimel et al., 2010). In studying privacy-preserving learning in this paper, we adopt

*Microsoft Research, Mountain View, CA, mailto:Ben.Rubinstein@microsoft.com

†Division of Computer Science and Department of Statistics, University of California, Berkeley, mailto:bartlett@cs.berkeley.edu

‡Intel Labs, Berkeley, CA, mailto:ling.huang@intel.com

§Technicolor, Palo Alto, CA, mailto:Nina.Taft@technicolor.com

the strong notion of differential privacy as formalized by Dwork et al. (2006).

In this paper we consider the practical goal of releasing a trained Support Vector Machine (SVM) classifier while maintaining the privacy of its training data. The SVM follows the principle of margin maximization and is built on a strong theoretical foundation (Burges, 1998; Cristianini and Shawe-Taylor, 2000; Schölkopf and Smola, 2001; Bishop, 2006). Over the last decade, there has been an uptake in the application of SVMs used for classification to a variety of application domains such as Internet traffic classification (Kim et al., 2008), recommendation systems (Xu and Araki, 2006), web search (Joachims, 2002), cancer diagnosis (Ramaswamy et al., 2001), and much more. Together these properties make the SVM an ideal candidate for privacy-preserving learning.

Our contributions are summarized as follows. First, we propose two mechanisms for differentially-private SVM learning, one for learning under finite-dimensional feature mappings (cf. Section 3) and one for learning with potentially infinite-dimensional feature mappings with translation-invariant kernels (cf. Section 4). Both mechanisms operate by adding noise to the output classifier; for each we prove the range of noise parameters required in order to guarantee privacy, and we also derive the conditions under which the mechanisms yield close approximations to the non-private SVM. These results are fairly general, since they cover not only finite and many infinite dimensional feature spaces, but also all convex loss functions. Our analysis applies to the case of hinge loss, the most frequent among loss functions used in SVM classification, that is not included in work on differentially-private SVMs done in parallel to our own (Chaudhuri et al., 2011). Second, in order to prove differential privacy, we develop a novel proof technique for privacy based upon algorithmic stability, a property of learning algorithms conventionally used for proving bounds on generalization error. Third, to handle the case of infinite-dimensional feature spaces, we propose using a technique of mapping data to a finite-dimensional random feature space instead of the target feature space, which results in only minor changes to the resulting SVM classifications.

Fourth, we define a notion of optimal differential privacy as the best privacy achievable among all mechanisms that approximate a non-private SVM. We combine the results on privacy and utility of our mechanisms in order to derive upper bounds on the optimal differential privacy, which states that the level of privacy achieved will be at least as good as the upper bound. We instantiate the upper bound in a case-study on the hinge loss (cf. Section 5). Fifth, we present two lower bounds on optimal differential privacy by proving impossibility results for privately learning with linear kernels and with the Radial Basis Function (RBF) kernel (cf. Section 6).

1.1 Related Work

An earlier version of this paper appeared as a technical report (Rubinstein et al., 2009). Independently, Sarwate et al. (2009) considered alternate privacy-preserving SVM mechanisms. Their mechanism for linear SVM guarantees differential privacy by adding a random term to the objective, as done previously by Chaudhuri and Monteleoni (2009)

for regularized logistic regression, and as is possible for a relatively general class of regularized empirical risk minimizers (Chaudhuri et al., 2011). For non-linear SVMs the authors exploit the same large-scale learning technique due to Rahimi and Recht (2008) we use here. It is noteworthy that preserving privacy via the randomized objective can only apply to convex differentiable loss functions, ruling out the most common case of the non-differentiable hinge loss; our mechanisms preserve privacy for any convex loss, which is a very weak condition since convexity is required in order for the formulation of SVM learning to be convex. In the recent journal version of their report (Chaudhuri et al., 2011), in addition to providing a more complete and unified treatment of the objective perturbation method, the authors experimentally show that their random objective method outperforms an output perturbation method based on random noise similar to our own on two benchmark datasets. Such a comparison between our methods is not easy analytically: while Sarwate et al. (2009) prove risk bounds (bounds on generalization error), our definition of utility measures the point-wise similarity of the private SVM classifier to the non-private SVM classifier. Thus, as Chaudhuri et al. (2011) note, our theoretical results are incomparable to theirs. Indeed it is noteworthy that for SVM learning with the hinge loss, guarantees on our notion of utility are strictly stronger than their risk bound measure (cf. Remark 7). Moreover our definition of utility offers a natural advantage: an arbitrary differentially-private mechanism that enjoys low risk is not necessarily an approximation of a given learning algorithm of interest; it is natural to expect that a private SVM approximates the classifications of a non-private SVM. Guarantees with respect to our measure of utility imply such approximation and (for the SVM) low risk. Sarwate et al. (2009) develop a method for tuning the regularization parameter while preserving privacy, a step of the learning process not considered here, using a comparison procedure due to McSherry and Talwar (2007). In addition to positive results on differentially-private SVM learning, we provide lower bounds on simultaneously achievable utility and privacy. Finally our proof of privacy is interesting due to its novel use of stability.

A rich literature of prior work on differential privacy exists. We overview some of this work and contrast it to our own.

Range Spaces Parametrizing Vector-Valued Statistics or Simple Functions. Early work on private interactive mechanisms focused on approximating real- and vector-valued statistics (e.g, Dinur and Nissim 2003; Blum et al. 2005; Dwork et al. 2006; Dwork 2006; Barak et al. 2007). McSherry and Talwar (2007) first considered private mechanisms with range spaces parametrizing sets more general than real-valued vectors, and used such differentially-private mappings for mechanism design. More related to our work are the private mechanisms for regularized logistic regression proposed and analyzed by Chaudhuri and Monteleoni (2009). There the mechanism's range space parametrizes the VC-dimension $d+1$ class of linear hyperplanes in $\mathbb{R}^d$. As stated above, one of their mechanisms injects a random term into the primal objective in order to achieve differential privacy. Their simpler mechanism adds noise to the learned weight vector, in a similar vein to our mechanism for SVM's with finite-dimensional feature mappings. Our stability-based calculation of SVM sensitivity (cf. Section 3) is a generalization of the derivation of the sensitivity of regularized logistic regression (Chaudhuri

and Monteleoni, 2009), to the setting of non-differentiable loss functions, with the condition on the gradient replaced by the Lipschitz condition and the condition on the Hessian replaced by strong convexity. Kasiviswanathan et al. (2008) show that discretized concept classes can be PAC or agnostically learned privately, albeit via an inefficient mechanism. Blum et al. (2008) show that non-interactive mechanisms can privately release anonymized data such that utility is guaranteed over classes of predicate queries with polynomial VC dimension, when the domain is discretized. Dwork et al. (2009) more recently characterized when utility and privacy can be achieved by efficient non-interactive mechanisms. In this paper we consider efficient mechanisms for private SVM learning, whose range spaces parametrize real-valued functions. One case covered by our analysis is learning with a Gaussian kernel, which corresponds to learning over a rich class of infinite dimension.

Practical Privacy-Preserving Learning (Mostly) via Subset-Sums. Most prior work in differential privacy has focused on the deep analysis of mechanisms for relatively simple statistics (with histograms and contingency tables as explored by Blum et al., 2005 and Barak et al., 2007 respectively, as examples) and learning algorithms (e.g., interval queries and half-spaces as explored by Blum et al., 2008), or on constructing learning algorithms that can be decomposed into subset-sum operations (e.g, perceptron, k-NN, ID3 as described by Blum et al., 2005, and various recommender systems [McSherry and Mironov, 2009]). By contrast, we consider the practical goal of SVM learning, which does not generally decompose into a subset sum (cf. Appendix 7). It is also notable that our mechanisms run in polynomial time. The most related work to our own in this regard is due to Chaudhuri and Monteleoni (2009), although their results hold only for differentiable loss, and finite feature mappings.

The Privacy-Utility Trade-Off. Like several prior studies, we consider the trade-off between privacy and utility. Barak et al. (2007) present a mechanism for releasing contingency tables that guarantees differential privacy and also guarantees a notion of accuracy: with high probability all marginals from the released table are close in L_1-norm to the true marginals. As mentioned above, Blum et al. (2008) develop a private non-interactive mechanism that releases anonymized data such that all predicate queries in a VC class take on similar values on the anonymized data and original data. Kasiviswanathan et al. (2008) consider utility as corresponding to PAC learning: with high probability the response and target concepts are close, averaged over the underlying measure.

Previous negative results show that any mechanism providing overly accurate responses cannot be private (Dinur and Nissim, 2003; Dwork and Yekhanin, 2008; Beimel et al., 2010; Hardt and Talwar, 2010). Dinur and Nissim (2003) show that if noise of rate only $o(\sqrt{n})$ is added to subset-sum queries on a database of bits, then an adversary can reconstruct a $1 - o(1)$ fraction of the bits. This threshold phenomenon says that if accuracy is too great, privacy cannot be guaranteed at all. We show a similar negative result for the case of private SVM learning: i.e., for all mechanisms we quantify the common intuition that requiring very high accuracy with respect to the SVM prevents high levels of privacy.

Our results are qualitatively closer to those of Hardt and Talwar (2010) and Beimel et al. (2010). The former work finds almost matching upper and lower bounds for the trade-off between differential privacy and accuracy through the lens of convex geometry in a setting that encompasses releasing histograms and recommender systems. Queries submitted to the interactive mechanism are linear mappings on a private database of reals. Non-private responses are the vector image of the query applied to the database and the mechanism's responses are a randomized version of this target image, and the mechanism's accuracy is the expected Euclidean distance between non-private and private responses. Beimel et al. (2010) focus on the notion of private learning (Kasiviswanathan et al., 2008) in which a private learner not only PAC learns, but the release of its hypothesis is differentially private with respect to the training data. Beimel et al. (2010) delve into the sample complexity of private learning and demonstrate separation results between proper and improper private learning[1]—which do not exist for non-private PAC learning—and between efficient and inefficient proper private learners. While both papers consider negative results on the trade-off between notions of utility and differential privacy, their analyses do not cover SVM learning for which the concept classes are not necessarily linear or have finite VC dimension.

The ϵ-packing proof technique used in our second lower bound for SVM learning with the RBF kernel, although discovered independently, is similar to the technique used by Hardt and Talwar (2010) to establish lower bounds for their setting of privately responding to linear map queries.

Connections between Stability and Differential Privacy. To prove differential privacy, we borrow a proof technique from algorithmic stability. In passing, Kasiviswanathan et al. (2008) predict a possible relationship between algorithmic stability and differential privacy, however do not exploit this.

2 Background and Definitions

Before developing our mechanisms for private SVM learning, we overview the relevant topics from machine learning and privacy. We assume basic knowledge of probability, analysis, and optimization.

2.1 Statistical Learning Theory

A *database* or training set D is a sequence of $n \in \mathbb{N}$ *rows* or *examples* $(\mathbf{x}_i, y_i) \in \mathbb{R}^d \times \{-1, 1\}$, which are pairs of points in d-dimensional *input space* and binary labels. A *learning map* $\mathcal{A}$ maps a database D to a classifier $f_D : \mathbb{R}^d \to \mathbb{R}$. A learned classifier produces binary predictions in $\{-1, 1\}$ by thresholding its real-valued output as in $\operatorname{sgn}(f_D(\mathbf{x}))$; when this is done, the real-valued prediction is often used as a measure of confidence for the binary classification. This paper develops randomized learning mappings that release classifiers while preserving the privacy of their training data D.

[1] A proper learner outputs a hypothesis from the target concept class.

In supervised learning tasks such as above, where the output of learning (e.g., a classifier) maps points (e.g., $\mathbf{x} \in \mathbb{R}^d$) to some response (e.g., $y \in \mathbb{R}$ or $y \in \{0, 1\}$), a *loss function* $\ell(y, \hat{y}) \in \mathbb{R}$ is used to measure the discrepancy or error in using prediction $\hat{y}$ for approximating true response y. In binary classification the most natural loss function is the 0-1 loss $\mathbf{1}[y = \hat{y}]$ for binary-valued responses or $\text{sgn}(y\hat{y})_+$ for real-valued responses. It is common to assume that the database's rows are drawn identically (and usually independently) from some fixed but unknown joint distribution μ on point-label pairs. A natural goal for learning then is to choose a classifier f that minimizes the expected loss with respect to a random draw from μ:

$$R[f] \quad = \quad \mathbb{E}_{(\mathbf{X},Y)\sim\mu}\left[\ell\left(Y, f(\mathbf{X})\right)\right].$$

This quantity is known as the *risk* of f. The achieved risk of both randomized and deterministic learning maps $R[\mathcal{A}(D)]$ is itself a random quantity due to the randomness in D. As learners do not have access to μ, it is necessary to minimize some empirical surrogate of the risk. The *empirical risk minimization* (ERM) principle involves minimizing the *empirical risk* of f on D.

$$R_{\text{emp}}[f] \quad = \quad \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, f(\mathbf{X}_i)),$$

which estimates the risk, and over the set of classifiers in the range space forms an empirical process. We refer the interested reader to the significant work in empirical process theory which has gone into studying these processes (van der Vaart and Wellner, 2000; van der Vaart, 2000; Pollard, 1984).

ERM can lead to overfitting or poor generalization (risk of the minimizer), so in theory and practice it is more desirable to perform regularized empirical risk minimization, which minimizes the sum of the empirical risk and a regularization term which imposes a soft smoothness constraint on the classifier. A well-known example is the soft-margin *Support Vector Machine* (SVM) which has the following primal program

$$\min_{\mathbf{w}\in\mathbb{R}^F} \qquad \frac{1}{2}\|\mathbf{w}\|_2^2 + \frac{C}{n}\sum_{i=1}^{n} \ell\left(y_i, f_{\mathbf{w}}(\mathbf{x}_i)\right),$$

where for chosen *feature mapping* $\phi : \mathbb{R}^d \to \mathbb{R}^F$ taking points in input space $\mathbb{R}^d$ to some (possibly infinite) F-dimensional *feature space*, and hyperplane normal $\mathbf{w} \in \mathbb{R}^F$, we define

$$f_{\mathbf{w}}(\mathbf{x}) \quad = \quad \langle \phi(\mathbf{x}), \mathbf{w} \rangle.$$

Parameter $C > 0$ is the soft-margin parameter that controls the amount of regularization. Let $\mathbf{w}^\star$ denote an optimizing weight vector. Then predictions are taken as the sign of $f^\star(\mathbf{x}) \;=\; f_{\mathbf{w}^\star}(\mathbf{x})$.We will refer to both $f_{\mathbf{w}}(\cdot)$ and $\text{sgn}\left(f_{\mathbf{w}}(\cdot)\right)$ as *classifiers*, with the exact meaning apparent from the context.

An overview of the relevant details on SVM learning follows; for full details see for example Burges (1998); Cristianini and Shawe-Taylor (2000); Schölkopf and Smola

(2001); Bishop (2006). In order for the primal to be convex and the process of SVM learning to be tractable, $\ell(y, \hat{y})$ is chosen to be a loss function that is convex in $\hat{y}$. A common convex surrogate for the 0-1 loss, and the loss most commonly associated with the SVM, is the hinge loss $\ell(y, \hat{y}) = (1 - y\hat{y})_+$ which upper bounds the 0-1 loss and is non-differentiable at $y\hat{y} = 1$. Other example losses include the square loss $(1 - y\hat{y})^2$ and the logistic loss $\log(1 + \exp(-y\hat{y}))$. We consider general convex losses in this paper, and a detailed case-study of private SVM learning under the hinge loss in Section 5.

