Characterizing the Sample Complexity of Private Learners[*]

Amos Beimel
Dept. of Computer Science
Ben-Gurion University
beimel@cs.bgu.ac.il

Kobbi Nissim
Dept. of Computer Science
Ben-Gurion University &
Harvard University
kobbi@cs.bgu.ac.il

Uri Stemmer
Dept. of Computer Science
Ben-Gurion University
stemmer@cs.bgu.ac.il

ABSTRACT

In 2008, Kasiviswanathan el al. defined *private learning* as a combination of PAC learning and differential privacy [16]. Informally, a private learner is applied to a collection of labeled individual information and outputs a hypothesis while preserving the privacy of each individual. Kasiviswanathan et al. gave a generic construction of private learners for (finite) concept classes, with sample complexity logarithmic in the size of the concept class. This sample complexity is higher than what is needed for non-private learners, hence leaving open the possibility that the sample complexity of private learning may be sometimes significantly higher than that of non-private learning.

We give a combinatorial characterization of the sample size sufficient and necessary to privately learn a class of concepts. This characterization is analogous to the well known characterization of the sample complexity of non-private learning in terms of the VC dimension of the concept class. We introduce the notion of *probabilistic representation* of a concept class, and our new complexity measure RepDim corresponds to the size of the smallest probabilistic representation of the concept class.

We show that any private learning algorithm for a concept class $\mathcal{C}$ with sample complexity m implies $\mathrm{RepDim}(\mathcal{C}) = O(m)$, and that there exists a private learning algorithm with sample complexity $m = O(\mathrm{RepDim}(\mathcal{C}))$. We further demonstrate that a similar characterization holds for the database size needed for privately computing a large class of optimization problems and also for the well studied problem of private data release.

Categories and Subject Descriptors

K.4.1 [**Computers and Society**]: Public Policy Issues—*Privacy*; F.2.m [**Analysis of Algorithms and Problem Complexity**]: Miscellaneous

General Terms

Theory

Keywords

Differential privacy, PAC learning, Sample complexity, Probabilistic Representation

1. INTRODUCTION

Motivated by the observation that learning generalizes many of the analyses applied to large collections of data, Kasiviswanathan el al. [16] defined in 2008 *private learning* as a combination of probably approximately correct (PAC) learning [19] and differential privacy [11]. A PAC learner is given a collection of labeled examples (sampled according to an unknown probability distribution and labeled according to an unknown concept) and generalizes the labeled examples into a hypothesis h that should predict with high accuracy the labeling of fresh examples taken from the same unknown distribution and labeled with the same unknown concept.

The privacy requirement is that the choice of h preserves differential privacy of sample points. Intuitively this means that this choice should not be significantly affected by any particular sample. Differential privacy is increasingly accepted as a standard for rigorous privacy and recent research has shown that differentially private variants exists to many analyses. We refer the reader to surveys of Dwork [9, 10].

The sample complexity required for learning a concept class $\mathcal{C}$ determines the amount of *labeled* data needed for learning a concept $c \in \mathcal{C}$. It is well known that the sample complexity of learning a concept class $\mathcal{C}$ (non-privately) is proportional to a complexity measure of the class $\mathcal{C}$ knowns as the VC-dimension [20, 6, 13]. Kasiviswanathan et al. [16] proved that a private learner exists for every *finite* concept class. The proof is via a generic construction that exhibits sample complexity logarithmic in the size of the concept class. The VC-dimension of a concept class is bounded by this quantity (and significantly lower for some interesting concept classes), and hence the results of [16] left open the possibility that the sample complexity of private learning may be significantly higher than that of non-private learning.

In analogy to the characterization of the sample complexity of (non-private) PAC learners via the VC-dimension, we give a combinatorial characterization of the sample size sufficient and necessary for private PAC learners. Towards obtaining this characterization, we introduce the notion of

probabilistic representation of a concept class. We note that our characterization, as the VC-dimension characterization, ignores the computation required by the learner. Some of our algorithms are, however, computationally efficient.

1.1   Related Work

We start with a short description of prior work on the sample complexity of private learning. To simplify the exposition, we ignore dependencies on the error, confidence and privacy parameters by considering them constants for this and the following section. The dependency on these parameters would be made explicit in the later sections of the paper.

Recall that the sample complexity of non-private learners for a class of functions $\mathcal{C}$ is proportional to the VC-dimension of the class [6, 13] – a combinatorial measure of the class that is equal to the size of the largest set of inputs that is shattered by the class. This characterization, as ours, ignores the computation required by the learner.

Kasiviswanathan et al. [16] showed, informally, that every finite concept class $\mathcal{C}$ can be learned privately (ignoring computational complexity). Their construction is based on the exponential mechanism of McSherry and Talwar [17], and the $O(\ln |\mathcal{C}|)$ bound on sample complexity results from the union bound argument used in the analysis of the exponential mechanism. Computationally efficient learners were shown to exist by Blum et al. [4] for all concept classes that can be efficiently learned in the *statistical queries* model. Kasiviswanathan et al. [16] showed an example of a concept class – the class of parity functions – that is not learnable in the statistical queries model but can be learned privately and efficiently. These positive results suggest that many "natural" computational learning tasks that are efficiently learned non-privately can be learned privately and efficiently.

Beimel et al. [3] studied the sample complexity of private learning. They examined the concept class of point functions $\texttt{POINT}_d$ where each concept evaluates to one on exactly one point of the domain and to zero otherwise. Note that the VC-dimension of $\texttt{POINT}_d$ is one. Beimel et al. proved lower bounds on the sample complexity of *properly* and privately learning the class $\texttt{POINT}_d$ (and related classes), implying that the VC dimension of a class does not characterize the sample complexity of private proper learning. On the other hand, they observed that the sample complexity can be improved for *improper* private learners whenever there exists a smaller hypothesis class $\mathcal{H}$ that represents $\mathcal{C}$ in the sense that for every concept $c \in \mathcal{C}$ and for every distribution on the examples, there is a hypothesis $h \in \mathcal{H}$ that is close to c. Using the exponential mechanism to choose among the hypotheses in $\mathcal{H}$ instead of $\mathcal{C}$, the sample complexity is reduced to $\ln |\mathcal{H}|$ (this is why the *size* of the representation $\mathcal{H}$ is defined to be $\ln |\mathcal{H}|$). For some classes this can dramatically improve the sample complexity, e.g., for the class $\texttt{POINT}_d$ (defined in Example 3.2), the sample complexity is improved from $O(\ln |\texttt{POINT}_d|) = O(d)$ to $O(\ln d)$. Using other techniques, Beimel et al. showed that the sample complexity of learning $\texttt{POINT}_d$ can be reduced even further to $O(1)$, hence showing the largest possible gap between proper and non proper private learning. Such a gap does not exists for non-private learning.

Chaudhuri and Hsu [7] studied the sample complexity needed for private learning infinite concept classes when the data is drawn from a continuous distribution. They showed that under these settings there exists a simple concept class for which any proper learner that uses a finite number of examples and guarantees differential privacy fails to satisfy accuracy guarantee for at least one data distribution. This implies that the results of Kasiviswanathan et al. [16] do not extend to infinite hypothesis classes. Interestingly, our results imply an improper private algorithm for an infinite extension of the class $\texttt{POINT}$ (that is, a class over the natural numbers of all boolean functions that return 1 on exactly one number).

Chaudhuri and Hsu [7] also study learning algorithms that are only required to protect the privacy of the labels (and do not necessarily protect the privacy of the examples themselves). They prove upper bounds and lower bounds on the sample complexity of such algorithms. In particular, they prove a lower bound on the sample complexity using the doubling dimension of the disagreement metric of the hypothesis class with respect to the unlabeled data distribution. This result does not imply our characterization as the privacy requirement in protecting the labels is much weaker than protecting the sample point and the label.

A line of research (started in [18]) that is very relevant to our paper is boosting learning algorithms, that is, taking learning algorithms that have a big classification error and producing a learning algorithm with small error. Dwork et al. [12] show how to privately boost accuracy, that is, given a *private* learning algorithms that have a big classification error, they produce a *private* learning algorithm with small error. In Lemma 3.14, we show how to boost the accuracy α for probabilistic representations. This gives an alternative private boosting, whose proof is simpler. However, as it uses the exponential mechanism, it is (generally) not computationally efficient.

1.2   Our Results

Beimel et al. [3] showed how to use a representation of a class to privately learn it. We make an additional step in improving the sample complexity by considering a *probabilistic* representation of a concept class $\mathcal{C}$. Instead of one collection $\mathcal{H}$ representing $\mathcal{C}$, we consider a list of collections $\mathcal{H}_1, \ldots, \mathcal{H}_r$ such that for every $c \in \mathcal{C}$ and every distribution on the examples, if we sample a collection $\mathcal{H}_i$ from the list, then with high probability there is a hypothesis $h \in \mathcal{H}_i$ that is close to c. To privately learn $\mathcal{C}$, the learning algorithm first samples $i \in \{1, \ldots, r\}$ and then uses the exponential mechanism to select a hypothesis from $\mathcal{H}_i$. This reduces the sample complexity to $O(\max_i \ln |\mathcal{H}_i|)$; the *size* of the probabilistic representation is hence defined to be $\max_i \ln |\mathcal{H}_i|$.

We show that for $\texttt{POINT}_d$ there exists a probabilistic representation of size $O(1)$. This results in a private learning algorithm with sample complexity $O(1)$, matching a different private algorithm for $\texttt{POINT}_d$ presented in [3]. Our new algorithm offers some improvement in the sample complexity compared to the algorithm of [3] when considering the learning and privacy parameters. Furthermore, our algorithm can be made computationally efficient without making any computational hardness assumptions, while the efficient version in [3] assumes the existence of one-way functions. Finally, it is conceptually simpler and in particular it avoids the sub-sampling technique used in [3].

One can ask if there are private learning algorithms with smaller sample complexity than the size of the smallest probabilistic representation. We show that the answer is no —

the size of the smallest probabilistic representation is a lower bound on the sample complexity. Thus, the size of the smallest probabilistic representation of a class $\mathcal{C}$, which we call the *representation dimension* and denote by $\mathrm{RepDim}(\mathcal{C})$, characterizes (up to constants) the sample size necessary and sufficient for privately learning the class $\mathcal{C}$. We also show that for concepts defined over a finite domain, the difference between the sizes of the best deterministic and probabilistic representation is bounded. Namely, that if $\mathcal{C}$ is a concept class over the domain $\{0,1\}^d$, then there exists a deterministic representation of $\mathcal{C}$ of size $O(\mathrm{RepDim}(\mathcal{C}) + \ln d)$. Thus, for classes whose smallest deterministic representation is of size $\omega(\ln d)$, the size of the smallest deterministic representation characterizes the sample complexity of private learning of the class.

The notion of probabilistic representation applies not only to private learning, but also to optimization problems. We consider a scenario where there is a domain X, a database S of m records, each taken from the domain X, a set of solutions $\mathcal{F}$, and a quality function $\mathsf{q} : X^* \times \mathcal{F} \to [0,1]$ that we wish to maximize. If the exponential mechanism is used for (approximately) solving the problem, then the size of the database should be $\Omega(\ln |\mathcal{F}|)$ in order to achieve a reasonable approximation. Using our notions of a representation of $\mathcal{F}$ and of a probabilistic representation of $\mathcal{F}$, one can reduce the size of the minimal database without paying too much in the quality of the solution. Interestingly, a similar notion to representation, called "solution list algorithms", was considered in [2] for constructing secure protocols for search problems while leaking only a few bits on the input. Curiously, their notion of leakage is very different from that of differential privacy.

We give two examples of such optimization problems. First, an example inspired by [2]: each record in the database is a clause with exactly 3 literals and we want to find an assignment satisfying at least 7/8 fraction of the clauses while protecting the privacy of the clauses. A construction of [2] yields a deterministic representation for this problem where the size of the database can be much smaller. Using a probabilistic representation, we can give a good assignment even for databases of constant size. This example is a simple instance of a scenario, where each individual has a preference on the solution and we want to choose a solution maximizing the number of individuals whose preference are met, while protecting the privacy of the preference. Another example of optimization is sanitization, where given a database we want to publish a synthetic database, which gives a similar utility as the original database while protecting the privacy of the individual records of the database. Using our techniques, we study the minimal database size for which sanitization gives reasonable performance with respect to a given family of queries.

Open Problem. We still do not know the relation between this dimension and the VC dimension. By Sauer's Lemma, if $\mathcal{C}$ is a concept class over $\{0,1\}^d$, then the number of functions in $\mathcal{C}$ is at most $\exp(d \cdot \mathrm{VC}(\mathcal{C}))$. By [16], there is a private learning algorithm for $\mathcal{C}$ whose sample size is $O(d \cdot \mathrm{VC}(\mathcal{C}))$, thus, the probabilistic representation dimension of $\mathcal{C}$ is $O(d \cdot \mathrm{VC}(\mathcal{C}))$. We do not know if there is a class $\mathcal{C}$ such that $\mathrm{RepDim}(\mathcal{C}) \gg \mathrm{VC}(\mathcal{C})$. A candidate for such separation appears in [1].

