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.aldà@rub.de

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

Abstract

We investigate the general problem of function release under differential privacy. First, we develop a mechanism leveraging the iterated Bernstein operator for polynomial approximation of the target function, followed by coefficient perturbation. We then prove $\varepsilon$-differential privacy using access to the target’s sensitivity and function evaluation only—admitting treatment of functions described explicitly or implicitly. Moreover, under weak regularity conditions—either Hölder continuity or bounded derivatives—we establish fast rates on utility measured by high-probability uniform approximation. We provide lower bounds on the utility achievable for any functional mechanism that is $\varepsilon$-differentially private. The generality of our mechanism is demonstrated by the analysis of a number of example learners, including non-parametric estimators and regularized empirical risk minimization. Competitive rates are demonstrated for kernel density estimation; and $\varepsilon$-differential privacy is achieved for a broader class of support vector machines than known previously.

1 Introduction

A major challenge in statistics and machine learning is balancing the conflicting objectives of releasing accurate data analyses while protecting data privacy. A U.S. hospital running an effective clinical trial must still guarantee patient privacy as demanded by HIPAA laws; while a media-streaming service’s recommendations that drive user engagement, should respect user privacy for risk of litigation. In recent years, differential privacy [10] has emerged as the standard paradigm for privacy-preserving statistical analysis. It provides formal guarantees that aggregate statistics output by a randomized mechanism are not significantly influenced by the presence or absence of an individual input datum.

In this paper, we aim to privately release functions that depend on privacy-sensitive training data, that can be subsequently evaluated on arbitrary test points. This non-interactive setting matches a wide variety of machine learning tasks such as non-parametric methods (kernel density estimation and regression) where the function of train and test data is explicit, to generalized linear models and support vector machines where the function is only implicitly defined by an algorithm. We develop a mechanism that addresses this goal, and 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. As a result, our mechanism applies to releasing explicitly and implicitly defined functions. Our technique can be regarded as a function-valued version of the popular Laplace mechanism [10], which generically privatizes vector-valued mappings assuming the same oracle access to function evaluation and sensitivity.

Polynomial approximation has proven useful in differential privacy settings outside function release [22, 4]. Towards the goal of private function release, few previous attempts have been made [24, 11]. Hall et al. [11] add Gaussian process noise which only yields a weaker form of
privacy, namely \((\varepsilon, \delta)\)-differential privacy, and does not admit general utility rates. Zhang et al. \[24\] 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.

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 yields approximations that are pointwise convex combinations of the function evaluations on a cover. As a result, performing privacy-preserving perturbations to the approximation’s coefficients, permit us to control utility and achieve fast convergence rates.

Wang et al. \[23\] propose a mechanism that releases a summary of data in a trigonometric basis, able to respond to subsequent queries that are smooth as in our setting, but are in addition required to be separable in the training dataset as in \[24\]. A natural application is kernel density estimation, which would achieve a rate of \(O \left( \log(1/\beta)/(n\varepsilon)^{h/(\ell+\delta)} \right)\) as does our approach. Private KDE has also been explored in various other settings \[9\] and under weaker notions of utility \[11\].

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 \[21, 13, 12, 1\] require finite-dimensional feature mapping or translation-invariant kernels. Hall et al. \[11\] consider more general mappings but provide \((\varepsilon, \delta)\)-differential privacy. Our treatment of regularized ERM extends to kernels that may be translation-invariant with infinite-dimensional mappings, while providing stronger privacy guarantees.

Finally, we provide a lower bound that fundamentally limits utility under private function release, partly resolving a question posed by Hall et al. \[11\]. This matches (up to logarithmic factors) our upper bound in the linear case.

2 Preliminaries

Problem setting. In this work, we consider \(X\) an arbitrary (possibly infinite) domain and \(D \in X^n\) a database consisting of \(n\) points in \(X\). We refer to \(n\) as the size of the database \(D\). For a positive integer \(\ell\), let \(Y = [0, 1]^\ell\) be a set of query points and \(F : X^n \times Y \rightarrow \mathbb{R}\) the function we aim to release. Once the database \(D\) is fixed, we denote by \(F_D = F(D, \cdot)\) the function parametrized by \(D\). For example, \(D\) might represent a training set—over \(X\) a product space of feature vectors and labels—with \(Y\) representing test points from the same feature space. In Section 3, we show how to privately release the function \(F_D\) and provide alternative error bounds depending on regularity of \(F\).

Definition 1. Let \(0 \leq \gamma \leq 1\) and \(L > 0\). A function \(f : [0, 1]^\ell \rightarrow \mathbb{R}\) is \((\gamma, L)\)-Hölder continuous if, for every \(x, y \in [0, 1]^\ell\), \(|f(x) - f(y)| \leq L|x - y|^\gamma\). When \(\gamma = 1\), we refer to \(f\) as \(L\)-Lipschitz.