Remark 1 *We say that a learning algorithm is* universally consistent *if for all distributions μ it is* consistent*: its expected risk converges to the minimum achievable (Bayes) risk with increasing sample size (Devroye et al., 1996). For universal consistency, the SVM's parameter C should grow like $\sqrt{n}$.*

When F is large the solution may be more easily obtained via the dual. For example, the following is the dual formulation on the n dual variables for learning with hinge loss

$$\begin{aligned} \max_{\boldsymbol{\alpha} \in \mathbb{R}^n} \quad & \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq \frac{C}{n} \; \forall i \in [n] \,, \end{aligned} \tag{1}$$

where $k(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle$ is the *kernel function*.

The vector of maximizing duals $\boldsymbol{\alpha}^\star$ parametrizes the function $f^\star = f_{\boldsymbol{\alpha}^\star}$ as

$$f_{\boldsymbol{\alpha}}(\cdot) \quad = \quad \sum_{i=1}^n \alpha_i y_i k(\cdot, \mathbf{x}_i).$$

The space of SVM classifiers endowed with the kernel function forms a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$.

Definition 2 *A* reproducing kernel Hilbert space *is a Hilbert space*[2] *of real-valued functions including, for each point $\mathbf{x}$, a point-evaluation function $k(\cdot, \mathbf{x})$ having the reproducing kernel property $\langle f, k(\cdot, \mathbf{x}) \rangle = f(\mathbf{x})$ for all $f \in \mathcal{H}$.*

In particular, $\langle k(\cdot, \mathbf{x}), k(\cdot, \mathbf{z}) \rangle = k(\mathbf{x}, \mathbf{z})$. The Representer Theorem (Kimeldorf and Wahba, 1971) implies that the minimizer $f^\star = \arg\min_{f \in \mathcal{H}} \frac{1}{2} \|f\|_{\mathcal{H}}^2 + \frac{C}{n} \sum_{i=1}^n \ell(y_i, f(\mathbf{x}_i))$ lies in the span of the functions $k(\cdot, \mathbf{x}_i) \in \mathcal{H}$. Indeed the above dual expansion shows that the coordinates in this subspace are given by the $\alpha_i^\star y_i$. We define the SVM *mechanism* to be the dual optimization that responds with the vector $\boldsymbol{\alpha}^\star$, as described by Algorithm 1.

A number of kernels/feature mappings have been proposed in the literature (Burges, 1998; Cristianini and Shawe-Taylor, 2000; Schölkopf and Smola, 2001; Bishop, 2006). The *translation-invariant kernels* are an important class of kernel that we study in the sequel (see Table 1 for examples).

[2] A Hilbert space is an inner-product space which is complete with respect to its norm-induced metric.

Kernel	$g(\Delta)$
RBF	$\exp\left(-\frac{\|\Delta\|_2^2}{2\sigma^2}\right)$
Laplacian	$\exp\left(-\|\Delta\|_1\right)$
Cauchy	$\prod_{i=1}^{d} \frac{2}{1+\Delta_i^2}$

Table 1: Example translation-invariant kernels and their g functions.

Algorithm 1 SVM

Inputs: database $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with $\mathbf{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$; kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$; convex loss ℓ; parameter $C > 0$.

1. $\boldsymbol{\alpha}^\star \leftarrow$ Solve the SVM's dual;
2. Return vector $\boldsymbol{\alpha}^\star$.

Definition 3 *A kernel function of the form $k(\mathbf{x}, \mathbf{y}) = g(\mathbf{x} - \mathbf{y})$, for some function g, is called* translation invariant.

In proving bounds on the differential privacy of our mechanisms for private SVM learning, we will exploit the uniform stability of regularized ERM as established by Bousquet and Elisseeff (2002).

We say that a pair of databases D_1, D_2 are *neighbors* if they differ on one entry, and define the learning stability with respect to neighboring databases as follows.

Definition 4 *A* learning map $\mathcal{A}$*, that takes databases D to classifiers, is said to have* γ-uniform stability *with respect to loss $\ell(\cdot, \cdot)$ if for all neighboring databases D, D', the losses of the classifiers trained on D and D' are close on all test examples $\|\ell(\cdot, \mathcal{A}(D)) - \ell(\cdot, \mathcal{A}(D'))\|_\infty \leq \gamma$.*

Stability corresponds to smoothness of the learning map, and the concept is typically used in statistical learning theory to yield tight risk bounds, sometimes when class capacity-based approaches (such as VC dimension-based approaches) do not apply (Devroye and Wagner, 1979; Kearns and Ron, 1999; Bousquet and Elisseeff, 2002; Kutin and Niyogi, 2002). Intuitively if a learning map is stable then it is not overly influenced by noise, and is less likely to suffer from overfitting.

2.2 Differential Privacy

We now begin the preliminary background on privacy, starting with the definition of differential privacy. Given access to database D, a *mechanism* M must release aggregate information about D while maintaining privacy of individual entries. We assume that the *response* $M(D)$, belonging to range space $\mathcal{T}_M$, is the only information released by the mechanism. We adopt the following strong notion of privacy due to Dwork et al. (2006).

Definition 5 *For any $\beta > 0$, a randomized mechanism M provides* β-differential privacy*, if, for all neighboring databases D_1, D_2 and all responses $t \in \mathcal{T}_M$ the mechanism*

satisfies

$$\log\left(\frac{\Pr\left(M(D_1)=t\right)}{\Pr\left(M(D_2)=t\right)}\right) \quad \leq \quad \beta.$$

The probability in the definition is over the randomization in M, not the databases. For continuous $\mathcal{T}_M$ we mean by this ratio a Radon-Nikodym derivative of the distribution of $M(D_1)$ with respect to the distribution of $M(D_2)$. In the sequel we assume without loss of generality that each pair of neighboring databases differ on their last entry. To understand the definition, consider a mechanism M preserving a high level of privacy. Even with knowledge of M up to randomness and knowledge of the first $n-1$ entries of D, an adversary cannot learn any additional information on the true identity of the n^{th} entry from a sublinear sample from $M(D)$. They may even calculate the distribution of $M(D')$ for every D-neighboring D' by simulating M with different choices for the n^{th} example; however, for sufficiently small β, these distributions will be closer than the $M(D)$ sampling error.

Intuitively, the more an 'interesting' mechanism M is perturbed to guarantee differential privacy, the less like M the resulting mechanism $\hat{M}$ will become. The next definition formalizes the notion of 'likeness'.

Definition 6 *Consider two mechanisms $\hat{M}$ and M with the same domains and with response spaces $\mathcal{T}_{\hat{M}}$ and $\mathcal{T}_M$. Let $\mathcal{X}$ be some set and let $\mathcal{F} \subseteq \mathbb{R}^{\mathcal{X}}$ be parametrized by the response spaces: for every $t \in \mathcal{T}_{\hat{M}} \cup \mathcal{T}_M$ define some corresponding function $f_t \in \mathcal{F}$. Finally, assume $\mathcal{F}$ is endowed with norm $\|\cdot\|_{\mathcal{F}}$. Then for $\epsilon > 0$ and $0 < \delta < 1$ we say that $\hat{M}$ is (ϵ, δ)-useful[3] with respect to M if, for all databases D,*

$$\Pr\left(\left\|f_{\hat{M}(D)} - f_{M(D)}\right\|_{\mathcal{F}} \leq \epsilon\right) \quad \geq \quad 1-\delta.$$

Typically $\hat{M}$ will be a privacy-preserving (perturbed) version of M. In the sequel we take $\|\cdot\|_{\mathcal{F}}$ to be $\|f\|_{\infty;\mathcal{M}} = \sup_{\mathbf{x}\in\mathcal{M}} |f(\mathbf{x})|$ for some $\mathcal{M} \subseteq \mathbb{R}^d$ containing the data. It will also be convenient to define $\|k\|_{\infty;\mathcal{M}} = \sup_{\mathbf{x},\mathbf{y}\in\mathcal{M}} |k(\mathbf{x},\mathbf{y})|$ for bivariate $k(\cdot,\cdot)$.

Remark 7 *In this paper we develop privacy-preserving mechanisms that are useful with respect to the SVM. Since the hinge loss is Lipschitz in the classifier output by the SVM, any mechanism $\hat{M}$ having utility with respect to the SVM also has expected hinge loss that is within ϵ of the SVM's hinge loss whp. i.e., (ϵ, δ)-usefulness with respect to the sup-norm is stronger than guaranteed closeness of risk.*

The following general notion, defined specifically for the SVM here, quantifies the highest level of privacy achievable over all (ϵ, δ)-useful mechanisms with respect to a target mechanism M. We present upper and lower bounds on $\beta(\epsilon, \delta, C, n, \ell, k)$ in Sections 5 and 6.

[3]Our definition of (ϵ, δ)-usefulness for releasing a single function is analogous to the notion of the same name introduced by Blum et al. (2008) for anonymization mechanisms.

Algorithm 2 PRIVATESVM-FINITE

Inputs: database $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with $\mathbf{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$; finite feature map $\phi : \mathbb{R}^d \to \mathbb{R}^F$ and induced kernel k; convex loss function ℓ; and parameters $\lambda, C > 0$.

1. $\boldsymbol{\alpha}^\star \leftarrow$ Run Algorithm 1 on D with parameter C, kernel k, and loss ℓ;
2. $\tilde{\mathbf{w}} \leftarrow \sum_{i=1}^n \alpha_i^\star y_i \phi(\mathbf{x}_i)$;
3. $\boldsymbol{\mu} \leftarrow$ Draw i.i.d. sample of F scalars from Laplace $(0, \lambda)$; and
4. Return $\hat{\mathbf{w}} = \tilde{\mathbf{w}} + \boldsymbol{\mu}$

Definition 8 *For $\epsilon, C > 0$, $\delta \in (0, 1)$, $n > 1$, loss function $\ell(y, \hat{y})$ convex in $\hat{y}$, and kernel k, the* optimal differential privacy for the SVM *is the function*

$$\beta^\star(\epsilon, \delta, C, n, \ell, k) \quad = \quad \inf_{\hat{M} \in \mathcal{I}} \sup_{(D_1, D_2) \in \mathcal{D}} \sup_{t \in \mathcal{T}_{\hat{M}}} \log \left(\frac{\Pr\left(\hat{M}(D_1) = t\right)}{\Pr\left(\hat{M}(D_2) = t\right)} \right),$$

where $\mathcal{I}$ is the set of all (ϵ, δ)-useful mechanisms with respect to the SVM with parameter C, loss ℓ, and kernel k; and $\mathcal{D}$ is the set of all pairs of neighboring databases with n entries.

3 Mechanism for Finite-Dimensional Feature Maps

In this section we consider differentially-private SVM learning with finite F-dimensional feature maps. We begin by describing the mechanism, then prove the range of noise parameters required in order to guarantee privacy (Theorem 10) and derive the conditions under which the mechanism yields close approximations to the non-private SVM (Theorem 10).

Algorithm 2 describes our PRIVATESVM-FINITE mechanism, which follows the established pattern of preserving differential privacy (Dwork et al., 2006). After forming the primal solution to the SVM—weight vector $\mathbf{w} \in \mathbb{R}^F$—the mechanism adds i.i.d. zero-mean, scale λ, Laplace noise to $\mathbf{w}$. Differential privacy follows from a two-step process of calculating the L_1-sensitivity Δ of $\mathbf{w}$ to data perturbations, then showing that β-differential privacy follows from sensitivity together with the choice of Laplace noise with scale $\lambda = \Delta/\beta$.

To calculate sensitivity—the change in $\mathbf{w}$ with respect to the L_1-norm when a training example is changed—we exploit the uniform stability of regularized ERM (cf. Definition 4).

Lemma 9 *Consider loss function $\ell(y, \hat{y})$ that is convex and L-Lipschitz in $\hat{y}$, and RKHS $\mathcal{H}$ induced by finite F-dimensional feature mapping ϕ with bounded kernel $k(\mathbf{x}, \mathbf{x}) \leq \kappa^2$*

for all $\mathbf{x} \in \mathbb{R}^d$. *For each database* $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, *define*

$$\mathbf{w}_S \quad \in \quad \arg\min_{\mathbf{w}\in\mathbb{R}^F} \frac{C}{n}\sum_{i=1}^{n} \ell\left(y_i, f_{\mathbf{w}}(\mathbf{x}_i)\right) + \frac{1}{2}\|\mathbf{w}\|_2^2.$$

Then for every pair of neighboring databases D, D' *of* n *entries, we have* $\|\mathbf{w}_D - \mathbf{w}_{D'}\|_2 \leq 4LC\kappa/n$, *and* $\|\mathbf{w}_D - \mathbf{w}_{D'}\|_1 \leq 4LC\kappa\sqrt{F}/n$.