2. PRELIMINARIES

Notation. We use $O_\gamma(g(n))$ as a shorthand for $O(h(\gamma) \cdot g(n))$ for some non-negative function h. Given a set $\mathcal{B}$ of cardinality r, and a distribution $\mathcal{P}$ on $\{1, 2, \ldots, r\}$, we use the notation $b \in_\mathcal{P} \mathcal{B}$ to denote a random element of $\mathcal{B}$ chosen according to $\mathcal{P}$.

2.1 Preliminaries from Privacy

A database is a vector $S = (z_1, \ldots, z_m)$ over a domain X, where each entry $z_i \in S$ represents information contributed by one individual. Databases S_1 and S_2 are called *neighboring* if they differ in exactly one entry. An algorithm preserves differential privacy if neighboring databases induce nearby outcome distributions. Formally,

DEFINITION 2.1 (DIFFERENTIAL PRIVACY [11]). *A randomized algorithm A is ϵ-differentially private if for all neighboring databases S_1, S_2, and for all sets $\mathcal{F}$ of outputs,*

$$\Pr[A(S_1) \in \mathcal{F}] \leq \exp(\epsilon) \cdot \Pr[A(S_2) \in \mathcal{F}]. \qquad (1)$$

The probability is taken over the random coins of A.

An immediate consequence of the definition is that for *any* two databases $S_1, S_2 \in X^m$, and for all sets $\mathcal{F}$ of outputs,

$$\Pr[A(S_1) \in \mathcal{F}] \geq \exp(-\epsilon m) \cdot \Pr[A(S_2) \in \mathcal{F}].$$

2.2 Preliminaries from Learning Theory

Let $X_d = \{0,1\}^d$. A concept $c : X_d \to \{0,1\}$ is a function that labels *examples* taken from the domain X_d by either 0 or 1. A *concept class* $\mathcal{C}$ over X_d is a class of concepts mapping X_d to $\{0,1\}$.

PAC learning algorithms are given examples sampled according to an unknown probability distribution $\mathcal{D}$ over X_d, and labeled according to an unknown *target* concept $c \in \mathcal{C}$. The *generalization error* of a hypothesis $h : X_d \to \{0,1\}$ is defined as

$$\mathrm{error}_\mathcal{D}(c, h) = \Pr_{x \in_\mathcal{D} X_d}[h(x) \neq c(x)].$$

For a labeled sample $S = (x_i, y_i)_{i=1}^m$, the *empirical error* of h is

$$\mathrm{error}_S(h) = \frac{1}{m} |\{i : h(x_i) \neq y_i\}|.$$

DEFINITION 2.2. *An α-good hypothesis for c and $\mathcal{D}$ is a hypothesis h such that $\mathrm{error}_\mathcal{D}(c, h) \leq \alpha$.*

DEFINITION 2.3 (PAC LEARNING [19]). *Algorithm A is an (α, β)-PAC learner for a concept class $\mathcal{C}$ over X_d using hypothesis class $\mathcal{H}$ and sample size m if for all concepts $c \in \mathcal{C}$, all distributions $\mathcal{D}$ on X_d, given an input of m samples $S = (z_1, \ldots, z_m)$, where $z_i = (x_i, c(x_i))$ and x_i are drawn i.i.d. from $\mathcal{D}$, algorithm A outputs a hypothesis $h \in \mathcal{H}$ satisfying*

$$\Pr[\mathrm{error}_\mathcal{D}(c, h) \leq \alpha] \geq 1 - \beta.$$

The probability is taken over the random choice of the examples in S according to $\mathcal{D}$ and the coin tosses of the learner A.

DEFINITION 2.4. *An algorithm satisfying Definition 2.3 with $\mathcal{H} \subseteq \mathcal{C}$ is called a proper PAC learner; otherwise it is called an improper PAC learner.*

2.3 Private Learning

As a private learner is a PAC learner, its outcome hypothesis should also be a good predictor of labels. Hence, the privacy requirement from a private learner is not that an application of the hypothesis h on a new sample (pertaining to an individual) should leak no information about the sample.

DEFINITION 2.5 (PRIVATE PAC LEARNING [16]). *Let A be an algorithm that gets an input $S = (z_1, \ldots, z_m)$. Algorithm A is an $(\alpha, \beta, \epsilon)$-PPAC learner for a concept class $\mathcal{C}$ over X_d using hypothesis class $\mathcal{H}$ and sample size m if*

PRIVACY. *Algorithm A is ϵ-differentially private (as formulated in Definition 2.1);*

UTILITY. *Algorithm A is an (α, β)-PAC learner for $\mathcal{C}$ using $\mathcal{H}$ and sample size m (as formulated in Definition 2.3).*

2.4 The Exponential Mechanism

We next describe the exponential mechanism of McSherry and Talwar [17]. We present its private learning variant; however, it can be used in more general scenarios. The goal here is to chooses a hypothesis $h \in \mathcal{H}$ approximately minimizing the empirical error. The choice is probabilistic, where the probability mass that is assigned to each hypothesis decreases exponentially with its empirical error.

Inputs: a privacy parameter ϵ, a hypothesis class $\mathcal{H}$, and m labeled samples $S = (x_i, y_i)_{i=1}^m$.

1. $\forall h \in \mathcal{H}$ define $q(S, h) = |\{i : h(x_i) = y_i\}|$.

2. Randomly choose $h \in \mathcal{H}$ with probability

$$\frac{\exp\left(\epsilon \cdot q(S, h)/2\right)}{\sum_{f \in \mathcal{H}} \exp\left(\epsilon \cdot q(S, f)/2\right)}.$$

PROPOSITION 2.6. *Denote $\hat{e} \triangleq \min_{f \in \mathcal{H}}\{\mathrm{error}_S(f)\}$. The probability that the exponential mechanism outputs a hypothesis h such that $\mathrm{error}_S(h) > \hat{e} + \Delta$ is at most $|\mathcal{H}| \cdot \exp(-\epsilon\Delta m/2)$. Moreover, The exponential mechanism is ϵ differentially private.*

2.5 Concentration Bounds

Let $X_1, \ldots, X_n$ be independent random variables where $\Pr[X_i = 1] = p$ and $\Pr[X_i = 0] = 1 - p$ for some $0 < p < 1$. Clearly, $\mathbb{E}[\sum_i X_i] = pn$. Chernoff and Hoeffding bounds show that the sum is concentrated around this expected value:

$$\Pr\left[\sum_i X_i > (1 + \delta)pn\right] \le \exp\left(-pn\delta^2/3\right) \quad \text{for } \delta > 0,$$

$$\Pr\left[\sum_i X_i < (1 - \delta)pn\right] \le \exp\left(-pn\delta^2/2\right) \quad \text{for } 0 < \delta < 1,$$

$$\Pr\left[\left|\sum_i X_i - pn\right| > \delta\right] \le 2\exp\left(-2\delta^2/n\right) \quad \text{for } \delta \ge 0.$$

The first two inequalities are known as the multiplicative Chernoff bounds [8], and the last inequality is known as the Hoeffding bound [15].

3. THE SAMPLE COMPLEXITY OF PRIVATE LEARNERS

In this section we present a combinatorial measure of a concept class $\mathcal{C}$ that characterizes the sample complexity

---

necessary and sufficient for privately learning $\mathcal{C}$. The measure is a *probabilistic representation* of the class $\mathcal{C}$. We start with the notation of deterministic representation from [3].

DEFINITION 3.1 ([3]). *A hypothesis class $\mathcal{H}$ is an α-representation for a class $\mathcal{C}$ if for every $c \in \mathcal{C}$ and every distribution $\mathcal{D}$ on X_d there exists a hypothesis $h \in \mathcal{H}$ such that $\mathrm{error}_\mathcal{D}(c, h) \le \alpha$.*

EXAMPLE 3.2 ($\texttt{POINT}_d$). *For $j \in X_d$, define $c_j : X_d \to \{0, 1\}$ as $c_j(x) = 1$ if $x = j$, and $c_j(x) = 0$ otherwise. Define $\texttt{POINT}_d = \{c_j\}_{j \in X_d}$. In [3] it was shown that for $\alpha < 1/2$, every α-representation for $\texttt{POINT}_d$ must be of cardinality at least d, and that an α-representation $\mathcal{H}_d$ for $\texttt{POINT}_d$ exists where $|\mathcal{H}_d| = O(d/\alpha^2)$.*

The above representation can be used for non-private learning, by taking a big enough sample and finding a hypothesis $h \in \mathcal{H}_d$ minimizing the empirical error. For *private* learning it was shown in [3] that a sample of size $O_{\alpha, \beta, \epsilon}(\log |\mathcal{H}_d|)$ suffices, with a learner that employs the exponential mechanism to choose a hypothesis from $\mathcal{H}_d$.

DEFINITION 3.3. *For a hypothesis class $\mathcal{H}$ we denote $\mathrm{size}(\mathcal{H}) = \ln |\mathcal{H}|$. We define the Deterministic Representation Dimension of a concept class $\mathcal{C}$ as*

$$\mathrm{DRepDim}(\mathcal{C}) = \min\left\{\mathrm{size}(\mathcal{H}) : \mathcal{H} \;\frac{1}{4}\text{-represents } \mathcal{C}\right\}.$$

EXAMPLE 3.4. *By the results of [3], stated in the previous example, $\mathrm{DRepDim}(\texttt{POINT}_d) = \theta(\ln(d))$.*

We are now ready to present the notion of a probabilistic representation. The idea behind this notion is that we have a list of hypothesis classes, such that for every concept c and distribution $\mathcal{D}$, if we sample a hypothesis class from the list, then with high probability it contains a hypothesis that is close to c.

DEFINITION 3.5. *Let $\mathcal{P}$ be a distribution over $\{1, 2, \ldots, r\}$, and let $\mathscr{H} = \{\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_r\}$ be a family of hypothesis classes (every $\mathcal{H}_i \in \mathscr{H}$ is a set of boolean functions). We say that $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for a class $\mathcal{C}$ if for every $c \in \mathcal{C}$ and every distribution $\mathcal{D}$ on X_d:*

$$\Pr_\mathcal{P}\left[\exists h \in \mathcal{H}_i \;\; s.t. \;\; \mathrm{error}_\mathcal{D}(c, h) \le \alpha\right] \ge 1 - \beta.$$

The probability is over randomly choosing a set $\mathcal{H}_i \in_\mathcal{P} \mathscr{H}$.

EXAMPLE 3.6 ($\texttt{POINT}_d$). *In Section 7 we construct for every α and every β a pair $(\mathscr{H}, \mathcal{P})$ that (α, β)- probabilistically represents the class $\texttt{POINT}_d$, where $\mathscr{H}$ contains all the sets of at most $\frac{4}{\alpha}\ln(1/\beta)$ boolean functions.*

DEFINITION 3.7. *Let $\mathscr{H} = \{\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_r\}$ be a family of hypothesis classes. We denote $|\mathscr{H}| = r$, and $\mathrm{size}(\mathscr{H}) = \max\{\ln |\mathcal{H}_i| : \mathcal{H}_i \in \mathscr{H}\}$. We define the Representation Dimension of a concept class $\mathcal{C}$ as*

$$\mathrm{RepDim}(\mathcal{C}) = \min\left\{\mathrm{size}(\mathscr{H}) \;:\; \begin{array}{c} \exists \mathcal{P} \; s.t. \; (\mathscr{H}, \mathcal{P}) \; is \; a \\ (\frac{1}{4}, \frac{1}{4})\text{-probabilistic} \\ representation \; for \; \mathcal{C} \end{array}\right\}.$$

EXAMPLE 3.8 ($\texttt{POINT}_d$). *The size of the probabilistic representation mentioned in Example 3.6 is $\ln(\frac{4}{\alpha}\ln(1/\beta))$. Placing $\alpha = \beta = \frac{1}{4}$, we see that the Representation Dimension of $\texttt{POINT}_d$ is constant.*

3.1 Equivalence of (α, β)-Probabilistic Representation and Private Learning

We now show that $\text{RepDim}(\mathcal{C})$ characterizes the sample complexity of private learners. We start by showing in Lemma 3.9 that an (α, β)-probabilistic representation of $\mathcal{C}$ implies a private learning algorithm whose sample complexity is the size of the representation. We then show in Lemma 3.12 that if there is a private learning algorithm with sample complexity m, then there is probabilistic representation of $\mathcal{C}$ of size $O(m)$; this lemma implies that $\text{RepDim}(\mathcal{C})$ is a lower bound on the sample complexity. Recall that $\text{RepDim}(\mathcal{C})$ is the size of the smallest probabilistic representation for $\alpha = \beta = 1/4$. Thus, to complete the proof we show in Lemma 3.14 that a probabilistic representation with $\alpha = \beta = 1/4$ implies a probabilistic representation for arbitrary α and β.

