Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

The Bernstein Mechanism: Function Release under Differential Privacy

Francesco Aldà
Horst Görtz Institute for IT Security
and Faculty of Mathematics
Ruhr-Universität Bochum, Germany
francesco.alda@rub.de

Benjamin I.P. Rubinstein
Dept. Computing and Information Systems
The University of Melbourne, Australia
brubinstein@unimelb.edu.au

Abstract

We address the problem of general function release under differential privacy, by developing a functional mechanism that applies under the weak assumptions of oracle access to target function evaluation and sensitivity. These conditions permit treatment of functions described explicitly or implicitly as algorithmic black boxes. We achieve this result by leveraging the iterated Bernstein operator for polynomial approximation of the target function, and polynomial coefficient perturbation. Under weak regularity conditions, we establish fast rates on utility measured by high-probability uniform approximation. We provide a lower bound on the utility achievable for any functional mechanism that is ε-differentially private. The generality of our mechanism is demonstrated by the analysis of a number of example learners, including naive Bayes, non-parametric estimators and regularized empirical risk minimization. Competitive rates are demonstrated for kernel density estimation; and ε-differential privacy is achieved for a broader class of support vector machines than known previously.

1 Introduction

In recent years, *differential privacy* (Dwork et al. 2006) has emerged as a leading paradigm for privacy-preserving statistical analyses. It provides formal guarantees that aggregate statistics output by a randomized mechanism are not significantly influenced by the presence or absence of an individual datum. *Where the Laplace mechanism (Dwork et al. 2006) is a de facto approach for converting vector-valued functions to differential privacy, in this paper we seek an equivalent approach for privatizing function-valued mappings.* We achieve our goal through the development of a novel Bernstein functional mechanism. Unlike existing mechanisms, ours applies to releasing explicitly and implicitly defined functions, and is characterized by a full theoretical analysis.

Our setting is the release of functions that depend on privacy-sensitive training data, and that can be subsequently evaluated on arbitrary test points. This non-interactive setting matches a wide variety of learning tasks from naive Bayes classification, non-parametric methods (kernel density estimation and regression) where the function of train and test data is explicit, to generalized linear models, support vector machines where the function is only implicitly defined by an iterative algorithm. Our generic mechanism is based on functional approximation by Bernstein basis polynomials, specifically via an iterated Bernstein operator. Privacy is guaranteed by sanitizing the coefficients of approximation, which requires only function evaluation. It is the very limited oracle access required by our mechanism—to non-private function evaluation and sensitivity—that grants it broad applicability akin to the Laplace mechanism.

The Bernstein polynomials central to our mechanism are used in the Stone-Weierstrass theorem to uniformly approximate any continuous function on a closed interval. Moreover, the Bernstein operator offers several advantages such as data-independent bounds, no requirement of access to target function derivatives, and yields approximations that are pointwise convex combinations of the function evaluations on a cover. As a result, applying privacy-preserving perturbations to the approximation's coefficients permits us to control utility and achieve fast convergence rates.

In addition to being analyzed in full, the Bernstein mechanism is easy to use. We demonstrate this with a variety of example analyses of the mechanism applied to learners. Finally, we provide a lower bound that fundamentally limits utility under private function release, partly resolving a question posed by Hall, Rinaldo, and Wasserman (2013). This matches (up to logarithmic factors) our upper bound in the linear case.

Related Work. Polynomial approximation has proven useful in differential privacy outside function release (Thaler, Ullman, and Vadhan 2012; Chandrasekaran et al. 2014). Few previous attempts have been made towards private function release. Hall, Rinaldo, and Wasserman (2013) add Gaussian process noise which only yields a weaker form of privacy, namely (ε, δ)-differential privacy, and does not admit general utility rates. Zhang et al. (2012) introduce a functional mechanism for the more specific task of perturbing the objective in private optimization, but they assume separability in the training data and do not obtain rates on utility.

Wang et al. (2013) propose a mechanism that releases a summary of data in a trigonometric basis, able to respond to queries that are smooth as in our setting, but are also required to be separable in the training dataset as assumed by Zhang et al. (2012). A natural application is kernel density estimation,

which would achieve a rate of $O\left(\log(1/\beta)/(n\varepsilon)\right)^{h/(\ell+h)}$ as does our approach. Private KDE has also been explored in various other settings (Duchi, Wainwright, and Jordan 2013) and under weaker notions of utility (Hall, Rinaldo, and Wasserman 2013). Zhang, Rubinstein, and Dimitrakakis (2016) explore discrete naive Bayes under differential privacy, while we investigate parametric Gaussian and non-parametric KDE for class-conditional likelihoods.

As an example of an implicitly defined function, we consider regularized empirical risk minimization such as logistic regression, ridge regression, and the SVM. Previous mechanisms for private SVM release and ERM more generally (Chaudhuri and Monteleoni 2008; Rubinstein et al. 2012; Chaudhuri, Monteleoni, and Sarwate 2011; Jain and Thakurta 2014; 2013; Bassily, Smith, and Thakurta 2014) require finite-dimensional feature mapping or translation-invariant kernels. Hall, Rinaldo, and Wasserman (2013) consider more general mappings but provide (ε, δ)-differential privacy. Our treatment of regularized ERM extends to kernels that may be translation-variant with infinite-dimensional mappings, while providing stronger privacy guarantees.

2 Preliminaries

Notation and Problem Setting. Throughout the paper, vectors are written in bold and the i-th component of a vector $\boldsymbol{x}$ is denoted by x_i. We consider $\mathcal{X}$ an arbitrary (possibly infinite) domain and $\mathcal{D} \in \mathcal{X}^n$ a database of n points in $\mathcal{X}$. We refer to n as the size of the database $\mathcal{D}$. For a positive integer ℓ, let $\mathcal{Y} = [0,1]^\ell$ be a set of query points and $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ the target function. Once the database $\mathcal{D}$ is fixed, we denote by $F_\mathcal{D} = F(\mathcal{D}, \cdot)$ the function parameterized by $\mathcal{D}$ that we aim to release. For example: $\mathcal{D}$ might represent a training set—over $\mathcal{X}$ a product space of feature vectors and labels—with $\mathcal{Y}$ representing test points from the same feature space; $F_\mathcal{D}$ would then be a classifier resulting from training on $\mathcal{D}$. Section 6 presents examples for F. In Section 3, we show how to privately release the function $F_\mathcal{D}$ and we provide alternative error bounds depending on the regularity of F.