Proof. The argument closely follows the proof of the SVM's uniform stability (Schölkopf and Smola, 2001, Theorem 12.4). For convenience we define for any training set S

$$\begin{aligned} R_{\text{reg}}(\mathbf{w}, S) &= \frac{C}{n}\sum_{i=1}^{n} \ell\left(y_i, f_{\mathbf{w}}(\mathbf{x}_i)\right) + \frac{1}{2}\|\mathbf{w}\|_2^2 \\ R_{\text{emp}}(\mathbf{w}, S) &= \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_{\mathbf{w}}(\mathbf{x}_i)). \end{aligned}$$

Then the first-order necessary KKT conditions imply

$$\mathbf{0} \quad \in \quad \partial_{\mathbf{w}} R_{\text{reg}}(\mathbf{w}_D, D) = C\partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_D, D) + \mathbf{w}_D\,, \tag{2}$$

$$\mathbf{0} \quad \in \quad \partial_{\mathbf{w}} R_{\text{reg}}(\mathbf{w}_{D'}, D') = C\partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_{D'}, D') + \mathbf{w}_{D'} \tag{3}$$

where $\partial_{\mathbf{w}}$ is the subdifferential operator wrt $\mathbf{w}$. Define the auxiliary risk function

$$\tilde{R}(\mathbf{w}) \quad = \quad C\langle \partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_D, D) - \partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_{D'}, D'),\, \mathbf{w} - \mathbf{w}_{D'}\rangle + \frac{1}{2}\|\mathbf{w} - \mathbf{w}_{D'}\|_2^2\,.$$

Note that $\tilde{R}(\cdot)$ maps to sets of reals. It is easy to see that $\tilde{R}(\mathbf{w})$ is strictly convex in $\mathbf{w}$. Substituting $\mathbf{w}_{D'}$ into $\tilde{R}(\mathbf{w})$ yields

$$\begin{aligned} \tilde{R}(\mathbf{w}_{D'}) &= C\left\langle \partial_{\mathbf{w}} R_{\text{emp}}\left(\mathbf{w}_D, D\right) - \partial_{\mathbf{w}} R_{\text{emp}}\left(\mathbf{w}_{D'}, D'\right), 0\right\rangle + \frac{1}{2}\|0\|_2^2 \\ &= \{0\}. \end{aligned}$$

And by Equation (3)

$$\begin{aligned} C\partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_D, D) + \mathbf{w} &\in C\partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_D, D) - C\partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_{D'}, D') + \mathbf{w} - \mathbf{w}_{D'} \\ &= \partial_{\mathbf{w}} \tilde{R}(\mathbf{w})\,, \end{aligned}$$

which combined with Equation (2) implies $\mathbf{0} \in \partial_{\mathbf{w}} \tilde{R}(\mathbf{w}_D)$, so that $\tilde{R}(\mathbf{w})$ is minimized at $\mathbf{w}_D$. Thus there exists some non-positive $r \in \tilde{R}(\mathbf{w}_D)$. Next simplify the first term of $\tilde{R}(\mathbf{w}_D)$, scaled by n/C for notational convenience. In what follows we denote by $\ell'(y, \hat{y})$

the subdifferential $\partial_{\hat{y}}\ell(y,\hat{y})$:

$$\begin{aligned}
& n\langle \partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_D, D) - \partial_{\mathbf{w}} R_{\text{emp}}(\mathbf{w}_{D'}, D'),\, \mathbf{w}_D - \mathbf{w}_{D'}\rangle \\
= & \sum_{i=1}^{n} \langle \partial_{\mathbf{w}} \ell\left(y_i, f_{\mathbf{w}_D}(\mathbf{x}_i)\right) - \partial_{\mathbf{w}} \ell\left(y_i', f_{\mathbf{w}_{D'}}(\mathbf{x}_i')\right),\, \mathbf{w}_D - \mathbf{w}_{D'}\rangle \\
= & \sum_{i=1}^{n-1} \left(\ell'\left(y_i, f_{\mathbf{w}_D}(\mathbf{x}_i)\right) - \ell'\left(y_i, f_{\mathbf{w}_{D'}}(\mathbf{x}_i)\right)\right)\left(f_{\mathbf{w}_D}(\mathbf{x}_i) - f_{\mathbf{w}_{D'}}(\mathbf{x}_i)\right) \\
& \quad + \ell'\left(y_n, f_{\mathbf{w}_D}(\mathbf{x}_n)\right)\left(f_{\mathbf{w}_D}(\mathbf{x}_n) - f_{\mathbf{w}_{D'}}(\mathbf{x}_n)\right) \\
& \quad - \ell'\left(y_n', f_{\mathbf{w}_{D'}}(\mathbf{x}_n')\right)\left(f_{\mathbf{w}_D}(\mathbf{x}_n') - f_{\mathbf{w}_{D'}}(\mathbf{x}_n')\right) \\
\geq & \ell'\left(y_n, f_{\mathbf{w}_D}(\mathbf{x}_n)\right)\left(f_{\mathbf{w}_D}(\mathbf{x}_n) - f_{\mathbf{w}_{D'}}(\mathbf{x}_n)\right) \\
& \quad - \ell'\left(y_n', f_{\mathbf{w}_{D'}}(\mathbf{x}_n')\right)\left(f_{\mathbf{w}_D}(\mathbf{x}_n') - f_{\mathbf{w}_{D'}}(\mathbf{x}_n')\right) \ .
\end{aligned}$$

Here the second equality follows from $\partial_{\mathbf{w}}\ell\left(y, f_{\mathbf{w}}(\mathbf{x})\right) = \ell'\left(y, f_{\mathbf{w}}(\mathbf{x})\right)\phi(\mathbf{x})$, and $\mathbf{x}_i' = \mathbf{x}_i$ and $y_i' = y_i$ for each $i \in [n-1]$. The inequality follows from the convexity of ℓ in its second argument.[4] Combined with the existence of non-positive $r \in \tilde{R}(\mathbf{w}_D)$ this yields that there exists

$$\begin{aligned}
g \ \in \ & \ell'\left(y_n', f_{\mathbf{w}_{D'}}(\mathbf{x}_n')\right)\left(f_{\mathbf{w}_D}(\mathbf{x}_n') - f_{\mathbf{w}_{D'}}(\mathbf{x}_n')\right) \\
& -\ell'\left(y_n, f_{\mathbf{w}_D}(\mathbf{x}_n)\right)\left(f_{\mathbf{w}_D}(\mathbf{x}_n) - f_{\mathbf{w}_{D'}}(\mathbf{x}_n)\right)
\end{aligned}$$

such that

$$\begin{aligned}
0 \ & \geq \ \frac{n}{C} r \\
& \geq \ g + \frac{n}{2C}\|\mathbf{w}_D - \mathbf{w}_{D'}\|_2^2 .
\end{aligned}$$

And since $|g| \leq 2L\left\|f_{\mathbf{w}_D} - f_{\mathbf{w}_{D'}}\right\|_\infty$ by the Lipschitz continuity of ℓ, this in turn implies

$$\frac{n}{2C}\|\mathbf{w}_D - \mathbf{w}_{D'}\|_2^2 \ \leq \ 2L\left\|f_{\mathbf{w}_D} - f_{\mathbf{w}_{D'}}\right\|_\infty \ . \tag{4}$$

Now by the reproducing property and Cauchy-Schwartz inequality we can upper bound the classifier difference's infinity norm by the Euclidean norm on the weight vectors: for each $\mathbf{x}$

$$\begin{aligned}
\left|f_{\mathbf{w}_D}(\mathbf{x}) - f_{\mathbf{w}_{D'}}(\mathbf{x})\right| \ & = \ |\langle \phi(\mathbf{x}), \mathbf{w}_D - \mathbf{w}_{D'}\rangle| \\
& \leq \ \|\phi(\mathbf{x})\|_2 \, \|\mathbf{w}_D - \mathbf{w}_{D'}\|_2 \\
& = \ \sqrt{k(\mathbf{x},\mathbf{x})}\, \|\mathbf{w}_D - \mathbf{w}_{D'}\|_2 \\
& \leq \ \kappa \, \|\mathbf{w}_D - \mathbf{w}_{D'}\|_2 \ .
\end{aligned}$$

Combining this with Inequality (4) yields $\|\mathbf{w}_D - \mathbf{w}_{D'}\|_2 \leq 4LC\kappa/n$ as claimed. The L_1-based sensitivity then follows from $\|\mathbf{w}\|_1 \leq \sqrt{F}\|\mathbf{w}\|_2$ for all $\mathbf{w} \in \mathbb{R}^F$. ■

[4]Namely for convex f and any $a, b \in \mathbb{R}$, $(g_a - g_b)(a - b) \geq 0$ for all $g_a \in \partial f(a)$ and all $g_b \in \partial f(b)$.

For SVM with Gaussian kernel, we have $L = 1$ and $\kappa = 1$. Then the bounds can be simplified as $\|\mathbf{w}_D - \mathbf{w}_{D'}\|_2 \leq 4C/n$ and $\|\mathbf{w}_D - \mathbf{w}_{D'}\|_1 \leq 4C\sqrt{F}/n$. With the weight vector's sensitivity in hand, differential privacy follows immediately from the proof technique established by Dwork et al. (2006).

Theorem 10 (Privacy of PrivateSVM-Finite) *For any $\beta > 0$, database D of size n, $C > 0$, loss function $\ell(y, \hat{y})$ that is convex and L-Lipschitz in $\hat{y}$, and finite F-dimensional feature map with kernel $k(\mathbf{x}, \mathbf{x}) \leq \kappa^2$ for all $\mathbf{x} \in \mathbb{R}^d$,* PrivateSVM-Finite *run on D with loss ℓ, kernel k, noise parameter $\lambda \geq 4LC\kappa\sqrt{F}/(\beta n)$, and regularization parameter C guarantees β-differential privacy.*

Proof. Let D_1, D_2 be a pair of neighboring size n DBs. For $i \in \{1, 2\}$, let $\boldsymbol{\mu}_\mathbf{i}$ denote i.i.d. zero-mean Laplace random variables with scale λ, and $\tilde{\mathbf{w}}_\mathbf{i}$ denote the SVM primal solution on D_i. Let $\hat{\mathbf{w}} \in \mathbb{R}^F$ be a response of PrivateSVM-Finite. The ratio of probabilities $\Pr(M(D_1) = \hat{\mathbf{w}})$ and $\Pr(M(D_2) = \hat{\mathbf{w}})$ can be bounded by

$$\frac{\Pr(\boldsymbol{\mu}_\mathbf{1} = \hat{\mathbf{w}} - \tilde{\mathbf{w}}_\mathbf{1})}{\Pr(\boldsymbol{\mu}_\mathbf{2} = \hat{\mathbf{w}} - \tilde{\mathbf{w}}_\mathbf{2})} = \frac{\exp(-\|\hat{\mathbf{w}} - \tilde{\mathbf{w}}_\mathbf{1}\|_1/\lambda)}{\exp(-\|\hat{\mathbf{w}} - \tilde{\mathbf{w}}_\mathbf{2}\|_1/\lambda)} \leq \exp\left(\frac{\|\tilde{\mathbf{w}}_\mathbf{1} - \tilde{\mathbf{w}}_\mathbf{2}\|_1}{\lambda}\right).$$

The equality holds by the noise's joint density with normalization canceling. The inequality follows by combining the two exponential terms and the triangle inequality, which allows the $\hat{\mathbf{w}}$ terms to be canceled. Taking logs we see that the choice of $\lambda \geq 4LC\kappa\sqrt{F}/(\beta n)$ guarantees β-differential privacy. ∎

This first main result states that higher levels of privacy require more noise, while more training examples reduce the level of required noise. We next establish the (ϵ, δ)-usefulness of PrivateSVM-Finite using the exponential tails of the noise vector $\boldsymbol{\mu}$. By contrast to privacy, utility demands that the noise not be too large.

Theorem 11 (Utility of PrivateSVM-Finite) *Consider any $C > 0$, $n > 1$, database D of n entries, arbitrary convex loss ℓ, and finite F-dimensional feature mapping ϕ with kernel k and $|\phi(\mathbf{x})_i| \leq \Phi$ for all $\mathbf{x} \in \mathcal{M}$ and $i \in [F]$ for some $\Phi > 0$ and $\mathcal{M} \subseteq \mathbb{R}^d$. For any $\epsilon > 0$ and $\delta \in (0, 1)$,* PrivateSVM-Finite *run on D with loss ℓ, kernel k, noise parameter $0 < \lambda \leq \frac{\epsilon}{2\Phi\left(F + \log_e \frac{1}{\delta}\right)}$, and regularization parameter C, is (ϵ, δ)-useful with respect to the* SVM *under the $\|\cdot\|_{\infty;\mathcal{M}}$-norm.*

In other words, run with arbitrary noise parameter $\lambda > 0$, PrivateSVM-Finite is (ϵ, δ)-useful for $\epsilon = \Omega\left(\lambda\Phi\left(F + \log_e \frac{1}{\delta}\right)\right)$.

Proof. Consider the SVM and PrivateSVM-Finite classifications on an arbitrary point $\mathbf{x} \in \mathcal{M}$:

$$\begin{aligned}\left|f_{\hat{M}(D)}(\mathbf{x}) - f_{M(D)}(\mathbf{x})\right| &= |\langle\hat{\mathbf{w}}, \phi(\mathbf{x})\rangle - \langle\tilde{\mathbf{w}}, \phi(\mathbf{x})\rangle| \\ &= |\langle\boldsymbol{\mu}, \phi(\mathbf{x})\rangle| \\ &\leq \|\boldsymbol{\mu}\|_1 \|\phi(\mathbf{x})\|_\infty \\ &\leq \Phi\|\boldsymbol{\mu}\|_1 .\end{aligned}$$

The absolute value of a zero-mean Laplace random variable with scale λ is exponentially distributed with scale λ^{-1}. Moreover the sum of q i.i.d. exponential random variables has Erlang q-distribution with the same scale parameter. Thus we have, for Erlang F-distributed random variable X and any $t > 0$,

$$\begin{aligned} \forall \mathbf{x} \in \mathcal{M},\ \left| f_{\hat{M}(D)}(\mathbf{x}) - f_{M(D)}(\mathbf{x}) \right| &\leq \Phi\, X \\ \Rightarrow\ \forall \epsilon > 0,\ \Pr\left(\left\| f_{\hat{M}(D)} - f_{M(D)} \right\|_{\infty;\mathcal{M}} > \epsilon \right) &\leq \Pr\left(X > \epsilon/\Phi\right) \\ &\leq \frac{\mathbb{E}\left[e^{tX}\right]}{e^{\epsilon t/\Phi}}. \end{aligned} \tag{5}$$

Here we have employed the Chernoff tail bound technique using Markov's inequality. The numerator of (5), the moment generating function of the Erlang F-distribution with parameter λ, is $(1-\lambda t)^{-F}$ for $t < \lambda^{-1}$. With the choice of $t = (2\lambda)^{-1}$, this gives

$$\begin{aligned} \Pr\left(\left\| f_{\hat{M}(D)} - f_{M(D)} \right\|_{\infty;\mathcal{M}} > \epsilon \right) &\leq (1-\lambda t)^{-F}\, e^{-\epsilon t/\Phi} \\ &= 2^F e^{-\epsilon/(2\lambda\Phi)} \\ &= \exp\left(F \log_e 2 - \frac{\epsilon}{2\lambda\Phi} \right) \\ &< \exp\left(F - \frac{\epsilon}{2\lambda\Phi} \right). \end{aligned}$$

And provided that $\epsilon \geq \left(2\lambda\Phi\left(F + \log_e \frac{1}{\delta}\right)\right)$ this probability is bounded by δ. ■

4 Mechanism for Translation-Invariant Kernels

We now consider the problem of privately learning in an RKHS $\mathcal{H}$ induced by an infinite-dimensional feature mapping ϕ. We begin the section by deriving the mechanism, then establish the range of noise parameters required to guarantee privacy (Corollary 12) and derive the conditions under which the mechanism yields close approximations to the non-private SVM (Theorem 13).

It is natural to look to the dual SVM as a starting point: an optimizing $f^\star \in \mathcal{H}$ must lie in the span of the data by the Representer Theorem (Kimeldorf and Wahba, 1971). While the coordinates with respect to this data basis—the $\alpha_i^\star$ dual variables—could be perturbed to guarantee differential privacy, the basis (i.e., the data itself) is also needed to parametrize $f^\star$. Instead, we approach the problem by approximating $\mathcal{H}$ with a random RKHS $\hat{\mathcal{H}}$ induced by a random finite-dimensional map $\hat{\phi}$, which admits a response based on a primal parametrization. Algorithm 3 summarizes this mechanism.

As noted recently by Rahimi and Recht (2008), the Fourier transform p of the kernel function g, a continuous positive-definite translation-invariant function, is a non-negative measure (Rudin, 1994). If the kernel g is properly scaled, Bochner's theorem guarantees that p is a proper probability distribution. Rahimi and Recht (2008) exploit this fact to construct a random RKHS $\hat{\mathcal{H}}$ by drawing $\hat{d}$ vectors $\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_{\hat{d}}$ from p, and

Kernel	$g(\Delta)$	$p(\omega)$
RBF	$\exp\left(-\frac{\|\Delta\|_2^2}{2\sigma^2}\right)$	$\frac{1}{(2\pi)^{d/2}}\exp\left(\frac{-\|\omega\|_2^2}{2}\right)$
Laplacian	$\exp\left(-\|\Delta\|_1\right)$	$\prod_{i=1}^{d}\frac{1}{\pi\left(1+\omega_i^2\right)}$
Cauchy	$\prod_{i=1}^{d}\frac{2}{1+\Delta_i^2}$	$\exp\left(-\|\Delta\|_1\right)$

Table 2: The translation-invariant kernels of Table 1, their g functions and the corresponding Fourier transforms p.