LEMMA 3.9. *If there a exists pair $(\mathcal{H}, \mathcal{P})$ that (α, β)-probabilistically represents a class $\mathcal{C}$, then for every ϵ there exists an algorithm A that $(6\alpha, 4\beta, \epsilon)$-PPAC learns $\mathcal{C}$ with a sample size $m = O\left(\frac{1}{\alpha\epsilon}(\text{size}(\mathcal{H}) + \ln(\frac{1}{\beta}))\right)$.*

PROOF. Let $(\mathcal{H}, \mathcal{P})$ be an (α, β)-probabilistic representation for the class $\mathcal{C}$, and consider the following algorithm A:

Inputs: $S = (x_i, y_i)_{i=1}^{m}$, and a privacy parameter ϵ.
1. Randomly choose $\mathcal{H}_i \in_{\mathcal{P}} \mathcal{H}$.
2. Choose $h \in \mathcal{H}_i$ using the exp. mechanism with ϵ.

By the properties of the exponential mechanism, A is ϵ-differentially private. We will show that with sample size $m = O\left(\frac{1}{\alpha\epsilon}(\text{size}(\mathcal{H}) + \ln(\frac{1}{\beta}))\right)$, algorithm A is a $(6\alpha, 4\beta)$-PAC learner for $\mathcal{C}$. Fix some $c \in \mathcal{C}$ and $\mathcal{D}$, and define the following 3 good events:

E_1 $\mathcal{H}_i$ chosen in step 1 contains at least one hypothesis h s.t. $\text{error}_S(h) \leq 2\alpha$.

E_2 For every $h \in \mathcal{H}_i$ s.t. $\text{error}_S(h) \leq 3\alpha$, it holds that $\text{error}_{\mathcal{D}}(c, h) \leq 6\alpha$

E_3 The exponential mechanism chooses an h such that $\text{error}_S(h) \leq \alpha + \min_{f \in \mathcal{H}_i} \{\text{error}_S(f)\}$.

We first show that if those 3 good events happen, algorithm A returns a 6α-good hypothesis. Event E_1 ensures the existence of a hypothesis $f \in \mathcal{H}_i$ s.t. $\text{error}_S(f) \leq 2\alpha$. Thus, event $E_1 \cap E_3$ ensures algorithm A chooses (using the exponential mechanism) a hypothesis $h \in \mathcal{H}_i$ s.t. $\text{error}_S(h) \leq 3\alpha$. Event E_2 ensures therefore that this h obeys $\text{error}_{\mathcal{D}}(c, h) \leq 6\alpha$.

We will now show that those 3 events happen with high probability. As $(\mathcal{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for the class $\mathcal{C}$, the chosen $\mathcal{H}_i$ contains a hypothesis h s.t. $\text{error}_{\mathcal{D}}(c, h) \leq \alpha$ with probability at least $1 - \beta$; by the Chernoff bound with probability at least $1 - \exp(-m\alpha/3)$ this hypothesis has empirical error at most 2α. Event E_1 happens with probability at least $(1 - \beta)(1 - \exp(-m\alpha/3)) > 1 - (\beta + \exp(-m\alpha/3))$, which is at least $(1 - 2\beta)$ for $m \geq \frac{3}{\alpha} \ln(1/\beta)$.

Using the Chernoff bound, the probability that a hypothesis h s.t. $\text{error}_{\mathcal{D}}(c, h) > 6\alpha$ has empirical error $\leq 3\alpha$ is less than $\exp(-m\alpha 3/4)$. Using the union bound, the probability that there is such a hypothesis in $\mathcal{H}_i$ is at most $|\mathcal{H}_i| \cdot$

$\exp(-m\alpha 3/4)$. Therefore, $\Pr[E_2] \geq 1 - |\mathcal{H}_i| \cdot \exp(-m\alpha 3/4)$. For $m \geq \frac{4}{3\alpha}(\ln(\frac{|\mathcal{H}_i|}{\beta}))$, this probability is at least $(1 - \beta)$.

The exponential mechanism ensures that the probability of event E_3 is at least $1 - |\mathcal{H}_i| \cdot \exp(-\epsilon\alpha m/2)$ (see Section 2.4), which is at least $(1 - \beta)$ for $m \geq \frac{2}{\alpha\epsilon} \ln(\frac{|\mathcal{H}_i|}{\beta})$.

All in all, by setting $m = \frac{3}{\alpha\epsilon}(\text{size}(\mathcal{H}) + \ln(\frac{1}{\beta}))$ we ensure that the probability of A failing to output a 6α-good hypothesis is at most 4β. $\square$

We will demonstrate the above lemma with two examples:

EXAMPLE 3.10 (EFFICIENT LEARNER FOR POINT$_d$). *As described in Example 3.6, there exists an $(\mathcal{H}, \mathcal{P})$ that $(\alpha/6, \beta/4)$-probabilistically represents the class POINT$_d$, where $\text{size}(\mathcal{H}) = O_{\alpha,\beta,\epsilon}(1)$. By Lemma 3.9, there exists an algorithm that $(\alpha, \beta, \epsilon)$-PPAC learns $\mathcal{C}$ with sample size $m = O_{\alpha,\beta,\epsilon}(1)$.*

The existence of an algorithm with sample complexity $O(1)$ was already proven in [3]. Moreover, assuming the existence of oneway functions, their learner is efficient. Our constructions yields an efficient learner, without assumptions. To see this, consider again algorithm A presented in the above proof, and note that as $\text{size}(\mathcal{H})$ is constant, step 2 could be done in constant time. Step 1 can be done efficiently as we can efficiently sample a set $\mathcal{H}_i \in_{\mathcal{P}} \mathcal{H}$. In Claim 7.1 we initially construct a probabilistic representation in which the description of every hypothesis is exponential in d. The representation is than revised using pairwise independence to yield a representation in which every hypothesis h has a short description, and given x the value $h(x)$ can be computed efficiently.

EXAMPLE 3.11 (POINT$_{\mathbb{N}}$). *Consider the class POINT$_{\mathbb{N}}$, which is exactly like POINT$_d$, only over the natural numbers. By results of [7, 3], it is impossible to properly PPAC learn the class POINT$_{\mathbb{N}}$. Our construction can yield an (inefficient) improper private learner for POINT$_{\mathbb{N}}$ with $O_{\alpha,\beta,\epsilon}(1)$ samples. The details are deferred to Section 7.*

The next lemma shows that a private learning algorithm implies a probabilistic representation. This lemma can be used to lower bound the sample complexity of private learners.

LEMMA 3.12. *Let $\alpha \leq 1/4$. If there exists an algorithm A that $(\alpha, \frac{1}{2}, \epsilon)$-PPAC learns a concept class $\mathcal{C}$ with a sample size m, then there exists a pair $(\mathcal{H}, \mathcal{P})$ that $(1/4, 1/4)$-probabilistically represents the class $\mathcal{C}$ such that $\text{size}(\mathcal{H}) = O(m\epsilon\alpha)$.*

PROOF. Let A be an $(\alpha, \frac{1}{2}, \epsilon)$-PPAC learner for the class $\mathcal{C}$ using hypothesis class $\mathcal{F}$ whose sample size is m. Without loss of generality, we can assume that $m \geq \frac{3 \ln(4)}{4\alpha}$ (since A can ignore part of the sample). For a target concept $c \in \mathcal{C}$ and a distribution $\mathcal{D}$ on X_d, we define

$$G_{\mathcal{D}}^{\alpha} = \{h \in \mathcal{F} : \text{error}_{\mathcal{D}}(c, h) \leq \alpha\}.$$

Fix some $c \in \mathcal{C}$ and a distribution $\mathcal{D}$ on X_d, and define the following distribution $\widetilde{\mathcal{D}}$ on X_d:

$$\Pr_{\widetilde{\mathcal{D}}}[x] = \begin{cases} 1 - 4\alpha + 4\alpha \cdot \Pr_{\mathcal{D}}[x], & x = 0^d. \\ 4\alpha \cdot \Pr_{\mathcal{D}}[x], & x \neq 0^d. \end{cases}$$

Note that for every $x \in X_d$,

$$\Pr_{\widetilde{\mathcal{D}}}[x] \geq 4\alpha \cdot \Pr_{\mathcal{D}}[x]. \tag{2}$$

As A is an $(\alpha, \frac{1}{2})$-PAC learner, it holds that

$$\Pr_{\widetilde{\mathcal{D}}, A} \left[A(S) \in G_{\widetilde{\mathcal{D}}}^{\alpha} \right] \geq \frac{1}{2},$$

where the probability is over A's randomness and over sampling the examples in S (according to $\widetilde{\mathcal{D}}$). In addition, by inequality (2), every hypothesis h with $\text{error}_{\mathcal{D}}(c, h) > 1/4$ has error strictly greater than α under $\widetilde{\mathcal{D}}$:

$$\text{error}_{\widetilde{\mathcal{D}}}(c, h) \geq 4\alpha \cdot \text{error}_{\mathcal{D}}(c, h) > \alpha.$$

So, every α-good hypothesis for c and $\widetilde{\mathcal{D}}$ is a $\frac{1}{4}$-good hypothesis for c and $\mathcal{D}$. That is, $G_{\widetilde{\mathcal{D}}}^{\alpha} \subseteq G_{\mathcal{D}}^{1/4}$. Therefore, $\Pr_{\widetilde{\mathcal{D}}, A} \left[A(S) \in G_{\mathcal{D}}^{1/4} \right] \geq \frac{1}{2}$.

We say that a database S of m labeled examples is *good* if the unlabeled example 0^d appears in S at least $(1 - 8\alpha)m$ times. Let S be a database constructed by taking m i.i.d. samples from $\widetilde{\mathcal{D}}$, labeled by c. By the Chernoff bound, S is good with probability at least $1 - \exp(-4\alpha m/3)$. Hence,

$$\Pr_{\widetilde{\mathcal{D}}, A} \left[(A(S) \in G_{\mathcal{D}}^{1/4}) \wedge (S \text{ is good}) \right] \geq \frac{1}{2} - \exp(-4\alpha m/3) \geq \frac{1}{4}.$$

Therefore, there exists a database S_{good} of m samples that contains the unlabeled sample 0^d at least $(1 - 8\alpha)m$ times, and $\Pr_A \left[A(S_{\text{good}}) \in G_{\mathcal{D}}^{1/4} \right] \geq \frac{1}{4}$, where the probability is only over the randomness of A. All of the examples in S_{good} (including the example 0^d) are labeled by c.

For $\sigma \in \{0, 1\}$, denote by $\vec{0}_\sigma$ a database containing m copies of the example 0^d labeled as σ. As A is ϵ-differentially private, and as the target concept c labels the example 0^d by either 0 or 1, for at least one $\sigma \in \{0, 1\}$ it holds that

$$\Pr_A[A(\vec{0}_\sigma) \in G_{\mathcal{D}}^{1/4}] \geq \exp(-8\alpha\epsilon m) \cdot \Pr_A \left[A(S_{\text{good}}) \in G_{\mathcal{D}}^{1/4} \right]$$
$$\geq \exp(-8\alpha\epsilon m) \cdot 1/4. \qquad (3)$$

That is, $\Pr_A[A(\vec{0}_\sigma) \notin G_{\mathcal{D}}^{1/4}] \leq 1 - \frac{1}{4}e^{-8\alpha\epsilon m}$. Now, consider a set $\mathcal{H}$ containing the outcomes of $4\ln(4)e^{8\alpha\epsilon m}$ executions of $A(\vec{0}_0)$, and the outcomes of $4\ln(4)e^{8\alpha\epsilon m}$ executions of $A(\vec{0}_1)$. The probability that $\mathcal{H}$ does not contain a $\frac{1}{4}$-good hypothesis for c and $\mathcal{D}$ is at most $(1 - \frac{1}{4}e^{-8\alpha\epsilon m})^{4\ln(4)e^{8\alpha\epsilon m}} \leq \frac{1}{4}$. Thus, $\mathcal{H} = \left\{ \mathcal{H} \subseteq \mathcal{F} : |\mathcal{H}| \leq 2 \cdot 4\ln(4)e^{8\alpha\epsilon m} \right\}$, and $\mathcal{P}$, the distribution induced by $A(\vec{0}_0)$ and $A(\vec{0}_1)$, are a $(1/4, 1/4)$-probabilistic representation for the class $\mathcal{C}$. Note that the value $c(0^d)$ is unknown, and can be either 0 or 1. Therefore the construction uses the two possible values (one of them correct).