Definition 2. Let \(h\) be a positive integer and \(T > 0\). A function \(f : [0, 1]^\ell \rightarrow \mathbb{R}\) is \((h, T)\)-smooth if it is \(C^h([0, 1]^\ell)\) and its partial derivatives up to order \(h\) are all bounded by \(T\).

Our goal is to develop a private release mechanism for the function \(F_D\) in the non-interactive setting. A non-interactive mechanism takes a function \(F\) and a database \(D\) as inputs and outputs a synopsis \(M\) which can be used to evaluate the function \(F_D\) on \(Y\) without accessing the database \(D\) further.

Differential privacy. To provide strong privacy guarantees on the release of \(F_D\), we adopt the well-established notion of differential privacy.

Definition 3 (\[10\]). Let \(\mathcal{R}\) be a (possibly infinite) set of responses. A mechanism \(\mathcal{M} : X^n \rightarrow \mathcal{Y}\) (meaning that, for every \(D \in X^n = \bigcup_{n \geq 0} X^n\), \(M(D)\) is an \(\mathcal{R}\)-valued random variable) is said to provide \(\varepsilon\)-differential privacy for \(\varepsilon > 0\) if, for every \(n \in \mathbb{N}\), for every pair \((D, D') \in X^n \times X^n\) of databases differing in one entry only (henceforth denoted by \(D \sim D'\)), and for every measurable \(S \subseteq \mathcal{R}\), we have \(\Pr[M(D) \in S] \leq e^{\varepsilon} \Pr[M(D') \in S]\).

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 \[10\] is a generic tool for differential privacy: adding zero-mean Laplace noise\(^4\) to a vector-valued function provides privacy if the noise is calibrated to the function’s sensitivity.

\(^4\)A \(\text{Lap}(\lambda)\)-distributed real random variable \(Y\) has probability density proportional to \(\exp(-|y|/\lambda)\).
We briefly introduce the univariate Bernstein basis polynomials and state some of their properties. Moreover, if \( f' \) is \( \frac{\gamma}{L} \)-Hölder continuous for every database \( D \in X^n \), with error rate scaling as \( \alpha = O \left( \frac{S(F)}{\varepsilon} \log(1/\beta) \right)^{\frac{1}{2}} \); or

(iii) If \( f' \) is linear for every database \( D \in X^n \), error scaling as \( \alpha = O \left( \frac{S(F)}{\varepsilon} \log(1/\beta) \right)^{\frac{1}{2}} \).

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

The proof of this result is detailed in Section 4. The mechanism makes use of the iterated Bernstein polynomial of \( f' \), which we introduce next (for a comprehensive survey refer to [16] [18]). This approximation consists of a linear combination of so-called Bernstein basis polynomials, whose coefficients are evaluations of target function \( f' \) on a lattice cover of \( Y \).
Proposition 2 (16). For every $x \in [0,1]$, any positive integer $k$ and $0 \leq \nu \leq k$, we have $b_{\nu, k}(x) \geq 0$ and $\sum_{\nu=0}^{k} b_{\nu, k}(x) = 1$.

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

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

We call $B_k$ the Bernstein operator. It maps a function $f$, defined on $[0,1]$, to $B_k f$, where the function $B_k f$ evaluated at $x$ is $B_k f(x)$. Note that the Bernstein operator is linear and if $f(x) \in [a_1, a_2]$ for every $x \in [0,1]$, then from Proposition 2 it follows that $B_k f(x) \in [a_1, a_2]$ for every positive integer $k$ and $x \in [0,1]$. Moreover, it is not hard to show that any linear function is a fixed point for $B_k$. For completeness, we provide a short proof in Appendix A.

Definition 8 (18). Let $h$ be a positive integer. The iterated Bernstein operator of order $h$ is defined as the sequence of linear operators $B_k^{(h)} = I - (I - B_h)^h = \sum_{i=1}^{h} \binom{h}{i} (-1)^{i-1} B_k^{i-1}$, where $I = B_k^0$ denotes the identity operator. The iterated Bernstein polynomial of order $h$ can then be computed as:

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

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

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 $\sum_{\nu=0}^{k} b_{\nu, k}^{(h)}(x) = 1$ for every $x \in [0,1]$.

4 Proof of the Main Theorem

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 $\varepsilon$-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.

4.1 Unidimensional case ($\ell = 1$)

Let us fix $k$, a positive integer. As described in Algorithm 1 the Bernstein mechanism perturbs the function $F_\mathcal{D}$ on a cover of the interval $[0,1]$.

Lemma 3. Let $\varepsilon > 0$. Then the Bernstein mechanism $\mathcal{M}$ provides $\varepsilon$-differential privacy.

