Private Prediction via Shrinkage

Chao Yan*
Technion

Abstract

We study *differentially private prediction* introduced by Dwork and Feldman [10] (COLT 2018): an algorithm receives one labeled sample set S and then answers a stream of unlabeled queries while the output transcript remains (ε, δ)-differentially private with respect to S. Standard composition yields a $\sqrt{T}$ dependence for T queries, i.e. $O(\mathrm{VC}(\mathcal{C}) \cdot \sqrt{T})$ sample complexity.

We show that this dependence can be reduced to *polylogarithmic* in T in streaming settings. For an oblivious online adversary and any concept class $\mathcal{C}$, we give a private predictor that answers T queries with $|S| = \tilde{O}(VC(\mathcal{C})^{3.5} \log^{3.5} T)$ labeled examples. For an adaptive online adversary and halfspaces over $\mathbb{R}^d$, we obtain $|S| = \tilde{O}\left(d^{5.5} \log T\right)$.

1 Introduction

In the PAC learning model [23], a learner receives a labeled sample $S = \{(x_i, y_i)\}_{i=1}^N$, where each x_i is drawn i.i.d. from an unknown distribution $\mathcal{D}$ over a domain X, and each label is generated by an unknown target concept $c \in \mathcal{C}$, where $\mathcal{C} \subseteq \{c : X \to \{\pm 1\}\}$. The learner outputs a hypothesis h intended to predict the labels of fresh unlabeled points. It is well known that the sample complexity of PAC learning is characterized (up to constant factors) by the VC dimension $VC(\mathcal{C})$.

In many applications, however, the sample S contains sensitive information, and the released hypothesis h may leak details about individual examples. Kasiviswanathan et al. [17] introduced private learning, which requires the learner to satisfy differential privacy with respect to the input sample S. They gave a generic differentially private learner with sample complexity $O(\log |\mathcal{C}|)$. Since $VC(\mathcal{C}) \leq \log |\mathcal{C}|$, this implies privacy can incur a potentially large overhead: the gap between $VC(\mathcal{C})$ and $\log |\mathcal{C}|$ can be arbitrarily large.

By now, it is well understood that private learning can require strictly more samples than non-private PAC learning. One example is the threshold class over an ordered domain X: $\mathcal{C} = \{c_t : c_t(x) = 1 \text{ if } x \geq t,\ t \in [X]\}$, which has $VC(\mathcal{C}) = 1$, and can be PAC learned from $O(1)$ examples. In contrast, under pure differential privacy, learning thresholds requires $\Theta(\log |X|)$ samples [13], and under approximate differential privacy it requires $\Theta(\log^* |X|)$ samples [3, 6, 15, 1, 8, 25]. In particular, for infinite domains, thresholds are PAC learnable but not privately learnable.

Dwork and Feldman [10] proposed an alternative paradigm for privacy-preserving prediction. Instead of outputting a hypothesis h, the algorithm interacts with a stream of unlabeled queries x and outputs a label only when a query arrives. For a single query, they showed that $|S| = \Theta(VC(\mathcal{C}))$ labeled examples suffice. For answering T queries, the composition theorem yields $|S| = O(VC(\mathcal{C}) \cdot \sqrt{T})$ [12]. (also observed by Bassily et al. [2])

A central question is whether one can improve the dependence on T. For an **offline** adversary, who reveals all queries in advance, Beimel et al. [4] showed that $O(VC(\mathcal{C}) \log T)$ labeled examples

*Department of Mathematics, Technion. cy399@georgetown.edu. Funded by the European Union (ERC, GENERALIZATION, 101039692). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. Research is supported in part by a gift from Georgetown University.

suffice to privately label T unlabeled points. For the **stochastic online** setting, where queries are drawn i.i.d. from the same distribution $\mathcal{D}$, Nandi and Bassily [18] obtained bounds independent of T via semi-supervised ideas, giving sample complexity $O((VC(\mathcal{C}))^{1.5})$. Naor et al. [19] further showed that one can even protect the privacy of the queries themselves with sample complexity $\tilde{O}((VC(\mathcal{C}))^2)$ (also see [22] for improved sample complexity on a specific concept class).

These latter results, however, rely crucially on the assumption that online queries are drawn from the underlying distribution $\mathcal{D}$. It remains unclear whether one can beat the $\sqrt{T}$ dependence when the queries are adversarially chosen in an online (streaming) manner. This motivates the following question:

> Can we privately answer a super-polynomial number of adversarially chosen queries in the online (streaming) setting?

1.1 Prediction Model

Here we formalize the prediction model. We follow the definition by Naor et al. [19][2] The learner never releases a single final classifier. Instead, it must answer each query as it arrives, while the transcript remains private and each round's predictor stays accurate under $\mathcal{D}$.

Private Prediction.