Definition 1. *Let h be a positive integer and $T > 0$. A function $f\colon [0,1]^\ell \to \mathbb{R}$ is (h,T)-smooth if it is $C^h([0,1]^\ell)$ and its partial derivatives up to order h are all bounded by T.*

Definition 2. *Let $0 \le \gamma \le 1$ and $L > 0$. A function $f\colon [0,1]^\ell \to \mathbb{R}$ is (γ, L)-Hölder continuous if, for every $\boldsymbol{x}, \boldsymbol{y} \in [0,1]^\ell$, $|f(\boldsymbol{x}) - f(\boldsymbol{y})| \le L\|\boldsymbol{x} - \boldsymbol{y}\|_\infty^\gamma$. When $\gamma = 1$, we refer to f as L-Lipschitz.*

Our goal is to develop a private release mechanism for the function $F_\mathcal{D}$ in the *non-interactive* setting. A non-interactive mechanism takes a function F and a database $\mathcal{D}$ as inputs and outputs a synopsis $\mathcal{A}$ which can be used to evaluate the function $F_\mathcal{D}$ on $\mathcal{Y}$ without accessing the database $\mathcal{D}$ further.

Differential Privacy. To provide strong privacy guarantees on the release of $F_\mathcal{D}$, we adopt the well-established notion of differential privacy.

Definition 3 (Dwork et al. 2006)**.** *Let $\mathcal{R}$ be a (possibly infinite) set of responses. A mechanism $\mathcal{M} : \mathcal{X}^\star \to \mathcal{R}$ (meaning that, for every $\mathcal{D} \in \mathcal{X}^\star = \bigcup_{n>0} \mathcal{X}^n$, $\mathcal{M}(\mathcal{D})$ is an $\mathcal{R}$-valued random variable) is said to provide (ε, δ)-*differential privacy *for $\varepsilon > 0$ and $0 \le \delta < 1$ if, for every $n \in \mathbb{N}$, for every pair $(\mathcal{D}, \mathcal{D}') \in \mathcal{X}^n \times \mathcal{X}^n$ of databases differing in one entry only (henceforth denoted by $\mathcal{D} \sim \mathcal{D}'$), and for every measurable $S \subseteq \mathcal{R}$, we have $\mathbb{P}[\mathcal{M}(\mathcal{D}) \in S] \le e^\varepsilon \mathbb{P}[\mathcal{M}(\mathcal{D}') \in S] + \delta$. If $\delta = 0$ we simply say that $\mathcal{M}$ provides ε-differential privacy.*

By limiting the influence of data on the induced response distribution, a powerful adversary (with knowledge of all but one input datum, the mechanism up to random source, and unbounded computation) cannot effectively identify an unknown input datum from mechanism responses. The Laplace mechanism (Dwork et al. 2006) is a generic tool for differential privacy: adding zero-mean Laplace noise[1] to a vector-valued function provides privacy if the noise is calibrated to the function's sensitivity.

Definition 4 (Dwork et al. 2006)**.** *The* sensitivity *of a function $f : \mathcal{X}^n \to \mathbb{R}^d$ is given by $S(f) = \sup_{\mathcal{D}\sim\mathcal{D}'} \|f(\mathcal{D}) - f(\mathcal{D}')\|_1$, where the supremum is taken over all $\mathcal{D}, \mathcal{D}' \in \mathcal{X}^n$ that differ in one entry only. The sensitivity of a function $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}^d$ is defined as $S(F) = \sup_{\boldsymbol{y}\in\mathcal{Y}} S(F(\cdot, \boldsymbol{y}))$.*

Lemma 1 (Dwork et al. 2006)**.** *Let $f : \mathcal{X}^n \to \mathbb{R}^d$ be a non-private function of finite sensitivity, and let $\boldsymbol{Z} \sim \mathrm{Lap}(S(f)/\varepsilon)^d$. Then, the random function $\tilde{f}(\mathcal{D}) = f(\mathcal{D}) + \boldsymbol{Z}$ provides ε-differential privacy.*

Given a mechanism, we measure its accuracy as follows.

Definition 5. *Let $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$. A mechanism $\mathcal{M}$ is (α, β)-*accurate *with respect to $F_\mathcal{D}$ if for any database $\mathcal{D} \in \mathcal{X}^n$ and $\mathcal{A} = \mathcal{M}(\mathcal{D})$, with probability at least $1 - \beta$ over the randomness of $\mathcal{M}$, $\sup_{\boldsymbol{y}\in\mathcal{Y}} |\mathcal{A}(\boldsymbol{y}) - F_\mathcal{D}(\boldsymbol{y})| \le \alpha$.*

3 The Bernstein Mechanism

Algorithm 1 introduces a differentially-private mechanism for releasing $F_\mathcal{D} : \mathcal{Y} \to \mathbb{R}$, a family of (h,T)-smooth or (γ, L)-Hölder continuous functions, parameterized by $\mathcal{D} \in \mathcal{X}^n$.

Algorithm 1 The Bernstein mechanism

Sanitization – **Inputs:** private dataset $\mathcal{D} \in \mathcal{X}^n$; sensitivity $S(F)$ and oracle access to target $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$
Parameters: cover size k, Bernstein order h positive integers; privacy budget $\varepsilon > 0$

1: $P \leftarrow (\{0, 1/k, 2/k, \ldots, 1\})^\ell$ ▷ Lattice cover of $\mathcal{Y}$
2: $\lambda \leftarrow S(F)(k+1)^\ell/\varepsilon$ ▷ Perturbation scale
3: For each $\boldsymbol{p} = (p_1, \ldots, p_\ell) \in P$:
4: $\quad \widetilde{F_\mathcal{D}}(\boldsymbol{p}) \leftarrow F_\mathcal{D}(\boldsymbol{p}) + Z$, where $Z \overset{i.i.d.}{\sim} \mathrm{Lap}(\lambda)$
5: **Return:** $\left\{\widetilde{F_\mathcal{D}}(\boldsymbol{p}) \mid \boldsymbol{p} \in P\right\}$

Evaluation – **Inputs:** query $\boldsymbol{y} \in \mathcal{Y}$; private response $\left\{\widetilde{F_\mathcal{D}}(\boldsymbol{p}) \mid \boldsymbol{p} \in P\right\}$

6: $b^{(h)}_{\nu_i,k} \leftarrow$ Compute basis ▷ See Definition 8
7: **Return:** $\sum_{j=1}^{\ell} \sum_{\nu_j=0}^{k} \widetilde{F_\mathcal{D}}\left(\frac{\nu_1}{k}, \cdots, \frac{\nu_\ell}{k}\right) \prod_{i=1}^{\ell} b^{(h)}_{\nu_i,k}(y_i)$

[1] A $\mathrm{Lap}(\lambda)$-distributed real random variable Z has probability density proportional to $\exp(-|y|/\lambda)$.

