Invited Open Problem: Does Differential Privacy Make PAC Learning Much Harder?

Kobbi Nissim KOBBI.NISSIM@GEORGETOWN.EDU
Georgetown University

Uri Stemmer U@URI.CO.IL
Tel Aviv University

Eliad Tsfadia ELIAD.TSFADIA@BIU.AC.IL
Bar-Ilan University

Editors: Steve Hanneke and Tor Lattimore

Abstract

What is the optimal sample complexity of differentially private (DP) PAC learning? Recent results establish that a concept class C is learnable under approximate DP if and only if it is online learnable. However, in any realistic computational model, C is finite, and it is well known that a sample complexity of $O(\log|C|)$ suffices for both online and DP learning. In contrast, non-private learning is characterized by the VC dimension of C, which can be significantly lower than $\log|C|$. While the gap between $\log|C|$ and $\mathrm{VC}(C)$ can be unavoidable for online learning (e.g., when learning thresholds over a finite domain), we currently lack evidence that the same holds true for DP learning. This leads to our central question: Is differentially private PAC learning much harder than non-private learning?

Keywords: Differential privacy, PAC learning, Littlestone dimension, VC dimension, sample complexity, online learning

1. Introduction

Non-private learning: VC dimension is the answer. The classical theory of PAC learning (Valiant, 1984) gives a clean and complete answer to the question of how many samples are needed to learn a hypothesis class C: the answer is $\Theta(\mathrm{VC}(C))$, where $\mathrm{VC}(C)$ is the Vapnik-Chervonenkis dimension. The VC dimension is thus the definitive complexity measure for non-private learning, and it can be much smaller than other natural measures of complexity, such as the description length of concepts from the class (i.e., $\log|C|$).

Private learning: how much more is needed? A randomized algorithm $A : (X \times \{0,1\})^n \to \mathcal{W}$ is (ε, δ)-*Differentially Private (DP)* (Dwork et al., 2006) if for every pair of neighboring datasets S, S' and every event $E \subseteq \mathcal{W}$ it holds that $\Pr[A(S) \in E] \leq e^{\varepsilon} \Pr[A(S') \in E] + \delta$. A *private learner* is a PAC learner that guarantees DP w.r.t. its training data.[1] It is well known that DP learning requires more samples than non-private learning for some classes. But the central question we ask is: *How much more* is needed? Is the answer close to $\mathrm{VC}(C)$, or could it be drastically larger?[2]

1. The standardly considered regime of parameters is to let ε be a small constant and let δ be a small function of the dataset size, much smaller than $1/|S|$.
2. For simplicity, throughout this paper we ignore the dependence of the sample complexity in the privacy and learning parameters, and focus on its dependence in complexity measures of the class C, such as $\mathrm{VC}(C)$, $\mathrm{LD}(C)$, and $\log|C|$.

© 2026 K. Nissim, U. Stemmer & E. Tsfadia.

In the original work that introduced the concept of private PAC learning, Kasiviswanathan, Lee, Nissim, Raskhodnikova, and Smith (2011) presented a generic construction showing that any concept class C can be privately learned with sample complexity linear in $\log|C|$. In a world where everything is finite, this suggests that *everything can be learned privately with "moderate" sample complexity*, which is already quite surprising. Still, $\log|C|$ can sometimes be very far from $\mathrm{VC}(C)$. This motivated a rich body of work aimed at understanding the quantitative difference between the sample complexity needed with or without privacy. To a large extent, this question still remains open.

Which classes *can* be learned privately? Following the work of Kasiviswanathan et al. (2011), several examples were given of *infinite* classes C (so $\log|C| = \infty$, and the generic construction of Kasiviswanathan et al. (2011) does not apply) that can nevertheless be learned with DP.[3] Thus, perhaps a more basic question than trying to characterize the sample complexity of DP learning is to first understand which classes *can* be learned privately. This question was resolved in a pair of landmark results by Alon, Livni, Malliaris, and Moran (2019) and Bun, Livni, and Moran (2020). Recall that the *Littlestone dimension* $\mathrm{LD}(C)$ (Littlestone, 1987) is a combinatorial parameter characterizing online learnability: C is online learnable if and only if $\mathrm{LD}(C) < \infty$, and the optimal mistake bound in the realizable setting equals $\mathrm{LD}(C)$. It is always the case that $\mathrm{VC}(C) \leq \mathrm{LD}(C)$, and the gap can be infinite (e.g., thresholds over $\mathbb{R}$ have $\mathrm{VC} = 1$ but $\mathrm{LD} = \infty$).