It holds that $\text{size}(\mathcal{H}) = \max\{ \ln|\mathcal{H}| : \mathcal{H} \in \mathcal{H} \} = \ln(8\ln(4)) + 8\alpha\epsilon m = O(m\epsilon\alpha)$. $\square$

Lemma 3.14 shows how to construct a probabilistic representation for an arbitrary α and β from a probabilistic representation with $\alpha = \beta = 1/4$; in other words we boost α and β. The proof of this lemma is combinatorial. It allows us to start with a private learning algorithm with constant α and β, move to a representation, use the combinatorial boosting, and move back to a private algorithm with small α and β. This should be contrasted with the private boosting of [12] which is algorithmic and more complicated (however, the algorithm of Dwork et al. [12] is computationally efficient).

We first show how to construct a probabilistic representation for arbitrary β from a probabilistic representation with $\beta = 1/4$.

CLAIM 3.13. *For every concept class $\mathcal{C}$ and for every β, there exists a pair $(\mathcal{H}, \mathcal{P})$ that $(1/4, \beta)$-probabilistically represents $\mathcal{C}$ where $\text{size}(\mathcal{H}) \leq \text{RepDim}(\mathcal{C}) + \ln\ln(1/\beta)$.*

PROOF. Let $\beta < 1/4$, and let $(\mathcal{H}^0, \mathcal{P}^0)$ be a $(\frac{1}{4}, \frac{1}{4})$-probabilistic representation for $\mathcal{C}$ with $\text{size}(\mathcal{H}^0) = \text{RepDim}(\mathcal{C}) \triangleq k_0$ (that is, for every $\mathcal{H}_i^0 \in \mathcal{H}^0$ it holds that $|\mathcal{H}_i^0| \leq e^{k_0}$). Denote $\mathcal{H}^0 = \{\mathcal{H}_1^0, \mathcal{H}_2^0, \ldots, \mathcal{H}_r^0\}$, and consider the following family of hypothesis classes:

$$\mathcal{H}^1 = \left\{ \mathcal{H}_{i_1}^0 \cup \cdots \cup \mathcal{H}_{i_{\ln(1/\beta)}}^0 : 1 \leq i_1 \leq \cdots \leq i_{\ln(1/\beta)} \leq r \right\}.$$

Note that for every $\mathcal{H}_i^1 \in \mathcal{H}^1$ it holds that $|\mathcal{H}_i^1| \leq \ln(1/\beta)e^{k_0}$ and so $\text{size}(\mathcal{H}^1) \triangleq k_1 \leq k_0 + \ln\ln(1/\beta)$. We will now show an appropriate distribution $\mathcal{P}^1$ on $\mathcal{H}^1$ s.t. $(\mathcal{H}^1, \mathcal{P}^1)$ is a $(\frac{1}{4}, \beta)$-probabilistic representation for $\mathcal{C}$. To this end, consider the following process for randomly choosing an $\mathcal{H}^1 \in \mathcal{H}^1$:

---

1. Denote $M = \ln(1/\beta)$
2. For $i = 1, \ldots, M$:
   Randomly choose $\mathcal{H}_i^0 \in_{\mathcal{P}_0} \mathcal{H}^0$.
3. Return $\mathcal{H}^1 = \bigcup_{i=1}^{M} \mathcal{H}_i^0$.

The above process induces a distribution on $\mathcal{H}^1$, denoted as $\mathcal{P}^1$. As $\mathcal{H}^0$ is a $(\frac{1}{4}, \frac{1}{4})$-probabilistic representation for $\mathcal{C}$, we have that

$$\Pr_{\mathcal{P}^1} \left[\nexists h \in \mathcal{H}^1 \ s.t. \ \text{error}_{\mathcal{D}}(c, h) \leq 1/4 \right] =$$
$$= \prod_{i=1}^{M} \Pr_{\mathcal{P}^0} \left[\nexists h \in \mathcal{H}_i^0 \ s.t. \ \text{error}_{\mathcal{D}}(c, h) \leq 1/4 \right] \leq$$
$$\leq \left(\frac{1}{4} \right)^M \leq \beta.$$

$\square$

LEMMA 3.14. *For every concept class $\mathcal{C}$, every α, and every β, there exists $(\mathcal{H}, \mathcal{P})$ that (α, β)-probabilistically represents $\mathcal{C}$ where*

$$\text{size}(\mathcal{H}) = O\left(\ln(\frac{1}{\alpha}) \cdot \left(\text{RepDim}(\mathcal{C}) + \ln\ln\ln(\frac{1}{\alpha}) + \ln\ln(\frac{1}{\beta}) \right) \right).$$

PROOF. Let $\mathcal{C}$ be a concept class, and let $(\mathcal{H}^1, \mathcal{P}^1)$ be a $(\frac{1}{4}, \beta/T)$-probabilistic representation for $\mathcal{C}$ (where T will be set later). By Claim 3.13, such a representation exists with $\text{size}(\mathcal{H}^1) \triangleq k_1 \leq \text{RepDim}(\mathcal{C}) + \ln\ln(T/\beta)$. We use $\mathcal{H}^1$ and $\mathcal{P}^1$ to create an (α, β)-probabilistic representation for $\mathcal{C}$. We begin with two notations:

1. For T hypotheses $h_1, \ldots, h_T$ we denote by $\text{maj}_{h_1, \ldots, h_T}$ the majority hypothesis. That is, $\text{maj}_{h_1, \ldots, h_T}(x) = 1$ if and only if $|\{h_i : h_i(x) = 1\}| \geq T/2$.

2. For T hypothesis classes $\mathcal{H}_1, \ldots, \mathcal{H}_T$ we denote
   $\text{MAJ}(\mathcal{H}_1, \ldots, \mathcal{H}_T) = \left\{ \text{maj}_{h_1, \ldots, h_T} : \forall_{1 \leq i \leq T} \ h_i \in \mathcal{H}_i \right\}$.

Consider the following family of hypothesis classes:

$$\mathcal{H} = \left\{ \text{MAJ}(\mathcal{H}_{i_1}, \ldots, \mathcal{H}_{i_T}) : \mathcal{H}_{i_1}, \ldots, \mathcal{H}_{i_T} \in \mathcal{H}^1 \right\}.$$

Moreover, denote the distribution on $\mathscr{H}$ induced by the following random process as $\mathcal{P}$:

> For $j = 1, \ldots, T$:
>   Randomly choose $\mathcal{H}_{i_j} \in_{\mathcal{P}^1} \mathscr{H}^1$
> Return $\mathrm{MAJ}(\mathcal{H}_{i_1}, \ldots, \mathcal{H}_{i_T})$.

Next we show that $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for $\mathcal{C}$: For a fixed pair of a target concept c and a distribution $\mathcal{D}$, randomly choose $\mathcal{H}_{i_1}, \ldots, \mathcal{H}_{i_T} \in_{\mathcal{P}^1} \mathscr{H}^1$. We now show that with probability at least $(1 - \beta)$ the set $\mathrm{MAJ}(\mathcal{H}_{i_1}, \ldots, \mathcal{H}_{i_T})$ contains at least one α-good hypothesis for $c, \mathcal{D}$.

To this end, denote $\mathcal{D}_1 = \mathcal{D}$ and consider the following thought experiment, inspired by the Adaboost Algorithm of [14]:

> For $t = 1 \ldots, T$:
>
> 1. Fail if $\mathcal{H}_{i_t}$ does not contain a $\frac{1}{4}$-good hypothesis for $c, \mathcal{D}_t$.
>
> 2. Denote by $h_t \in \mathcal{H}_{i_t}$ a $\frac{1}{4}$-good hypothesis for $c, \mathcal{D}_t$.
>
> 3. $\mathcal{D}_{t+1}(x) = \begin{cases} 2\mathcal{D}_t(x), & \text{if } h_t(x) \neq c(x). \\ \left(1 - \frac{\mathrm{error}_{\mathcal{D}_t}(c, h_t)}{1 - \mathrm{error}_{\mathcal{D}_t}(c, h_t)}\right)\mathcal{D}_t(x), & \text{otherwise.} \end{cases}$

Note that as $\mathcal{D}_1$ is a probability distribution on X_d; the same is true for $\mathcal{D}_2, \mathcal{D}_3, \ldots, \mathcal{D}_T$. As $(\mathscr{H}^1, \mathcal{P}^1)$ is a $(\frac{1}{4}, \beta/T)$-probabilistic representation for $\mathcal{C}$, the failure probability of every iteration is at most β/T. Thus (using the union bound), with probability at least $(1 - \beta)$ the whole thought experiment will succeed, and in this case we show that the error of $h_{\mathrm{fin}} = \mathrm{maj}_{h_1, \ldots, h_T}$ is at most α.

Consider the set $R = \{x \; : \; h_{\mathrm{fin}}(x) \neq c(x)\} \subseteq X_d$. This is the set of points on which at least $T/2$ of $h_1, \ldots, h_T$ err. Next consider the partition of R to the following sets:

$$R_t = \left\{x \in R \; : \; \left(h_t(x) \neq c(x)\right) \wedge \left(\forall_{i>t} \, h_i(x) = c(x)\right)\right\}.$$

That is, R_t contains the points $x \in R$ on which h_t is last to err. Clearly $\mathcal{D}_t(R_t) \leq 1/4$, as R_t is a subset of the set of points on which h_t errs. Moreover,

$$
\begin{aligned}
\mathcal{D}_t(R_t) & \geq \mathcal{D}_1(R_t) \cdot 2^{T/2} \cdot \left(1 - \frac{\mathrm{error}_{\mathcal{D}_t}(c, h_t)}{1 - \mathrm{error}_{\mathcal{D}_t}(c, h_t)}\right)^{t - T/2} \\
& \geq \mathcal{D}_1(R_t) \cdot 2^{T/2} \cdot \left(1 - \frac{1/4}{1 - 1/4}\right)^{t - T/2} \\
& \geq \mathcal{D}_1(R_t) \cdot 2^{T/2} \cdot \left(1 - \frac{1/4}{1 - 1/4}\right)^{T/2} \\
& = \mathcal{D}(R_t) \cdot \left(\frac{4}{3}\right)^{T/2},
\end{aligned}
$$

so,

$$\mathcal{D}(R_t) \leq \mathcal{D}_t(R_t) \cdot \left(\frac{4}{3}\right)^{-T/2} \leq \frac{1}{4} \cdot \left(\frac{4}{3}\right)^{-T/2}.$$

Finally,

$$\mathrm{error}_{\mathcal{D}}(c, h_{\mathrm{fin}}) = \mathcal{D}(R) = \sum_{t=T/2}^{T} \mathcal{D}(R_t) \leq$$

$$\leq \frac{T}{2} \cdot \frac{1}{4} \cdot \left(\frac{4}{3}\right)^{-T/2} = \frac{T}{8} \cdot \left(\frac{4}{3}\right)^{-T/2}.$$

Choosing $T = 14 \ln(\frac{2}{\alpha})$, we get that $\mathrm{error}_{\mathcal{D}}(c, h_{\mathrm{fin}}) \leq \alpha$. Hence, $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for $\mathcal{C}$. Moreover, for every $\mathcal{H}_i \in \mathscr{H}$ we have that $|\mathcal{H}_i| \leq \left(e^{k_1}\right)^T$, and so

$$
\begin{aligned}
\mathrm{size}(\mathscr{H}) & \leq k_1 \cdot T \leq \left(\mathrm{RepDim}(\mathcal{C}) + \ln \ln(T/\beta)\right)T \\
& = O\Big(\ln(\frac{1}{\alpha}) \cdot \left(\mathrm{RepDim}(\mathcal{C}) + \ln \ln \ln(\frac{1}{\alpha}) + \ln \ln(\frac{1}{\beta})\right)\Big).
\end{aligned}
$$

$\square$

The next theorem states the main result of this section – RepDim characterizes the sample complexity of private learning.