The mechanism makes use of the iterated Bernstein polynomial of $F_{\mathcal{D}}$, which we introduce next (for a comprehensive survey refer to Lorentz 1953, Micchelli 1973). This approximation consists of a linear combination of so-called Bernstein basis polynomials, whose coefficients are evaluations of target $F_{\mathcal{D}}$ on a (lattice) cover P.

We briefly introduce the univariate Bernstein basis polynomials and state some of their properties.

Definition 6. *Let k be a positive integer. The* Bernstein basis polynomials *of degree k are defined as $b_{\nu,k}(y) = \binom{k}{\nu} y^{\nu}(1-y)^{k-\nu}$ for $\nu = 0, \ldots, k$.*

Proposition 2 (Lorentz 1953)**.** *For every $y \in [0,1]$, any positive integer k and $0 \leq \nu \leq k$, we have $b_{\nu,k}(y) \geq 0$ and $\sum_{\nu=0}^{k} b_{\nu,k}(y) = 1$.*

In order to introduce the iterated Bernstein polynomials, we first need to recall the Bernstein operator.

Definition 7. *Let $f\colon [0,1] \to \mathbb{R}$ and k be a positive integer. The* Bernstein polynomial *of f of degree k is defined as $B_k(f;y) = \sum_{\nu=0}^{k} f(\nu/k)\, b_{\nu,k}(y)$.*

The *Bernstein operator* B_k maps a function f, defined on $[0,1]$, to $B_k f$, where the function $B_k f$ evaluated at y is $B_k(f;y)$. Note that the Bernstein operator is linear and if $f(y) \in [a_1, a_2]$ for every $y \in [0,1]$, then from Proposition 2 it follows that $B_k(f;y) \in [a_1, a_2]$ for every positive integer k and $y \in [0,1]$. Moreover, it is not hard to show that any linear function is a fixed point for B_k (*cf.* the full report Aldà and Rubinstein 2016).

Definition 8 (Micchelli 1973)**.** *Let h be a positive integer. The* iterated Bernstein operator *of order h is defined as the sequence of linear operators $B_k^{(h)} = I - (I - B_k)^h = \sum_{i=1}^{h} \binom{h}{i} (-1)^{i-1} B_k^i$, where $I = B_k^0$ denotes the identity operator and B_k^i is defined inductively as $B_k^i = B_k \circ B_k^{i-1}$ for $i \geq 1$. The* iterated Bernstein polynomial *of order h can then be computed as:*

$$B_k^{(h)}(f;y) = \sum_{\nu=0}^{k} f\left(\frac{\nu}{k}\right) b_{\nu,k}^{(h)}(y)\,,$$

where $b_{\nu,k}^{(h)}(y) = \sum_{i=1}^{h} \binom{h}{i} (-1)^{i-1} B_k^{i-1}(b_{\nu,k};y)$.

We observe that $B_k^{(1)} = B_k$. Although the bases $b_{\nu,k}^{(h)}$ are not always positive for $h \geq 2$, we still have $\sum_{\nu=0}^{k} b_{\nu,k}^{(h)}(y) = 1$ for every $y \in [0,1]$. The iterated Bernstein polynomial of a multivariate function $f\colon [0,1]^{\ell} \to \mathbb{R}$ is analogously defined.

Definition 9. *Assume $f\colon [0,1]^{\ell} \to \mathbb{R}$ and let $k_1, \ldots, k_{\ell}, h$ be positive integers. The (multivariate) iterated Bernstein polynomial of f (of order h) is defined as*

$$B_{k_1,\ldots,k_{\ell}}^{(h)}(f;\boldsymbol{y}) = \sum_{j=1}^{\ell} \sum_{\nu_j=0}^{k_j} f\left(\frac{\nu_1}{k_1}, \cdots, \frac{\nu_{\ell}}{k_{\ell}}\right) \prod_{i=1}^{\ell} b_{\nu_i,k_i}^{(h)}(y_i).$$

For ease of exposition, we fix user-selected $k \in \mathbb{N}$ such that $k_1 = \ldots = k_{\ell} = k$. The Bernstein mechanism perturbs the evaluation of $F_{\mathcal{D}}$ on a lattice cover P of $\mathcal{Y} = [0,1]^{\ell}$ parameterized by k.

4 Analysis of Mechanism Privacy and Utility

In the following result, we assume ℓ to be an arbitrary but fixed constant with $\mathcal{Y} = [0,1]^{\ell}$. We underline that this is a common assumption in the differential privacy literature, especially when dealing with Euclidean spaces (Blum, Ligett, and Roth 2008; Dwork and Lei 2009; Wasserman and Zhou 2010; Lei 2011; Wang et al. 2013).

Theorem 3 (Main Theorem)**.** *Let $\ell, h \in \mathbb{N}_+$, $0 < \gamma \leq 1$, $L > 0$ and $T > 0$ be constants. Let $\mathcal{X}$ be an arbitrary space and $\mathcal{Y} = [0,1]^{\ell}$. Let furthermore $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ with $S(F) = o(1)$. For $\varepsilon > 0$, the Bernstein mechanism $\mathcal{M}$ provides ε-differential privacy. Moreover, for $0 < \beta < 1$ the mechanism $\mathcal{M}$ is (α, β)-accurate with error scaling as follows, where hidden constants depend on ℓ, L, γ, T, h.*

(i) If $F_{\mathcal{D}}$ is $(2h, T)$-smooth for every $\mathcal{D} \in \mathcal{X}^n$, there exists $k = k(S(F), \varepsilon, \beta, \ell, h, T)$ such that $\alpha = O\left(\frac{S(F)}{\varepsilon} \log(1/\beta)\right)^{\frac{h}{\ell+h}}$;

(ii) If $F_{\mathcal{D}}$ is (γ, L)-Hölder continuous for every $\mathcal{D} \in \mathcal{X}^n$, there exists $k = k(S(F), \varepsilon, \beta, \ell, \gamma, L)$ such that $\alpha = O\left(\frac{S(F)}{\varepsilon} \log(1/\beta)\right)^{\frac{\gamma}{2\ell+\gamma}}$; and

(iii) If $F_{\mathcal{D}}$ is linear for every $\mathcal{D} \in \mathcal{X}^n$, there exists a constant k such that $\alpha = O\left(\frac{S(F)}{\varepsilon} \log(1/\beta)\right)$.