Theorem 1 (Alon et al. (2019); Bun et al. (2020)) *A class C is DP PAC learnable if and only if $\mathrm{LD}(C) < \infty$.*

For the remainder of this paper, we therefore restrict attention to classes with $LD(C) < \infty$ and ask how the DP sample complexity compares to $VC(C)$.

Existing quantitative bounds. We next state the quantitative results behind Theorem 1.

Theorem 2 (Lower bound, Alon et al. (2019)) *Let C be a concept class. Any DP PAC learner for C requires sample complexity $\Omega(\mathrm{VC}(C) + \log^*(\mathrm{LD}(C)))$.*

Theorem 3 (Upper bound, Bun et al. (2020); Ghazi et al. (2021); Lyu (2025)) *Let C be a concept class. There exists a DP PAC learner for C with sample complexity $\tilde{O}(\mathrm{LD}^5(C))$.*

Simplified proof sketches for Theorems 2 and 3 are given in Sections A and B, respectively.

2. Open Questions

Theorems 2 and 3, along with Kasiviswanathan et al. (2011), place the sample complexity of DP learning somewhere between $\Omega(\mathrm{VC}(C) + \log^*(\mathrm{LD}(C)))$ and $\min\{O(\log|C|), \mathrm{poly}(\mathrm{LD}(C))\}$. Because these bounds can be a tower-of-exponentials apart, the quest for understanding the sample complexity of private learning is far from over. The big question in this context would be to fully understand this sample complexity, either in terms of both $\mathrm{VC}(C)$ and $\mathrm{LD}(C)$, or possibly using different combinatorial measures.

3. See, e.g., Beimel et al. (2010, 2013a,b); Feldman and Xiao (2014).

Open Question 1 Characterize the sample complexity of private learning. That is, identify a combinatorial measure[4] of a class that determines the sample complexity of privately learning it, analogously to the characterization of non-private learning in terms of the VC dimension.

While we leave the question somewhat amorphous, we can consider more concrete formulations using *known* measures. For example, can private PAC learning be tightly characterized in terms of $\mathrm{VC}(C)$ and $\mathrm{LD}(C)$? An intermediate step toward this goal could be to reduce the gap between the current upper and lower bounds stated in Theorems 2 and 3. For example, showing a generic upper bound of $\mathrm{poly}(\mathrm{VC}(C), \log^*(\mathrm{LD}(C)))$. Even a generic upper bound of $\mathrm{poly}(\mathrm{VC}(C), \log(\mathrm{LD}(C)))$ would constitute major progress.

We remark that Open Question 1 was studied in special cases. For example, Beimel et al. (2013a) and Feldman and Xiao (2014) presented characterizations of the sample complexity of *pure DP* learning (i.e., when restricting the privacy parameter δ to be 0). As another example, Yan (2025) studied the sample complexity of privately learning classes C with $\mathrm{VC}(C) = 1$ and showed that it is essentially characterized by $\log^*(\mathrm{LD}(C))$. However, the unrestricted setting is still far from being well understood.

A significant effort has also been devoted to analyzing the sample complexity of DP learning of *specific* concept classes of interest. Simply put, before trying to understand the sample complexity of DP learning at large, maybe we should first understand it "only" for, say, learning halfspaces. By now, several interesting such examples where presented. In all of them, the DP sample complexity is much closer to the lower bound rather than to the upper bound. These examples include:

- **Thresholds over finite domains**: Cohen et al. (2023) showed that $\tilde{O}(\log^*(\mathrm{LD}(C)))$ samples suffice for privately learning 1-dimensional thresholds, essentially matching the lower bound.
- **Axis-aligned rectangles**: Sadigurschi and Stemmer (2021) and Cohen et al. (2023) showed that $\tilde{O}(\mathrm{VC}(C) \cdot \log^*(\mathrm{LD}(C)))$ samples suffice for privately learning high-dimensional axis-aligned rectangles.
- **Halfspaces**: Beimel et al. (2019) and Kaplan et al. (2020) showed that $\tilde{O}(\mathrm{VC}^{2.5}(C)) \cdot 2^{O(\log^*(\mathrm{LD}(C)))}$ samples suffice for privately learning high-dimensional halfspaces. Recently, Nissim et al. (2026) showed that $\tilde{O}(\mathrm{VC}^{5.5}(C) \cdot \log^*(\mathrm{LD}(C)))$ samples also suffice.