The proof of Lemma 3 follows from an application of Lemma 1. We provide the full argument in Appendix B. In order to analyze the accuracy of our mechanism, we denote by $B_k^{(h)}(F_\mathcal{D}; x) = \sum_{\nu=0}^{k} [F_\mathcal{D}(\nu/k) + Y_\nu] b_{\nu, k}^{(h)}(x)$ the iterated Bernstein polynomial of order $h$ constructed using the coefficients output by the mechanism $\mathcal{M}$. The error $\alpha$ introduced by the mechanism can be expressed as follows:

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

$$\leq \max_{x \in [0,1]} \left| B_k^{(h)}(F_\mathcal{D}; x) - B_k^{(h)}(F_\mathcal{D}; x) \right| + \max_{x \in [0,1]} \left| F_\mathcal{D}(x) - B_k^{(h)}(F_\mathcal{D}; x) \right|. \quad (2)$$

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

Proposition 4. Let $Y_0, \ldots, Y_k \overset{iid}{\sim} \text{Lap}(\lambda), \, \delta \geq 0,$ and $C_h$ be a constant depending on $h$ only. Then:

$$\Pr \left[ \max_{x \in [0,1]} \left| \sum_{\nu=0}^{k} Y_\nu b_{\nu, k}^{(h)}(x) \right| \geq \delta \right] \leq \exp \left( \frac{-\delta}{C_h \lambda} \right).$$

4
The proof of Proposition 4 follows from a result of Proschan \cite{proschan2010}. For completeness, we give the full proof in Appendix C. Proposition 4 implies that with probability at least $1 - \beta$ the first summand in Equation (2) is bounded by $O(S(F)k \log(1/\beta)/\varepsilon)$. In order to bound the second summand we make use of the following convergence rates.

**Theorem 5** \cite{14,17}. Let $0 < \gamma \leq 1$ and $L > 0$. If $f : [0, 1] \to \mathbb{R}$ is a $(\gamma, L)$-Hölder continuous function, then $\left| f(x) - B_k^{(1)}(f; x) \right| \leq L(4k)^{\gamma/2}$ for all positive integers $k$ and $x \in [0, 1]$.

**Theorem 6** \cite{18}. Let $h$ be a positive integer and $T > 0$. If $f : [0, 1] \to \mathbb{R}$ is a $(2h, T)$-smooth function, then, for all positive integers $k$ and $x \in [0, 1]$, $\left| f(x) - B_k^{(h)}(f; x) \right| \leq TD_hk^{-h}$, where $D_h$ is a constant independent of $k, f$, and $x \in [0, 1]$.

According to the regularity of $F_D$, the second summand in Equation (2) can then be bounded by a decreasing function $g(k)$. All in all, the error in Equation (1) can be bounded as follows:

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

Since 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

$$g(k) = \frac{S(F)k}{\varepsilon} \log(1/\beta). \quad (4)$$

Solving Equation (4) with the bounds for $g(k)$ provided in Theorems 5 and 6 and substituting the thus obtained value of $k$ into (3) prove the first two statements. The bound when $F_D$ is linear follows from the fact that for $h = 1$ and $k = 1$ the approximation error in Equation (2) is zero, since linear functions are fixed points of $B_1^{(1)}$. The error is thus bounded by $O(S(F)\log(1/\beta)/\varepsilon)$. The running time of the mechanism and the running time for answering a query is linear in $k$ and hence upper bounded by a polynomial in $n$ and $1/\varepsilon$, if $1/S(F) \leq \text{poly}(n, 1/\varepsilon)$.

### 4.2 Multidimensional case ($\ell > 1$)

In order to extend the proof of the Main Theorem to $\ell > 1$, we need to introduce the iterated Bernstein polynomial of a multivariate function $f : [0, 1]^\ell \to \mathbb{R}$.

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

$$B_{k_1,\ldots,k_\ell}^{(h)}(f; x_1, \ldots, x_\ell) = \sum_{j=1}^{\ell} \sum_{\nu_j=0}^{k_j} \binom{k_j}{\nu_j} f \left( \frac{\nu_j}{k_j}, \ldots, \frac{\nu_\ell}{k_\ell} \right) \prod_{i=1}^\ell b_{\nu_i,k_i}^{(h)}(x_i).$$

For ease of exposition, we consider $k_1 = \ldots = k_\ell = k$. By induction, it is possible to show that the approximation error of $B_k^{(h)}$ can be bounded by $\ell g(k)$, if the error of the corresponding univariate polynomial is bounded by $g(k)$. For completeness, we provide a proof in Appendix C. Moreover, note that the mechanism $\mathcal{M}$ now outputs a $(k + 1)^\ell$-dimensional vector, where each component is perturbed by Laplace-distributed noise with parameter $\lambda = S(F)(k + 1)^\ell/\varepsilon$. Privacy then follows immediately from Lemma 11. In order to conclude the error analysis, we observe that the error bound provided in Proposition 4 can be extended, too.