Moreover, if $1/S(F) \leq \mathrm{poly}(n)$, then the running-time of the mechanism and the running-time per evaluation are both polynomial in n and $1/\varepsilon$.

4.1 Proof of the Main Theorem[2]

To prove privacy we note that only the coefficients of the Bernstein polynomial of $F_{\mathcal{D}}$ are sensitive and need to be protected. In order to provide ε-differential privacy, these coefficients—evaluations of target $F_{\mathcal{D}}$ on a cover—are perturbed by means of Lemma 1. In this way, we can release the sanitized coefficients and use them for unlimited, efficient evaluation of the approximation of $F_{\mathcal{D}}$ over $\mathcal{Y}$, without further access to the data $\mathcal{D}$. To establish utility, we separately analyze error due to the polynomial approximation of $F_{\mathcal{D}}$ and error due to perturbation.

In order to analyze the accuracy of our mechanism, we denote by $\widetilde{B_k^{(h)}}(F_{\mathcal{D}};\boldsymbol{y})$ the iterated Bernstein polynomial of order h constructed using the coefficients output by the mechanism $\mathcal{M}$. The error α introduced by the mechanism can be expressed as follows:

$$\alpha = \max_{\boldsymbol{y} \in [0,1]^{\ell}} \left| F_{\mathcal{D}}(\boldsymbol{y}) - \widetilde{B_k^{(h)}}(F_{\mathcal{D}};\boldsymbol{y}) \right| \tag{1}$$

$$\leq \max_{\boldsymbol{y} \in [0,1]^{\ell}} \left| \widetilde{B_k^{(h)}}(F_{\mathcal{D}};\boldsymbol{y}) - B_k^{(h)}(F_{\mathcal{D}};\boldsymbol{y}) \right| \tag{2}$$

$$+ \max_{\boldsymbol{y} \in [0,1]^{\ell}} \left| F_{\mathcal{D}}(\boldsymbol{y}) - B_k^{(h)}(F_{\mathcal{D}};\boldsymbol{y}) \right|.$$

[2]For sake of clarity, in the full report Aldà and Rubinstein (2016) we provide a self-contained proof of Theorem 3 for $\ell = 1$. Although it is not a prerequisite to the general result, it reflects the building blocks used in this section.

For every $\boldsymbol{y} \in [0,1]^\ell$, the first summand in Equation (2) consists of the absolute value of an affine combination of independent Laplace-distributed random variables.

Proposition 4. *Let $\Gamma = \{\boldsymbol{\nu} \in \mathbb{N}^\ell \mid 0 \leq \nu_j \leq k \text{ for } 1 \leq j \leq \ell\}$. For every $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_\ell) \in \Gamma$ let $Z_{\boldsymbol{\nu}} \overset{i.i.d.}{\sim} \mathrm{Lap}(\lambda)$. Moreover, let $\tau \geq 0$ and constant $C_{h,\ell}$ depend only on h, ℓ. Then:*

$$\mathbb{P}\left[\max_{\boldsymbol{y}\in[0,1]^\ell} \left|\sum_{j=1}^{\ell}\sum_{\nu_j=0}^{k} Z_{\boldsymbol{\nu}} \prod_{i=1}^{\ell} b^{(h)}_{\nu_i,k}(y_i)\right| \geq \tau\right] \leq e^{-\tau/(C_{h,\ell}\lambda)}.$$

The proof of Proposition 4 follows from a result of Proschan (1965) on the concentration of convex combinations of random variables drawn i.i.d. from a log-concave symmetric distribution (*cf.* the full report Aldà and Rubinstein 2016). Proposition 4 implies that with probability at least $1-\beta$ the first summand in Equation (2) is bounded by $O\left(S(F)k^\ell \log(1/\beta)/\varepsilon\right)$. In order to bound the second summand we make use of the following (unidimensional) convergence rates.

Theorem 5 (Micchelli 1973)**.** *Let h be a positive integer and $T > 0$. If $f\colon [0,1] \to \mathbb{R}$ is a $(2h,T)$-smooth function, then, for all positive integers k and $y \in [0,1]$, $\left|f(y) - B^{(h)}_k(f;y)\right| \leq T D_h k^{-h}$, where D_h is a constant independent of k, f and $y \in [0,1]$.*

Theorem 6 (Kac 1938; Mathé 1999)**.** *Let $0 < \gamma \leq 1$ and $L > 0$. If $f\colon [0,1] \to \mathbb{R}$ is a (γ, L)-Hölder continuous function, then $\left|f(y) - B^{(1)}_k(f;y)\right| \leq L\,(4k)^{-\gamma/2}$ for all positive integers k and $y \in [0,1]$.*

By induction, it is possible to show that the approximation error of the multivariate iterated Bernstein polynomial can be bounded by $O(\ell g(k)) = O(g(k))$, if the error of the corresponding univariate polynomial is bounded by $g(k)$ (*cf.* the full report Aldà and Rubinstein 2016).

All in all, the error α introduced by the mechanism can thus be bounded by

$$\alpha = O\left(g(k) + \frac{S(F)k^\ell}{\varepsilon}\log(1/\beta)\right). \tag{3}$$

Since $g(k)$ is a decreasing function in k and the second summand in Equation (3) is an increasing function in k, the optimal value for k (up to a constant factor) is achieved when k satisfies

$$g(k) = \frac{S(F)k^\ell}{\varepsilon}\log(1/\beta)\,. \tag{4}$$

Solving Equation (4) with the bound for $g(k)$ provided in Theorem 5 yields

$$k = \max\left\{1, \left(\frac{\varepsilon}{S(F)\log(1/\beta)}\right)^{\frac{1}{h+\ell}}\right\}$$

and substituting the thus obtained value of k into Equation (3) yields the first statement of Theorem 3. Similarly, using the bound for $g(k)$ provided in Theorem 6 we get the result for Hölder continuous functions. The bound for linear functions follows from the fact that the approximation error is zero for $h = 1$ and $k = 1$, since linear functions are fixed points of $B^{(1)}_1$. Finally, the analysis of the running time follows from observing that, for the optimal cover size k we computed, k^ℓ is always upper bounded by $\varepsilon/(S(F)\log(1/\beta))$ and thus by $\mathrm{poly}(n)$.