Note that in all examples above, the overhead is polynomial in $\mathrm{VC}(C)$ and a tiny function of $\mathrm{LD}(C)$. We currently do not have any example where the DP sample complexity must be "far" from the generic lower bound. For example, we currently do not know if there is a concept class C that requires sample complexity $\Omega\left(\mathrm{VC}(C) \cdot 2^{\log^*(\mathrm{LD}(C))}\right)$. Note, however, that this would still be tantalizingly close to the non-private sample complexity. What would constitute a "convincing" example showing that DP learning can be "significantly" harder than non-private learning? We formulate this in the following question:

4. We acknowledge that what constitutes a "combinatorial measure" is inherently open-ended. We intentionally avoid a rigid syntactic definition to prevent ruling out unforeseen structural properties. To borrow a famous phrase, we will "know it when we see it".

Open Question 2 Does there exist a sequence of finite-size classes $C = \{C_\kappa\}_{\kappa\in\mathbb{N}}$ where: (1) $\lim_{\kappa\to\infty} |C_\kappa| = \infty$, (2) $\log|C_\kappa|$ is superpolynomial[5] in $\mathrm{VC}(C_\kappa)$, and (3) The number of samples required to learn C_κ under differential privacy is $\Omega(\log|C_\kappa|)$.

An intermediate step toward this goal could be to find a class sequence C, satisfying Items (1) and (2) above, for which the private sample complexity is closer to $\log|C|$ than to $\mathrm{VC}(C)$ on a $\log$ scale, or even on a $\log\log$ scale. In other words, we measure the significance of the privacy overhead by where it falls on the spectrum between the non-private baseline of $\mathrm{VC}(C)$ and the $\log|C|$ upper bound. Of course, the reference to $\log|C|$ could be replaced with other quantities, such as $\mathrm{LD}(C)$ or $\mathrm{poly}(\mathrm{LD}(C))$. We chose $\log|C|$ as the reference point because it is a known upper bound of the sample complexity of DP learning that does not require a special combinatorial structure of C, making the problem more natural.

We note that for the specific concept classes mentioned earlier (thresholds, axis-aligned rectangles, and halfspaces), $\mathrm{LD}(C)$ is effectively $\log|C|$. In addition, Item (2) in Open Question 2 states that $\mathrm{VC}(C)$ must be tiny compared to $\log|C|$. Therefore, the known upper bounds for these classes are negligible compared to $\log|C|$, and are much closer to $\mathrm{VC}(C)$ than to $\log|C|$ in any reasonable scale. Hence, these specific classes cannot constitute a positive answer to Open Question 2 or to any of its relaxed forms (e.g., on a logarithmic scale).

References

Noga Alon, Roi Livni, Maryanthe Malliaris, and Shay Moran. Private PAC learning implies finite littlestone dimension. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019*, pages 852–860. ACM, 2019.

Amos Beimel, Shiva Prasad Kasiviswanathan, and Kobbi Nissim. Bounds on the sample complexity for private learning and private data release. In *TCC*, Lecture Notes in Computer Science, pages 437–454. Springer, 2010.

Amos Beimel, Kobbi Nissim, and Uri Stemmer. Characterizing the sample complexity of private learners. In *ITCS*, pages 97–110. ACM, 2013a.

Amos Beimel, Kobbi Nissim, and Uri Stemmer. Private learning and sanitization: Pure vs. approximate differential privacy. In *APPROX-RANDOM*, Lecture Notes in Computer Science, pages 363–378. Springer, 2013b.

Amos Beimel, Shay Moran, Kobbi Nissim, and Uri Stemmer. Private center points and learning of halfspaces. In *COLT*, Proceedings of Machine Learning Research, pages 269–282. PMLR, 2019.

Mark Bun, Kobbi Nissim, Uri Stemmer, and Salil P. Vadhan. Differentially private release and learning of threshold functions. In *FOCS*, pages 634–649, 2015.

Mark Bun, Roi Livni, and Shay Moran. An equivalence between private classification and online prediction. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 389–402, 2020.

5. Let f and g be functions of κ that are eventually non-negative. We say that $f(\kappa)$ is *superpolynomial* in $g(\kappa)$ if for every $p \in \mathbb{N}$, there exists an integer κ_0 such that for all $\kappa \geq \kappa_0$ it holds that $f(\kappa) > (g(\kappa))^p$.

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

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284, 2006.