Algorithm 3 PRIVATESVM

Inputs: database $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with $\mathbf{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$; translation-invariant kernel $k(\mathbf{x}, \mathbf{y}) = g(\mathbf{x} - \mathbf{y})$ with Fourier transform $p(\boldsymbol{\omega}) = 2^{-1}\int e^{-j\langle\boldsymbol{\omega},\mathbf{x}\rangle} g(\mathbf{x})\, d\mathbf{x}$; convex loss function ℓ; parameters $\lambda, C > 0$ and $\hat{d} \in \mathbb{N}$.

1. $\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_{\hat{d}} \leftarrow$ Draw i.i.d. sample of $\hat{d}$ vectors in $\mathbb{R}^d$ from p;
2. $\hat{\boldsymbol{\alpha}} \leftarrow$ Run Algorithm 1 on D with parameter C, kernel $\hat{k}$ induced by map (6), and loss ℓ;
3. $\tilde{\mathbf{w}} \leftarrow \sum_{i=1}^n y_i \hat{\alpha}_i \hat{\phi}(\mathbf{x}_i)$ where $\hat{\phi}$ is defined in Equation (6);
4. $\boldsymbol{\mu} \leftarrow$ Draw i.i.d. sample of $2\hat{d}$ scalars from Laplace $(0, \lambda)$; and
5. Return $\hat{\mathbf{w}} = \tilde{\mathbf{w}} + \boldsymbol{\mu}$ and $\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_{\hat{d}}$.

defining the random $2\hat{d}$-dimensional feature map

$$\hat{\phi}(\cdot) \quad = \quad \hat{d}^{-1/2}\left[\cos\left(\langle\boldsymbol{\rho}_1, \cdot\rangle\right), \sin\left(\langle\boldsymbol{\rho}_1, \cdot\rangle\right), \ldots, \cos\left(\langle\boldsymbol{\rho}_{\hat{d}}, \cdot\rangle\right), \sin\left(\langle\boldsymbol{\rho}_{\hat{d}}, \cdot\rangle\right)\right]^T . \qquad (6)$$

Table 2 presents three translation-invariant kernels and their transformations. Inner products in the random feature space $\hat{k}(\cdot,\cdot)$ approximate $k(\cdot,\cdot)$ uniformly, and to arbitrary precision depending on parameter $\hat{d}$, as restated in Lemma 18. Rahimi and Recht (2008) apply this approximation to large-scale learning, finding good approximations for $\hat{d} \ll n$. We perform regularized ERM in $\hat{\mathcal{H}}$, not to avoid complexity in n, but to provide a direct finite representation $\tilde{\mathbf{w}}$ of the primal solution in the case of infinite-dimensional feature spaces. Subsequently, Laplace noise is added to the primal solution $\tilde{\mathbf{w}}$ to guarantee differential privacy as before.

Unlike PRIVATESVM-FINITE, PRIVATESVM must release a parametrization of feature map $\hat{\phi}$—the sample $\{\boldsymbol{\rho}_i\}_{i=1}^{\hat{d}}$—in order to classify as $\hat{f}^\star(\cdot) = \langle\hat{\mathbf{w}}, \hat{\phi}(\cdot)\rangle$. Of PRIVATESVM's response, only $\hat{\mathbf{w}}$ depends on D; the $\boldsymbol{\rho}_i$ are data-independent draws from the kernel's transform p, which we assume to be known by the adversary (to wit the adversary knows the mechanism, including k). Thus to establish differential privacy we

need only consider the weight vector, as we did for PrivateSVM-Finite.

Corollary 12 (Privacy of PrivateSVM) *For any $\beta > 0$, database D of size n, $C > 0$, $\hat{d} \in \mathbb{N}$, loss function $\ell(y, \hat{y})$ that is convex and L-Lipschitz in $\hat{y}$, and translation-invariant kernel k,* PrivateSVM *run on D with loss ℓ, kernel k, noise parameter $\lambda \geq 2^{2.5} LC\sqrt{\hat{d}}/(\beta n)$, approximation parameter $\hat{d}$, and regularization parameter C guarantees β-differential privacy.*

Proof. The result follows from Theorem 10 since $\tilde{\mathbf{w}}$ is the primal solution of SVM with kernel $\hat{k}$, the response vector $\hat{\mathbf{w}} = \tilde{\mathbf{w}} + \boldsymbol{\mu}$, and $\hat{k}(\mathbf{x}, \mathbf{x}) = 1$ for all $\mathbf{x} \in \mathbb{R}^d$. The extra factor of $\sqrt{2}$ comes from the fact that $\hat{\phi}(\cdot)$ is a $2\hat{d}$-dimensional feature map. ■

This result is surprising in that PrivateSVM is able to guarantee privacy for regularized ERM over a function class of infinite dimension, where the obvious way to return the learned classifier (responding with the dual variables and feature mapping) reveals all the entries corresponding to the support vectors, completely.

The remainder of this section is spent proving the following main result which states that PrivateSVM is useful with respect to the SVM .

Theorem 13 (Utility of PrivateSVM) *Consider any database D, compact set $\mathcal{M} \subset \mathbb{R}^d$ containing D, convex loss ℓ, translation-invariant kernel k, and scalars $C, \epsilon > 0$, and $\delta \in (0, 1)$. Suppose the SVM with loss ℓ, kernel k, and parameter C has dual variables with L_1-norm bounded by Λ. Then Algorithm 3 run on D with loss ℓ, kernel k, parameters $\hat{d} \geq \frac{4(d+2)}{\theta(\epsilon)} \log_e \left(\frac{2^9 (\sigma_p \mathrm{diam}(\mathcal{M}))^2}{\delta \theta(\epsilon)} \right)$ where $\theta(\epsilon) = \min \left\{ 1, \frac{\epsilon^4}{2^4 \left(\Lambda + 2\sqrt{(CL + \Lambda/2)\Lambda} \right)^4} \right\}$, $\lambda \leq \min \left\{ \frac{\epsilon}{2^4 \log_e 2\sqrt{\hat{d}}}, \frac{\epsilon\sqrt{\hat{d}}}{8 \log_e \frac{2}{\delta}} \right\}$ and parameter C, is (ϵ, δ)-useful with respect to Algorithm 1 run on D with loss ℓ, kernel k, and parameter C, with respect to the $\| \cdot \|_{\infty;\mathcal{M}}$-norm.*

Remark 14 *Theorem 13 introduces the assumption that the SVM has a dual solution vector with bounded L_1-norm. The motivation for this condition is the most common case for SVM classification of learning with the hinge loss. Under this loss the dual program* (1) *has box constraints which ensure that this condition is satisfied.*

The result of Theorem 13 bounds the pointwise distance between classifiers $f^\star$ output by SVM and $\hat{f}^\star$ output by PrivateSVM whp. Let $\tilde{f}$ be the function parametrized by intermediate weight vector $\tilde{\mathbf{w}}$. Then we establish the main result by proving that both $f^\star$ and $\hat{f}^\star$ are close to $\tilde{f}$ whp and applying the triangle inequality. We begin by relating $\tilde{f}$ and $f^\star$. As $f^\star$ is the result of adding Laplace noise to $\tilde{\mathbf{w}}$, the task of relating these two classifiers is almost the same as proving utility of PrivateSVM-Finite (cf. Theorem 11).

Corollary 15 *Consider a run of Algorithms 1 and 3 with $\hat{d} \in \mathbb{N}$, $C > 0$, convex loss, and translation-invariant kernel. Denote by $\hat{f}^\star$ and $\tilde{f}$ the classifiers parametrized by weight vectors $\hat{\mathbf{w}}$ and $\tilde{\mathbf{w}}$, respectively, where these vectors are related by $\hat{\mathbf{w}} = \tilde{\mathbf{w}} + \boldsymbol{\mu}$ with $\boldsymbol{\mu} \overset{iid}{\sim} \text{Laplace}(0, \lambda)$ in Algorithm 3. For any $\epsilon > 0$ and $\delta \in (0,1)$, if $0 < \lambda \leq \min\left\{ \frac{\epsilon}{2^4 \log_e 2\sqrt{\hat{d}}}, \frac{\epsilon\sqrt{\hat{d}}}{8 \log_e \frac{2}{\delta}} \right\}$ then $\Pr\left(\left\| \hat{f}^\star - \tilde{f} \right\|_\infty \leq \frac{\epsilon}{2} \right) \geq 1 - \frac{\delta}{2}$.*

Proof. As in the proof of Theorem 11 we can use the Chernoff trick to show that, for Erlang $2\hat{d}$-distributed random variable X, the choice of $t = (2\lambda)^{-1}$, and for any $\epsilon > 0$

$$\begin{aligned} \Pr\left(\left\| \hat{f}^\star - \tilde{f} \right\|_\infty > \epsilon/2 \right) &\leq \frac{\mathbb{E}\left[e^{tX}\right]}{e^{\epsilon t \sqrt{\hat{d}}/2}} \\ &\leq (1 - \lambda t)^{-2\hat{d}} \, e^{-\epsilon t \sqrt{\hat{d}}/2} \\ &= 2^{2\hat{d}} e^{-\epsilon\sqrt{\hat{d}}/(4\lambda)} \\ &= \exp\left(\hat{d} \log_e 4 - \epsilon\sqrt{\hat{d}}/(4\lambda) \right) . \end{aligned}$$

Provided that $\lambda \leq \epsilon / \left(2^4 \log_e 2\sqrt{\hat{d}}\right)$, this is bounded by $\exp\left(-\epsilon\sqrt{\hat{d}}/(8\lambda)\right)$. Moreover, if $\lambda \leq \epsilon\sqrt{\hat{d}} / \left(8 \log_e \frac{2}{\delta}\right)$, then the claim follows. ∎

To relate $f^\star$ and $\tilde{f}$, we exploit smoothness of regularized ERM with respect to small changes in the RKHS itself. We begin with a technical lemma that we will use to exploit the convexity of the regularized empirical risk functional; it shows a kind of converse to Remark 7, that functions with close risks are themselves close in proximity.

Lemma 16 *Let R be a functional on Hilbert space $\mathcal{H}$ satisfying $R[f] \geq R[f^\star] + \frac{a}{2}\|f - f^\star\|_{\mathcal{H}}^2$ for some $a > 0$, $f^\star \in \mathcal{H}$ and all $f \in \mathcal{H}$. Then $R[f] \leq R[f^\star] + \epsilon$ implies $\|f - f^\star\|_{\mathcal{H}} \leq \sqrt{\frac{2\epsilon}{a}}$, for all $\epsilon > 0$, $f \in \mathcal{H}$.*

Proof. By assumption and the antecedent $\|f - f^\star\|_{\mathcal{H}}^2 \leq \frac{2}{a}\left(R[f] - R[f^\star]\right) \leq \frac{2}{a}\left(R[f^\star] + \epsilon - R[f^\star]\right) = \frac{2\epsilon}{a}$. Taking square roots of both sides yields the result. ∎

Provided that the kernels $k, \hat{k}$ are uniformly close, we now show that $f^\star$ and $\tilde{f}$ are pointwise close, using insensitivity of regularized ERM to feature mapping perturbation.

Lemma 17 *Let $\mathcal{H}$ be an RKHS with translation-invariant kernel k, and let $\hat{\mathcal{H}}$ be the random RKHS corresponding to feature map (6) induced by k. Let C be a positive scalar and loss $\ell(y, \hat{y})$ be convex and L-Lipschitz continuous in $\hat{y}$. Consider the regularized empirical risk minimizers in each RKHS, where $R_{\text{emp}}[f] = n^{-1} \sum_{i=1}^n \ell\left(y_i, f(\mathbf{x}_i)\right)$,*

$$\begin{aligned} f^\star &\in \arg\min_{f \in \mathcal{H}} C R_{\text{emp}}[f] + \frac{1}{2}\|f\|_{\mathcal{H}}^2 , \\ g^\star &\in \arg\min_{g \in \hat{\mathcal{H}}} C R_{\text{emp}}[g] + \frac{1}{2}\|g\|_{\hat{\mathcal{H}}}^2 . \end{aligned}$$

Let $\mathcal{M} \subseteq \mathbb{R}^d$ be any set containing $\mathbf{x}_1, \ldots, \mathbf{x}_n$. For any $\epsilon > 0$, if the dual variables from both optimizations have L_1-norms bounded by some $\Lambda > 0$ and $\left\|k - \hat{k}\right\|_{\infty;\mathcal{M}} \leq \min\left\{1, \frac{\epsilon^2}{2^2\left(\Lambda + 2\sqrt{(CL+\Lambda/2)\Lambda}\right)^2}\right\}$ then $\|f^\star - g^\star\|_{\infty;\mathcal{M}} \leq \epsilon/2$.