4.2 Discussion

Comparison to Baseline. Algorithm 1 is based on a relatively simple approach: it evaluates the target function on a lattice cover, adding Laplace noise for privacy. One might be tempted to approximate the input function by rounding a query point $\boldsymbol{y}$ to the nearest lattice point $\boldsymbol{p}$ and releasing the corresponding noisy evaluation $\widetilde{F_{\mathcal{D}}(\boldsymbol{p})}$. Although it is straightforward to prove that, for (γ, L)-Hölder continuous functions, such a piecewise constant approximation achieves error $O(1/k^\gamma)$, this upper bound is essentially tight, as it can be shown by considering the approximation error it achieves for linear functions. Therefore, this method has two main disadvantages: the output function is not even continuous (although we always consider continuous input functions) and for highly smooth input functions it cannot achieve the fast convergence rates of the Bernstein mechanism. In Section 6, we offer further examples supporting this argument.

(ε, δ)-Differential Privacy. We note that our analysis can be easily extended to the relaxed notion of *approximate differential privacy* using advanced composition theorems (see for example Dwork and Roth 2014) instead of sequential composition (Dwork et al. 2006). Specifically, it suffices to choose the perturbation scale $\lambda_\delta = 2S(F)\sqrt{2(k+1)^\ell \log(1/\delta)}/\varepsilon$.

Theorem 7. *Let $0 < \delta < 1$. Under the same assumptions of Theorem 3, the Bernstein mechanism $\mathcal{M}$ (with perturbation scale λ_δ) provides (ε, δ)-differential privacy and is (α, β)-accurate with error scaling as follows.*

(i) If $F_{\mathcal{D}}$ is $(2h,T)$-smooth for every $\mathcal{D} \in \mathcal{X}^n$, there exists $k = k(S(F), \varepsilon, \delta, \beta, \ell, h, T)$ such that $\alpha = O\left(\frac{S(F)}{\varepsilon}\log(1/\beta)\sqrt{\log(1/\delta)}\right)^{\frac{2h}{\ell+2h}}$; and

(ii) If $F_{\mathcal{D}}$ is (γ, L)-Hölder continuous for every $\mathcal{D} \in \mathcal{X}^n$, there exists $k = k(S(F), \varepsilon, \delta, \beta, \ell, \gamma, L)$ such that $\alpha = O\left(\frac{S(F)}{\varepsilon}\log(1/\beta)\sqrt{\log(1/\delta)}\right)^{\frac{\gamma}{\ell+\gamma}}$.

Even though this relaxation allows for improved accuracy, in this work we explore a different point on the privacy-utility Pareto front and focus our attention on ε-differential privacy, since there is generally a significant motivation for achieving stronger privacy guarantees. Moreover, to the best of our knowledge, it is unknown whether previous solutions (Hall, Rinaldo, and Wasserman 2013) even apply to this framework.

5 Lower Bound

In this section we present a lower bound on the error that any ε-differentially private mechanism approximating a function $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ must introduce.

Theorem 8. *For $\varepsilon > 0$, there exists a function $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ such that the error that any ε-differentially private mechanism approximating F introduces is $\Omega\left(S(F)/\varepsilon\right)$, with probability arbitrarily close to* 1.

Proof. In order to prove Theorem 8, we consider $\mathcal{X} \subset [0,1]^\ell$ to be a finite set and without loss of generality we view the database $\mathcal{D}$ as an element of $\mathcal{X}^n$ or as an element of $\mathbb{N}^{|\mathcal{X}|}$, *i.e.,* a histogram over the elements of $\mathcal{X}$, interchangeably. We can then make use of a general result provided by De (2012).

Proposition 9 (De 2012)**.** *Assume $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_{2^s} \in \mathbb{N}^N$ such that, for every i, $\|\mathcal{D}_i\|_1 \leq n$ and, for $i \neq j$, $\|\mathcal{D}_i - \mathcal{D}_j\|_1 \leq \Delta$. Moreover, let $f\colon \mathbb{N}^N \to \mathbb{R}^t$ be such that for any $i \neq j$, $\|f(\mathcal{D}_i) - f(\mathcal{D}_j)\|_\infty \geq \eta$. If $\Delta \leq (s-1)/\varepsilon$, then any mechanism which is ε-differentially private for the query f on databases of size n introduces an error which is $\Omega(\eta)$, with probability arbitrarily close to* 1.

Therefore, we only need to show that there exists a suitable sequence of databases $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_{2^s}$, a function $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ and a $\boldsymbol{y} \in \mathcal{Y}$ such that $F(\cdot, \boldsymbol{y})$ satisfies the assumptions of Proposition 9. We actually show that this holds for every $\boldsymbol{y} \in \mathcal{Y}$. Let $\varepsilon > 0$ and V be a non-negative integer. We define $\mathcal{X} = (\{0, 1/(V+8), 2/(V+8), \ldots, 1\})^\ell$. Note that $N = |\mathcal{X}| = (V+9)^\ell$. Let furthermore $c = \lfloor 1/\varepsilon \rfloor$ and $n = V + c$. The function $F\colon \mathcal{X}^n \times [0,1]^\ell \to \mathbb{R}$ we consider is defined as follows:

$$F(\mathcal{D}, \boldsymbol{y}) = \eta(d_0 + \ldots + d_{N-7} + 2d_{N-6} + \ldots + 8d_N + \langle \boldsymbol{y}, \mathbf{1} \rangle),$$

where d_i corresponds to the number of entries in $\mathcal{D}$ whose value is x_i, for every $x_i \in \mathcal{X}$. For $s = 3$, we consider the sequence of databases $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_8$, where, for $j \in \{1, 2, \ldots, 8\}$, $d_i \in \mathcal{D}_j$ is such that

$$d_i = \begin{cases} 1, & \text{for } i \in \{0, 1, \ldots, V-1\} \\ c, & \text{for } i = N - j + 8 \\ 0, & \text{otherwise} \end{cases}.$$