Vitaly Feldman and David Xiao. Sample complexity bounds on differentially private learning via communication complexity. In *COLT*, JMLR Workshop and Conference Proceedings, pages 1000–1019. JMLR.org, 2014.

Badih Ghazi, Noah Golowich, Ravi Kumar, and Pasin Manurangsi. Sample-efficient proper pac learning with approximate differential privacy. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pages 183–196, 2021.

Haim Kaplan, Yishay Mansour, Uri Stemmer, and Eliad Tsfadia. Private learning of halfspaces: Simplifying the construction and reducing the sample complexity. *Advances in Neural Information Processing Systems*, 33:13976–13985, 2020.

Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM J. Comput.*, 40(3):793–826, 2011.

Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. In *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*, pages 68–77, 1987. doi: 10.1109/SFCS.1987.37.

Xin Lyu. Private learning of littlestone classes, revisited, 2025.

Kobbi Nissim, Eliad Tsfadia, and Chao Yan. Differentially private quasi-concave optimization: Bypassing the lower bound and application to geometric problems. In *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 5824–5842, 2026.

Menachem Sadigurschi and Uri Stemmer. On the sample complexity of privately learning axis-aligned rectangles. In *NeurIPS*, pages 28286–28297, 2021.

Saharon Shelah. *Classification theory: and the number of non-isomorphic models*. Elsevier, 1990.

L. G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984.

Chao Yan. An õptimal differentially private learner for concept classes with VC dimension 1. *CoRR*, abs/2505.06581, 2025.

Appendix A. Proof Sketch of Theorem 2

Proof sketch. A classical result of Shelah (1990) shows that every class C with Littlestone dimension d has a collection of $\Omega(\log d)$ 1-dimensional threshold functions embedded in it as a subclass. Thus, a lower bound on the sample complexity of privately learning 1-dimensional thresholds implies a lower bound for C. Here, we provide an informal description of a lower bound by Bun, Nissim, Stemmer,

and Vadhan (2015) that holds only for *proper* learners (i.e., learners that output a hypothesis that is itself a threshold function). As we mentioned, an unrestricted lower bound that holds also for *improper* learners was later proven by Alon et al. (2019).

Limiting ourselves to proper learners allows us to focus on the following simple problem, called the *Private Interior Point (PIP) problem*: Given n input points from a 1-dimensional grid of size d, the goal is to privately identify a point between the smallest and largest input points. This problem directly reduces to proper learning of 1-dimensional thresholds by labeling the lower half of the (sorted) inputs as 1 and the upper half as 0, thereby forcing any accurate proper learner to produce a threshold that "breaks" between the minimal and maximal inputs. Thus, it suffices to show a lower bound for the PIP problem. Towards this, observe that the PIP problem is trivially unsolvable when $n = 1$ (i.e., just 1 input point) on a grid of size (say) 10, as DP prevents the output from depending significantly on a single uniform input. We now "lift" this impossibility result inductively to larger input sizes, by showing that an impossibility result for n inputs over a grid of size d implies an impossibility result for $n + 1$ inputs over a grid of size $\exp(d)$. This shows that privately solving the interior point problem over a domain of size d requires at least $\log^*(d)$ input points.

To this end, suppose that we have a distribution $\mathcal{D}$ on n-row databases over the grid $[d]$ such that any DP algorithm fails (with high probability) to solve the interior point problem on datasets sampled from $\mathcal{D}$. Consider the following distribution $\mathcal{D}'$ on $(n+1)$-row databases over a larger grid $[d']$, where $d' = 10^d$: First, sample $(x_1, ..., x_n) \in [d]^n$ according to $\mathcal{D}$. Next, sample a uniformly random $y_0 \in [d']$. For $i = 1, \ldots, n$, let y_i be a uniformly random string that agrees with y_0 on the first x_i coordinates in base 10 notation. Output $(y_0, ..., y_n)$.

Suppose, for the sake of contradiction, that we have a DP mechanism M' on $(n+1)$-row databases over universe $[d']$. Consider the following mechanism M on n-row databases over $[d]$: on input $(x_1, .., x_n)$, sample $y_0, y_1, \ldots, y_n$ as in the distribution $\mathcal{D}'$ (given $x_1, ..., x_n$). Run M' on $(y_0, \ldots, y_n)$, and let its output be y. Output x, defined as the length of the longest prefix for which y agrees with y_0 (in base 10 notation).