Let $\mathcal{C} \subseteq \{\pm 1\}^X$ be a concept class over a domain X. An unknown target concept $c \in \mathcal{C}$ and an unknown distribution $\mathcal{D}$ over X generate a labeled sample

$$S = \{(x_i, y_i)\}_{i=1}^N \in (X \times \{\pm 1\})^N, \qquad x_i \overset{i.i.d.}{\sim} D, \;\; y_i = c(x_i).$$

The interaction then proceeds for T prediction rounds. In round $j \in [T]$:

1. The algorithm $\mathcal{A}$ (possibly using internal randomness and the sample S) computes a hypothesis $h_j : X \to \{\pm 1\}$.
2. $\mathcal{A}$ receives a query point $x_j \in X$ (chosen by an adversary, possibly adaptively).
3. $\mathcal{A}$ outputs the predicted label $\mathsf{Label}_j := h_j(x_j)$.

Definition 1. *We say that $\mathcal{A}$ is an $(\varepsilon, \delta, \alpha, \beta)$-differentially private predictor for $\mathcal{C}$ over X if it satisfies:*

1. ***Privacy.*** *The transcript of outputs $((x_1, \mathsf{Label}_1), \ldots, (x_T, \mathsf{Label}_T))$ is (ε, δ)-differentially private with respect to the sample S.*
2. ***Accuracy.*** *With probability at least $1 - \beta$ over the randomness of $\mathcal{A}$ and the draw of S,*

$$\mathrm{err}_{\mathcal{D}}(c, h_j) \le \alpha \quad \textit{for all } j \in [T],$$

where $\mathrm{err}_{\mathcal{D}}(c, h) := \Pr_{x \sim \mathcal{D}}[h(x) \neq c(x)]$.

That is, the predictor may change from round to round, but at every round we require the current predictor to have a small error on fresh samples from D.

1.2 Adversary models.

We consider several standard models for how the query sequence $x_1, \ldots, x_T$ is generated.

- **Offline adversary.** The adversary selects the entire query sequence $(x_1, \ldots, x_T)$ in advance and reveals it to the predictor before any interaction takes place.
- **Oblivious online adversary.** The adversary selects the entire sequence $(x_1, \ldots, x_T)$ in advance, but reveals the queries one at a time. Equivalently, x_j may depend on public parameters (and on j), but not on the predictor's internal randomness or prior outputs.

[2] In the setting of [19], the predictor predicts an infinite number of queries and requires protecting the privacy of queries. We limit it to be T and only protect the privacy of the dataset.

Adversary model	Labeled sample size	Reference
Online prediction (general), T queries	$O(\mathrm{VC}(\mathcal{C})\sqrt{T})$	[10, 2]
Offline adversary (queries revealed in advance)	$O(\mathrm{VC}(\mathcal{C})\log T)$	[4]
Online stochastic adversary (queries are sampled from $\mathcal{D}$)	$O(\mathrm{VC}(\mathcal{C})^{1.5})$ (indep. of T)	[18]
Online stochastic + query privacy	$\widetilde{O}(\mathrm{VC}(\mathcal{C})^2)$ (indep. of T)	[19]
Oblivious online adversary	$\widetilde{O}(\mathrm{VC}(\mathcal{C})^{3.5}\log^{3.5} T)$	this work
Adaptive online adversary, halfspaces over $\mathbb{R}^d$	$\widetilde{O}(d^{5.5}\log T)$	this work

Table 1: Comparison of labeled-sample complexity.

- **Stochastic online adversary.** Each query x_j is drawn i.i.d. from the same underlying distribution $\mathcal{D}$ that generates the labeled sample S (and is independent of the predictor's randomness). This is sometimes called the *online i.i.d.* query model.
- **Adaptive online adversary.** The adversary chooses queries sequentially, where x_j may depend on the predictor's prior outputs. This is the strongest (streaming) adversary model considered in this work.

Remark. Here "online" refers only to the sequential arrival of the queries $x_1, \ldots, x_T$: the learner must answer each query before seeing the next one. Our accuracy guarantee is *not* a mistake or regret bound; rather, following the private prediction model, we require that each round's hypothesis h_j have small population error under the underlying distribution D.

1.3 Our Results

In this work, we show two main results:

1. **Oblivious online adversary.** We give a differentially private predictor that can answer T prediction queries using $\tilde{O}((VC(\mathcal{C})^{3.5} \cdot \log^{3.5} T)$ labeled samples.
2. **Adaptive online adversary for halfspaces.** For an adaptive online adversary and the class of halfspaces over $\mathbb{R}^d$, we give a differentially private predictor that answers T queries using $\tilde{O}(d^{5.5} \cdot \log T)$ labeled samples.

For comparison, we list the previous results in Table 1.

1.4 Related Work

Our work is also related to the literature on interactive query answering under differential privacy. Roth and Roughgarden [21] introduced the median mechanism, showing that in interactive private query release one can exploit correlations among online predicate queries by identifying a small number of "hard" queries and answering the remaining "easy" ones implicitly. Their guarantees are stated for predicate queries over a finite domain and depend explicitly on the domain size.