Proof. Define regularized empirical risk functional $R_{\text{reg}}[f] = C\, R_{\text{emp}}[f] + \|f\|^2/2$, for the appropriate RKHS norm. Let minimizer $f^\star \in \mathcal{H}$ be given by parameter vector $\boldsymbol{\alpha}^\star$, and let minimizer $g^\star \in \hat{\mathcal{H}}$ be given by parameter vector $\boldsymbol{\beta}^\star$. Let $g_{\alpha^\star} = \sum_{i=1}^n \alpha_i^\star y_i \hat{\phi}(\mathbf{x}_i)$ and $f_{\boldsymbol{\beta}^\star} = \sum_{i=1}^n \beta_i^\star y_i \phi(\mathbf{x}_i)$ denote the images of $f^\star$ and $g^\star$ under the natural mapping between the spans of the data in RKHS's $\hat{\mathcal{H}}$ and $\mathcal{H}$, respectively. We will first show that these four functions have arbitrarily close regularized empirical risk in their respective RKHS, and then that this implies uniform proximity of the functions themselves. Observe that for any $g \in \hat{\mathcal{H}}$

$$\begin{aligned} R_{\text{reg}}^{\hat{\mathcal{H}}}[g] &= C\, R_{\text{emp}}[g] + \frac{1}{2}\|g\|_{\hat{\mathcal{H}}}^2 \\ &\geq C\langle \partial_g R_{\text{emp}}[g^\star], g - g^\star\rangle_{\hat{\mathcal{H}}} + C\, R_{\text{emp}}[g^\star] + \frac{1}{2}\|g\|_{\hat{\mathcal{H}}}^2 \\ &= \langle \partial_g R_{\text{reg}}^{\hat{\mathcal{H}}}[g^\star], g - g^\star\rangle_{\hat{\mathcal{H}}} - \langle g^\star, g - g^\star\rangle_{\hat{\mathcal{H}}} + C\, R_{\text{emp}}[g^\star] + \frac{1}{2}\|g\|_{\hat{\mathcal{H}}}^2 . \end{aligned}$$

The inequality follows from the convexity of $R_{\text{emp}}[\cdot]$ and holds for all elements of the subdifferential $\partial_g R_{\text{emp}}[g^\star]$. The subsequent equality holds by $\partial_g R_{\text{reg}}^{\hat{\mathcal{H}}}[g] = C\, \partial_g R_{\text{emp}}[g] + g$. Now since $\mathbf{0} \in \partial_g R_{\text{reg}}^{\hat{\mathcal{H}}}[g^\star]$, it follows that

$$\begin{aligned} R_{\text{reg}}^{\hat{\mathcal{H}}}[g] &\geq C\, R_{\text{emp}}[g^\star] + \frac{1}{2}\|g\|_{\hat{\mathcal{H}}}^2 - \langle g^\star, g - g^\star\rangle_{\hat{\mathcal{H}}} \\ &= R_{\text{reg}}^{\hat{\mathcal{H}}}[g^\star] + \frac{1}{2}\|g\|_{\hat{\mathcal{H}}}^2 - \langle g^\star, g\rangle_{\hat{\mathcal{H}}} + \frac{1}{2}\|g^\star\|_{\hat{\mathcal{H}}}^2 \\ &= R_{\text{reg}}^{\hat{\mathcal{H}}}[g^\star] + \frac{1}{2}\|g - g^\star\|_{\hat{\mathcal{H}}}^2 . \end{aligned}$$

With this, Lemma 16 states that for any $g \in \hat{\mathcal{H}}$ and $\epsilon' > 0$,

$$R_{\text{reg}}^{\hat{\mathcal{H}}}[g] \leq R_{\text{reg}}^{\hat{\mathcal{H}}}[g^\star] + \epsilon' \quad \Rightarrow \quad \|g - g^\star\|_{\hat{\mathcal{H}}} \leq \sqrt{2\epsilon'} . \tag{7}$$

Next we will show that the antecedent is true for $g = g_{\boldsymbol{\alpha}^\star}$. Conditioned on $\left\{\left\|k - \hat{k}\right\|_{\infty;\mathcal{M}} \leq \epsilon'\right\}$, for all $\mathbf{x} \in \mathcal{M}$

$$\begin{aligned} |f^\star(\mathbf{x}) - g_{\boldsymbol{\alpha}^\star}(\mathbf{x})| &= \left|\sum_{i=1}^n \alpha_i^\star y_i \left(k(\mathbf{x}_i, \mathbf{x}) - \hat{k}(\mathbf{x}_i, \mathbf{x})\right)\right| \\ &\leq \sum_{i=1}^n |\alpha_i^\star| \left|k(\mathbf{x}_i, \mathbf{x}) - \hat{k}(\mathbf{x}_i, \mathbf{x})\right| \\ &\leq \epsilon' \|\boldsymbol{\alpha}^\star\|_1 \\ &\leq \epsilon'\Lambda , \end{aligned} \tag{8}$$

by the bound on $\|\boldsymbol{\alpha}^\star\|_1$. This and the Lipschitz continuity of the loss lead to

$$\begin{aligned}
\left|R^{\mathcal{H}}_{\text{reg}}[f^\star] - R^{\hat{\mathcal{H}}}_{\text{reg}}[g_{\boldsymbol{\alpha}^\star}]\right| &= \left|C\,R_{\text{emp}}[f^\star] - C\,R_{\text{emp}}[g_{\boldsymbol{\alpha}^\star}] + \frac{1}{2}\|f^\star\|^2_{\mathcal{H}} - \frac{1}{2}\|g_{\boldsymbol{\alpha}^\star}\|^2_{\hat{\mathcal{H}}}\right| \\
&\leq \frac{C}{n}\sum_{i=1}^{n}\left|\ell\left(y_i, f^\star(\mathbf{x}_i)\right) - \ell\left(y_i, g_{\boldsymbol{\alpha}^\star}(\mathbf{x}_i)\right)\right| \\
&\quad + \frac{1}{2}\left|\boldsymbol{\alpha}^{\star\prime}\left(\mathbf{K} - \hat{\mathbf{K}}\right)\boldsymbol{\alpha}^\star\right| \\
&\leq CL\,\|f^\star - g_{\boldsymbol{\alpha}^\star}\|_{\infty;\mathcal{M}} + \frac{1}{2}\,\|\boldsymbol{\alpha}^\star\|_1\left\|\left(\mathbf{K} - \hat{\mathbf{K}}\right)\boldsymbol{\alpha}^\star\right\|_\infty \\
&\leq CL\epsilon'\Lambda + \Lambda^2\epsilon'/2 \\
&= \left(CL + \frac{\Lambda}{2}\right)\Lambda\epsilon'\,.
\end{aligned}$$

Similarly,

$$\left|R^{\hat{\mathcal{H}}}_{\text{reg}}[g^\star] - R^{\mathcal{H}}_{\text{reg}}[f_{\boldsymbol{\beta}^\star}]\right| \leq (CL + \Lambda/2)\Lambda\epsilon'$$

by the same argument. And since $R^{\mathcal{H}}_{\text{reg}}[f_{\boldsymbol{\beta}^\star}] \geq R^{\mathcal{H}}_{\text{reg}}[f^\star]$ and $R^{\hat{\mathcal{H}}}_{\text{reg}}[g_{\boldsymbol{\alpha}^\star}] \geq R^{\hat{\mathcal{H}}}_{\text{reg}}[g^\star]$ we have proved that

$$\begin{aligned}
R^{\hat{\mathcal{H}}}_{\text{reg}}[g_{\boldsymbol{\alpha}^\star}] &\leq R^{\mathcal{H}}_{\text{reg}}[f^\star] + (CL + \Lambda/2)\Lambda\epsilon' \\
&\leq R^{\mathcal{H}}_{\text{reg}}[f_{\boldsymbol{\beta}^\star}] + (CL + \Lambda/2)\Lambda\epsilon' \\
&\leq R^{\hat{\mathcal{H}}}_{\text{reg}}[g^\star] + 2(CL + \Lambda/2)\Lambda\epsilon'.
\end{aligned}$$

And by implication (7),

$$\|g_{\boldsymbol{\alpha}^\star} - g^\star\|_{\hat{\mathcal{H}}} \leq 2\sqrt{\left(CL + \frac{\Lambda}{2}\right)\Lambda\epsilon'}\,. \tag{9}$$

Now $\hat{k}(\mathbf{x}, \mathbf{x}) = 1$ for each $\mathbf{x} \in \mathbb{R}^d$ implies

$$\begin{aligned}
|g_{\boldsymbol{\alpha}^\star}(\mathbf{x}) - g^\star(\mathbf{x})| &= \left\langle g_{\boldsymbol{\alpha}^\star} - g^\star, \hat{k}(\mathbf{x}, \cdot)\right\rangle_{\hat{\mathcal{H}}} \\
&\leq \|g_{\boldsymbol{\alpha}^\star} - g^\star\|_{\hat{\mathcal{H}}}\sqrt{\hat{k}(\mathbf{x}, \mathbf{x})} \\
&= \|g_{\boldsymbol{\alpha}^\star} - g^\star\|_{\hat{\mathcal{H}}}\,.
\end{aligned}$$

This combines with Inequality (9) to yield $\|g_{\boldsymbol{\alpha}^\star} - g^\star\|_{\infty;\mathcal{M}} \leq 2\sqrt{\left(CL + \frac{\Lambda}{2}\right)\Lambda\epsilon'}$. Together with Inequality (8) this finally implies that $\|f^\star - g^\star\|_{\infty;\mathcal{M}} \leq \epsilon'\Lambda + 2\sqrt{(CL + \Lambda/2)\,\Lambda\epsilon'}$, conditioned on event $P_{\epsilon'} = \left\{\left\|k - \hat{k}\right\|_\infty \leq \epsilon'\right\}$. For desired accuracy $\epsilon > 0$, conditioning on event $P_{\epsilon'}$ with $\epsilon' =$

$\min\left\{\epsilon/\left[2\left(\Lambda+2\sqrt{(CL+\Lambda/2)\,\Lambda}\right)\right],\ \epsilon^2/\left[2\left(\Lambda+2\sqrt{(CL+\Lambda/2)\,\Lambda}\right)\right]^2\right\}$ yields bound $\|f^\star-g^\star\|_{\infty;\mathcal{M}}\leq\epsilon/2$: if $\epsilon'\leq 1$ then $\epsilon/2\geq\sqrt{\epsilon'}\left(\Lambda+2\sqrt{(CL+\Lambda/2)\,\Lambda}\right)\geq\epsilon'\Lambda+2\sqrt{(CL+\Lambda/2)\,\Lambda\epsilon'}$ provided that $\epsilon'\leq\epsilon^2/\left[2\left(\Lambda+2\sqrt{(CL+\Lambda/2)\,\Lambda}\right)\right]^2$. Otherwise if $\epsilon'>1$ then we have $\epsilon/2\geq\epsilon'\left(\Lambda+2\sqrt{(CL+\Lambda/2)\,\Lambda}\right)\geq\epsilon'\Lambda+2\sqrt{(CL+\Lambda/2)\,\Lambda\epsilon'}$ provided $\epsilon'\leq\epsilon/\left[2\left(\Lambda+2\sqrt{(CL+\Lambda/2)\,\Lambda}\right)\right]$. Since for any $H>0$, $\min\left\{H,H^2\right\}\geq\min\left\{1,H^2\right\}$, the result follows. ■

We now recall the result due to Rahimi and Recht (2008) that establishes the non-asymptotic uniform convergence of the kernel functions required by the previous Lemma (i.e., an upper bound on the probability of event $P_{\epsilon'}$).

Lemma 18 (Rahimi and Recht 2008, Claim 1) *For any $\epsilon>0$, $\delta\in(0,1)$, translation-invariant kernel k and compact set $\mathcal{M}\subset\mathbb{R}^d$, if $\hat{d}\geq\frac{4(d+2)}{\epsilon^2}\log_e\left(\frac{2^8(\sigma_p\mathrm{diam}(\mathcal{M}))^2}{\delta\epsilon^2}\right)$, then Algorithm 3's random feature mapping $\hat{\phi}$ defined in Equation (6) satisfies $\Pr\left(\left\|\hat{k}-k\right\|_\infty<\epsilon\right)\geq 1-\delta$, where $\sigma_p^2=\mathbb{E}\left[\langle\boldsymbol{\omega},\boldsymbol{\omega}\rangle\right]$ is the second moment of the Fourier transform p of k's g function.*

Combining these ingredients establishes utility for PrivateSVM.

Proof of Theorem 13. Lemma 17 and Corollary 15 combined via the triangle inequality with Lemma 18 together establish the result as follows. Define P to be the conditioning event regarding the approximation of k by $\hat{k}$, denote the events in Lemma's 17 and 11 by Q and R, and the target event in the theorem by S.

$$P = \left\{\left\|\hat{k}-k\right\|_{\infty;\mathcal{M}}<\min\left\{1,\frac{\epsilon^2}{2^2\left(\Lambda+2\sqrt{\left(CL+\frac{\Lambda}{2}\right)\Lambda}\right)^2}\right\}\right\}$$

$$\begin{aligned} Q &= \left\{\left\|f^\star-\tilde{f}\right\|_{\infty;\mathcal{M}}\leq\frac{\epsilon}{2}\right\} \\ R &= \left\{\left\|\hat{f}^\star-\tilde{f}\right\|_\infty\leq\frac{\epsilon}{2}\right\} \\ S &= \left\{\left\|f^\star-\hat{f}^\star\right\|_{\infty;\mathcal{M}}\leq\epsilon\right\}. \end{aligned}$$

The claim is a bound on $\Pr(S)$. By the triangle inequality, events Q and R together imply S. Second, note that event R is independent of P and Q. Thus $\Pr(S\mid P)\geq\Pr(Q\cap R\mid P)=\Pr(Q\mid P)\Pr(R)\geq 1\cdot(1-\delta/2)$, for sufficiently small λ. Finally, Lemma 18 bounds $\Pr(P)$: provided that $\hat{d}\geq 4(d+2)\log_e\left(2^9\left(\sigma_p\mathrm{diam}(\mathcal{M})\right)^2/\left(\delta\theta(\epsilon)\right)\right)/\theta(\epsilon)$ where

$\theta(\epsilon) = \min\left\{1, \epsilon^4 / \left[2\left(\Lambda + 2\sqrt{(CL + \Lambda/2)\,\Lambda}\right)\right]^4\right\}$ we have $\Pr(P) \geq 1 - \delta/2$. Together this yields $\Pr(S) = \Pr(S \mid P)\Pr(P) \geq (1 - \delta/2)^2 \geq 1 - \delta$. ■

5 Hinge Loss & Upper Bounds on Optimal Differential Privacy

We now present a short case study on using the above analysis for the hinge loss. We begin by plugging hinge loss $\ell(y, \hat{y}) = (1 - y\hat{y})_+$ into the main results on privacy and utility of the previous sections. Similar computations can be done for other convex losses; we select hinge loss for this example as it is the most common among SVM classification losses. We then proceed to combine the obtained privacy and utility bounds into an upper bound on the optimal differential privacy for SVM learning with the hinge loss. Again, the details of this second step are not specific to the hinge loss, however we are motivated by comparing positive results with the lower bounds for SVM learning with the hinge loss in the next section.

Combining Theorems 10 and 11 immediately establishes the upper bound on the optimal differential privacy $\beta^\star$ for mechanisms achieving a given desired level (ϵ, δ) of usefulness.

Corollary 19 *The optimal differential privacy $\beta^\star$ among all mechanisms that are (ϵ, δ)-useful with respect to the* SVM *with finite F-dimensional feature mapping inducing bounded norms $k(\mathbf{x}, \mathbf{x}) \leq \kappa^2$ and $\|\phi(\mathbf{x})\|_\infty \leq \Phi$ for all $\mathbf{x} \in \mathbb{R}^d$, hinge loss, parameter $C > 0$, on n training, is at most*

$$\begin{aligned} \beta^\star &\leq \frac{8\kappa\Phi C\left(F \log_e 2 + \log_e \frac{1}{\delta}\right)}{n\epsilon} \\ &= O\left(\frac{C}{\epsilon n} \log \frac{1}{\delta}\right) . \end{aligned}$$

Proof. The proof is a straightforward calculation for general L-Lipschitz loss. In the general case the bound has numerator leading coefficient $8\kappa\Phi CL$. The result then follows from the fact that hinge loss is 1-Lipschitz on $\mathbb{R}$: i.e., $\partial_{\hat{y}}\ell = \mathbf{1}[1 \geq y\hat{y}] \leq 1$. ■

Observe that $\Phi \geq \kappa/\sqrt{F}$, so κ could be used in place of Φ to simplify the result's statement, however, doing so would yield a slightly looser bound. Also note that by this result, if we set $C = \sqrt{n}$ (needed for universal consistency, cf. Remark 1) and fix β and δ, then the error due to preserving privacy is on the same order as the error in estimating the "true" parameter $\mathbf{w}$.

