CS839: Differential Privacy and Learning April 16, 2026

Lecture 24: Interior Point and PAC Learning

Instructor: Gavin Brown *Scribe:* Gavin Brown

Disclaimer: This document is intended as an informal supplement to in-class note-taking. It has not been given the level of scrutiny expected in polished lecture notes, let alone that reserved for peer-reviewed publications.

Today we'll review the lower bounds for private interior point, talk a little about algorithms for the problem, and then discuss PAC learning and private learning of thresholds. Next week we will see how to use interior point as part of a larger algorithm to privately approximate (almost) any univariate function.

1 Review: Private Interior Point

Recall the setting: $\mathcal{X}$ is a totally ordered domain, e.g., $\mathcal{X} \subseteq [0,1]$ or $\mathcal{X} = \{0, 1, 2, \ldots, |\mathcal{X}| - 1\}$.

Definition 1.1. An algorithm $A : \mathcal{X}^n \to \mathcal{X}$ solves the *interior point problem* on $\mathcal{X}$ with error probability β if for all $X \in \mathcal{X}^n$

$$\Pr\left[\min_{i\in[n]} x_i \le A(X) \le \max_{i\in[n]} x_i\right] \ge 1 - \beta.$$

This is trivial without privacy (output any data point). For DP algorithms, however, we have the following lower bound of Bun et al. [2015].

Theorem 1.2 (BNSV)**.** *Fix $\varepsilon \in (0, 1/4)$ and $\delta(n) \le 1/(50n^2)$. For every $n \in \mathbb{N}$, solving the interior point problem on $\mathcal{X}$ with error less than $1/4$ and $(\varepsilon, \delta(n))$-DP requires $n = \Omega(\log^*(|\mathcal{X}|))$.*

Here $\log^*$ is the *iterated logarithm*, the minimum integer k such that

$$\underbrace{\log(\log(\cdots \log(z))}_{k \text{ times}} \le 1.$$

This is an extremely slow-growing function: for $z \in (16, 65536]$, $\log^*(z) = 4$, and for $z \in (65536, 2^{65536}]$, $\log^*(z) = 5$. As we said last class, the theorem is both remarkably weak (since $\log^*$ grows so slowly) and remarkably strong (as an impossibility result for many real-valued tasks).

It was only implicit last class, but the theorem is ignorant to the scale of $\mathcal{X}$: it applies to any totally ordered domain. So the private interior point problem is as impossible over $\mathcal{X} = [0, 1]$ as it is over $\mathcal{X} = \mathbb{R}$. Contrast this with our statistical results: if you want to estimate the mean of data from a Gaussian distribution $\mathcal{N}\mu, 1)$ and are promised that $\mu \in [0, 1]$, then the problem is almost trivial: clip the data (say, to $[-5, +6]$) and add Laplace noise to the empirical mean. Without such prior knowledge we need more complicated techniques. But this type of assumption doesn't help for private interior point.

Another type of assumption that doesn't help is assuming that the data comes i.i.d. from some fixed but unknown distribution. This is a natural assumption to make, since it's the standard setting for statistics and

machine learning. But for any dataset $X = (x_1, \ldots, x_n)$,

$$p_X = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i}$$

is a distribution. (Here δ_{x_i} means a point mass on x_i.) And any interior point for data from p_X is an interior point for X.

Exercise 1.3. Can you formalize the above reasoning as a corollary?

There are assumptions that can sidestep Theorem 1.2: if we assume the data come from a Gaussian distribution, for example, then any non-trivial mean estimate will usually be an interior point of the samples. Weaker assumptions suffice as well [Aliakbarpour et al., 2023].

2 Algorithms for Private Interior Point

Now we will discuss some algorithms that meet Theorem 1.2 on its own terms. The best known algorithm comes from Cohen et al. [2023].

Theorem 2.1. *There is an (ε, δ)-DP algorithm that solves the private interior point problem on $\mathcal{X}$ with error δ using $n = O(\log^*(|\mathcal{X}|) \log^2(1/\delta)/\varepsilon)$ samples.*

The algorithm is complicated and we will not cover it here. However, we have already seen a couple of algorithms which solve the task, albeit with far worse dependence on $|\mathcal{X}|$.

The simplest is the StableHistogram algorithm ("Approx DP Histogram" on Homework 1). Once we have approximately $\log(1/\delta)/\varepsilon$ copies of the same point in our data, we can deterministically release its identity. Thus, if $n \approx |\mathcal{X}| \cdot \log(1/\delta)/\varepsilon$, this algorithm will reliably give us an interior point.