**Proposition 7.** For $j \in \{1, \ldots, \ell\}$ and $\nu_j = 0, \ldots, k$, let $Y_{\nu_j} \overset{iid}{\sim} \text{Lap}(\lambda)$, let $\delta \geq 0$, and constant $C_{h,\ell}$ depend only on $h, \ell$. Then:

$$\Pr \left[ \max_{x \in [0, 1]^\ell} \left| \sum_{j=1}^{\ell} \sum_{\nu_j=0}^{k_j} Y_{\nu_j} \prod_{i=1}^{\ell} b_{\nu_i,k_i}^{(h)}(x_i) \right| \geq \delta \right] \leq \exp \left( -\frac{\delta}{C_{h,\ell} \lambda} \right).$$
We first observe that, for every \( j \), we actually show that this holds for every \( y \). Therefore, we only need to show that there exist a suitable sequence of databases whose value is \( x \), \( s \) and \( k \) substituting the thus obtained value of \( k \) into (5) give the first two statements of the Main Theorem. The bound for linear functions follows from the fact that the approximation error is zero for \( h = 1 \) and \( k = 1 \). The analysis of the running time of the mechanism and the running time for answering a query is straightforward and hence omitted.

5 Lower bound

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

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

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

**Proposition 9** ([8]). Assume \( D_1, D_2, \ldots, D_{2^d} \in (\mathbb{Z}^+)^N \) such that, for every \( i \), \( \|D_i\|_1 \leq n \) and, for \( i \neq j \), \( \|D_i - D_j\|_1 \leq \Delta \). Moreover, let \( f : (\mathbb{Z}^+)^N \to \mathbb{R} \) be such that for any \( i \neq j \), \( \|f(D_i) - f(D_j)\|_\infty \geq \eta \). If \( \Delta \leq (n + 1)/\varepsilon \), then any mechanism which is \( \varepsilon \)-differentially private for the query \( f \) on databases of size \( n \) introduces an error which is \( \Omega(\eta) \), with high probability.

Therefore, we only need to show that there exist a suitable sequence of databases \( D_1, D_2, \ldots, D_{2^d} \), a function \( F : \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R} \) and a \( y \in \mathcal{Y} \) such that \( F(\cdot,y) \) satisfies the assumptions of Proposition 9. We actually show that this holds for every \( y \in \mathcal{Y} \). Let \( \varepsilon > 0 \) and \( V \) be a non-negative integer. We define \( \mathcal{X} = \{0,1/(V + 8),2/(V + 8),\ldots,1\}^d \). Note that \( N = |\mathcal{X}| = (V + 9)^d \). Let furthermore \( \eta = |1/\varepsilon| \) and \( n = V + c \). The function \( F : \mathcal{X}^n \times [0,1]^d \to \mathbb{R} \) we consider is defined as follows:

\[
F(D,y) = \eta(d_0 + d_1 + \ldots + d_{N-7} + 2d_{N-6} + \ldots + 8d_N + \langle y, \bar{y} \rangle),
\]

where \( d_i \) corresponds to the number of entries in \( D \) whose value is \( x_i \), for every \( x_i \in \mathcal{X} \). For \( s = 3 \), we consider the sequence of databases \( D_1, D_2, \ldots, D_8 \), where, for \( j \in \{1,2,\ldots,8\} \), \( d_i \in D_j \) is such that

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

We first observe that, for every \( j \in \{1,2,\ldots,8\} \), \( \|D_j\|_1 = n \). Moreover, for \( i \neq j \), \( \|D_i - D_j\|_1 = 2\eta \leq 2/\varepsilon \). Finally, for \( i \neq j \), \( |F(D_i,y) - F(D_j,y)| \geq c\eta \) for every \( y \in [0,1]^d \). Since \( S(F) = 7\eta \), Proposition 9 implies that, with high probability, any \( \varepsilon \)-differentially private mechanism approximating \( F \) must introduce error \( \Omega(S(F)/\varepsilon) \).

6 Examples

In this section, we demonstrate the versatility of the Bernstein mechanism through the analysis of a number of example learners.
Kernel density estimation. Let $\mathcal{X} = \mathcal{Y} = [0, 1]^\ell$ and $D = (d_1, d_2, \ldots, d_n) \in \mathcal{X}^n$. For a given kernel $K_H$, with bandwidth $H$ (a symmetric and positive definite $\ell \times \ell$ matrix), the kernel density estimator $F_H : \mathcal{X}^n \times \mathcal{Y} \to \mathbb{R}$ is defined as $F_H(D, y) = \frac{1}{n} \sum_{i=1}^{n} K_H(y - d_i)$. It is easy to see that $S(F_H) \leq \sup_{x \in [-1,1]} K_H(x)/n$. 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 $F_H(D, \cdot)$ is an $(h, T)$-smooth function for any positive integer $h$. Hence the error introduced by the mechanism is