Recall the dual program for learning under hinge loss from Section 2 repeated here

for convenience:

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^n} \quad \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j k(\mathbf{x}_i,\mathbf{x}_j) \tag{10}$$
$$\text{s.t.} \quad 0 \le \alpha_i \le \frac{C}{n}\ \forall i \in [n]\ .$$

We split the calculation of the upper bound for the translation-invariant kernel case into the following two steps as they are slightly more involved than the finite-dimensional feature mapping case.

Corollary 20 *Consider any database D of size n, scalar $C > 0$, and translation-invariant kernel k. For any $\beta > 0$ and $\hat{d} \in \mathbb{N}$,* PrivateSVM *run on D with hinge loss, noise parameter $\lambda \geq \frac{2^{2.5}C\sqrt{\hat{d}}}{\beta n}$, approximation parameter $\hat{d}$, and regularization parameter C, guarantees β-differential privacy. Moreover for any compact set $\mathcal{M} \subset \mathbb{R}^d$ containing D, and scalars $\epsilon > 0$ and $\delta \in (0,1)$,* PrivateSVM *run on D with hinge loss, kernel k, noise parameter $\lambda \leq \min\left\{\frac{\epsilon}{2^4\log_e 2\sqrt{\hat{d}}}, \frac{\epsilon\sqrt{\hat{d}}}{8\log_e\frac{2}{\delta}}\right\}$, approximation parameter $\hat{d} \geq \frac{4(d+2)}{\theta(\epsilon)}\log_e\left(\frac{2^9(\sigma_p \mathrm{diam}(\mathcal{M}))^2}{\delta\theta(\epsilon)}\right)$ with $\theta(\epsilon) = \min\left\{1, \frac{\epsilon^4}{2^{12}C^4}\right\}$, and parameter C, is (ϵ,δ)-useful with respect to hinge loss* SVM *run on D with kernel k and parameter C.*

Proof. The first result follows from Theorem 10 and the fact that hinge loss is convex and 1-Lipschitz on $\mathbb{R}$ (as justified in the proof of Corollary 19). The second result follows almost immediately from Theorem 13. For hinge loss we have that feasible α_i's are bounded by C/n (and so $\Lambda = C$) by the dual's box constraints and that $L = 1$, implying we take $\theta(\epsilon) = \min\left\{1, \frac{\epsilon^4}{2^4C^4\left(1+\sqrt{6}\right)^4}\right\}$. This is bounded by the stated $\theta(\epsilon)$. ■

Combining the competing requirements on λ upper-bounds optimal differential privacy of hinge loss SVM .

Theorem 21 *The optimal differential privacy for hinge loss* SVM *learning on translation-invariant kernel k is bounded by $\beta^\star(\epsilon,\delta,C,n,\ell,k) = O\left(\frac{C}{\epsilon^3 n}\log^{1.5}\frac{C}{\delta\epsilon}\right)$.*

Proof. Consider hinge loss in Corollary 20. Privacy places a lower bound of $\beta \geq 2^{2.5}C\sqrt{\hat{d}}/(\lambda n)$ for any chosen λ, which we can convert to a lower bound on β in terms of ϵ and δ as follows. For small ϵ, we have $\theta(\epsilon) = O(\epsilon^4/C^4)$ and so to achieve (ϵ,δ)-usefulness we must take $\hat{d} = \Omega\left(\frac{1}{\epsilon^4}\log_e\left(\frac{C^4}{\delta\epsilon^4}\right)\right)$. There are two cases for utility. The

first case is with $\lambda = \epsilon / \left(2^4 \log_e \left(2\sqrt{\hat{d}}\right)\right)$, yielding

$$\begin{aligned}
\beta &= O\left(\frac{C\sqrt{\hat{d}}\log\sqrt{\hat{d}}}{\epsilon n}\right) \\
&= O\left(\frac{C}{\epsilon^3 n}\sqrt{\log\frac{C}{\delta\epsilon}}\left(\log\frac{1}{\epsilon} + \log\log\frac{C}{\delta\epsilon}\right)\right) \\
&= O\left(\frac{C}{\epsilon^3 n}\log^{1.5}\frac{C}{\delta\epsilon}\right) \quad .
\end{aligned}$$

In the second case, $\lambda = \frac{\epsilon\sqrt{\hat{d}}}{8\log_e\frac{2}{\delta}}$ yields $\beta = O\left(\frac{C}{\epsilon n}\log\frac{1}{\delta}\right)$ which is dominated by the first case as $\epsilon \downarrow 0$. ■

A natural question arises from this discussion: given any mechanism that is (ϵ, δ)-useful with respect to hinge SVM , for how small a β can we possibly hope to guarantee β-differential privacy? In other words, what lower bounds exist for the optimal differential privacy for the SVM?

6 Lower-Bounding Optimal Differential Privacy

In this section we present lower bounds on the level of differential privacy for any mechanism approximating SVMs with high accuracy. We begin with a lower bound in Theorem 23 for approximating hinge loss linear SVMs. We then apply an extension of the proof technique from this lower bound to produce a lower bound for private SVM learning with the Radial Basis Function kernel in Theorem 26.

6.1 Private SVM Learning with the Linear Kernel

The following lemma establishes a negative sensitivity result for the SVM mechanism run with the hinge loss and linear kernel.

Lemma 22 *For any $C > 0$ and $n > 1$, there exists a pair of neighboring databases D_1, D_2 on n entries, such that the functions $f_1^\star, f_2^\star$ parametrized by SVM run with parameter C, linear kernel, and hinge loss on D_1, D_2, respectively, satisfy $\|f_1^\star - f_2^\star\|_\infty > \frac{\sqrt{C}}{n}$.*

Proof. We construct the two databases on the line as follows. Let $0 < m < M$ be scalars to be chosen later. Both databases share negative examples $x_1 = \ldots = x_{\lfloor n/2 \rfloor} = -M$ and positive examples $x_{\lfloor n/2 \rfloor + 1} = \ldots = x_{n-1} = M$. Each database has $x_n = M - m$, with $y_n = -1$ for D_1 and $y_n = 1$ for D_2. In what follows we use subscripts to denote an example's parent database, so $(x_{i,j}, y_{i,j})$ is the j^{th} example from

D_i. Consider the result of running primal SVM on each database:

$$\begin{aligned} w_1^\star &= \arg\min_{w\in\mathbb{R}} \frac{1}{2}w^2 + \frac{C}{n}\sum_{i=1}^{n}(1-y_{1,i}wx_{1,i})_+ \\ w_2^\star &= \arg\min_{w\in\mathbb{R}} \frac{1}{2}w^2 + \frac{C}{n}\sum_{i=1}^{n}(1-y_{2,i}wx_{2,i})_+ \ . \end{aligned}$$

Each optimization is strictly convex and unconstrained, so the optimizing $w_1^\star, w_2^\star$ are characterized by the first-order KKT conditions $0 \in \partial_w f_i(w)$ for f_i being the objective function for learning on D_i, and ∂_w denoting the subdifferential operator. Now for each $i \in [2]$

$$\partial_w f_i(w) = w - \frac{C}{n}\sum_{j=1}^{n} y_{i,j}x_{i,j}\tilde{\mathbf{1}}\left[1-y_{i,j}wx_{i,j}\right] \ ,$$

where

$$\tilde{\mathbf{1}}[x] = \begin{cases} \{0\} \ , & \text{if } x<0 \\ [0,1] \ , & \text{if } x=0 \\ \{1\} \ , & \text{if } x>0 \end{cases}$$

is the subdifferential of $(x)_+$.

Thus for each $i \in [2]$, we have that $w_i^\star \in \frac{C}{n}\sum_{j=1}^{n} y_{i,j}x_{i,j}\tilde{\mathbf{1}}\left[1-y_{i,j}w_i^\star x_{i,j}\right]$ which is equivalent to

$$\begin{aligned} w_1^\star &\in \frac{CM(n-1)}{n}\tilde{\mathbf{1}}\left[\frac{1}{M}-w_1^\star\right] + \frac{C(m-M)}{n}\tilde{\mathbf{1}}\left[w_1^\star - \frac{1}{m-M}\right] \\ w_2^\star &\in \frac{CM(n-1)}{n}\tilde{\mathbf{1}}\left[\frac{1}{M}-w_2^\star\right] + \frac{C(M-m)}{n}\tilde{\mathbf{1}}\left[\frac{1}{M-m}-w_2^\star\right] \ . \end{aligned}$$

The RHSs of these conditions correspond to decreasing piecewise-constant functions, and the conditions are met when the corresponding functions intersect with the diagonal $y = x$ line, as shown in Figure 1. If $\frac{C(M(n-2)+m)}{n} < \frac{1}{M}$ then $w_1^\star = \frac{C(M(n-2)+m)}{n}$. And if $\frac{C(Mn-m)}{n} < \frac{1}{M}$ then $w_2^\star = \frac{C(Mn-m)}{n}$. So provided that $\frac{1}{M} > \frac{C(Mn-m)}{n} = \max\left\{\frac{C(M(n-2)+m)}{n}, \frac{C(Mn-m)}{n}\right\}$, we have $|w_1^\star - w_2^\star| = \frac{2C}{n}|M-m|$. So taking $M = \frac{2n\epsilon}{C}$ and $m = \frac{n\epsilon}{C}$, this implies

$$\begin{aligned} \|f_1^\star - f_2^\star\|_\infty &\geq |f_1^\star(1) - f_2^\star(1)| \\ &= |w_1^\star - w_2^\star| \\ &= 2\epsilon \ , \end{aligned}$$

provided $\epsilon < \frac{\sqrt{C}}{2n}$. In particular taking $\epsilon = \frac{\sqrt{C}}{2n}$ yields the result. ■

With this negative sensitivity result in hand, we can prove the following lower bound on the optimal differential privacy for any mechanism approximating the SVM with hinge loss.

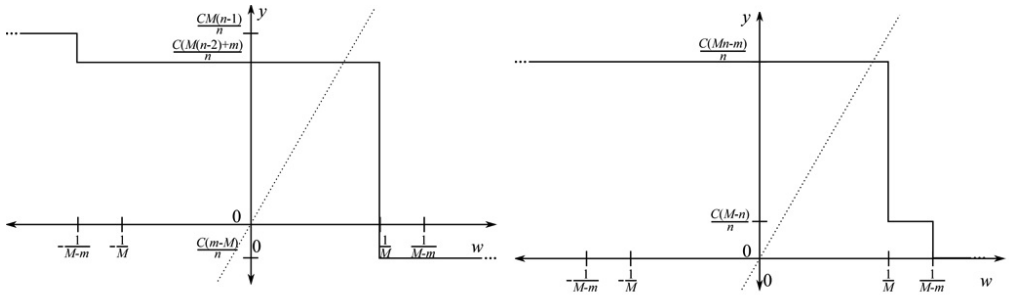

Figure 1: For each $i \in [2]$, the SVM's primal solution $w_i^\star$on database D_i constructed in the proof of Lemma 22, corresponds to the crossing point of line $y = w$ with $y = w - \partial_w f_i(w)$. Database D_1 is shown on the left, database D_2 is shown on the right.

Theorem 23 (Lower bound on optimal differential privacy for linear SVM) *For any $C > 0$, $n > 1$, $\delta \in (0,1)$, and $\epsilon \in \left(0, \frac{\sqrt{C}}{2n}\right)$, the optimal differential privacy for the hinge loss* SVM *with linear kernel is lower bounded by* $\log_e \frac{1-\delta}{\delta} = \Omega(\log \frac{1}{\delta})$.

Proof. Consider (ϵ, δ)-useful mechanism $\hat{M}$ with respect to SVM learning mechanism M with parameter $C > 0$, hinge loss, and linear kernel on n training examples, where $\delta > 0$ and $\frac{\sqrt{C}}{2n} > \epsilon > 0$. By Lemma 22 there exists a pair of neighboring databases D_1, D_2 on n entries, such that $\|f_1^\star - f_2^\star\|_\infty > 2\epsilon$ where $f_i^\star = f_{M(D_i)}$ for each $i \in [2]$. Let $\hat{f}_i = f_{\hat{M}(D_i)}$ for each $i \in [2]$. Then by the utility of $\hat{M}$,

$$\Pr\left(\hat{f}_1 \in \mathcal{B}_\epsilon^\infty\left(f_1^\star\right)\right) \;\geq\; 1 - \delta\,, \tag{11}$$

$$\Pr\left(\hat{f}_2 \in \mathcal{B}_\epsilon^\infty\left(f_1^\star\right)\right) \;\leq\; \Pr\left(\hat{f}_2 \notin \mathcal{B}_\epsilon^\infty\left(f_2^\star\right)\right) \;<\; \delta\,. \tag{12}$$

Let $\hat{\mathcal{P}}_1$ and $\hat{\mathcal{P}}_2$ be the distributions of $\hat{M}(D_1)$ and $\hat{M}(D_2)$, respectively, so that $\hat{\mathcal{P}}_i(t) = \Pr\left(\hat{M}(D_i) = t\right)$. Then by Inequalities (11) and (12),

$$\mathbb{E}_{T\sim\mathcal{P}_1}\left[\frac{d\mathcal{P}_2(T)}{d\mathcal{P}_1(T)}\,\middle|\, T \in \mathcal{B}_\epsilon^\infty\left(f_1^\star\right)\right] \;=\; \frac{\int_{\mathcal{B}_\epsilon^\infty(f_1^\star)} \frac{d\mathcal{P}_2(t)}{d\mathcal{P}_1(t)} d\mathcal{P}_1(t)}{\int_{\mathcal{B}_\epsilon^\infty(f_1^\star)} d\mathcal{P}_1(t)} \;\leq\; \frac{\delta}{1-\delta}\,.$$

Thus there exists a t such that $\log \frac{\Pr\left(\hat{M}(D_1)=t\right)}{\Pr\left(\hat{M}(D_2)=t\right)} \geq \log \frac{1-\delta}{\delta} = \Omega(\log \frac{1}{\delta})$. ∎

Remark 24 *Equivalently this result can be written as follows. For any $C > 0, \beta > 0$, and $n > 1$, if a mechanism $\hat{M}$ is (ϵ, δ)-useful and β-differentially private then either $\epsilon \geq \frac{\sqrt{C}}{2n}$ or $\delta \geq \exp(-\beta)$.*

We have now presented both upper and lower bounds on the optimal differential privacy for the case of the linear SVM with hinge loss, as the upper bound for this case is covered by Corollary 19 where we can take $L = 1$ for the hinge loss. Ignoring constants and using the scaling of C (cf. Remark 1) we have that

$$\Omega\left(\log\frac{1}{\delta}\right) \;=\; \beta^\star \;=\; O\left(\frac{1}{\epsilon\sqrt{n}}\log\frac{1}{\delta}\right).$$

It is noteworthy that the bounds agree in their scaling on utility confidence δ but that they disagree on linear and square-root terms in their dependence on ϵ and n, respectively. Moreover under the appropriate scaling of C, the lower bound holds only for $\epsilon = O\left(n^{-0.75}\right)$, under which the upper asymptotic bound becomes $O\left(n^{0.25}\log(1/\delta)\right)$. Finding better-matching bounds remains an interesting open problem.