Much better is the exponential mechanism for the median: sample from

$$p(y) \propto \exp\left(-\frac{\varepsilon}{2}\left|\frac{n}{2} - \#\{i : x_i \le y\}\right|\right).$$

This is ε-DP and we can use our off-the-shelf utility analysis: there is at least one median $x^* \in X$ with weight $\exp((\varepsilon/2)|n/2 - n/2|) = e^0$ and points outside the data receive weight $\exp((\varepsilon/2) \cdot (n/2))$. So we bound

$$\begin{aligned}
\Pr[A(X) \text{ not IP}] &\le \frac{\Pr[A(X) \text{ not IP}]}{\Pr[A(X) = y^*]} \\
&\le \frac{\#\{\text{candidates outside data}\} \cdot \exp(\varepsilon n/4)}{1} \\
&\le |\mathcal{X}| \cdot \exp(\varepsilon n/4).
\end{aligned}$$

So this algorithm will succeed on any dataset with probability $1 - \beta$ once $n \ge \frac{4 \log(|\mathcal{X}|/\beta)}{\varepsilon}$. This is actually optimal for pure-DP [Beimel et al., 2019, Feldman and Xiao, 2014].

3 PAC Learning at a Glance

The PAC learning model is the most well-studied setting in learning theory. "PAC" stands for *probably approximately correct*. We have a data domain $\mathcal{X}$, possibly unordered, a binary label space $\{0, 1\}$ and a hypothesis class $\mathcal{H} \subseteq \{f : \mathcal{X} \to \{0, 1\}\}$. For a given function and data distribution D over $\mathcal{X} \times \{0, 1\}$, we use the misclassification loss:

$$L_D(f) = \Pr_{(x,y)\sim D}[f(x) \ne y].$$

Definition 3.1. A distribution D over $\mathcal{X} \times \{0,1\}$ is *realizable* by $\mathcal{H}$ if there exists $f^* \in \mathcal{H}$ such that $L_D(f*) = 0$.

Definition 3.2. An algorithm $A : \mathcal{X}^n \to \mathcal{H}$ is an (α, β)-*PAC learner* if for all realizable D, with probability at least $1 - \beta$, for $X \sim D^{\otimes n}$, $A(X) = f$ with $L_D(f) \leq \alpha$.

In other words, a PAC learner probably outputs a hypothesis which is approximately correct.

Our motivating example is the hypothesis class of *threshold functions*: for a totally ordered $\mathcal{X}$, let

$$\mathcal{H}_{\text{thresh}} = \{f_t^{\text{thresh}} : t \in \mathcal{X}\}, \qquad \text{where} \qquad f_t^{\text{thresh}}(x) = \begin{cases} 1 & \text{if } x \geq t \\ 0 & \text{otherwise} \end{cases}.$$

Exercise 3.3. Draw a picture of this. (You do have pencil and paper out, don't you?)

Without privacy, there is an (α, β)-PAC learner for thresholds using

$$n = \theta\left(\frac{\log(1/\beta)}{\alpha}\right) \tag{1}$$

samples. There is a very simple learner: output any f_t^{thresh} that labels all data points correctly (this is called *empirical risk minimization*).

The key fact we will use is that any hypothesis which correctly labels at least a $1 - O(\alpha)$ fraction of the dataset will work.

3.1 Aside: Vapnik–Chervonenkis Dimension

The non-private sample complexity in Equation (1) has no dependence on the size of the domain/hypothesis class. This itself is pretty surprising! It is part of a beautiful set of results that form the bedrock of what we call learning theory today. It is beyond the scope of this class, but if you are reading these notes you should definitely learn about it. As a teaser, I will include a fundamental result. The key definition is VC dimension, named for Vapnik and Chervonenkis [1982].

Definition 3.4 (Shattering)**.** A set $X = \{x_1, \ldots, x_n\} \subseteq \mathcal{X}$ is *shattered* by a hypothesis class $\mathcal{H}$ if, for any possible labeling $y_1, \ldots, y_n \in \{0,1\}$ there exists an $f \in \mathcal{H}$ with $f(x_i) = y_i$ for all i.

Definition 3.5. The *VC dimension* of a hypothesis class $\mathcal{H}$, denoted $\text{VC}(\mathcal{H})$, is the size of the largest set $X \subset \mathcal{X}$ which is shattered by $\mathcal{H}$.

Theorem 3.6 (Hanneke [2016])**.** *For any hypothesis class $\mathcal{H}$, there is an (α, β)-PAC learner using*

$$n = \Theta\left(\frac{\text{VC}(\mathcal{H}) + \log(1/\beta)}{\alpha}\right)$$

samples.

There is much more to say about this and related questions, both with privacy and without. For now, we will mention that the existence of a quantity which fully characterizes the sample complexity of PAC learning under approximate DP is one of the major open questions in learning theory.

4 Private Interior Point ⇔ Private Threshold Learning

Privately finding an interior point and privately learning a threshold are essentially equivalent, in the sense that we can use an algorithm for one to solve the other. We will give a highly informal sketch of one direction.