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

with probability at least $1 - \beta$. In Figure 1 we display the utility (averaged over 1000 repeats) of the Bernstein mechanism 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. Accuracy improves for increasing $h$, except for sufficiently large perturbations (very small $\varepsilon$) which affect approximations with larger derivatives (larger $h$) greater. Private cross validation \cite{7, 6} can be used to tune $h$. We conclude noting that the same error bounds can be provided by the mechanism of Wang et al. \cite{23}, since the function $F_H(D, \cdot)$ is separable in the training set $D$, i.e. $F_H(D, \cdot) = \sum_{d \in D} f_H(d, \cdot)$, however, this assumption is overly restrictive for many applications. In the following, we discuss a few such cases and show how the Bernstein mechanism can still be successfully applied.

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 $D = ((d_1, l_1), (d_2, l_2), \ldots, (d_n, l_n)) \in \mathcal{X}^n$, 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) $0 < c \leq n$. Small values of $c$ restrict the data space under consideration, whereas $c = n$ corresponds to the general case $D \in \mathcal{X}^n$. For kernel $K$ and bandwidth $\delta > 0$, the Priestley-Chao kernel estimator \cite{19, 2} is defined as $F_\delta(D, y) = \frac{1}{\delta} \sum_{i=2}^{n} (d_i - d_{i-1}) K((y - d_i)/\delta) l_i$. This function is not separable in $D$ and $\sup_{y \in \mathcal{Y}} S(F_\delta) = \sup_{y \in \mathcal{Y}} S(F_\delta(y, \cdot)) \leq \frac{4Bc}{n\delta} \sup_{x \in [-1,1]} K \left( \frac{x}{\delta} \right)$. If $K$ is the Gaussian kernel, then with probability at least $1 - \beta$ the error introduced by the mechanism can be bounded by

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

Logistic regression. In the next two examples, the functions we aim to release are implicitly defined by an algorithm. Let $X = \{ x \in [0, 1]^\ell : \|x\|_2 \leq 1 \}$. Let furthermore $\mathcal{X} = X \times [0, 1]$ and $\mathcal{Y} = [0, 1]^\ell$. The logistic regressor can be seen as a function $F : \mathcal{X}^n \times \mathcal{Y} \to [0, 1]$ such that, for $D = ((d_1, l_1), (d_2, l_2), \ldots, (d_n, l_n)) \in \mathcal{X}^n$, $F_D(y) = 1/(1 + \exp(-\langle w^*, y \rangle))$, where $w^*$ is such
that
\[ w^* \in \arg \min_{w \in \mathbb{R}^d} \frac{C}{n} \sum_{i=1}^n \log \left( 1 + e^{-l_i(w, d_i)} \right) + \frac{1}{2} \|w\|_2^2. \]

In order to compute \( S(F) \), we first observe that the sigmoid function is \( 1/4 \)-Lipschitz. Denoting by \( w \sim w' \) the minimizers obtained from input databases \( D \sim D' \), we have
\[ S(F) \leq \sup_{y \in Y, w \sim w'} \frac{1}{4} |(w - w', y)| \leq \sup_{y \in Y, w \sim w'} \frac{1}{4} \|w - w'\|_2 \|y\|_2, \]
where the last inequality follows from an application of the Cauchy-Schwarz inequality. Chaudhuri et al. [15] implies that the minimizer \( \varepsilon \sim \gamma \) demonstrating the Bernstein mechanism’s versatility.

\[ \varepsilon \sim \gamma \]

We note that defining \( G_D(y) = (w^*, y) \) the previous bound can be improved to
\[ O \left( \frac{C}{n \varepsilon} \log(1/\beta) \right)^{\frac{h}{2-\nu}}, \]

since \( S(G) \leq 2C\sqrt{\lambda}/n \) and \( G_D(y) \) is a linear function. The prediction with the sigmoid function achieves the same error bound, being \( 1/4 \)-Lipschitz.

**Regularized empirical risk minimization.** Let now \( X = [0, 1]^d, \mathcal{X} = X \times [0, 1] \) and \( Y = X \). Let \( L \) be a convex and locally \( M \)-Lipschitz (in the first argument) loss function. For \( D = ((d_1, l_1), (d_2, l_2), \ldots, (d_n, l_n)) \in \mathcal{X}^n \), a regularized empirical risk minimization program with loss function \( L \) is defined as
\[ w^* \in \arg \min_{w \in \mathbb{R}^d} \frac{C}{n} \sum_{i=1}^n L(l_i, f_w(d_i)) + \frac{1}{2} \|w\|_2^2, \]