6.2 Private SVM Learning with the RBF Kernel

We now turn to lower bounding the level β of differential privacy achievable for any (ϵ, δ)-useful mechanism approximating an SVM equipped with an RBF kernel. To do so we first state a negative sensitivity result for the SVM. While the analogous result of the previous section is witnessed by a pair of neighboring databases on which the SVM produces very different results, here we construct a sequence of N pairwise-neighboring databases whose images under SVM learning form an ϵ-packing. Indeed the RBF kernel is key to achieving such a packing for any N.

The *Radial Basis Function* (a.k.a. the Gaussian) kernel as given in Table 1, corresponds to a mapping ϕ with infinite-dimensional range space in which all points have norm one. It is one of the most popular non-linear kernels in practice (it is the default kernel in the popular `libsvm` package [Chang and Lin, 2001]). It is of particular interest to study private learning with the RBF kernel, particularly the effect of the kernel's hyperparameter (the variance, or kernel width) on the lower bound.

Lemma 25 *For any $C > 0$, $n > C$, $0 < \epsilon < \frac{C}{4n}$, and $0 < \sigma < \sqrt{\frac{1}{2\log_e 2}}$, there exists a set of $N = \left\lfloor \frac{2}{\sigma}\sqrt{\frac{2}{\log_e 2}} \right\rfloor$ pairwise-neighboring databases $\{D_i\}_{i=1}^N$ on n examples, such that the functions $f_i^\star$ parametrized by hinge loss* SVM *run on D_i with parameter C and RBF kernel with parameter σ, satisfy $\left\|f_i^\star - f_j^\star\right\|_\infty > 2\epsilon$ for $i \neq j$.*

Proof. Construct $N > 1$ pairwise-neighboring databases each on n examples in $\mathbb{R}^2$ as follows. Each database i has $n-1$ negative examples $\mathbf{x}_{i,1} = \ldots = \mathbf{x}_{i,n-1} = \mathbf{0}$, and database D_i has positive example $\mathbf{x}_{i,n} = (\cos\theta_i, \sin\theta_i)$, where $\theta_i = \frac{2\pi i}{N}$. Consider the result of running SVM with hinge loss and RBF kernel on each D_i. For each database $k(\mathbf{x}_{i,s}, \mathbf{x}_{i,t}) = 1$ and $k(\mathbf{x}_{i,s}, \mathbf{x}_{i,n}) = \exp\left(-\frac{1}{2\sigma^2}\right) =: \gamma$ for all $s, t \in [n-1]$. Notice that the range space of γ is $(0, 1)$. Since the inner products and labels are database-independent,

the SVM dual variables are also database-independent. Each involves solving

$$\begin{array}{ll} \max\limits_{\boldsymbol{\alpha}\in\mathbb{R}^n} & \boldsymbol{\alpha}'\mathbf{1} - \frac{1}{2}\boldsymbol{\alpha}'\begin{pmatrix} 1 & -\gamma \\ -\gamma & 1 \end{pmatrix}\boldsymbol{\alpha} \\ \text{s.t.} & \mathbf{0} \leq \boldsymbol{\alpha} \leq \frac{C}{n}\mathbf{1}. \end{array}$$

This reduces to the equivalent two-variable program by symmetry $\alpha_1^\star = \ldots = \alpha_{n-1}^\star$

$$\begin{array}{ll} \max\limits_{\boldsymbol{\alpha}\in\mathbb{R}^2} & \boldsymbol{\alpha}'\begin{pmatrix} n-1 \\ 1 \end{pmatrix} - \frac{1}{2}\boldsymbol{\alpha}'\begin{pmatrix} (n-1)^2 & -\gamma(n-1) \\ -\gamma(n-1) & 1 \end{pmatrix}\boldsymbol{\alpha} \\ \text{s.t.} & \mathbf{0} \leq \boldsymbol{\alpha} \leq \frac{C}{n}\mathbf{1}. \end{array}$$

Consider first the unconstrained program, for which the necessary first-order KKT condition is that

$$\mathbf{0} = \begin{pmatrix} n-1 \\ 1 \end{pmatrix} - \begin{pmatrix} (n-1)^2 & -\gamma(n-1) \\ -\gamma(n-1) & 1 \end{pmatrix}\boldsymbol{\alpha}^\star .$$

This implies

$$\begin{aligned} \boldsymbol{\alpha}^\star &= \begin{pmatrix} (n-1)^2 & -\gamma(n-1) \\ -\gamma(n-1) & 1 \end{pmatrix}^{-1}\begin{pmatrix} n-1 \\ 1 \end{pmatrix} \\ &= \frac{1}{(n-1)^2(1-\gamma^2)}\begin{pmatrix} 1 & \gamma(n-1) \\ \gamma(n-1) & (n-1)^2 \end{pmatrix}\begin{pmatrix} n-1 \\ 1 \end{pmatrix} \\ &= \frac{1}{(n-1)^2(1-\gamma)(1+\gamma)}\begin{pmatrix} 1 & \gamma(n-1) \\ \gamma(n-1) & (n-1)^2 \end{pmatrix}\begin{pmatrix} n-1 \\ 1 \end{pmatrix} \\ &= \frac{1}{(n-1)^2(1-\gamma)(1+\gamma)}\begin{pmatrix} (n-1)(1+\gamma) \\ (n-1)^2(1+\gamma) \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{(n-1)(1-\gamma)} \\ \frac{1}{1-\gamma} \end{pmatrix} . \end{aligned}$$

Since this solution is strictly positive, it follows that at most two (upper) constraints can be active. Thus four cases are possible: the solution lies in the interior of the feasible set, or one or both upper box-constraints hold with equality. Noting that $\frac{1}{(n-1)(1-\gamma)} \leq \frac{1}{1-\gamma}$, it follows that $\boldsymbol{\alpha}^\star$ is feasible iff $\frac{1}{1-\gamma} \leq \frac{C}{n}$. This is equivalent to $C \geq \frac{1}{1-\gamma}n > n$, since $\gamma \in (0,1)$. This corresponds to under-regularization.

If both constraints hold with equality we have $\boldsymbol{\alpha}^\star = \frac{C}{n}\mathbf{1}$, which is always feasible.

In the case where the first constraint holds with equality $\alpha_1^\star = \frac{C}{n}$, the second dual

variable is found by optimizing

$$\begin{aligned}
\alpha_2^\star &= \max_{\alpha_2 \in \mathbb{R}} \boldsymbol{\alpha}' \begin{pmatrix} n-1 \\ 1 \end{pmatrix} - \frac{1}{2}\boldsymbol{\alpha}' \begin{pmatrix} (n-1)^2 & -\gamma(n-1) \\ -\gamma(n-1) & 1 \end{pmatrix} \boldsymbol{\alpha} \\
&= \max_{\alpha_2 \in \mathbb{R}} \frac{C(n-1)}{n} + \alpha_2 - \frac{1}{2}\left(\left(\frac{C(n-1)}{n} \right)^2 - 2\frac{C\gamma(n-1)}{n}\alpha_2 + \alpha_2^2 \right) \\
&= \max_{\alpha_2 \in \mathbb{R}} -\frac{1}{2}\alpha_2^2 + \alpha_2 \left(1 + \frac{C\gamma(n-1)}{n} \right) ,
\end{aligned}$$

implying $\alpha_2^\star = 1 + C\gamma\frac{n-1}{n}$. This solution is feasible provided $1 + C\gamma\frac{n-1}{n} \leq \frac{C}{n}$ iff $n \leq \frac{C(1+\gamma)}{1+C\gamma}$. Again this corresponds to under-regularization.

Finally in the case where the second constraint holds with equality $\alpha_2^\star = \frac{C}{n}$, the first dual is found by optimizing

$$\begin{aligned}
\alpha_2^\star &= \max_{\alpha_1 \in \mathbb{R}} \boldsymbol{\alpha}' \begin{pmatrix} n-1 \\ 1 \end{pmatrix} - \frac{1}{2}\boldsymbol{\alpha}' \begin{pmatrix} (n-1)^2 & -\gamma(n-1) \\ -\gamma(n-1) & 1 \end{pmatrix} \boldsymbol{\alpha} \\
&= \max_{\alpha_1 \in \mathbb{R}} (n-1)\alpha_1 + \frac{C}{n} - \frac{1}{2}\left((n-1)^2\alpha_1^2 - 2C\gamma\frac{n-1}{n}\alpha_1 + \frac{C^2}{n^2} \right) \\
&= \max_{\alpha_2 \in \mathbb{R}} -\frac{1}{2}(n-1)^2\alpha_1^2 + \alpha_1 \left(1 + \frac{C\gamma}{n} \right) ,
\end{aligned}$$

implying $\alpha_1^\star = \frac{1+\frac{C\gamma}{n}}{(n-1)^2}$. This is feasible provided $\frac{1+\frac{C\gamma}{n}}{(n-1)^2} \leq \frac{C}{n}$. Passing back to the program on n variables, by the invariance of the duals to the database, for any pair D_i, D_j

$$\begin{aligned}
|f_i(\mathbf{x}_{i,n}) - f_j(\mathbf{x}_{i,n})| &= \alpha_n^\star (1 - k(\mathbf{x}_{i,n}, \mathbf{x}_{j,n})) \\
&\geq \alpha_n^\star \left(1 - \max_{q \neq i} k(\mathbf{x}_{i,n}, \mathbf{x}_{q,n}) \right) .
\end{aligned}$$

Now a simple argument shows that this maximum is equal to $\gamma^4 \exp\left(\sin^2\frac{\pi}{N}\right)$ for all i. The maximum objective is optimized when $|q-i| = 1$. In this case $|\theta_i - \theta_q| = \frac{2\pi}{N}$. The norm $\|\mathbf{x}_{i,n} - \mathbf{x}_{q,n}\| = 2\sin\frac{|\theta_i - \theta_q|}{2} = 2\sin\frac{\pi}{N}$ by basic geometry. Thus $k(\mathbf{x}_{i,n}, \mathbf{x}_{q,n}) = \exp\left(-\frac{\|\mathbf{x}_{i,n} - \mathbf{x}_{q,n}\|^2}{2\sigma^2}\right) = \exp\left(-\frac{2}{\sigma^2}\sin^2\frac{\pi}{N}\right) = \gamma^4 \exp\left(\sin^2\frac{\pi}{N}\right)$ as claimed. Notice that $N \geq 2$ so the second term is in $(1, e]$, while the first term is in $(0, 1)$. In summary we have shown that for any $i \neq j$

$$|f_i(\mathbf{x}_{i,n}) - f_j(\mathbf{x}_{i,n})| \geq \left(1 - \exp\left(-\frac{2}{\sigma^2}\sin^2\frac{\pi}{N} \right) \right) \alpha_n^\star .$$

Assume $\gamma < \frac{1}{2}$. If $n > C$ then $n > \frac{C}{2} > (1-\gamma)C$, which implies case 1 is infeasible. Similarly since $C\gamma\frac{n-1}{n} > 0$, $n > C$ implies $1 + C\gamma\frac{n-1}{n} > 1 > \frac{C}{n}$ which implies case 3 is infeasible. Thus provided that $\gamma < \frac{1}{2}$ and $n > C$, we have that either case 2 or case 4

must hold. In both cases $\alpha_n^\star = \frac{C}{n}$ giving

$$|f_i(\mathbf{x}_{i,n}) - f_j(\mathbf{x}_{i,n})| \quad \geq \quad \left(1 - \exp\left(-\frac{2}{\sigma^2}\sin^2\frac{\pi}{N}\right)\right)\frac{C}{n}\,.$$

Provided that $\sigma \leq \sqrt{\frac{2}{\log 2}}\sin\frac{\pi}{N}$, we have $\left(1-\exp\left(-\frac{2}{\sigma^2}\sin^2\frac{\pi}{N}\right)\right)\frac{C}{n} \geq \left(1-\frac{1}{2}\right)\frac{C}{n} = \frac{C}{2n}$. Now, for small x we can take the linear approximation $\sin x \geq \frac{x}{\pi/2}$ for $x \in [0, \pi/2]$. If $N \geq 2$ then $\sin\frac{\pi}{N} \geq \frac{2}{N}$. Thus in this case we can take $\sigma \leq \sqrt{\frac{2}{\log 2}}\frac{2}{N}$ to imply $|f_i(\mathbf{x}_{i,n}) - f_j(\mathbf{x}_{i,n})| \geq \frac{C}{2n}$. This bound on σ in turn implies the following bound on γ: $\gamma = \exp\left(-\frac{1}{2\sigma^2}\right) \leq \exp\left(-\frac{N^2\log_e 2}{2^4}\right)$. Thus taking $N > 4$ in conjunction with $\sigma \leq \sqrt{\frac{2}{\log 2}}\frac{2}{N}$ implies $\gamma \leq \frac{1}{2}$. Rather than selecting N which bounds σ, we can choose N in terms of σ. $\sigma \leq \sqrt{\frac{2}{\log 2}}\frac{2}{N}$ is implied by $N = \frac{2}{\sigma}\sqrt{\frac{2}{\log_e 2}}$. So for small σ we can construct more databases leading to the desired separation. Finally, $N > 4$ implies that we must constrain $\sigma < \sqrt{\frac{1}{2\log_e 2}}$.

In summary, if $n > C$ and $\sigma < \sqrt{\frac{1}{2\log_e 2}}$ then $|f_i(\mathbf{x}_{i,n}) - f_j(\mathbf{x}_{i,n})| \geq \frac{C}{2n}$ for each $i \neq j \in [N]$ where $N = \left\lfloor \frac{2}{\sigma}\sqrt{\frac{2}{\log_e 2}} \right\rfloor$. Moreover if $\epsilon \leq \frac{C}{4n}$ then for any $i \neq j$ this implies $\|f_i - f_j\|_\infty \geq 2\epsilon$ as claimed. ■

We can now state and prove the lower bound on optimal differential privacy for any mechanism that well-approximates the SVM with RBF kernel.