We first observe that, for every $j \in \{1, 2, \ldots, 8\}$, $\|\mathcal{D}_j\|_1 = n$. Moreover, for $i \neq j$, $\|\mathcal{D}_i - \mathcal{D}_j\|_1 = 2c \leq 2/\varepsilon$. Finally, for $i \neq j$, $|F(\mathcal{D}_i, \boldsymbol{y}) - F(\mathcal{D}_j, \boldsymbol{y})| \geq c\eta$ for every $\boldsymbol{y} \in [0,1]^\ell$. Since $S(F) = 7\eta$, Proposition 9 implies that, with high probability, any ε-differentially private mechanism approximating F must introduce an error of order $\Omega\left(S(F)/\varepsilon\right)$. □

6 Examples

In this section, we demonstrate the versatility of the Bernstein mechanism through the analysis of a range of example learners.

Kernel Density Estimation. Let $\mathcal{X} = \mathcal{Y} = [0,1]^\ell$ and $\mathcal{D} = (\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_n) \in \mathcal{X}^n$. For a given kernel K_H, with bandwidth H (a symmetric and positive definite $\ell \times \ell$ matrix), the kernel density estimator $F_H\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ is defined as $F_H(\mathcal{D}, \boldsymbol{y}) = \frac{1}{n}\sum_{i=1}^n K_H(\boldsymbol{y} - \boldsymbol{d}_i)$. It is easy to see that $S(F_H) \leq \sup_{\boldsymbol{y} \in [-1,1]^\ell} K_H(\boldsymbol{y})/n$. For instance, if K_H is the Gaussian kernel with covariance matrix H, then $S(F_H) \leq 1/(n\sqrt{(2\pi)^\ell \det(H)})$. Moreover, observe that $F_H(\mathcal{D}, \cdot)$ is an (h, T)-smooth function for any positive integer h. Hence the error introduced by the mechanism is

$$O\left(\frac{1}{n\varepsilon\sqrt{\det(H)}}\log(1/\beta)\right)^{\frac{h}{\ell+h}},$$

with probability at least $1 - \beta$. In Figure 1 we display the utility (averaged over 1000 repeats) of the Bernstein mechanism ($k = 20$) on 5000 points drawn from a mixture of two normal distributions $N(0.5, 0.02)$ and $N(0.75, 0.005)$ with weights $0.4, 0.6$, respectively. We first observe that for every privacy budget ε there is a suitable choice of h such that our mechanism always achieves better utility compared to the baseline (*cf.* Section 4.2). Moreover, accuracy improves for increasing h, except for sufficiently large perturbations (small ε) which more significantly affect higher-order basis functions (larger h). Private cross validation (Chaudhuri, Monteleoni, and Sarwate 2011; Chaudhuri and Vinterbo 2013) can be used to tune h. We conclude noting that the same error bounds can be provided by the mechanism of Wang et al. (2013), since the function $F_H(\mathcal{D}, \cdot)$ is *separable* in the training set $\mathcal{D}$, *i.e.,* $F_H(\mathcal{D}, \cdot) = \sum_{\boldsymbol{d} \in \mathcal{D}} f_H(\boldsymbol{d}, \cdot)$. However, this assumption is overly restrictive for many applications. In the following, we discuss how the Bernstein mechanism can be successfully applied to several such cases.

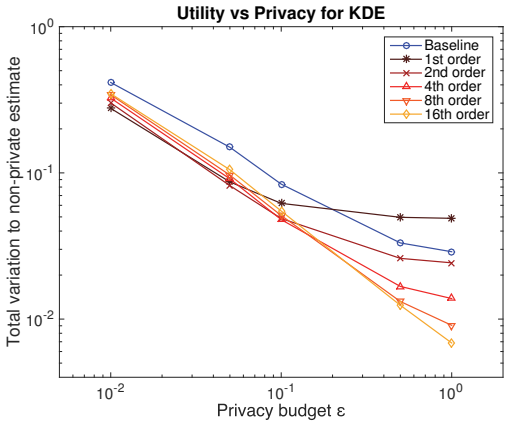

Figure 1: Private KDE with Gaussian kernel

Priestley-Chao Kernel Regression. For ease of exposition, consider $\ell = 1$. For constant $B > 0$, let $\mathcal{X} = [0,1] \times [-B, B]$ and $\mathcal{Y} = [0,1]$. Without loss of generality, consider datasets $\mathcal{D} = ((d_1, l_1), (d_2, l_2), \ldots, (d_n, l_n)) \in \mathcal{X}^n$, where $d_1 \leq d_2 \leq \ldots \leq d_n$, and for every $i \in \{1, \ldots, n\}$ there exists $j \neq i$ such that $|d_i - d_j| \leq c/n$, for a given (and publicly known) $0 < c = o(n)$. Small values of c restrict the data space under consideration, whereas $c = n$ would correspond to the general case $\mathcal{D} \in \mathcal{X}^n$. For kernel K and bandwidth $b > 0$, the Priestley-Chao kernel estimator (Priestley and Chao 1972; Benedetti 1977) is defined as $F_b(\mathcal{D}, y) = \frac{1}{b}\sum_{i=2}^n (d_i - d_{i-1}) K\left((y - d_i)/b\right) l_i$. This

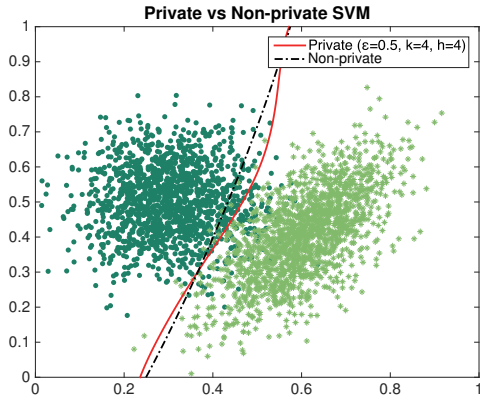

Figure 2: Private SVM with Gaussian kernel

function is not separable in $\mathcal{D}$ and

$$S(F_b) = \sup_{y \in \mathcal{Y}} S(F_b(\cdot, y)) \leq \frac{4Bc}{nb} \sup_{y \in [-1,1]} K\left(\frac{y}{b}\right).$$