Suppose we have an algorithm A which (i) is (ε, δ)-DP and (ii) is an (α, β)-PAC learner for thresholds when given $N \approx \frac{n}{\alpha}$ samples.[1] Given a dataset X consisting of n examples from $\mathcal{X}$, our goal is to produce an interior point of X.

The approach is displayed in Figure 1. Let $N \approx \frac{n}{\alpha}$ be the number of examples our threshold learner expects. We take our n real samples and pad them with $N - n$ dummy examples, half at $\min \mathcal{X}$ and half at $\max \mathcal{X}$. This means our real samples represent an $n/N = n(\alpha/n) = \alpha$ fraction of the overall data. The top half of this expanded dataset receives label "1" and the bottom half receives label "0". This is the dataset we feed to algorithm A. Observe that there exists a threshold function which labels all the data points correctly (this is important, since A is only assumed to work for realizable distributions).

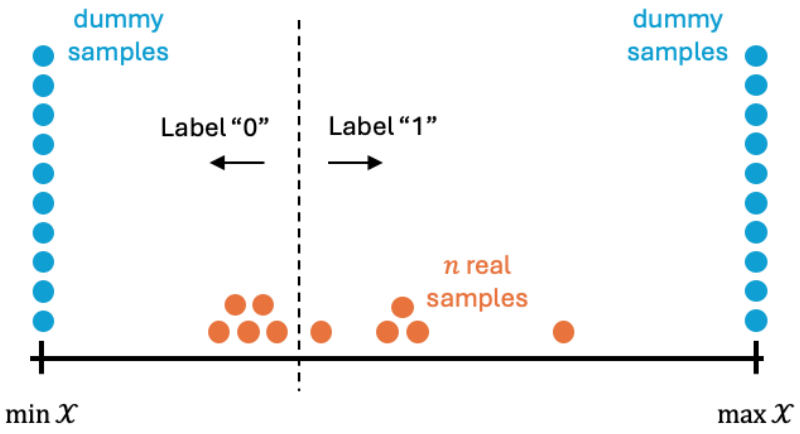

Figure 1: Threshold learning ⇒ interior point, via padding with dummy samples.

Now our argument becomes extremely informal: suppose A returns a good threshold function f_t. What do we mean by "good"? That it labels the input dataset almost perfectly. By the (α, β)-PAC-learning guarantee, we expect that f_t should correctly label all but an $\approx \alpha$ fraction of the data. Since our real samples represent the middle α fraction of the dataset, a good function f_t means that t is an interior point of the real data.

We leave the details and other direction as productive, but non-trivial, exercise.

Exercise 4.1. Make the above argument rigorous.

Exercise 4.2. Show how a DP algorithm for interior point on n examples can be used for private threshold learning on $O(n/\alpha)$ samples.

5 Further Reading

The above discussion in fully contained in Bun et al. [2015] and Bun [2016]. Our understanding of the sample complexity of differentially private PAC learning has grown dramatically in the past few years, with much of the work focusing on a quantity called *Littlestone dimension*, a quantity which characterizes the difficulty of

[1] We omit the failure probability β; for now one can assume it's a small constant.

learning a hypothesis class in an online setting. See Lyu [2026] for a recent result and many pointers. For more on PAC learning and VC dimension, see Shalev-Shwartz and Ben-David [2014].

References

Maryam Aliakbarpour, Rose Silver, Thomas Steinke, and Jonathan Ullman. Differentially private medians and interior points for non-pathological data. *arXiv preprint arXiv:2305.13440*, 2023.

Amos Beimel, Kobbi Nissim, and Uri Stemmer. Characterizing the sample complexity of pure private learners. *Journal of Machine Learning Research*, 20(146):1–33, 2019.

Mark Bun, Kobbi Nissim, Uri Stemmer, and Salil Vadhan. Differentially private release and learning of threshold functions. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 634–649. IEEE, 2015.

Mark Mar Bun. *New Separations in the Complexity of Differential Privacy*. PhD thesis, 2016.

Edith Cohen, Xin Lyu, Jelani Nelson, Tamás Sarlós, and Uri Stemmer. Optimal differentially private learning of thresholds and quasi-concave optimization. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 472–482, 2023.

Vitaly Feldman and David Xiao. Sample complexity bounds on differentially private learning via communication complexity. In *Conference on Learning Theory*, pages 1000–1019. PMLR, 2014.

Steve Hanneke. The optimal sample complexity of pac learning. *Journal of Machine Learning Research*, 17 (38):1–15, 2016.

Xin Lyu. Private learning of littlestone classes, revisited. In *Proceedings of the 58th Annual ACM Symposium on Theory of Computing*, pages 2218–2229, 2026.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Vladimir N Vapnik and A Ya Chervonenkis. Necessary and sufficient conditions for the uniform convergence of means to their expectations. *Theory of Probability & Its Applications*, 26(3):532–553, 1982.