Theorem 26 (Lower bound on optimal differential privacy for RBF SVM) *For $C > 0$, $n > C$, $\delta \in (0,1)$, $\epsilon \in \left(0, \frac{C}{4n}\right)$, and $\sigma \in \left(0, \sqrt{\frac{1}{2\log_e 2}}\right)$, the optimal differential privacy for the hinge* SVM *with RBF kernel having parameter σ is lower-bounded by $\log_e \frac{(1-\delta)(N-1)}{\delta}$, where $N = \left\lfloor \frac{2}{\sigma}\sqrt{\frac{2}{\log_e 2}} \right\rfloor$. That is, under these conditions, all mechanisms that are (ϵ, δ)-useful with respect to hinge* SVM *with RBF kernel for any σ do not achieve differential privacy at any level.*

Proof. Consider (ϵ, δ)-useful mechanism $\hat{M}$ with respect to hinge SVM mechanism M with parameter $C > 0$ and RBF kernel with parameter $0 < \sigma < \sqrt{\frac{1}{2\log_e 2}}$ on n training examples, where $\delta > 0$ and $\frac{C}{4n} > \epsilon > 0$. Let $N = \left\lfloor \frac{2}{\sigma}\sqrt{\frac{2}{\log_e 2}} \right\rfloor > 4$. By Lemma 25 there exist pairwise-neighboring databases $D_1, \ldots, D_N$ of n entries, such that $\{f_i^\star\}_{i=1}^N$ is an ϵ-packing with respect to the L_∞-norm, where $f_i^\star = f_{M(D_i)}$. So $\hat{M}$'s

utility, for each $i \in [N]$, satisfies

$$\begin{aligned} \Pr\left(\hat{f}_i \in \mathcal{B}_\epsilon^\infty\left(f_i^\star\right)\right) &\geq 1-\delta\,, & (13)\\ \sum_{j\neq 1}\Pr\left(\hat{f}_1 \in \mathcal{B}_\epsilon^\infty\left(f_j^\star\right)\right) &\leq \Pr\left(\hat{f}_1 \notin \mathcal{B}_\epsilon^\infty\left(f_1^\star\right)\right) < \delta\,, \\ \Rightarrow\ \exists j\neq 1,\ \Pr\left(\hat{f}_1 \in \mathcal{B}_\epsilon^\infty\left(f_j^\star\right)\right) &< \frac{\delta}{N-1}\,. & (14) \end{aligned}$$

Let $\hat{\mathcal{P}}_1$ and $\hat{\mathcal{P}}_j$ be the distributions of $\hat{M}(D_1)$ and $\hat{M}(D_j)$, respectively, so that for each, $\hat{\mathcal{P}}_i(t) = \Pr\left(\hat{M}(D_i) = t\right)$. Then by Inequalities (13) and (14),

$$\mathbb{E}_{T\sim\mathcal{P}_j}\left[\frac{d\mathcal{P}_1(T)}{d\mathcal{P}_j(T)}\,\middle|\, T\in\mathcal{B}_\epsilon^\infty\left(f_j^\star\right)\right] = \frac{\int_{\mathcal{B}_\epsilon^\infty(f_j^\star)}\frac{d\mathcal{P}_1(t)}{d\mathcal{P}_j(t)}d\mathcal{P}_j(t)}{\int_{\mathcal{B}_\epsilon^\infty(f_j^\star)}d\mathcal{P}_j(t)} \leq \frac{\delta}{(1-\delta)(N-1)}\,.$$

Thus there exists a t such that $\log\frac{\Pr\left(\hat{M}(D_j)=t\right)}{\Pr\left(\hat{M}(D_1)=t\right)} \geq \log\frac{(1-\delta)(N-1)}{\delta}$. ■

Note that $n > C$ is a weak condition, by Remark 1. Also note that this negative result is consistent with our upper bound on optimal differential privacy: σ affects σ_p, increasing the upper bounds as $\sigma \downarrow 0$.

We can again compare upper and lower bounds on the optimal differential privacy now for the case of the SVM with hinge loss and RBF kernel. Using the upper bound in Theorem 21, ignoring constants and using the scaling of C (cf. Remark 1) we have that

$$\Omega\left(\log\frac{1}{\delta}\right) = \beta^\star = O\left(\frac{1}{\epsilon^3\sqrt{n}}\log^{1.5}\frac{\sqrt{n}}{\epsilon\delta}\right).$$

Again the lower bound holds only for small $\epsilon = O(n^{-0.5})$ with the appropriately scaled C. With this growth of ϵ the upper asymptotic bound becomes $O(n\log^{1.5}(n/\delta))$. Compared to the linear-kernel case, the gap here is significantly larger due to mismatching growth with n. Again, an interesting open problem is to improve these bounds.

7 Conclusions

In this paper we present a pair of mechanisms for private SVM learning, each of which releases a classifier based on a privacy-sensitive database of training data. In each case we establish differential privacy of our mechanisms via the algorithmic stability of regularized ERM—a property that is typically used in learning theory to prove risk bounds of learning algorithms.

In addition to measuring the training data privacy preserved by our mechanisms, we also study their utility: the similarity of the classifiers released by private and

non-private SVM. This form of utility implies good generalization error of the private SVM. To achieve utility under infinite-dimensional feature mappings we perform regularized empirical risk minimization (ERM) in a random reproducing kernel Hilbert space (RKHS) whose kernel approximates the target kernel. This trick, borrowed from large-scale learning, permits the mechanism to privately respond with a finite representation of a maximum-margin hyperplane classifier. We establish the high-probability, pointwise similarity between the resulting function and the non-private SVM classifier through a smoothness result of regularized ERM with respect to perturbations of the RKHS.

An interesting direction for future research is to extend our mechanisms and proof techniques to other kernel methods. A general connection between algorithmic stability and global sensitivity would immediately suggest a number of practical privacy-preserving learning mechanisms for which calculations on stability are available: stability would dictate the level of (possibly Laplace) noise required for differential privacy, and for finite-dimensional feature spaces utility would likely follow a similar pattern as presented here for the SVM. Without a general connection, it may be necessary to modify existing stability calculations to yield global sensitivities as we have done here. The application of the random RKHS with kernel approximating a target kernel would also be a useful tool in making kernelized learners differentially private for translation-invariant kernels.

Our bounds on differential privacy and utility combine to upper bound the optimal level of differential privacy possible among all mechanisms that are (ϵ, δ)-useful with respect to the hinge loss SVM . We derive a lower bound on this quantity which establishes that any mechanism that is too accurate with respect to the hinge SVM with RBF kernel, with any non-trivial probability, cannot be β-differentially private for small β. Moreover the lower bound explicitly depends on the RBF kernel's variance. We also present a lower bound for learning with the linear kernel. Interesting open problems are to derive lower bounds holding for moderate to large ϵ, and to reduce the gaps between our upper and lower bounds particularly for both the linear and RBF kernel cases.

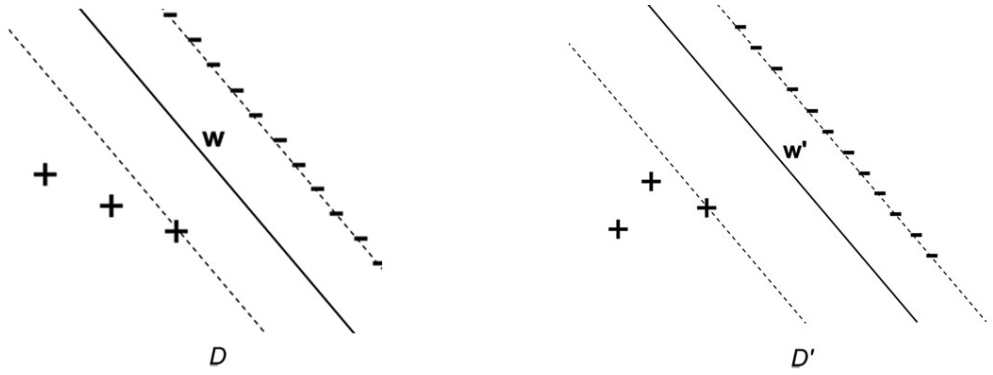

Figure 2: Depiction of the database used in the proof of Proposition 27. Positive and negative training examples are shown as '-' and '+' signs; the SVM decision boundary is shown by the solid line and the margins are shown by the dashed lines. Examples on the margins are support vectors. (Left) displays the original database and (right) shows the same database with the positive support vector modified so that one of the non-support positives becomes a support vector.

Appendix A: SVM Learning and the Statistical Query Model

We now briefly show that SVM learning does not generally fit the Statistical Query model (Kearns, 1998) and so cannot simply be made differentially private by directly applying the existing pattern of making Statistical Query model learners privacy preserving (Blum et al., 2005). However, note that the SVM primal can be iteratively optimized where each iterate is a function of sums over the data (Chapelle, 2007; Chu et al., 2006).

Proposition 27 *The output* $\mathbf{w}$ *of* SVM *learning cannot in general be represented as a sum of a fixed function over the training data* D*.*

Proof. A proof sketch is as follows. Consider linearly-separable D, as depicted in Figure 2, containing three co-linear positive and a large number of co-linear negative examples (but not jointly co-linear), all in $\mathbb{R}^2$. Suppose that the groups of points are close together as shown in the figure, so that in the original configuration of points, the maximum-margin hyperplane must be such that all the negative examples are support vectors, while only one positive is a support vector; and such that when the one positive support vector is moved away from the negative points, the middle positive point becomes a support vector (only).

Denote by S and N the support vectors and the non-support vectors, so that S, N forms a disjoint partition of D. Consider the max-margin normal vector $\mathbf{w}$ for D. Suppose that $\mathbf{0} \neq \mathbf{w} = \sum_{(\mathbf{x},y)\in D} g(\mathbf{x}, y)$ for some g. While holding the support vectors fixed, $\mathbf{w}$ is invariant to perturbing non-support vectors (so long as they do not become

support vectors). Thus $\sum_{(\mathbf{x},y)\in N} g(\mathbf{x},y)$ must be constant in neighborhoods around $(\mathbf{x},y) \in N$. However, if the positive support vector in D is perturbed as in the figure to form D', then $\mathbf{w}$ changes to $\mathbf{w'}$ as shown. Note that both vectors from N are unchanged, and so $\sum_{(\mathbf{x},y)\in N} g(\mathbf{x},y)$ must remain the same as we move from D to D'.

Now suppose that we take D' and perturb the new positive support vector slightly so that the set of support vectors goes unchanged, but $\mathbf{w'}$ changes to some $\mathbf{w''}$. Denote the new configuration of examples by D''. Since the weight vector changes, it must be the case that $\sum_{(\mathbf{x},y)\in D''} g(\mathbf{x},y) \neq \sum_{(\mathbf{x},y)\in D'} g(\mathbf{x},y)$. However, the only term changed is the summand corresponding to the positive support vector. In particular this implies that $\sum_{(\mathbf{x},y)\in N} g(\mathbf{x},y)$ has changed in a neighborhood around a $(\mathbf{x},y) \in N$, which is a contraction. It follows that $\mathbf{w}$ cannot be decomposed as a sum. ■

References

Barak, B., Chaudhuri, K., Dwork, C., Kale, S., McSherry, F., and Talwar, K. (2007). Privacy, accuracy, and consistency too: A holistic solution to contingency table release. In *Proceedings of the Twenty-Sixth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS'07)*. 273–282.

Beimel, A., Kasiviswanathan, S., and Nissim, K. (2010). Bounds on the sample complexity for private learning and private data release. In D. Micciancio (ed.), *Theory of Cryptography*, vol. 5978 of *LNCS*. Springer. 437–454.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer-Verlag.

Blum, A., Dwork, C., McSherry, F., and Nissim, K. (2005). Practical privacy: the SuLQ framework. In *Proceedings of the Twenty-Fourth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS'05)*. 128–138.

Blum, A., Ligett, K., and Roth, A. (2008). A learning theory approach to non-interactive database privacy. In *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC'08)*. 609–618.

Bousquet, O. and Elisseeff, A. (2002). Stability and generalization. *Journal of Machine Learning Research*, 2(Mar):499–526.

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

Chang, C.-C. and Lin, C.-J. (2001). LIBSVM: A library for support vector machines. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Chapelle, O. (2007). Training a support vector machine in the primal. *Neural Computation*, 19(5):1155–1178.

Chaudhuri, K. and Monteleoni, C. (2009). Privacy-preserving logistic regression. In *Advances in Neural Information Processing Systems (NIPS'08)*. 289–296.

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

Chu, C.-T., Kim, S. K., Lin, Y.-A., Yu, Y., Bradski, G. R., Ng, A. Y., and Olukotun, K. (2006). Map-reduce for machine learning on multicore. In *Advances in Neural Information Processing Systems 19 (NIPS '06)*. 281–288.

Cristianini, N. and Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines*. Cambridge University Press.

Devroye, L., Györfi, L., and Lugosi, G. (1996). *A Probabilistic Theory of Pattern Recognition*. Applications of Mathematics: Stochastic Modelling and Applied Probability. Springer.

Devroye, L. P. and Wagner, T. J. (1979). Distribution-free performance bounds for potential function rules. *IEEE Transactions on Information Theory*, 25(5):601–604.

Dinur, I. and Nissim, K. (2003). Revealing information while preserving privacy. In *Proceedings of the Symposium on Principles of Database Systems (PODS'03)*. 202–210.

Dwork, C. (2006). Differential privacy. In *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP'06)*. 1–12.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Theory of Cryptology Conference (TCC'06)*. 265–284.

Dwork, C., Naor, M., Reingold, O., Rothblum, G. N., and Vadhan, S. (2009). On the complexity of differentially private data release: Efficient algorithms and hardness results. In *Proceedings of the 41st ACM Symposium on Theory of Computing (STOC'09)*. 381–390.

Dwork, C. and Yekhanin, S. (2008). New efficient attacks on statistical disclosure control mechanisms. In *Proceedings of the 28th International Cryptology Conference (CRYPTO'08)*. 469–480.

Hardt, M. and Talwar, K. (2010). On the geometry of differential privacy. In *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC'10)*. ACM. 705–714.

Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'02)*. 133–142.

Kasiviswanathan, S. P., Lee, H. K., Nissim, K., Raskhodnikova, S., and Smith, A. (2008). What can we learn privately? In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS'08)*. 531–540.

Kearns, M. (1998). Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45:983–1006.

Kearns, M. and Ron, D. (1999). Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. *Neural Computation*, 11:1427–1453.

Kim, H., Claffy, K., Fomenkov, M., Barman, D., Faloutsos, M., and Lee, K. (2008). Internet traffic classification demystified: myths, caveats, and the best practices. In *Proceedings of the 2008 ACM CoNEXT Conference (CoNEXT '08)*.

Kimeldorf, G. and Wahba, G. (1971). Some results on Tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33(1):82–95.

Kutin, S. and Niyogi, P. (2002). Almost-everywhere algorithmic stability and generalization error. Technical report TR-2002-03, Computer Science Department, University of Chicago.

McSherry, F. and Mironov, I. (2009). Differentially private recommender systems: building privacy into the net. In *Proceedings of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'09)*. 627–636.

McSherry, F. and Talwar, K. (2007). Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*. 94–103.

Pollard, D. (1984). *Convergence of Stochastic Processes*. Springer-Verlag.

Rahimi, A. and Recht, B. (2008). Random features for large-scale kernel machines. In *Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems (NIPS'08)*. 1177–1184.

Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C.-H., Angelo, M., Ladd, C., Reich, M., Latulippe, E., Mesirov, J. P., Poggio, T., Gerald, W., Loda, M., Lander, E. S., and Golub, T. R. (2001). Multiclass cancer diagnosis using tumor gene expression signatures. *Proceedings of the National Academy of Sciences*, 98(26):15149–15154.

Rubinstein, B. I. P., Bartlett, P. L., Huang, L., and Taft, N. (2009). Learning in a large function space: Privacy-preserving mechanisms for SVM learning. *CoRR*, abs/0911.5708. Submitted 30 Nov 2009.

Rudin, W. (1994). *Fourier Analysis on Groups*. Wiley-Interscience, reprint edition.

Sarwate, A. D., Chaudhuri, K., and Monteleoni, C. (2009). Differentially private support vector machines. *CoRR*, abs/0912.0071. Submitted 1 Dec 2009.

Schölkopf, B. and Smola, A. J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press.

van der Vaart, A. W. (2000). *Asymptotic Statistics*. Cambridge University Press.

van der Vaart, A. W. and Wellner, J. A. (2000). *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer, 2nd ed.

Xu, J. A. and Araki, K. (2006). A SVM-based personal recommendation system for TV programs. In *Proceedings of the 12th International Multi-Media Modelling Conference*.