where \( f_w(x) = \langle \phi(x), w \rangle \) for a chosen feature mapping \( \phi: X \rightarrow \mathbb{R}^F \) taking points from \( X \) to some (possibly infinite) \( F \)-dimensional feature space and a hyperplane normal \( w \in \mathbb{R}^F \). Let \( K(x, y) = \langle \phi(x), \phi(y) \rangle \) be the kernel function induced by the feature mapping \( \phi \). The Representer Theorem [13] implies that the minimizer \( w^* \) lies in the span of the functions \( K(\cdot, d_i) \in \mathcal{H} \), where \( \mathcal{H} \) is a reproducing Hilbert space (RKHS). Therefore, we consider \( F: \mathcal{X}^n \times Y \rightarrow \mathcal{Y} \) such that \( F_D(y) = f_{w^*}(y) = \sum_{i=1}^n \alpha_i d_i K(y, d_i) \), for some \( \alpha_i \in \mathbb{R} \). An upper bound on the sensitivity of this function follows from an argument provided by Hall et al. [11] based on a technique of Bousquet and Elisseeff [3]. In particular, we have
\[ S(F) = \sup_{y \in Y, w \sim w'} |f_w(y) - f_w'(y)| \leq \frac{MC}{n} \sup_{y \in Y} K(y, y). \]

If \( K \) is \((2h, T)\)-smooth, the error introduced is bounded, with probability at least \( 1 - \beta \), by
\[ O \left( \frac{MC \sup_{y \in Y} K(y, y)}{n \varepsilon} \log(1/\beta) \right)^{\frac{h}{2-\nu}}, \]

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 3 depicts SVM learning with RBF kernel \((C = \sigma = 1)\) on 1000 each of positive (negative) Gaussian synthetic data with mean \([0.3, 0.5]\) \([(0.6, 0.4)]\), covariance \([0.01, 0; 0, 0.01]\) \((0.01 * [1, 0.8; 0.8, 1.5])\). On a 200 point i.i.d. test, non-private vs. private SVM both achieve 0.01 misclassification rate.

**7 Conclusions**

In this paper we have considered the setting of releasing 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 \( \varepsilon \)-differential privacy and utility rates are proved in general for the mechanism, with corresponding lower bounds provided, and a number of example learners 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
[1] R. Bassily, A. Smith, and A. Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In Proceedings of the 55th Annual Symposium on Foundations of Computer Science, pages 464–473. IEEE, 2014.

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

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

[4] K. Chandrasekaran, J. Thaler, J. Ullman, and A. Wan. Faster private release of marginals on small databases. In Proceedings of the 5th Conference on Innovations in Theoretical Computer Science, pages 387–402. ACM, 2014.

[5] K. Chaudhuri and C. Monteleoni. Privacy-preserving logistic regression. In Advances in Neural Information Processing Systems, pages 289–296, 2008.

[6] K. Chaudhuri and S. A. Vinterbo. A stability-based validation procedure for differentially private machine learning. In Advances in Neural Information Processing Systems, pages 2652–2660, 2013.

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

[8] A. De. Lower bounds in differential privacy. In Proceedings of the 9th Theory of Cryptography Conference, pages 321–338, 2012.

[9] J. Duchi, M. J. Wainwright, and M. I. Jordan. Local privacy and minimax bounds: Sharp rates for probability estimation. In Advances in Neural Information Processing Systems, pages 1529–1537, 2013.

[10] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In Proceedings of the 3rd Theory of Cryptography Conference, pages 265–284, 2006.

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

[12] P. Jain and A. Thakurta. Differentially private learning with kernels. In Proceedings of the 30th International Conference on Machine Learning, pages 118–126, 2013.

[13] P. Jain and A. G. Thakurta. (Near) dimension independent risk bounds for differentially private learning. In Proceedings of the 31st International Conference on Machine Learning, pages 476–484, 2014.

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

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

[16] G. G. Lorentz. Bernstein polynomials. University of Toronto Press, 1953.

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

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

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

[20] F. Prosch. Peakedness of distributions of convex combinations. The Annals of Mathematical Statistics, pages 1703–1706, 1965.
A Fixed points of the Bernstein operator

Although this is a classical result, we show for completeness that linear functions are fixed points of the Bernstein operator \( B_k = B_k^{(1)} \), for \( k \geq 1 \). Let \( f(x) = mx + q \), for \( m, q \in \mathbb{R} \) and \( x \in [0, 1] \). We have

\[
B_k(f; x) = \sum_{\nu=0}^{k} f\left(\frac{\nu}{k}\right) b_{\nu,k}(x) \\
= \frac{m}{k} \sum_{\nu=0}^{k} \nu b_{\nu,k}(x) + q \sum_{\nu=0}^{k} b_{\nu,k}(x) \\
= mx + q,
\]

since \( \sum_{\nu=0}^{k} b_{\nu,k}(x) = 1 \) and \( \sum_{\nu=0}^{k} \nu b_{\nu,k}(x) = kx \).