Bun, Steinke, and Ullman [7] study threshold queries in the interactive query-answering model under differential privacy, and introduce the `BetweenThresholds` primitive for adaptively chosen low-sensitivity queries. Since threshold queries correspond to the one-dimensional halfspace case, their work provides a natural one-dimensional precursor to our adaptive result. Our adaptive theorem can be viewed as extending this perspective to private prediction for halfspaces in $\mathbb{R}^d$, where the main new ingredient is a geometric shrinkage argument based on the linear-feasibility / cdepth framework.

Unlike these query-answering works, our setting is private prediction from a labeled sample, with guarantees stated in terms of labeled sample complexity and population error.

1.5 The main ideas

1. **Ideas from previous works.** We use the framework by Dwork and Feldman [10] and Naor et al. [19]. More details can be found in Section 2.
   (a) **Subsample-and-aggregation** We follow the standard subsample-and-aggregate framework appearing in a line of works [10, 2, 18, 19]. We randomly partition the labeled dataset into many disjoint blocks, each of size $O(VC(\mathcal{C}))$. On each block,

we run a non-private PAC learner to obtain a hypothesis. Given a query point x, we predict using a noisy majority vote over the ensemble of hypotheses.

(b) **Sparse vector** To answer exponentially many queries while preserving privacy, we use the Sparse Vector technique. Specifically, we use the algorithm `BetweenThresholds` introduced by Bun et al. [7]. Informally, `BetweenThresholds` allows us to perform a large number of noisy majority votes and only pays privacy when a query is hard (i.e., when approximately half of the hypotheses vote for label 1). A limitation of prior analyses is that the number of hard queries can be large. We solve it by a shrinkage approach.

2. **Shrinking the concept class to bound the number of hard queries.** The main new ingredient is a shrinkage argument controlling the number of hard queries. The key idea is to ensure that the sparse vector (in this work, we use `BetweenThresholds` algorithm) stops only $O(\log T)$ times by progressively shrinking the set of candidate hypotheses. Each time `BetweenThresholds` declares a query hard, we update (shrink) the hypothesis space so that the number of remaining label patterns decreases.

   (a) **Oblivious adversary (adversarially selected queries but fixed in advance).** Suppose the query points $x_1, \ldots, x_T$ are fixed in advance. Consider the set of possible label sequences $\mathcal{L} = \big\{(h(x_1), \ldots, h(x_T)) : h \in \mathcal{C}\big\}$. By Sauer's lemma, $|\mathcal{L}| \leq O(T^{\mathrm{VC}(\mathcal{C})})$. For each hard query x_t, we guess a label $\hat{y}_t \in \{\pm 1\}$ uniformly at random and then remove all hypotheses h inconsistent with the guess, i.e., those with $h(x_t) = -\hat{y}_t$. With probability $1/2$, this removes at least half of the remaining label sequences in $\mathcal{L}$. Therefore, after $O(\mathrm{VC}(\mathcal{C}) \cdot \log T)$ hard queries, the remaining hypotheses agree on all future queries, and no query will be hard anymore.

   (b) **Halfspaces under an adaptive adversary.** For halfspaces we leverage geometric techniques of Nissim et al. [20] and Kaplan et al. [16]. We cast prediction as maintaining the feasibility of a system of linear constraints corresponding to the target halfspace. Each hard query can be interpreted as a hyperplane constraint. By restricting attention to the intersection structure induced by hard queries, we show that after $O(d)$ such events, the remaining feasible region collapses to a single consistent solution, after which all future queries are easy, and no query will be hard anymore.

1.6 Discussion and Open Questions

Our results show that, for fixed privacy and accuracy parameters, one can privately answer super-polynomially many queries even when the queries arrive in an adversarial stream. We highlight several directions for future work.

1. **Improving the dependence on $\mathrm{VC}(\mathcal{C})$ for oblivious adversaries.** Compared to the baseline $O(\mathrm{VC}(\mathcal{C})\sqrt{T})$ bound obtained via composition, our guarantees achieve only polylogarithmic dependence on T but incur a larger polynomial dependence on $\mathrm{VC}(\mathcal{C})$. Can one obtain a bound of the form $\widetilde{O}(\mathrm{VC}(\mathcal{C}) \cdot \mathrm{poly}(\log T))$? Besides being a natural target in its own right, such an improvement would also sharpen bounds in the stochastic setting: combined with techniques of semi-supervised learning [4, 18], it would potentially reduce the labeled-sample complexity from $O(\mathrm{VC}(\mathcal{C})^{1.5})$ to $\widetilde{O}(\mathrm{VC}(\mathcal{C}))$.
2. **Adaptive adversaries beyond halfspaces.** Our adaptive-adversary result relies on geometric structure specific to halfspaces. Can one obtain polylogarithmic dependence on T for *arbitrary* VC classes against adaptive online adversaries?
3. **Query privacy.** In our model, the predictor's future behavior depends on past queries and outputs, and we only guarantee differential privacy with respect to the labeled sample S. Is it possible to additionally protect the privacy of the query stream itself (as in [19]), while still achieving polylogarithmic dependence on T against adversarial streams?
4. **Lower bounds.** Cohen et al. [9] prove an $\Omega(\min\{n, \log T\})$ lower bound in the continual-observation setting, and show that it extends to the online thresholds problem, yielding a lower bound that is tight with respect to the dependence on T. This provides strong evidence that logarithmic dependence on T is the right benchmark in closely related threshold settings. Via standard connections between online threshold queries and threshold-based