If K is the Gaussian kernel, then with probability at least $1 - \beta$ the error introduced by the mechanism can be bounded by

$$O\left(\frac{c}{n\varepsilon b} \log(1/\beta)\right)^{\frac{h}{1+h}}.$$

Naive Bayes Classification. In this example we apply the Bernstein mechanism to a probabilistic learner. Without loss of generality, assume $X = [0,1]^\ell$, $\mathcal{X} = X \times \{l^+, l^-\}$, $\mathcal{Y} = X$ and $\mathcal{D} = ((\boldsymbol{d}_1, l_1), (\boldsymbol{d}_2, l_2), \ldots, (\boldsymbol{d}_n, l_n)) \in \mathcal{X}^n$. A naive Bayes classifier can be interpreted as $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ such that $F_{\mathcal{D}}(\boldsymbol{y}) \propto \mathbb{P}(\boldsymbol{y}|l^+, \mathcal{D})\mathbb{P}(l^+|\mathcal{D}) - \mathbb{P}(\boldsymbol{y}|l^-, \mathcal{D})\mathbb{P}(l^-|\mathcal{D})$. Predictions can then be made by assigning the instance $\boldsymbol{y}$ to the class l^+ (resp. l^-) if $F_{\mathcal{D}}(\boldsymbol{y}) \geq 0$ (resp. $F_{\mathcal{D}}(\boldsymbol{y}) < 0$). Since, for a class l, $\mathbb{P}(\boldsymbol{y}|l, \mathcal{D}) \propto \prod_{i=1}^{\ell} \mathbb{P}(y_i|l, \mathcal{D})$, it is easy to show that $F_{\mathcal{D}}(\cdot)$ is an (h, T)-smooth function whenever each likelihood is estimated using a Gaussian distribution or KDE (John and Langley 1995) (with a sufficiently smooth kernel). In the latter case, using a Gaussian kernel, the sensitivity of F can be bounded by $S(F) \leq 2(1/n + (2^\ell - 1)/(n\sqrt{2\pi}b))$, where b is the chosen bandwidth (*cf.* the full report Aldà and Rubinstein 2016). The error introduced by the Bernstein mechanism is thus bounded by

$$O\left(\frac{1}{n\varepsilon b} \log(1/\beta)\right)^{\frac{h}{\ell+h}},$$

with probability at least $1 - \beta$.

Regularized Empirical Risk Minimization. In the next examples, the functions we aim to release are implicitly defined by an algorithm. Let $X = [0,1]^\ell$, $\mathcal{X} = X \times [0,1]$ and $\mathcal{Y} = X$. Let L be a convex and locally M-Lipschitz (in the first argument) loss function. For $\mathcal{D} = ((\boldsymbol{d}_1, l_1), (\boldsymbol{d}_2, l_2), \ldots, (\boldsymbol{d}_n, l_n)) \in \mathcal{X}^n$, a regularized empirical risk minimization program with loss function L is defined as

$$\boldsymbol{w}^\star \in \arg\min_{\boldsymbol{w} \in \mathbb{R}^r} \frac{C}{n} \sum_{i=1}^{n} L(l_i, f_{\boldsymbol{w}}(\boldsymbol{d}_i)) + \frac{1}{2}\|\boldsymbol{w}\|_2^2, \quad (5)$$

where $f_{\boldsymbol{w}}(\boldsymbol{x}) = \langle \phi(\boldsymbol{x}), \boldsymbol{w} \rangle$ for a chosen feature mapping $\phi\colon X \to \mathbb{R}^r$ taking points from X to some (possibly infinite) r-dimensional feature space and a hyperplane normal $\boldsymbol{w} \in \mathbb{R}^r$. Let $K(\boldsymbol{x}, \boldsymbol{y}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{y}) \rangle$ be the kernel function induced by the feature mapping ϕ. The Representer Theorem (Kimeldorf and Wahba 1971) implies that the minimizer $\boldsymbol{w}^\star$ lies in the span of the functions $K(\cdot, \boldsymbol{d}_i) \in \mathcal{H}$, where $\mathcal{H}$ is a reproducing kernel Hilbert space (RKHS). Therefore, we consider $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ such that $F_{\mathcal{D}}(\boldsymbol{y}) = f_{\boldsymbol{w}^\star}(\boldsymbol{y}) = \sum_{i=1}^{n} \alpha_i l_i K(\boldsymbol{y}, \boldsymbol{d}_i)$, for some $\alpha_i \in \mathbb{R}$. An upper bound on the sensitivity of this function follows from an argument provided by Hall, Rinaldo, and Wasserman (2013) based on a technique of Bousquet and Elisseeff (2002). In particular, we have

$$S(F) = \sup_{\boldsymbol{y} \in \mathcal{Y}, \boldsymbol{w} \sim \boldsymbol{w}'} |f_{\boldsymbol{w}}(\boldsymbol{y}) - f_{\boldsymbol{w}'}(\boldsymbol{y})| \leq \frac{MC}{n} \sup_{\boldsymbol{y} \in \mathcal{Y}} K(\boldsymbol{y}, \boldsymbol{y})$$

If K is $(2h, T)$-smooth, the error introduced is bounded, with probability at least $1 - \beta$, by

$$O\left(\frac{MC \sup_{\boldsymbol{y} \in \mathcal{Y}} K(\boldsymbol{y}, \boldsymbol{y})}{n\varepsilon} \log(1/\beta)\right)^{\frac{h}{\ell+h}},$$

Note that this result holds with very mild assumptions, namely for any convex and locally M-Lipschitz loss function (*e.g.,* square-loss, log-loss, hinge-loss) and any bounded kernel K. Figure 2 depicts SVM learning with RBF kernel ($C = \sigma = 1$) on 1500 each of positive (negative) Gaussian data with mean $[0.3, 0.5]$ ($[0.6, 0.4]$) and covariance $[0.01, 0; 0, 0.01]$ ($0.01 * [1, 0.8; 0.8, 1.5]$) and demonstrates the mechanism's uniform approximation of predictions, best seen geometrically with the classifier's decision boundary.