THEOREM 3.15. *Let $\mathcal{C}$ be a concept class.* $\widetilde{\Theta}_\beta \left(\frac{\mathrm{RepDim}(\mathcal{C})}{\alpha \epsilon}\right)$ *samples are necessary and sufficient for the private learning of the class $\mathcal{C}$.*

PROOF. Fix some $\alpha \leq 1/4, \beta \leq 1/2$, and ϵ. By Lemma 3.14, there exists a pair $(\mathscr{H}, \mathcal{P})$ that $(\frac{\alpha}{6}, \frac{\beta}{4})$-represent class $\mathcal{C}$, where $\mathrm{size}(\mathscr{H}) = O\Big(\ln(1/\alpha) \cdot \left(\mathrm{RepDim}(\mathcal{C}) + \ln \ln \ln(1/\alpha) + \ln \ln(1/\beta)\right)\Big)$. Therefore, by Lemma 3.9, there exists an algorithm A that $(\alpha, \beta, \epsilon)$-PPAC learns the class $\mathcal{C}$ with a sample size

$$m = O_\beta \left(\frac{1}{\alpha \epsilon} \ln(\frac{1}{\alpha}) \cdot \left(\mathrm{RepDim}(\mathcal{C}) + \ln \ln \ln(\frac{1}{\alpha})\right)\right).$$

For the lower bound, let A be an $(\alpha, \beta, \epsilon)$-PPAC learner for the class $\mathcal{C}$ with a sample size m, where $\alpha \leq 1/4$ and $\beta \leq 1/2$. By Lemma 3.12, there exists an $(\mathscr{H}, \mathcal{P})$ that $(\frac{1}{4}, \frac{1}{4})$-probabilistically represents the class $\mathcal{C}$ and $\mathrm{size}(\mathscr{H}) = \ln(8) + \ln \ln(4) + 8\alpha \epsilon m$. Therefore, by definition, $\mathrm{RepDim}(\mathcal{C}) \leq \ln(8 \ln(4)) + 8\alpha \epsilon m$. Thus,

$$m \geq \frac{1}{8\alpha \epsilon} \cdot \left(\mathrm{RepDim}(\mathcal{C}) - \ln(8 \ln(4))\right) = \Omega \left(\frac{\mathrm{RepDim}(\mathcal{C})}{\alpha \epsilon}\right).$$

$\square$

4. FROM A PROBABILISTIC REPRESENTATION TO A DETERMINISTIC REPRESENTATION

In this section we will establish a connection between the (probabilistic) representation dimension of a class and its deterministic representation dimension.

OBSERVATION 4.1. *Let $(\mathscr{H}, \mathcal{P})$ be an (α, β)-probabilistic representation for a concept class $\mathcal{C}$. Then, $\mathcal{B} = \bigcup_{\mathcal{H}_i \in \mathscr{H}} \mathcal{H}_i$ is an α-representation of $\mathcal{C}$.*

PROOF. As $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for $\mathcal{C}$, for every c and every $\mathcal{D}$

$$\Pr_{\mathcal{P}}[\exists h \in \mathcal{H}_i \; s.t \; \mathrm{error}_{\mathcal{D}}(c, h) \leq \alpha] \geq 1 - \beta > 0.$$

The probability is over choosing a set $\mathcal{H}_i \in_{\mathcal{P}} \mathscr{H}$. In particular, for every c and every $\mathcal{D}$ there exists an $\mathcal{H}_i \in \mathscr{H}$ that contains an α-good hypothesis. $\square$

The simple construction in Observation 4.1 may result in a very large deterministic representation. For example, in Claim 7.1 we show an $(\mathscr{H}, \mathcal{P})$ that (α, β)- probabilistically represents the class $\texttt{POINT}_d$, where $\mathscr{H}$ contains all the sets

of at most $\frac{4}{\alpha}\ln(\frac{1}{\beta})$ boolean functions. While $\bigcup_{\mathcal{H}_i \in \mathscr{H}} \mathcal{H}_i = 2^{X_d}$ is indeed an α-representation for POINT$_d$, it is extremely over-sized.

We will show that it is not necessary to take the union of all the $\mathcal{H}_i$'s in $\mathscr{H}$ in order to get an α-representation for $\mathcal{C}$. As $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation, for every c and every $\mathcal{D}$, with probability at least $1 - \beta$ a randomly chosen $\mathcal{H}_i \in_{\mathcal{P}} \mathscr{H}$ contains an α-good hypothesis. The straight forward strategy here is to first boost β as in Claim 3.13, and then use the union bound over all possible $c \in \mathcal{C}$ and over all possible distributions $\mathcal{D}$ on X_d. Unfortunately, there are infinitely many such distributions, and the proof will be somewhat more complicated.

DEFINITION 4.2. *Let $\mathscr{H} = \{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_r\}$ be a family of hypothesis classes, and $\mathcal{P}$ be a distribution over $\{1, \dots, r\}$. We will denote the following non private algorithm as $Learner(\mathscr{H}, \mathcal{P}, m, \gamma)$:*

Input: a sample $S = (x_i, y_i)_{i=1}^m$.
1. *Randomly choose $\mathcal{H}_i \in_{\mathcal{P}} \mathscr{H}$.*
2. *If for every $h \in \mathcal{H}_i$ $\mathrm{error}_S(h) > \gamma$, then fail.*
3. *Return $h \in \mathcal{H}_i$ minimizing $\mathrm{error}_S(h)$.*

We will say that $Learner(\mathscr{H}, \mathcal{P}, m, \gamma)$ is β-successful for a class $\mathcal{C}$ over X_d, if for every $c \in \mathcal{C}$ and every distribution $\mathcal{D}$ on X_d, given an input sample drawn i.i.d. according to $\mathcal{D}$ and labeled by c, algorithm Learner fails with probability at most β.

CLAIM 4.3. *If $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for a class $\mathcal{C}$, then, for $m \geq \frac{3}{\alpha}\ln(1/\beta)$, algorithm $Learner(\mathscr{H}, \mathcal{P}, m, 2\alpha)$ is 2β-successful for $\mathcal{C}$.*

PROOF. We will show that with probability at least $1 - 2\beta$, the set $\mathcal{H}_i$ (chosen in Step 1) contains at least one hypothesis h s.t. $\mathrm{error}_S(h) \leq 2\alpha$. As $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for class $\mathcal{C}$, the chosen $\mathcal{H}_i$ will contain a hypothesis h s.t. $\mathrm{error}_{\mathcal{D}}(c, h) \leq \alpha$ with probability at least $1 - \beta$; by the Chernoff bound with probability at least $1 - \exp(-m\alpha/3)$ this hypothesis has empirical error at most 2α. The set $\mathcal{H}_i$ contains a hypothesis h s.t. $\mathrm{error}_S(h) \leq 2\alpha$ with probability at least $(1 - \beta)(1 - \exp(-m\alpha/3)) > 1 - (\beta + \exp(-m\alpha/3))$, which is at least $(1 - 2\beta)$ for $m \geq \frac{3}{\alpha}\ln(1/\beta)$. $\square$

CLAIM 4.4. *Let $\mathscr{H}$ be a family of hypothesis classes, and $\mathcal{P}$ a distribution on it. Let γ, β and m be such that $m \geq \frac{4}{\gamma}(\mathrm{size}(\mathscr{H}) + \ln(\frac{1}{\beta}))$. If $Learner(\mathscr{H}, \mathcal{P}, m, \gamma)$ is β-successful for a class $\mathcal{C}$ over X_d, then there exists $\widetilde{\mathscr{H}} \subseteq \mathscr{H}$ and a distribution $\widehat{\mathcal{P}}$ on it, s.t. $Learner(\widetilde{\mathscr{H}}, \widehat{\mathcal{P}}, m, \gamma)$ is a $(2\gamma, 3\beta)$-PAC learner for $\mathcal{C}$ and $\left|\widetilde{\mathscr{H}}\right| = \frac{d \cdot m}{\beta^2}$.*

PROOF. For every input $S = (x_i, y_i)_{i=1}^m$, denote by p_S the probability of $Learner(\mathscr{H}, \mathcal{P}, m, \gamma)$ failing on step 2 (the probability is only over the choice of $\mathcal{H}_i \in_{\mathcal{P}} \mathscr{H}$ in the first step). As $Learner(\mathscr{H}, \mathcal{P}, m, \gamma)$ is β-successful,

$$\Pr_{\mathcal{P}, \mathcal{D}}\left[Learner(\mathscr{H}, \mathcal{P}, m, \gamma) \text{ fails}\right] = \sum_S \Pr_{\mathcal{D}}[S] \cdot p_S \leq \beta.$$

Consider the following process, denoted by Proc, for ran-

domly choosing a multiset $\widetilde{\mathscr{H}}$ of size t (t will be set later):

For $i = 1, \dots, t$:
    Randomly choose $\mathcal{H}_i \in_{\mathcal{P}} \mathscr{H}$
Return $\widetilde{\mathscr{H}} = (\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_t)$.

Denote by $\mathcal{U}_t$ the uniform distribution on $\{1, 2, \dots, t\}$. As before, for every input $S = (x_i, y_i)_{i=1}^m$, denote by $\widetilde{p_S}$ the probability of $Learner(\widetilde{\mathscr{H}}, \mathcal{U}_t, m, \gamma)$ failing on its second step (again, the probability is only over the choice of $\mathcal{H}_i \in_{\mathcal{U}_t} \widetilde{\mathscr{H}}$ in the first step). Using those notations:

$$\Pr_{\mathcal{U}_t, \mathcal{D}}\left[Learner(\widetilde{\mathscr{H}}, \mathcal{U}_t, m, \gamma) \text{ fails}\right] = \sum_S \Pr_{\mathcal{D}}[S] \cdot \widetilde{p_S}.$$

Fix a sample S. As the choice of $\mathcal{H}_i \in_{\mathcal{U}_t} \widetilde{\mathscr{H}}$ is uniform,

$$\widetilde{p_S} = \frac{\left|\left\{\mathcal{H}_i \in \widetilde{\mathscr{H}} \ : \ \forall h \in \mathcal{H}_i \ \mathrm{error}_S(h) > \gamma\right\}\right|}{\left|\widetilde{\mathscr{H}}\right|}.$$

Using the Hoeffding bound,

$$\Pr_{Proc}\left[|\widetilde{p_S} - p_S| \geq \beta\right] \leq 2e^{-2t\beta^2}.$$

The probability is over choosing the multiset $\widetilde{\mathscr{H}}$. There are at most $2^{m(d+1)}$ samples of size m (as every entry in the sample is an element of X_d, concatenated with a label bit). Using the union bound over all possible samples S,

$$\Pr_{Proc}\left[\exists S \text{ s.t. } |\widetilde{p_S} - p_S| \geq \beta\right] \leq 2^{m(d+1)} \cdot 2 \cdot e^{-2t\beta^2}.$$

For $t \geq \frac{m \cdot d}{\beta^2}$ the above probability is strictly less than 1. This means that for $t = \frac{m \cdot d}{\beta^2}$ there exists a multiset $\widetilde{\mathscr{H}}$ such that $|\widetilde{p_S} - p_S| \leq \beta$ for every sample S. We will show that for this $\widetilde{\mathscr{H}}$, $Learner(\widetilde{\mathscr{H}}, \mathcal{U}_t, m, \gamma)$ is a $(2\gamma, 3\beta)$-PAC learner. Fix a target concept $c \in \mathcal{C}$ and a distribution $\mathcal{D}$ on X_d. Define the following two good events:

E_1 $Learner(\widehat{\mathscr{H}}, \mathcal{U}_t, m, \gamma)$ outputs a hypothesis h such that $\mathrm{error}_S(h) \leq \gamma$.

E_2 For every $h \in \mathcal{H}_i$ s.t. $\mathrm{error}_S(h) \leq \gamma$, it holds that $\mathrm{error}_{\mathcal{D}}(c, h) \leq 2\gamma$.

Note that if those two events happen, $Learner(\widehat{\mathscr{H}}, \mathcal{U}_t, m, \gamma)$ returns a 2γ-good hypothesis for c and $\mathcal{D}$. We will show that those two events happen with high probability. We start by bounding the failure probability of $Learner(\widehat{\mathscr{H}}, \mathcal{U}_t, m, \gamma)$.

$$\Pr_{\mathcal{U}_t, \mathcal{D}}\left[Learner(\widehat{\mathscr{H}}, \mathcal{U}_t, m, \gamma) \text{ fails}\right]$$
$$= \sum_S \Pr_{\mathcal{D}}[S] \cdot \widehat{p_S}$$
$$\leq \sum_S \Pr_{\mathcal{D}}[S] \cdot (p_S + \beta)$$
$$= \Pr_{\mathcal{P}, \mathcal{D}}\left[Learner(\mathscr{H}, \mathcal{P}, m, \gamma) \text{ fails}\right] + \beta \leq 2\beta.$$

When $Learner(\widehat{\mathscr{H}}, \mathcal{U}_t, m, \gamma)$ does not fail, it returns a hypothesis h with empirical error at most γ. Thus, $\Pr[E_1] \geq 1 - 2\beta$.

Using the Chernoff bound, the probability that a hypothesis h with $\mathrm{error}_{\mathcal{D}}(c, h) > 2\gamma$ has empirical error $\leq \gamma$ is less

than $\exp(-m\gamma/4)$. Using the union bound, the probability that there is such a hypothesis in $\mathcal{H}_i$ is at most $|\mathcal{H}_i| \cdot \exp(-m\gamma/4)$. Therefore, $\Pr[E_2] \geq 1 - |\mathcal{H}_i| \cdot \exp(-m\gamma/4)$. For $m \geq \frac{4}{\gamma}\ln(\frac{|\mathcal{H}_i|}{\beta})$, this probability is at least $(1 - \beta)$.