private prediction, this also suggests an $\Omega(\log T)$ barrier for threshold functions (equivalently, one-dimensional halfspaces) against adaptive adversaries. To the best of our knowledge, however, no matching lower bound is currently known for the oblivious-adversary case in our model.

2 A Generic Framework

In this section, we present a generic prediction framework, building on the approaches of Dwork and Feldman [10] and Naor et al. [19]. The core idea is to randomly partition the labeled dataset into several disjoint subsets, learn a hypothesis non-privately from each subset, and then answer each query by aggregating the resulting hypotheses. Concretely, given a query point, we predict its label using a noisy majority vote over the ensemble.

As observed by Naor et al. [19], the aggregation step can be implemented using the sparse vector technique, which allows us to answer many queries while spending privacy budget only on a limited number of 'hard' rounds. Consequently, both the privacy and accuracy guarantees of the framework are governed by the number of times the sparse vector subroutine triggers i.e., its stopping behavior.

Algorithm 1: GenericPredictor

Inputs: A labeled database $S \in (X \times \{\pm 1\})^N$ where we set $k = \frac{64}{\varepsilon}\left(\log(T+1) + \log(1/\beta)\right)$ and $N = k \cdot m$

1. Randomly partition S to $S_1, \ldots, S_k$, each of size $|S_i| = m$.
2. For $j = 1, \ldots, T$:
   (a) For $i = 1, \ldots, k$, compute a hypothesis $G(S_i, transcript_j) = f_{i,j} \in \mathcal{C}$. Let $F_j = \{f_{1,j}, \ldots, f_{k,j}\}$
   /* $transcript_j$ includes all prediction queries and predicted labels before step j
   We view F_j as a dataset with size k, change one example of S will change one entry of F_j, so the query sensitivity is $1/k$ */
   (b) Receive a prediction query point $x_j \in X$.
   (c) Set parameter $t_\ell = 1/2 - 1/8$ and $t_u = 1/2 + 1/8$. Define the counting query $q_{x_j}(F_j) = \frac{1}{2}\left(1 + \frac{1}{k}\sum_{i=1}^{k} f_{i,j}(x_j)\right)$, and run `BetweenThresholds` on q_{x_j} to obtain an output $y_j \in \{L, \top, R\}$.
   (d) If $y_j = L$, respond with the label $\mathsf{Label}_j = -1$.
   (e) If $y_j = R$, respond with the label $\mathsf{Label}_j = 1$.
   (f) If $y_j = \top$, uniformly randomly respond with the label $\mathsf{Label}_j \in \{\pm 1\}$.
   /* When `BetweenThresholds` outputs $\top$, we reinitialize it with fresh noise and continue.*/

The following observation shows that both privacy and accuracy depend only on the number of times `BetweenThresholds` outputs $\top$. We defer the proof to the Appendix C.1.

Observation 1 ([19])**.** *Let v denote the number of times that `BetweenThresholds` in Step 2c outputs $\top$. Then Algorithm 1 is equivalent to v composition of `BetweenThresholds`.*

Claim 1 (Privacy of Generic Predictor [19])**.** *If `BetweenThresholds` in Step 2c outputs $\top$ at most v times, then Algorithm 1 is $\left(O(\sqrt{v \cdot \ln(1/\delta)} \cdot \varepsilon), O(v\delta)\right)$-differentially private.*

Claim 2 (Accuracy of Generic Predictor [19])**.** *If all $f_{i,j}$ are α-good with respect to the underlying distribution $\mathcal{D}$, then with probability $1 - v\beta$, all predictors defined above are 4α-good with respect to the distribution $\mathcal{D}$.*

3 Private Prediction with Oblivious Adversary

In this section, we construct a hypothesis generator for an oblivious adversary. The key idea is that whenever `BetweenThresholds` outputs $\top$, we record this query and use it to shrink the cur-

rent version space. In particular, each hard query induces a random constraint that (with constant probability) eliminates at least half of the remaining feasible labelings of the fixed query sequence.