B Proof of Lemma 3

Let \( D' \in \mathcal{X}^n \) be a second database differing from \( D \) in one entry only. Let furthermore \( \psi: \mathcal{X}^n \rightarrow \mathbb{R}^{k+1} \) be the map defined by

\[
\psi(D) = \left( F_D\left(\frac{0}{k}\right), F_D\left(\frac{1}{k}\right), \ldots, F_D\left(\frac{k}{k}\right) \right) .
\]

Then

\[
S(\psi) = \sup_{D \sim D'} \| \psi(D) - \psi(D') \|_1 \leq \sum_{\nu=0}^{k} \sup_{D \sim D'} \left| F_D\left(\frac{\nu}{k}\right) - F_D'\left(\frac{\nu}{k}\right) \right| \leq S(F)(k + 1).
\]

According to Lemma 1 (applied with \( k + 1 \) in place of \( d \)), the mechanism \( \mathcal{M} \) provides \( \varepsilon \)-differential privacy.

C Proof of Proposition 4

In order to prove the proposition, we make use of the following result.

**Theorem 10 (20).** Suppose that \( f: \mathbb{R} \rightarrow [0, 1] \) is a log-concave density function such that \( f(x) = f(-x) \) for every \( x \in \mathbb{R} \). Let \( Y_1, \ldots, Y_m \) be i.i.d random variables with density \( f \), and suppose that \( (a_1, \ldots, a_m), (b_1, \ldots, b_m) \in [0, 1]^m \) satisfy

1. \( a_1 \geq a_2 \geq \ldots \geq a_m, b_1 \geq b_2 \geq \ldots \geq b_m \);
2. \( \sum_{i=1}^{k} b_i \leq \sum_{i=1}^{k} a_i \) for \( k = 1, \ldots, m - 1 \);
3. \( \sum_{i=1}^{m} a_i = \sum_{i=1}^{m} b_i = 1 \).

Then, for all \( \delta \geq 0 \)

\[
\Pr \left[ \left| \sum_{i=1}^{m} b_i Y_i \right| \geq \delta \right] < \Pr \left[ \left| \sum_{i=1}^{m} a_i Y_i \right| \geq \delta \right].
\]
Choosing \( a_1 = 1 \) and \( a_j = 0 \) for \( j = 2, \ldots, m \), Theorem 10 implies
\[
\Pr \left[ \sum_{i=1}^{m} b_i Y_i \geq \delta \right] < \Pr \left[ \|Y_1\| \geq \delta \right].
\]
for every \( (b_1, \ldots, b_m) \in [0, 1]^m \) which satisfies \( \sum_{i=1}^{m} b_i = 1 \). We then observe that the density function \( h(y) = \exp(-|y|/\lambda)/(2\lambda) \) of the Laplace distribution is symmetric and log-concave. If \( Y_i \sim \text{Lap}(\lambda) \) are i.i.d. random variables for \( i = 1, \ldots, m \), the right-hand side of Equation (6) satisfies
\[
\Pr \left[ \|Y_1\| \geq \delta \right] = \exp \left( -\frac{\delta}{\lambda} \right).
\]

Although the bases \( b^{(h)}_{\nu,k} \) are not always positive for \( h \geq 2 \), we observe that, for \( x \in [0, 1] \), \( Z(x) = \sum_{\nu=0}^{k} Y_{\nu} b^{(h)}_{\nu,k}(x) \) and \( Z'(x) = \sum_{\nu=0}^{k} Y_{\nu} |b^{(h)}_{\nu,k}(x)| \) have the same distribution, since the random variables \( Y_{\nu} \) are i.i.d. and symmetric around zero. We can thus restrict our analysis to \( Z'(x) \). For \( x \in [0, 1] \), let \( U(x) = \sum_{\nu=0}^{k} |b^{(h)}_{\nu,k}(x)| \). We first note that
\[
U(x) = \sum_{\nu=0}^{k} \left( \sum_{i=1}^{h} \binom{h}{i} (-1)^{i-1} B_k^{i-1}(b_{\nu,k};x) \right)
\leq \sum_{\nu=0}^{k} \left( \sum_{i=1}^{h} \binom{h}{i} B_k^{i-1}(b_{\nu,k};x) \right)
= k \sum_{\nu=0}^{k} \left( \sum_{i=1}^{h} \binom{h}{i} B_k^{i-1}(b_{\nu,k};x) \right)
= \sum_{i=1}^{h} \left( \sum_{\nu=0}^{k} \binom{h}{i} B_k^{i-1}(b_{\nu,k};x) \right)
= \sum_{i=1}^{h} \binom{h}{i} B_k^{i-1}(b_{\nu,k};x)
= 2^h - 1.
\]

According to Equations (6) and (7), for every \( x \in [0, 1] \) and \( \delta' \geq 0 \) we have
\[
\Pr \left[ \frac{1}{U(x)} \sum_{\nu=0}^{k} Y_{\nu} |b^{(h)}_{\nu,k}(x)| \geq \delta' \right] \leq \exp \left( -\frac{\delta'}{\lambda} \right).
\]
Choosing \( \delta = U(x)\delta' \), we get
\[
\Pr \left[ \sum_{\nu=0}^{k} Y_{\nu} |b^{(h)}_{\nu,k}(x)| \geq \delta \right] \leq \exp \left( -\frac{\delta}{U(x)\lambda} \right) \leq \exp \left( -\frac{\delta}{(2^h - 1)\lambda} \right),
\]
for every \( x \in [0, 1] \), concluding the proof.