All in all, the probability of $Learner(\mathcal{H}, \mathcal{P}, m, \gamma)$ failing to output a 2γ-good hypothesis is at most 3β. $\square$

THEOREM 4.5. *If there exists a pair $(\mathcal{H}, \mathcal{P})$ that (α, β)-probabilistically represents a class $\mathcal{C}$ over X_d (where $|\mathcal{H}|$ might be very big), then there exists a pair $(\widehat{\mathcal{H}}, \widehat{\mathcal{P}})$ that $(4\alpha, 6\beta)$-probabilistically represents $\mathcal{C}$, where $\widehat{\mathcal{H}} \subseteq \mathcal{H}$, and*

$$\left|\widehat{\mathcal{H}}\right| = \frac{3d}{4\alpha\beta^2}\left(\text{size}(\mathcal{H}) + \ln(\frac{1}{\beta})\right).$$

PROOF. Let $(\mathcal{H}, \mathcal{P})$ be an (α, β)-probabilistic representation for a class $\mathcal{C}$. Set $m = \frac{3}{\alpha}(\text{size}(\mathcal{H}) + \ln(\frac{1}{\beta}))$. By Claim 4.3, $Learner(\mathcal{H}, \mathcal{P}, m, 2\alpha)$ is 2β-successful for class $\mathcal{C}$. By Claim 4.4, there exists an $\widehat{\mathcal{H}} \subseteq \mathcal{H}$ and a distribution $\widehat{\mathcal{P}}$ on it, such that $Learner(\widehat{\mathcal{H}}, \widehat{\mathcal{P}}, m, 2\alpha)$ is a $(4\alpha, 6\beta)$-PAC learner for $\mathcal{C}$ and $\left|\widehat{\mathcal{H}}\right| = \frac{d \cdot m}{4\beta^2} = \frac{3d}{4\alpha\beta^2}(\text{size}(\mathcal{H}) + \ln(\frac{1}{\beta}))$.

Assume towards contradiction that $(\widehat{\mathcal{H}}, \widehat{\mathcal{P}})$ does not $(4\alpha, 6\beta)$-represent $\mathcal{C}$. So, there exist a concept $c \in \mathcal{C}$ and a distribution $\mathcal{D}$ s.t., with probability strictly greater than 6β, a randomly chosen $\mathcal{H}_i \in_{\widehat{\mathcal{P}}} \widehat{\mathcal{H}}$ does not contain a 4α-good hypothesis for $c, \mathcal{D}$. Therefore, for those c and $\mathcal{D}$, $Learner(\widehat{\mathcal{H}}, \widehat{\mathcal{P}}, m, 2\alpha)$ will fail to return a 4α-good hypothesis with probability strictly greater than 6β. $\square$

THEOREM 4.6. *For every class $\mathcal{C}$ over X_d there exists a $\frac{1}{4}$-representation $\mathcal{B}$ such that $\text{size}(\mathcal{B}) = O(\ln(d) + \text{RepDim}(\mathcal{C}))$.*

PROOF. By Lemma 3.14, there exists a pair $(\mathcal{H}, \mathcal{P})$ that $(\frac{1}{16}, \frac{1}{12})$-probabilistically represents $\mathcal{C}$ such that $\text{size}(\mathcal{H}) = O(\text{RepDim}(\mathcal{C}))$. Using Theorem 4.5, there exists a pair $(\widehat{\mathcal{H}}, \widehat{\mathcal{P}})$ that $(\frac{1}{4}, \frac{1}{2})$-probabilistically represents $\mathcal{C}$, such that $\text{size}(\widehat{\mathcal{H}}) = \text{size}(\mathcal{H})$ and

$$\left|\widehat{\mathcal{H}}\right| = O\left(d \cdot \text{size}(\mathcal{H})\right).$$

We can now use Observation 4.1 and construct the set $\mathcal{B} = \bigcup_{\mathcal{H}_i \in \widehat{\mathcal{H}}} \mathcal{H}_i$ which is a $\frac{1}{4}$-representation for the class $\mathcal{C}$. In addition,

$$|\mathcal{B}| = O\left(\left|\widehat{\mathcal{H}}\right| \cdot e^{\text{size}(\mathcal{H})}\right) = O\left(d \cdot \text{size}(\mathcal{H}) \cdot e^{\text{size}(\mathcal{H})}\right).$$

Thus, $\text{size}(\mathcal{B}) = \ln|\mathcal{B}| = O\left(\ln(d) + \text{RepDim}(\mathcal{C})\right)$. $\square$

COROLLARY 4.7. *For every concept class $\mathcal{C}$ over X_d, $\text{DRepDim}(\mathcal{C}) = O(\ln(d) + \text{RepDim}(\mathcal{C}))$.*

COROLLARY 4.8. *There exists a constant N s.t. for every concept class C over X_d where $\text{DRepDim}(\mathcal{C}) \geq N\log(d)$, the sample complexity that is necessary and sufficient for privately learning $\mathcal{C}$ is $\Theta_{\alpha,\beta}(\text{DRepDim}(\mathcal{C}))$.*

5. PROBABILISTIC REPRESENTATION FOR PRIVATELY SOLVING OPTIMIZATION PROBLEMS

The notion of probabilistic representation applies not only to private learning, but also to a broader task of optimization problems. We consider the following scenario:

DEFINITION 5.1. *An* optimization problem OPT *over a universe X and a set of solutions $\mathcal{F}$ is defined by a quality function $q : X^* \times \mathcal{F} \to [0, 1]$. Given a database S, the task is to choose a solution $f \in \mathcal{F}$ such that $q(S, f)$ is maximized.*

Notation. We will refer to the optimization problem defined by a quality function q as OPT_q.

DEFINITION 5.2. *An α-good* solution for a database S is a solution s such that $q(S, s) \geq \max_{f \in \mathcal{F}}\{q(S, f)\} - \alpha$.

Given an optimization problem OPT_q, one can use the exponential mechanism to choose a solution $s \in \mathcal{F}$. In general, this method achieves a reasonable solution only for databases of size $\Omega(\log|\mathcal{F}|/\epsilon)$. To see this, consider a case where there exists a database S of m records such that exactly one solution $t \in \mathcal{F}$ has a quality of $q(S, t) = 1$, and every other $f \in \mathcal{F}$ has a quality of $q(S, f) = 1/2$. The probability of the exponential mechanism choosing t is:

$$\Pr[t \text{ is chosen}] = \frac{\exp(\epsilon m/2)}{(|\mathcal{F}| - 1) \cdot \exp(\epsilon m/4) + \exp(\epsilon m/2)}.$$

Unless

$$m \geq \frac{4}{\epsilon}\ln(|\mathcal{F}| - 1) = \Omega(\frac{1}{\epsilon}\ln|\mathcal{F}|), \quad (4)$$

the above probability is strictly less than $1/2$. Using our notations of probabilistic representation, it might be possible to reduce the necessary database size.

Consider using the exponential mechanism for choosing a solution s, not out of $\mathcal{F}$, but rather from a smaller set of solutions $\mathcal{B}$. Roughly speaking, the factor of $\ln|\mathcal{F}|$ in requirement (4) will now be replaced with $\ln|\mathcal{B}|$, which corresponds to size of the representation. Therefore, the database size m should be at least $\ln|\mathcal{B}|/\epsilon$. So m needs to be bigger than the size of the representation by at least a factor of $1/\epsilon$.

In the following analysis we will denote this required gap, i.e., $m/\ln|\mathcal{B}|$, as Δ. We will see that the existence of a private approximation algorithm implies a probabilistic representation with $1 < \Delta \approx \frac{1}{\epsilon}$, and that a probabilistic representation with $\Delta > 1$ implies a private approximation algorithm. Bigger Δ corresponds to better privacy; however, it might be harder to achieve.

DEFINITION 5.3. *Let OPT_q be an optimization problem over a universe X and a set of solutions $\mathcal{F}$. Let $\mathcal{B}$ be a set of solutions, and denote $\text{size}(\mathcal{B}) = \ln|\mathcal{B}|$. We say that $\mathcal{B}$ is an α-deterministic representation of OPT_q for databases of m elements if for every $S \in X^m$ there exists a solution $s \in \mathcal{B}$ such that $q(S, s) \geq \max_{f \in \mathcal{F}}\{q(S, f)\} - \alpha$.*

DEFINITION 5.4. *Let $\mathcal{B}$ be an α-deterministic representation of OPT_q for databases of m elements. Denote $\Delta \triangleq \frac{m}{\text{size}(\mathcal{B})}$. If $\Delta > 1$, then we say that the* ratio *of $\mathcal{B}$ is Δ.*

An α-deterministic representation $\mathcal{B}$ with ratio Δ is required to support all the databases of $m = \Delta \cdot \text{size}(\mathcal{B})$ elements. That is, for every $S \in X^m$, the set $\mathcal{B}$ is required to contain at least one α-good solution.

Fix $S \in X^m$. Intuitively, Δ controls the ratio between m and number of bits needed to represent an α-good solution for S. As $\mathcal{B}$ contains an α-good solution for S, and assuming $\mathcal{B}$ is publicly known, this solution could be represented with $\ln|\mathcal{B}| = \text{size}(\mathcal{B}) = m/\Delta$ bits.

DEFINITION 5.5. *Let* OPT_q *be an optimization problem over a universe X and a set of solutions $\mathcal{F}$. Let $\mathcal{P}$ be a distribution over $\{1, 2, \ldots, r\}$, and let $\mathcal{B} = \{\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_r\}$ be a family of solution sets for OPT_q. We denote $\mathrm{size}(\mathcal{B}) = \max\{\ln|\mathcal{B}_i| : \mathcal{B}_i \in \mathcal{B}\}$. We say that $(\mathcal{B}, \mathcal{P})$ is an (α, β)-probabilistic representation of OPT_q for databases of m elements if for every $S \in X^m$:*

$$\Pr_{\mathcal{P}} \left[\exists s \in \mathcal{B}_i \ \ s.t. \ \ q(S, s) \geq \max_{f \in \mathcal{F}}\{q(S, f)\} - \alpha \right] \geq 1 - \beta.$$

DEFINITION 5.6. *Let $(\mathcal{B}, \mathcal{P})$ be an (α, β)-probabilistic representation of OPT_q for databases of m elements. Denote $\Delta \triangleq \frac{m}{\mathrm{size}(\mathcal{B})}$. If $\Delta > 1$, then we say that the* ratio *of the representation is Δ.*

DEFINITION 5.7. *An optimization problem OPT_q is* bounded *if $\left| |S_1| \cdot q(S_1, f) - |S_2| \cdot q(S_2, f) \right| \leq 1$ for every solution f and every two neighboring databases S_1, S_2.*

We are interested in approximating bounded optimization problems, while guaranteeing differential privacy:

DEFINITION 5.8. *Let OPT_q be a bounded optimization problem over a universe X and a set of solutions $\mathcal{F}$. An algorithm A is an $(\alpha, \beta, \epsilon)$-private approximation algorithm for OPT_q with a database of m records if:*

1. *Algorithm A is ϵ-differentially private (as formulated in Definition 2.1);*

2. *For every $S \in X^m$, algorithm A outputs with probability at least $(1 - \beta)$ a solution s such that $q(S, s) \geq \max_{f \in \mathcal{F}}\{q(S, f)\} - \alpha$.*

EXAMPLE 5.9 (SANITIZATION). *Consider a class of predicates $\mathcal{C}$ over X. A database S contains points taken from X. A predicate query Q_c for $c \in \mathcal{C}$ is defined as $Q_c(S) = \frac{1}{|S|} \cdot |\{x_i \in S : c(x_i) = 1\}|$. Blum et al. [5] defined a sanitizer (or data release mechanism) as a differentially private algorithm that, on input a database S, outputs another database $\hat{S}$ with entries taken from X. A sanitizer A is (α, β)-useful for predicates in the class $\mathcal{C}$ if for every database S it holds that*

$$\Pr_A \left[\forall c \in C \ \ |Q_c(S) - Q_c(\hat{S})| \leq \alpha \right] \geq 1 - \beta.$$

This scenario can be viewed as a bounded optimization problem: The solutions are sanitized databases. For an input database S and and a sanitized database $\hat{S}$, the quality function is

$$q(S, \hat{S}) = 1 - \max_{c \in C} \left\{ |Q_c(S) - Q_c(\hat{S})| \right\}.$$

To see that this optimization problem is bounded, note that for every two neighboring databases S_1, S_2 of m elements, and every $c \in C$ it holds that $|Q_c(S_1) - Q_c(S_2)| \leq \frac{1}{m}$. Therefore, for every sanitized database f,

$$m \cdot |q(S_1, f) - q(S_2, f)|$$
$$= m \cdot \left| \max_{c \in C}\{|Q_c(S_1) - Q_c(f)|\} - \max_{c \in C}\{|Q_c(S_2) - Q_c(f)|\} \right|$$
$$\leq 1$$

The next two lemmas establish an equivalence between a private approximation algorithm and a probabilistic representation for a bounded optimization problem.