Since the adversary is oblivious, the queries $x_1, \ldots, x_T$ are fixed in advance, and the number of distinct label sequences realizable by $\mathcal{C}$ on these points is at most $O(T^{VC(\mathcal{C})})$ by Sauer's lemma. Therefore, after at most $O(VC(\mathcal{C}) \log T)$ hard queries, only one labeling remains and all remaining hypotheses agree on future queries. Consequently, the number of times `BetweenThresholds` outputs $\top$ is bounded by $O(\mathrm{VC}(\mathcal{C}) \log T)$ (see illustration in Figure 1 and 2).

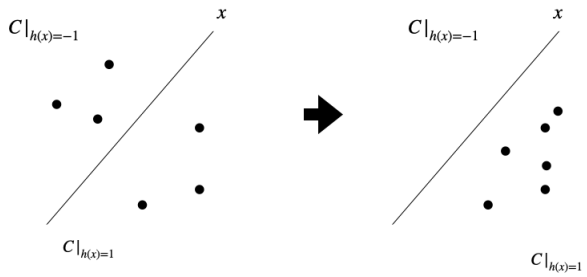


Figure 1: In the left figure, we illustrate that when a "hard" query x occurs, the current hypothesis set splits into two subspaces, $\mathcal{C}|_{h(x)=1}$ and $\mathcal{C}|_{h(x)=-1}$. We then guess a label uniformly at random (say, 1) and update the hypothesis set by restricting to $\mathcal{C}|_{h(x)=1}$, as shown in the right figure.

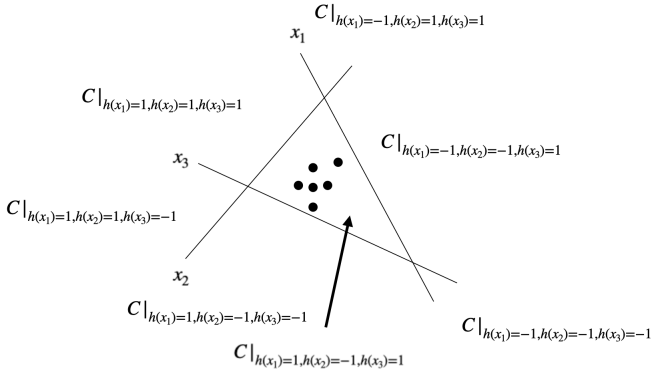


Figure 2: After answering $O(VC(\mathcal{C}) \log T)$ "hard" queries, the remaining hypothesis space collapses to a set of hypotheses that induce the same labeling on the entire query sequence $x_1, \ldots, x_T$ (including queries that have not yet appeared in the stream).

3.1 Hypotheses Generator for Oblivious Adversary

We use the generic framework of Section 2 and design a hypothesis generator for oblivious adversaries.

Algorithm 2: $G_{oblivious}$: Hypotheses Generator for Oblivious Adversary

Inputs: A labeled database S_i,
$transcript_j = ((x_1, y_1, \mathsf{Label}_1), \ldots, (x_{j-1}, y_{j-1}, \mathsf{Label}_{j-1}))$

1. Let $HardQuery = \{(x, y, \mathsf{Label}) \in transcript_j | y = \top\}$. Let $\mathcal{C}|_{HardQuery} = \{c \in \mathcal{C} : c(x_{j^*}) = \mathsf{Label}_{j^*} \text{ for all } (x_{j^*}, y_{j^*}, \mathsf{Label}_{j^*}) \in HardQuery\}$
2. Output a hypothesis $f_{i,j} = \arg\min_{h \in \mathcal{C}|_{HardQuery}} [\mathrm{err}_{S_i}(h)]$.

3.2 Analysis

Here we recall that we partition the dataset S into k subsets $S_1, \ldots, S_k$, each with size $|S_i| = m$.

Claim 3. *If $m \geq O\left(\frac{d \cdot \log \frac{d}{\alpha} + \log \frac{1}{\beta}}{\alpha^2}\right)$, with probability $1 - k\beta$, we have*

1. $|\mathrm{err}_S(h) - \mathrm{err}_{S_i}(h)| \leq \alpha$ *for all* $i \in [k]$ *and all* $h \in \mathcal{C}$.
2. $\left|\mathrm{err}_{S_j}(h) - \mathrm{err}_{S_i}(h)\right| \leq 2\alpha$ *for all* $i, j \in [k]$ *and all* $h \in \mathcal{C}$.

Proof. By Theorem 2, with probability $1 - \beta$, each S_i is an α-approximation of S with respect to the concept class $\mathcal{C}$, and thus $|\mathrm{err}_S(h) - \mathrm{err}_{S_i}(h)| \leq \alpha$. The first statement can follow by union bound. The second statement is the corollary of the first statement. □

Claim 4. *Assume* `BetweenThresholds` *satisfies the accuracy guarantee of Lemma 2. Then $HardQuery$ is a set of realizable labeled points. That is, there exists a hypothesis $h \in \mathcal{C}$ satisfying $h(x_i) = \mathsf{Label}_i$ for all pairs $(x_i, \top, \mathsf{Label}_i) \in HardQuery$.*

Proof. We prove it by induction on $|HardQuery|$. In the base case of $|HardQuery| = 0$, $HardQuery$ is an empty set, and the result is trivial.