D Proof of Proposition 7

The proof of the proposition follows from the same argument provided in Appendix C with some minor changes. In particular, it suffices to provide a tail bound for
\[
\max_{x \in [0, 1]^{\ell}} \left| \sum_{j=1}^{k} \sum_{\nu_j=0}^{\nu} Y_{\nu_j} \prod_{i=1}^{\ell} b^{(h)}_{\nu_i,k}(x_i) \right|,
\]
since, as observed in Appendix C, the random variables $Y_{\nu_j}$ are i.i.d. and symmetric around zero. In order to apply Theorem 10 and conclude the proof, we need to upper bound

$$U(x) = \sum_{j=1}^k \sum_{\nu_j=0}^k \prod_{i=1}^\ell \left| b_{\nu_j,k}^{(h)}(x_i) \right|,$$

for every $x \in [0, 1]^\ell$. We have

$$U(x) = \sum_{j=1}^k \sum_{\nu_j=0}^k \prod_{i=1}^\ell \left| b_{\nu_j,k}^{(h)}(x_i) \right|$$

$$= \sum_{\nu_1=0}^k \left| b_{\nu_1,k}^{(h)}(x_1) \right| \left( \sum_{\nu_2=0}^k \left| b_{\nu_2,k}^{(h)}(x_2) \right| \cdots \left( \sum_{\nu_{\ell-1}=0}^k \left| b_{\nu_{\ell-1},k}^{(h)}(x_{\ell-1}) \right| \right) \right)$$

$$\leq (2^h - 1)^\ell,$$

since, according to Equation (8), $\sum_{\nu_j=0}^k \left| b_{\nu_j,k}^{(h)}(x_j) \right| \leq (2^h - 1)$ for every $j \in \{1, \ldots, \ell\}$. The rest of the proof follows from the same computations done at the end of Appendix C.

### E Approximation error of multivariate Bernstein polynomials

In what follows, we assume that $f : [0, 1]^\ell \to \mathbb{R}$ is a $(\gamma, L)$-Hölder continuous function. The proof for $(h, T)$-smooth functions follows the same argument, with minor changes. The argument we present here is by induction on $\ell$. The base case ($\ell = 1$) follows from the fact that the Bernstein polynomial $B_k(f; x_1)$ converge uniformly to $f$ in the interval $[0, 1]$, as shown in Theorem 5. Assume now

$$|B_k(f; x_1, \ldots, x_\ell) - f(x_1, \ldots, x_\ell)| \leq \ell L \left( \frac{1}{4k} \right)^{\gamma/2},$$

for every $(x_1, \ldots, x_\ell) \in [0, 1]^\ell$. Let $f : [0, 1]^{\ell+1} \to \mathbb{R}$ be a $(\gamma, L)$-Hölder continuous function and let $B_k(f; x_1, \ldots, x_{\ell+1})$ be the corresponding Bernstein polynomial. For every $(x_1, \ldots, x_{\ell+1}) \in [0, 1]^{\ell+1}$, the error

$$|B_k(f; x_1, \ldots, x_{\ell+1}) - f(x_1, \ldots, x_{\ell+1})|$$

can be bounded by

$$\leq |B_k(f; x_1, \ldots, x_{\ell+1}) - B_k(f; x_1, \ldots, x_\ell)| + |B_k(f; x_1, \ldots, x_\ell) - f(x_1, \ldots, x_{\ell+1})| \tag{9}$$

$$\leq L \left( \frac{1}{4k} \right)^{\gamma/2} + \ell L \left( \frac{1}{4k} \right)^{\gamma/2}$$

$$= (\ell + 1)L \left( \frac{1}{4k} \right)^{\gamma/2}. $$

In fact, the second term of Equation (9) is the error of the Bernstein polynomial of $f$ seen as a function of $x_1, \ldots, x_\ell$ only. The corresponding bound then follows from the inductive step. On the other hand, the first summand corresponds to the approximation error of the (univariate) Bernstein polynomial of $B_k(f; x_1, \ldots, x_\ell)$, as a function of the remaining variable $x_{\ell+1}$. The proof for $(h, T)$-smooth functions is obtained by replacing $B_k$ with $B_k^{(h)}$ and using the bound of Theorem 6.