LEMMA 5.10. *Let OPT_q be a bounded optimization problem over a universe X. If there exists a pair $(\mathcal{B}, \mathcal{P})$ that (α, β)-probabilistically represents OPT_q for databases of m elements, s.t. the ratio of $(\mathcal{B}, \mathcal{P})$ is $\Delta > 1$, then for every $\hat{\alpha}, \hat{\beta}, \epsilon$ satisfying*

$$\Delta \geq \frac{2}{\epsilon \hat{\alpha}} \left(1 + \frac{\ln(1/\hat{\beta})}{\mathrm{size}(\mathcal{B})} \right),$$

there exists an $\big((\alpha + \hat{\alpha}), (\beta + \hat{\beta}), \epsilon\big)$-approximation algorithm for OPT_q with a database of size m.

PROOF. Consider the following algorithm A:

> Inputs: a database $S \in X^m$, and a privacy parameter ϵ.
>
> 1. Randomly choose $\mathcal{B}_i \in_{\mathcal{P}} \mathcal{B}$.
>
> 2. Choose $s \in \mathcal{B}_i$ using the exponential mechanism, that is, with probability
> $$\frac{\exp(\epsilon \cdot m \cdot q(S, s)/2)}{\sum_{f \in \mathcal{B}_i} \exp(\epsilon \cdot m \cdot q(S, f)/2)}.$$

By the properties of the exponential mechanism, A is ϵ-differentially private. Fix a database $S \in X^m$, and define the following 2 bad events:

E_1 The set $\mathcal{B}_i$ chosen in step 1 does not contain a solution s s.t. $q(S, s) \geq \max_{f \in \mathcal{F}}\{q(S, f)\} - \alpha$.

E_2 The solution s chosen in step 2 is such that $q(S, s) < \max_{t \in \mathcal{B}_i} q(S, t) - \hat{\alpha}$.

Note that if those two bad events do not occur, algorithm A outputs a solution s such that $q(S, s) \geq \max_{f \in \mathcal{F}}\{q(S, f)\} - \alpha - \hat{\alpha}$. As $(\mathcal{B}, \mathcal{P})$ is an (α, β)-probabilistic representation of OPT_q for databases of size m, event E_1 happens with probability at most β. By the properties of the exponential mechanism, the probability of event E_2 is bounded by $|\mathcal{B}_i| \cdot \exp(-\epsilon m \hat{\alpha}/2)$. As $m = \Delta \mathrm{size}(\mathcal{B})$, this probability is at most

$$\Pr[E_2] \leq \mathrm{size}(\mathcal{B}) \cdot \exp(-\epsilon m \hat{\alpha}/2)$$
$$= \mathrm{size}(\mathcal{B}) \cdot \exp(-\epsilon \Delta \mathrm{size}(\mathcal{B})\hat{\alpha}/2)$$
$$\leq \mathrm{size}(\mathcal{B}) \cdot \exp\left(-\left(1 + \frac{\ln(1/\hat{\beta})}{\mathrm{size}(\mathcal{B})} \right) \mathrm{size}(\mathcal{B}) \right)$$
$$= \mathrm{size}(\mathcal{B}) \cdot \exp(-\mathrm{size}(\mathcal{B}) - \ln(1/\hat{\beta})) = \hat{\beta}.$$

Therefore, algorithm A outputs an $(\alpha + \hat{\alpha})$-good solution with probability at least $(1 - \beta - \hat{\beta})$. $\square$

LEMMA 5.11. *Let OPT_q be an optimization problem. If there exists an $(\alpha, \beta, \epsilon)$-private approximation algorithm for OPT_q with a database of m records, then for every $\hat{\beta}$ satisfying*

$$\Delta \triangleq \frac{m}{\ln(\frac{1}{1-\beta}) + \ln\ln(\frac{1}{\hat{\beta}}) + m \cdot \epsilon} > 1,$$

there exists a pair $(\mathcal{B}, \mathcal{P})$ that $(\alpha, \hat{\beta})$-probabilistically represents OPT_q for databases of m elements, where the ratio of the representation is Δ.

PROOF. Let A be an $(\alpha, \beta, \epsilon)$-private approximation algorithm for OPT_q, with a sample size m. Fix an arbitrary input database $S \in X^m$. Define G as the set of all solutions s, possibly outputted by A, such that $q(S, s) \geq \max_{f \in \mathcal{F}} \{ q(S, f) \} - \alpha$. As A is an $(\alpha, \beta, \epsilon)$-approximation algorithm, $\Pr_A [A(S) \in G] \geq 1 - \beta$. As A is ϵ-differentially private, $\Pr_A \left[A(\vec{0}) \in G \right] \geq (1 - \beta) e^{-m\epsilon}$, where $\vec{0}$ is a database with m zeros. That is, $\Pr_A \left[A(\vec{0}) \notin G \right] \leq 1 - (1 - \beta) e^{-m\epsilon}$. Now, consider a set $\mathcal{B}$ containing the outcomes of $\Gamma \triangleq \frac{1}{1-\beta} \ln(\frac{1}{\hat{\beta}}) e^{m\epsilon}$ executions of $A(\vec{0})$. The probability that $\mathcal{B}$ does not contain a solutions $s \in G$ is at most $(1 - (1 - \beta) e^{-m\epsilon})^\Gamma \leq \hat{\beta}$. Thus, $\mathscr{B} = \{ \mathcal{B} \subseteq support(A) : |\mathcal{B}| \leq \Gamma \}$, and $\mathcal{P}$, the distribution induced by $A(\vec{0})$, are an $(\alpha, \hat{\beta})$-probabilistic representation of OPT_q for databases with m elements. Moreover, the ratio of the representation is

$$\frac{m}{\mathrm{size}(\mathscr{B})} = \frac{m}{\max\{ \ln|\mathcal{B}| : \mathcal{B} \in \mathscr{B} \}}$$
$$= \frac{m}{\ln(\frac{1}{1-\beta}) + \ln\ln(\frac{1}{\hat{\beta}}) + m\epsilon} = \Delta.$$

$\square$

5.1 Exact 3SAT

Consider the following bounded optimization problem, denoted as $\mathrm{OPT}_{\mathrm{E3SAT}}$: The universe X is the set of all possible clauses with exactly 3 different literals over n variables, and the set of solutions $\mathcal{F}$ is the set of all possible 2^n assignments. Given a database $S = (\sigma_1, \sigma_2, \ldots, \sigma_m)$ containing m E3CNF clauses, the quality of an assignment $a \in \mathcal{F}$ is

$$q(S, a) = \frac{|\{i : a(\sigma_i) = 1\}|}{m}.$$

Aiming at the (very different) objective of secure protocols for search problems, Beimel et al. [2] defined the notation of solution-list algorithms, which corresponds to our notation of deterministic representation. We next rephrase their results using our notations.

R1 For every $\alpha > 0$ and every $\Delta > 1$, there exists a set $\mathcal{B}$ that $(\alpha + 1/8)$-deterministically represents $\mathrm{OPT}_{\mathrm{E3SAT}}$ for databases of size $m = O\big(\Delta(\ln\ln(n) + \ln(1/\alpha))\big)$, and a ratio of Δ.

R2 Let $\alpha < 1/2$ and $\Delta > 1$. For every set $\mathcal{B}$ that α-deterministically represents $\mathrm{OPT}_{\mathrm{E3SAT}}$ for databases of size m with a ratio of Δ, it holds that $m = \Omega\big(\ln\ln(n)\big)$.

Using $(R1)$ and a deterministic version of Lemma 5.10, for every $\alpha, \beta, \epsilon > 0$, there exists an $\big((1/8 + \alpha), \beta, \epsilon\big)$- approximation algorithm for $\mathrm{OPT}_{\mathrm{E3SAT}}$ with a database of $m = O_{\alpha, \beta, \epsilon}(\ln\ln(n))$ clauses. By $(R2)$, this is the best possible using a deterministic representation.

We can reduce the necessary database size, using a probabilistic representation. Fix a clause with three different literals. If we pick an assignment at random, then with probability at least $7/8$ it satisfies the clause. Now, fix any exact 3CNF formula. If we pick an assignment at random, then the expected fraction of satisfied clauses is at least $7/8$. Moreover, for every $0 < \alpha < 7/8$, the fraction of satisfied clauses is at least $(7/8 - \alpha)$ with probability at least $\frac{\alpha}{\alpha + 1/8}$. So, if we pick $t = \frac{\ln(1/\beta)}{\ln(\alpha + 1/8) + \ln(1/\alpha)}$ random assignments, the

probability that none of them will satisfy at least $(7/8 - \alpha)m$ clauses is at most $\left(\frac{\alpha}{\alpha + 1/8} \right)^t = \beta$. So, for every $\Delta > 1$,

$$\mathscr{B} = \{ \mathcal{B} : \mathcal{B} \text{ is a set of at most } t \text{ assignments} \},$$

and $\mathcal{P}$, the distribution induced on $\mathscr{B}$ by randomly picking t assignments, are an $\big((1/8 + \alpha), \beta\big)$-probabilistic representation of $\mathrm{OPT}_{\mathrm{E3SAT}}$ for databases of size $\Delta \cdot \ln(t)$ and a ratio of Δ. By Lemma 5.11, for every ϵ there exists an $\big((1/8 + \alpha), \beta, \epsilon\big)$-approximation algorithm for $\mathrm{OPT}_{\mathrm{E3SAT}}$ with a database of $m = o_{\alpha, \beta, \epsilon}(1)$ clauses.

6. EXTENSIONS

6.1 (ϵ, δ)-Differential Privacy

The notation of ϵ-differential privacy was generalized to (ϵ, δ)-differential privacy, where the requirement in inequality (1) is changed to

$$\Pr[A(S_1) \in \mathcal{F}] \leq \exp(\epsilon) \cdot \Pr[A(S_2) \in \mathcal{F}] + \delta.$$

The proof of Lemma 3.12 remains valid even if algorithm A is only (ϵ, δ)-differential private for

$$\delta \leq \tfrac{1}{8} e^{-8\alpha\epsilon m}(1 - e^{-\epsilon}). \qquad (5)$$

To see this, note that inequality (3) changes to

$$\Pr_A \left[A(\vec{0}) \in G \right] \geq$$
$$\geq \left(\left(\left(\Pr_A [A(S) \in G] \cdot e^{-\epsilon} - \delta \right) e^{-\epsilon} - \delta \right) \cdots \right) e^{-\epsilon} - \delta$$
$$\geq \frac{1}{4} e^{-8\alpha\epsilon m} - \delta \left(\sum_{i=0}^{8\alpha m - 1} e^{-i\epsilon} \right)$$
$$\geq \frac{1}{4} e^{-8\alpha\epsilon m} - \delta \left(\frac{1}{1 - e^{-\epsilon}} \right) \geq \frac{1}{8} e^{-8\alpha\epsilon m}.$$

The rest of the proof remains almost intact (only minor changes in the constants). With that in mind, we see that the lower bound showed in Theorem 3.15 for ϵ-differentially private (that is, with $\delta = 0$) learners also applies for (ϵ, δ)-differentially private learners satisfying inequality (5). That is, every such learner for a class $\mathcal{C}$ must use $\Omega \left(\frac{\mathrm{RepDim}(\mathcal{C})}{\alpha\epsilon} \right)$ samples.

When using (ϵ, δ)-differential privacy, δ should be negligible in the security parameter, that is, in d – the representation length of elements in X_d. Therefore, using (ϵ, δ)-differential privacy instead of ϵ-differential privacy cannot reduce the sample complexity for PPAC learning a concept class $\mathcal{C}$ whenever $\mathrm{RepDim}(\mathcal{C}) = O(\log(d))$.

6.2 Probabilistic Representation Using a Hypothesis Class

We will now consider a generalization of our representation notations that can be useful when considering PPAC learners that use a specific hypothesis class. In particular, those notation can be useful when considering proper-PPAC learners, that is, a learner that learns a class $\mathcal{C}$ using a hypothesis class $\mathcal{B} \subseteq \mathcal{C}$.