For the inductive step, suppose the claim holds for all transcripts with at most $a - 1$ hard queries. Consider a transcript with a hard queries, and let $(x_{a^*}, \mathsf{Label}_{a^*})$ be the most recent one. At round a^*, `BetweenThresholds` output $\top$ in Step 2c, which means the ensemble vote $q_{x_{a^*}} = \frac{1}{2}(1 + \frac{1}{k}\sum_{i=1}^{k} f_{i,a^*}(x_{a^*}))$ lies in the uncertain region, so in particular there exists at least one hypothesis f_{i,a^*} with $f_{i,a^*}(x_{a^*}) = \mathsf{Label}_{a^*}$. By construction, every f_{i,a^*} is chosen from $\mathcal{C}|_{HardQuery \setminus (x_{a^*}, \top, \mathsf{Label}_{a^*})}$, which is nonempty by the induction hypothesis. Hence, there exists a concept in $\mathcal{C}$ consistent with all previous hard queries and also labeling x_{a^*} as Label_{a^*}, as required. □

Claim 5. *Assume* `BetweenThresholds` *satisfies Lemma 2. Then with probability $1 - \beta$,* `BetweenThresholds` *outputs $\top$ at most $O(VC(\mathcal{C}) \cdot \log T + \log(1/\beta))$ times.*

Proof. Since all queries are selected in advance, we can consider all possible labels of $\mathcal{L} = \{h(x_1), \ldots, h(x_T)\}_{h \in \mathcal{C}}$. By Sauer's lemma, we know $|\mathcal{L}| \leq O(T^{VC(\mathcal{C})})$.

Each time `BetweenThresholds` outputs $\top$, the algorithm outputs a uniformly random label. Conditioned on the history, with probability $1/2$ this random label matches the target label sequence, and the restriction $\mathcal{C}|_{\mathsf{HardQuery}}$ eliminates at least half of the remaining label sequences in $\mathcal{L}$. Let E be the event that a given hard query halves $|\mathcal{L}|$, then $\Pr[E] \geq 1/2$.

Therefore, after $O(\mathrm{VC}(\mathcal{C}) \log T)$ successful halving events, the set of remaining label sequences has size 1. On the other hand, when the number of hard queries is $O(VC(\mathcal{C}) \cdot \log T + \log(1/\beta))$, the Chernoff bound implies that with probability at least $1 - \beta$, the event E happens at least $O(\mathrm{VC}(\mathcal{C}) \log T)$ times.

At the end, once $|\mathcal{L}| = 1$, all remaining hypotheses agree on all future queries, all $f_{i,j}$ predict the same label for new x_j. under the guarantee of Lemma 2, `BetweenThresholds` will not output $\top$ anymore. □

Let $v = O(VC(\mathcal{C}) \cdot \log T + \log(1/\beta))$, we have the privacy guarantee of the predictor with an oblivious adversary.

Corollary 1 (Privacy Guarantee)**.** *Using $G_{oblivious}$, Algorithm 1 is $(O(\sqrt{v \cdot \ln(1/\delta)} \cdot \varepsilon), O(v \cdot \delta))$-differentially private, where $v = O(VC(\mathcal{C}) \cdot \log T + \log(1/\beta))$.*

Claim 6. *Under the guarantee of Claim 3 and Claim 5, all $f_{i,j}$ satisfies $\mathrm{err}_{\mathcal{D}}(f_{i,j}) \leq O(v \cdot \alpha)$, where $v = O(VC(\mathcal{C}) \cdot \log T + \log(1/\beta))$.*

Proof. By Theorem 3, it suffices to prove $\mathrm{err}_S(f_{i,j}) \leq O(v \cdot \alpha)$ for all $f_{i,j}$. Claim 5 implies that $|HardQuery| \leq v$.

We prove it by induction. In the base case of $|HardQuery| = 0$, all $f_{i,j}$ are selected from $\mathcal{C}$ and $\mathrm{err}_{S_i}(f_{i,j}) = 0$