Logistic Regression. Let now $X = \{\boldsymbol{x} \in [0,1]^\ell \colon \|\boldsymbol{x}\|_2 \leq 1\}$. Let furthermore $\mathcal{X} = X \times [0,1]$ and $\mathcal{Y} = [0,1]^\ell$. The logistic regressor can be seen as a function $F\colon \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ such that $F_{\mathcal{D}}(\boldsymbol{y}) = \langle \boldsymbol{w}^\star, \boldsymbol{y} \rangle$, where $\boldsymbol{w}^\star$ is the minimizer of (5) when ϕ is the identity mapping and the loss function is $L(l, \langle \boldsymbol{w}, \boldsymbol{d} \rangle) = \log\left(1 + e^{-l\langle \boldsymbol{w}, \boldsymbol{d} \rangle}\right)$. It is then possible to show that the error introduced by the Bernstein mechanism is bounded, with probability at least $1 - \beta$, by

$$O\left(\frac{C}{n\varepsilon} \log(1/\beta)\right),$$

since $F_{\mathcal{D}}(\boldsymbol{y})$ is a linear function. The prediction with the sigmoid function achieves the same error bound, since it is $1/4$-Lipschitz (*cf.* the full report Aldà and Rubinstein 2016).

7 Conclusions

In this paper we have considered the release of functions of test data and privacy-sensitive training data. We have presented a simple yet effective mechanism for this general

setting, that makes use of iterated Bernstein polynomials to approximate any regular function with perturbations applied to the resulting coefficients. Both ε-differential privacy and utility rates are proved in general for the mechanism, with corresponding lower bounds provided. A number of example learners are analyzed, demonstrating the Bernstein mechanism's versatility.

Acknowledgments. This work was partially completed while F. Aldà was visiting the University of Melbourne. Moreover, he acknowledges support of the DFG Research Training Group GRK 1817/1. The work of B. Rubinstein was supported by the Australian Research Council (DE160100584).

References

Aldà, F., and Rubinstein, B. I. P. 2016. The Bernstein mechanism: Function release under differential privacy. Technical report, ArXiv. https://arxiv.org/abs/1507.04499.

Bassily, R.; Smith, A.; and Thakurta, A. 2014. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *Proceedings of FOCS 2014*, 464–473. IEEE.

Benedetti, J. K. 1977. On the nonparametric estimation of regression functions. *Journal of the Royal Statistical Society. Series B (Methodological)* 248–253.

Blum, A.; Ligett, K.; and Roth, A. 2008. A learning theory approach to non-interactive database privacy. In *Proceedings of STOC 2008*, 609–618. ACM.

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

Chandrasekaran, K.; Thaler, J.; Ullman, J.; and Wan, A. 2014. Faster private release of marginals on small databases. In *Proceedings of ITCS 2014*, 387–402. ACM.

Chaudhuri, K., and Monteleoni, C. 2008. Privacy-preserving logistic regression. In *Proceedings of NIPS 2008*, 289–296.

Chaudhuri, K., and Vinterbo, S. A. 2013. A stability-based validation procedure for differentially private machine learning. In *Proceedings of NIPS 2013*, 2652–2660.

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

De, A. 2012. Lower bounds in differential privacy. In *Proceedings of TCC 2012*, 321–338.

Duchi, J.; Wainwright, M. J.; and Jordan, M. I. 2013. Local privacy and minimax bounds: Sharp rates for probability estimation. In *Proceedings of NIPS 2013*, 1529–1537.

Dwork, C., and Lei, J. 2009. Differential privacy and robust statistics. In *Proceedings of STOC 2009*, 371–380. ACM.

Dwork, C., and Roth, A. 2014. The algorithmic foundations of differential privacy. *Theoretical Computer Science* 9(3-4):211–407.

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. 2006. Calibrating noise to sensitivity in private data analysis. In *Proceedings of TCC 2006*, 265–284.

Hall, R.; Rinaldo, A.; and Wasserman, L. 2013. Differential privacy for functions and functional data. *The Journal of Machine Learning Research* 14(1):703–727.

Jain, P., and Thakurta, A. 2013. Differentially private learning with kernels. In *Proceedings of ICML 2013*, 118–126.

Jain, P., and Thakurta, A. G. 2014. (Near) dimension independent risk bounds for differentially private learning. In *Proceedings of ICML 2014*, 476–484.

John, G. H., and Langley, P. 1995. Estimating continuous distributions in Bayesian classifiers. In *Proceedings of UAI 1995*, 338–345.

Kac, M. 1938. Une remarque sur les polynomes de M. S. Bernstein. *Studia Mathematica* 7(1):49–51.

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

Lei, J. 2011. Differentially private m-estimators. In *Proceedings of NIPS 2011*, 361–369.

Lorentz, G. G. 1953. *Bernstein polynomials*. University of Toronto Press.

Mathé, P. 1999. Approximation of Hölder continuous functions by Bernstein polynomials. *American Mathematical Monthly* 106(6):568–574.

Micchelli, C. 1973. The saturation class and iterates of the Bernstein polynomials. *Journal of Approximation Theory* 8(1):1–18.

Priestley, M., and Chao, M. 1972. Non-parametric function fitting. *Journal of the Royal Statistical Society. Series B (Methodological)* 385–392.

Proschan, F. 1965. Peakedness of distributions of convex combinations. *The Annals of Mathematical Statistics* 1703–1706.

Rubinstein, B. I. P.; Bartlett, P. L.; Huang, L.; and Taft, N. 2012. Learning in a large function space: Privacy-preserving mechanisms for SVM learning. *Journal of Privacy and Confidentiality* 4(1):4.

Thaler, J.; Ullman, J.; and Vadhan, S. 2012. Faster algorithms for privately releasing marginals. In *Automata, Languages, and Programming*. Springer. 810–821.

Wang, Z.; Fan, K.; Zhang, J.; and Wang, L. 2013. Efficient algorithm for privately releasing smooth queries. In *Proceedings of NIPS 2013*, 782–790.

Wasserman, L., and Zhou, S. 2010. A statistical framework for differential privacy. *Journal of the American Statistical Association* 105(489):375–389.

Zhang, J.; Zhang, Z.; Xiao, X.; Yang, Y.; and Winslett, M. 2012. Functional mechanism: regression analysis under differential privacy. *Proceedings of the VLDB Endowment* 5(11):1364–1375.

Zhang, Z.; Rubinstein, B. I. P.; and Dimitrakakis, C. 2016. On the differential privacy of bayesian inference. In *Proceedings of AAAI 2016*, 2365–2371.