DEFINITION 6.1. *We define the α-Deterministic Representation Dimension of a concept class $\mathcal{C}$ using a hypothesis*

class $\mathcal{B}$ as

$$\mathrm{DRepDim}_\alpha(\mathcal{C}, \mathcal{B}) = \min \left\{ \mathrm{size}(\mathcal{H}) : \begin{array}{c} \mathcal{H} \subseteq \mathcal{B} \text{ is an} \\ \alpha\text{-representation} \\ \text{for class } \mathcal{C} \end{array} \right\}.$$

Note that $\mathrm{DRepDim}_{\frac{1}{4}}(\mathcal{C}, 2^{X_d}) = \mathrm{DRepDim}(\mathcal{C})$. The dependency on α in the above definition is necessary: if $\mathcal{C}$ is not contained in $\mathcal{B}$ then for every small enough α, the hypothesis class $\mathcal{B}$ itself does not α-represents $\mathcal{C}$ (and therefore no subset $\mathcal{H} \subseteq \mathcal{B}$ can α-represent $\mathcal{C}$). Moreover, when considering the notations of representation using a hypothesis class, our boosting technique for α does not work (as the boosting uses more complex hypotheses).

EXAMPLE 6.2. *Beimel et al. [3] showed that for every $\alpha < 1$, every subset $\mathcal{H} \subsetneq \mathrm{POINT}_d$ does not α-represent the class POINT_d. Therefore, $\mathrm{DRepDim}_\alpha(\mathrm{POINT}_d, \mathrm{POINT}_d) = \theta(d)$ for every $\alpha < 1$.*

DEFINITION 6.3. *A pair $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for a concept class C using a hypothesis class $\mathcal{B}$ if:*

1. *$(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for the class C, as formulated in Definition 3.5.*

2. *Every $\mathcal{H}_i \in \mathscr{H}$ is a subset of $\mathcal{B}$.*

Note that whenever $\mathcal{B} = 2^{X_d}$, this definition is identical to Definition 3.5. Using this general notation, we can restate Lemma 3.9 and Lemma 3.12 as follows:

LEMMA 6.4. *If there exists a pair $(\mathscr{H}, \mathcal{P})$ that (α, β)- probabilistically represents a class $\mathcal{C}$ using a hypothesis class $\mathcal{B}$, then for every ϵ and every γ there exists an algorithm A that $(\alpha + \gamma, 3\beta, \epsilon)$-PPAC learns $\mathcal{C}$ using $\mathcal{B}$ and a sample size $m = O((\mathrm{size}(\mathscr{H}) + \ln(\frac{1}{\beta})) \max\{\frac{1}{\gamma\epsilon}, \frac{1}{\gamma^2}\})$.*

Note that in the above lemma the resulting algorithm A has accuracy $(\alpha + \gamma)$ as opposed to 6α in lemma 3.9, where γ is arbitrary. While in section 3 we did not mind the multiplicative factor of 6 in the accuracy parameter (as we could boost it back), replacing it with an additive factor of γ might be of value in this section as our boosting technique for the accuracy parameter does not work here. As an example, consider a representation with $\alpha = \frac{1}{10}$. Without boosting capabilities, this change makes the difference between the ability to generate an algorithm with $\alpha = \frac{6}{10}$, or an algorithm with $\alpha = \frac{1}{10} + \frac{1}{1000}$.

LEMMA 6.5. *If there exists an algorithm A that $(\alpha, \frac{1}{2}, \epsilon)$-PPAC learns a concept class $\mathcal{C}$ using a hypothesis class $\mathcal{B}$ and a sample size m, then there exists a pair $(\mathscr{H}, \mathcal{P})$ that $(\alpha, 1/4)$-probabilistically represents the class $\mathcal{C}$ using the hypothesis class $\mathcal{B}$ where $\mathrm{size}(\mathscr{H}) = O(m\epsilon)$.*

DEFINITION 6.6. *We define the α-Probabilistic Representation Dimension of a concept class $\mathcal{C}$ using a hypothesis class $\mathcal{B}$ as*

$$\mathrm{RepDim}_\alpha(\mathcal{C}, \mathcal{B}) = \min \left\{ \mathrm{size}(\mathscr{H}) : \begin{array}{c} \exists \mathcal{P} \text{ s.t. } (\mathscr{H}, \mathcal{P}) \\ \text{is an } (\alpha, \frac{1}{4})\text{-prob.} \\ \text{representation} \\ \text{for } \mathcal{C} \text{ using } \mathcal{B} \end{array} \right\}.$$

EXAMPLE 6.7. *Beimel et al. [3] showed that for every $\alpha < 1$, every proper-PPAC learner for POINT_d requires $\Omega((d + \log(1/\beta))/(\epsilon\alpha))$ labled examples. Using Lemma 6.4, we get that $\mathrm{RepDim}_\alpha(\mathrm{POINT}_d, \mathrm{POINT}_d) = \Omega(d)$.*

We still do not know the relation between the representation dimension of a concept class and its VC dimension. However, the above example shows a strong separation between the VC dimension of the class POINT_d and $\mathrm{RepDim}_\alpha(\mathrm{POINT}_d, \mathrm{POINT}_d)$.

7. A PROBABILISTIC REPRESENTATION FOR POINTS

Example 3.6 states the existence of a constant size probabilistic representation for the class POINT_d. We now give the construction.

CLAIM 7.1. *There exists an (α, β)-probabilistic representation for POINT_d of size $\ln(4/\alpha) + \ln\ln(1/\beta)$. Furthermore, each hypothesis h in each $\mathcal{H}_i$ has a short description and given x, the value $h(x)$ can be computed efficiently.*

PROOF. Consider the following set of hypothesis classes

$$\mathscr{H} = \left\{ \mathcal{H} \subseteq 2^{X_d} : |\mathcal{H}| \leq \frac{4}{\alpha} \ln(\frac{1}{\beta}) \right\}.$$

That is, $\mathcal{H} \in \mathscr{H}$ if $\mathcal{H}$ contains at most $\frac{4}{\alpha} \ln(\frac{1}{\beta})$ boolean functions. We will show an appropriate distribution $\mathcal{P}$ s.t. $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation of the class POINT_d. To this end, fix a target concept $c_j \in \mathrm{POINT}_d$ and a distribution $\mathcal{D}$ on X_d (remember that j is the unique point on which $c_j(j) = 1$). We need to show how to randomly choose an $\mathcal{H} \in_R \mathscr{H}$ such that with probability at least $(1-\beta)$ over the choice of $\mathcal{H}$, there will be at least one $h \in \mathcal{H}$ such that $\mathrm{error}_\mathcal{D}(c_j, h) \leq \alpha$. Consider the following process for randomly choosing an $\mathcal{H} \in \mathscr{H}$:

---

1. Denote $M = \frac{4}{\alpha} \ln(\frac{1}{\beta})$
2. For $i = 1, \ldots, M$ construct hypothesis h_i as follows:
   For each $x \in X_d$ (independently):
       Let $h_i(x) = 1$ with probability $\alpha/2$,
       and $h_i(x) = 0$ otherwise.
3. Return $\mathcal{H} = \{h_1, h_2, \ldots, h_M\}$.

The above process induces a distribution on $\mathscr{H}$, denoted as $\mathcal{P}$. We will next analyze the probability that the returned $\mathcal{H}$ does not contain an α-good hypothesis. We start by fixing some i and analyzing the expected error of h_i, conditioned on the event that $h_i(j) = 1$. The probability is taken over the random coins used to construct h_i.

$$\mathop{\mathbb{E}}_{h_i}\left[\mathrm{error}_\mathcal{D}(c_j, h_i) \mid h_i(j) = 1 \right] =$$

$$= \mathop{\mathbb{E}}_{h_i}\left[\mathop{\mathbb{E}}_{x \in \mathcal{D}}\left[|c_j(x) - h_i(x)| \right] \mid h_i(j) = 1 \right]$$

$$= \mathop{\mathbb{E}}_{x \in \mathcal{D}}\left[\mathop{\mathbb{E}}_{h_i}\left[|c_j(x) - h_i(x)| \mid h_i(j) = 1 \right] \right] \leq \frac{\alpha}{2}.$$

Using Markov's Inequality,

$$\mathop{\Pr}_{h_i}\left[\mathrm{error}_\mathcal{D}(c_j, h_i) \geq \alpha \mid h_i(j) = 1 \right] \leq \frac{1}{2}.$$

So, the probability that h_i is α-good for c_j and $\mathcal{D}$ is:

$$\Pr_{h_i}\left[\text{error}_{\mathcal{D}}(c_j, h_i) \leq \alpha\right] \geq$$

$$\geq \Pr_{h_i}\left[h_i(j) = 1\right] \cdot \Pr_{h_i}\left[\text{error}_{\mathcal{D}}(c_j, h_i) \leq \alpha \ \middle| \ h_i(j) = 1\right]$$

$$\geq \frac{\alpha}{2} \cdot \frac{1}{2} = \frac{\alpha}{4}.$$

Thus, the probability that $\mathcal{H}$ fails to contain an α-good hypothesis is at most $\left(1 - \frac{\alpha}{4}\right)^M$, which is less than β for our choice of M. This concludes the proof that $(\mathscr{H}, \mathcal{P})$ is an (α, β)-probabilistic representation for $\texttt{POINT}_d$.

When a hypothesis $h_i()$ was constructed in the above random process, the value of $h_i(x)$ was independently drawn for every $x \in X_d$. This results in a hypothesis whose description size is $O(2^d)$, which in turn, will result in a non efficient learning algorithm. We next construct hypotheses whose description is short. To achieve this goal, we note that in the above analysis we only care about the probability that $h_i(x) = 0$ given that $h_i(j) = 1$. Thus, we can choose the values of h_i in a pairwise independent way, e.g., using a random polynomial of degree 2. The size of the description in this case is $O(d)$. $\quad\square$

OBSERVATION 7.2. *Consider the class* $\texttt{POINT}_{\mathbb{N}}$, *defined in Example 3.11. The above construction can be adjusted to yield an (inefficient) improper private learner for* $\texttt{POINT}_{\mathbb{N}}$ *with* $O_{\alpha,\beta,\epsilon}(1)$ *samples. The only adjustments necessary are in the construction of the* (α, β)-*probabilistic representation. Specifically, we need to specify how to randomly draw a boolean function* h *over the natural numbers, such that for every* $x \in \mathbb{N}$ *the probability of* $h(x) = 1$ *is* $\alpha/2$, *and the values of* h *on every two distinct points in* $\mathbb{N}$ *are independent. This can be done easily, as a random real number could be interpreted as a random function over* $\mathbb{N}$.

8. REFERENCES

[1] A. Beimel, H. Brenner, S. P. Kasiviswanathan, and K. Nissim. Bounds on the sample complexity for private learning and private data release. Full version of [3], in submition, 2012.

[2] A. Beimel, P. Carmi, K. Nissim, and E. Weinreb. Private approximation of search problems. *SIAM J. Comput.*, 38(5):1728–1760, 2008.

[3] A. Beimel, S. P. Kasiviswanathan, and K. Nissim. Bounds on the sample complexity for private learning and private data release. In *7th Theory of Cryptography Conference*, pages 437–454, 2010.

[4] A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: The SuLQ framework. In *PODS*, pages 128–138. ACM, 2005.

[5] A. Blum, K. Ligett, and A. Roth. A learning theory approach to non-interactive database privacy. In *STOC*, pages 609–618. ACM, 2008.

[6] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the Association for Computing Machinery*, 36(4):929–965, 1989.

[7] K. Chaudhuri and D. Hsu. Sample complexity bounds for differentially private learning. *Journal of Machine Learning Research – COLT 2011 Proceedings*, 19:155–186, 2011.

[8] H. Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Ann. Math. Statist.*, 23:493–507, 1952.

[9] C. Dwork. The differential privacy frontier. In O. Reingold, editor, *TCC*, volume 5444 of *LNCS*, pages 496–502. Springer, 2009.

[10] C. Dwork. A firm foundation for private data analysis. *Commun. of the ACM*, 54(1):86–95, 2011.

[11] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In S. Halevi and T. Rabin, editors, *TCC*, volume 3876 of *LNCS*, pages 265–284. Springer, 2006.

[12] C. Dwork, G. N. Rothblum, and S. P. Vadhan. Boosting and differential privacy. In *51th Annual IEEE Symposium on Foundations of Computer Science*, pages 51–60, 2010.

[13] A. Ehrenfeucht, D. Haussler, M. J. Kearns, and L. G. Valiant. A general lower bound on the number of examples needed for learning. *Inf. Comput.*, 82(3):247–261, 1989.

[14] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119 – 139, 1997.

[15] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[16] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith. What can we learn privately? In *48th Annual IEEE Symposium on Foundations of Computer Science*, pages 531–540. IEEE Computer Society, 2008.

[17] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science*, pages 94–103. IEEE, 2007.

[18] R. E. Schapire. The strength of weak learnability. *Mach. Learn.*, 5(2):197–227, 1990.

[19] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27:1134–1142, 1984.

[20] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16:264, 1971.