Assume when $|HardQuery| \leq c-1$, we have $\min_{h \in \mathcal{C}|_{HardQuery}}[\mathrm{err}_{S_i}(h)] \leq 2(c-1)\alpha$. For the case of $|HardQuery| = c$, let $(x_{c^*}, \top, \mathsf{Label}_{c^*})$ be the c-th element of $HardQuery$. On the c^*-th query, `BetweenThresholds` output $\top$ in Step 2c. It happens when there exists $f_{i^*,c^*}(x_{c^*}) = \mathsf{Label}_{c^*}$ in query $q_{x_{c^*}}$ for some $i^* \in [k]$. Since f_{i^*,c^*} is selected from $\mathcal{C}|_{HardQuery \setminus (x_{c^*}, \top, \mathsf{Label}_{c^*})}$, By the induction assumption, we have $\min_{h \in \mathcal{C}|_{HardQuery}}[\mathrm{err}_{S_{i^*}}(h)] \leq 2(c-1)\alpha$. By the selection strategy, we have $\mathrm{err}_{S_{i^*}}(f_{i^*,c^*}) \leq 2(c-1)\alpha$. It also works for $f_{i^*,j}$, where $j \geq c^*$ and conditioned on $|HardQuery| = c$ because $HardQuery$ doesn't change. Thus by the second statement of Claim 3, we have $\mathrm{err}_{S_{i^*}}(f_{i,x_j}) \leq 2c\alpha$ for all $i \in [k]$. At the end, this claim can result from $c \leq v$ and the first statement of Claim 3.

□

Using Claim 2, we have the following corollary.

Corollary 2 (Accuracy Guarantee)**.** *Using $G_{oblivious}$, with probability $1 - O((v+k)\cdot\beta)$, every query x_j is predicted by a predictor h_j satisfying $\mathrm{err}_D(c, h_j) \leq O(v \cdot \alpha)$*

Select the appropriate parameters, and we can have the main result. We put the details in Appendix C.2

Theorem 1. *For any concept class $\mathcal{C}$ with VC dimension d, there exists an (ε, δ)-differentially private predictor that can (α, β)-predict T queries if the given dataset has size*

$$N \quad = \tilde{O}_{\alpha,\beta,\varepsilon,\delta}\left(d^{3.5} \cdot \log^{3.5} T\right)$$

4 Prediction for Halfspaces with Adaptive Adversary

In this section, we construct a hypothesis generator specialized to the halfspace class under an adaptive adversary. Our construction is based on the reduction from PAC learning halfspaces to linear feasibility problem and the corresponding definition $cdepth$, which are introduced by Kaplan et al. [16]. As in the generic framework, whenever `BetweenThresholds` outputs $\top$ we treat the corresponding query as a hard query and record it. The crucial difference is that, for halfspaces, each hard query can be interpreted as a geometric constraint in an associated linear-feasibility representation: the hard query identifies a hyperplane, and we restrict attention to the intersection of the approximate feasible region with this hyperplane.

By repeatedly intersecting with the hyperplanes induced by hard queries, we progressively collapse the approximate feasible region. In $\mathbb{R}^{d+1}$, after at most $d+1$ such independent intersections, the feasible set reduces to a single candidate, implying that all remaining hypotheses agree on every future query. Consequently, BETWEENTHRESHOLDS can output $\top$ only $d+1$ times.

To make the argument more concrete, we record the main invariant maintained by the adaptive-halfspace construction. Let

$$P_j = \{(a_1, \ldots, a_d, w) \in \mathbb{R}^{d+1} : \langle a, x_{j^*} \rangle = w \text{ for every } (x_{j^*}, \top, \mathrm{Label}_{j^*}) \in \mathrm{HardQuery}\}$$

denote the current feasible region after round j. The proof maintains that: (i) P_j is always nonempty; (ii) for every block S_i, the region P_j contains a point of large $cdepth_{S_i}$, so the hypothesis selected by $G_{\mathrm{halfspace}}$ has low empirical error on S_i; and (iii) every new hard query induces a nonredundant hyperplane constraint, so intersecting with this hyperplane strictly decreases the affine dimension of P_j. Since $P_j \subseteq \mathbb{R}^{d+1}$, this can happen at most $d+1$ times. Once the affine dimension drops to 0, all remaining feasible hypotheses agree on every future query, and `BetweenThresholds` can no longer output $\top$.

The full proofs of the nonemptiness, high-$cdepth$, and dimension-drop claims are given in Appendix B.

References

[1] Noga Alon, Roi Livni, Maryanthe Malliaris, and Shay Moran. Private PAC learning implies finite littlestone dimension. In Moses Charikar and Edith Cohen, editors, *Proceedings of the*

51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019, pages 852–860. ACM, 2019.

[2] Raef Bassily, Abhradeep Guha Thakurta, and Om Dipakbhai Thakkar. Model-agnostic private learning. In *NeurIPS*, pages 7102–7112, 2018.

[3] Amos Beimel, Kobbi Nissim, and Uri Stemmer. Private learning and sanitization: Pure vs. approximate differential privacy. In *APPROX-RANDOM*, pages 363–378, 2013.

[4] Amos Beimel, Kobbi Nissim, and Uri Stemmer. Learning privately with labeled and unlabeled examples. In *SODA*, pages 461–477. SIAM, 2015.

[5] Anselm Blumer, A. Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *J. ACM*, 36(4):929–965, October 1989.