Whenever M succeeds, x cannot be smaller than $\min_i\{x_i\}$, or else y would be larger than all the elements $y_1, ..., y_n$ or smaller than all of them (depending on whether y's symbol is larger or smaller than that of y_0 in the first coordinate where they disagree). Now we will use the privacy of M' to argue that x cannot be larger than $\max_i\{x_i\}$. To see this, note that excluding y_0 itself, all the other inputs given to M' do not contain any information about the $\max_i\{x_i\} + 1$ coordinate of y_0. Hence, as M' is DP, even with y_0 as one of its inputs, the probability of it guessing this coordinate is $\lesssim 1/10$. (The actual construction leverages a larger base in order to keep this probability sufficiently small.) So $\min_i\{x_i\} \leq x \leq \max_i\{x_i\}$ is a valid solution for $\mathcal{D}$, contradicting its hardness. ■

Appendix B. Proof Sketch of Theorem 3

Proof sketch. We give an informal description of the construction of Bun, Livni, and Moran (2020) which provides an upper bound of $\approx 2^{2^{\mathrm{LD}(C)}}$. Let C be a class with $\mathrm{LD}(C) = d$, which implies the existence of an online learner $\mathcal{A}_{\text{online}}$ that makes at most d mistakes on any realizable sequence. We construct a DP learner for C by first transforming $\mathcal{A}_{\text{online}}$ into a (non-private) *globally-stable* learner, meaning a learner that outputs the *exact same* hypothesis with noticeable probability even when its training dataset is completely resampled. (Such a learner is easily made private via standard techniques, such as running it on disjoint samples and privately outputting the most frequent result.)

As $\mathcal{A}_{\text{online}}$ makes at most d mistakes, there must exist an index $i^* \in \{0, 1, 2, \ldots, d\}$ such that the probability of it making *exactly* i^* mistakes on a fresh sample is $\gtrsim 1/d$. Consider sampling $k = 2^{d-i^*}$ independent samples $\mathcal{S} = (S_1, S_2, \ldots, S_k)$. Then, with probability $\gtrsim (1/d)^k \geq (1/d)^{2^d}$ algorithm $\mathcal{A}_{\text{online}}$ makes exactly i^* mistakes on each of these samples. Suppose that this is indeed the case. We may assume that there are not too many duplicates among the hypotheses $h_1 = \mathcal{A}_{\text{online}}(S_1), \ldots, h_k = \mathcal{A}_{\text{online}}(S_k)$, as otherwise $\mathcal{A}_{\text{online}}$ itself would be globally-stable and our job would be complete. (Here we treat $\mathcal{A}_{\text{online}}$ as a batch learner by letting $\mathcal{A}_{\text{online}}(S)$ denote its final hypothesis after processing the sequence S.) For this simplified proof sketch, let us further assume that these hypotheses are in fact all distinct.

We next combine pairs of samples from $\mathcal{S}$ to generate input sequences on which $\mathcal{A}_{\text{online}}$ makes $i^* + 1$ mistakes. For concreteness, let us illustrate this on S_1 and S_2. To this end, let x be a point such that $h_1(x) \neq h_2(x)$ (such a point exists as we assumed that $h_1 \neq h_2$). Now observe that each of the input sequences $S_1 \circ (x, h_2(x))$ and $S_2 \circ (x, h_1(x))$ would cause $\mathcal{A}_{\text{online}}$ to make exactly $i^* + 1$ mistakes. Furthermore, one of these two sequences is realizable, and we can "guess" which one it is with probability 1/2. This way, by pairing samples from $\mathcal{S}$ (and guessing), with probability $\geq (1/2)^{k/2}$ we generate $k/2$ realizable input sequences, each forcing exactly $i^* + 1$ mistakes.

The process continues by an inductive argument: Either there is a "stable" hypothesis among the newly obtained $k/2$ hypotheses, or else we randomly generate realizable sequences that force $i^* + 2$ mistakes, and so on. Observe that if we reach sequences that force d mistakes, meaning that no additional mistake could ever be made, then these sequences must produce the actual target concept, and global-stability is obtained. Overall, this construction guarantees that a stable hypothesis is identified with probability $\gtrsim (1/d)^{2^d} \cdot (1/2)^{2^d}$, which results in a DP learner with sample complexity roughly 2^{2^d}. This construction was later improved by Ghazi, Golowich, Kumar, and Manurangsi (2021) and by Lyu (2025) to obtain a DP learner with sample complexity $\text{poly}(d)$.

■