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

[7] Mark Bun, Thomas Steinke, and Jonathan Ullman. Make up your mind: The price of online queries in differential privacy. *CoRR*, abs/1604.04618, 2016.

[8] 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.

[9] Edith Cohen, Xin Lyu, Jelani Nelson, Tamás Sarlós, and Uri Stemmer. Lower bounds for differential privacy under continual observation and online threshold queries. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 1200–1222. PMLR, 2024.

[10] Cynthia Dwork and Vitaly Feldman. Privacy-preserving prediction. In Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet, editors, *Conference On Learning Theory, COLT 2018, Stockholm, Sweden, 6-9 July 2018*, volume 75 of *Proceedings of Machine Learning Research*, pages 1693–1702. PMLR, 2018.

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

[12] Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. Boosting and differential privacy. In *FOCS*, pages 51–60, 2010.

[13] Vitaly Feldman and David Xiao. Sample complexity bounds on differentially private learning via communication complexity. *SIAM J. Comput.*, 44(6):1740–1764, 2015.

[14] David Haussler and Emo Welzl. Epsilon-nets and simplex range queries. In *SCG '86*, 1986.

[15] Haim Kaplan, Katrina Ligett, Yishay Mansour, Moni Naor, and Uri Stemmer. Privately learning thresholds: Closing the exponential gap. In *COLT*, pages 2263–2285, 2020.

[16] 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.

[17] 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.

[18] Anupama Nandi and Raef Bassily. Privately answering classification queries in the agnostic PAC model. In *ALT*, volume 117 of *Proceedings of Machine Learning Research*, pages 687–703. PMLR, 2020.

[19] Moni Naor, Kobbi Nissim, Uri Stemmer, and Chao Yan. Private everlasting prediction. In A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 54785–54804. Curran Associates, Inc., 2023.

[20] Kobbi Nissim, Eliad Tsfadia, and Chao Yan. Differentially private quasi-concave optimization: Bypassing the lower bound and application to geometric problems. *arXiv preprint arXiv:2504.19001*, 2025.

[21] Aaron Roth and Tim Roughgarden. Interactive privacy via the median mechanism. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 765–774, New York, NY, USA, 2010. ACM.

[22] Uri Stemmer. Private truly-everlasting robust-prediction. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

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

[24] Vladimir N. Vapnik and Alexey Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16(2):264–280, 1971.

[25] Chao Yan. An õptimal differentially private learner for concept classes with vc dimension 1. *CoRR*, 2025.

A Preliminaries

A.1 Learning Theory

Definition 2 (Vapnik-Chervonenkis dimension [24, 14])**.** *Let X be a set and R be a set of subsets of X. Let $S \subseteq X$ be a subset of X. Define $\Pi_R(S) = \{S \cap r \mid r \in R\}$. If $|\Pi_R(S)| = 2^{|S|}$, then we say S is shattered by R. The Vapnik-Chervonenkis dimension of (X, R) is the largest integer d such that there exists a subset S of X with size d that is shattered by R.*

Definition 3 (α-approximation[3] [24, 14])**.** *Let X be a set and R be a set of subsets of X. Let $S \in X^N$ be a finite multi-set of elements in X. For any $0 \leq \alpha \leq 1$ and $S' \subseteq S$, S' is an α-approximation of S for R if for all $r \in R$, it holds that $\left|\frac{|S \cap r|}{|S|} - \frac{|S' \cap r|}{|S'|}\right| \leq \alpha$.*

Theorem 2 ([24, 14])**.** *Let (X, R) have VC dimension d. Let $S \subseteq X$ be a subset of X. Let $0 < \alpha, \beta \leq 1$. There exists a constant c, such that if $S' \subseteq S$ is a random subset of S with size at least $c \cdot \left(\frac{d \cdot \log \frac{d}{\alpha} + \log \frac{1}{\beta}}{\alpha^2}\right)$, then with probability at least $1 - \beta$, S' is an α-approximation of S for R.*

Definition 4 (Generalization and empirical error)**.** *Let $\mathcal{D}$ be a distribution, c be a concept and h be a hypothesis. The error of h w.r.t. c over $\mathcal{D}$ is defined as*

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

For a finite dataset S, the error of h w.r.t. c over S is defined as

$$\mathrm{err}_S(c, h) = \frac{|\{x \in S \mid c(x) \neq h(x)\}|}{|S|}.$$

*For convenience, we abbreviate $\mathrm{err}_{\mathcal{D}}(c, h)$ as $\mathrm{err}_{\mathcal{D}}(h)$ and abbreviate $\mathrm{err}_S(c, h)$ as $\mathrm{err}_S(h)$. We say a hypothesis h is α-**good** with respect to a distribution $\mathcal{D}$ (dataset S) if $\mathrm{err}_{\mathcal{D}}(c, h) \leq \alpha$ ($\mathrm{err}_S(c, h) \leq \alpha$, respectively).*

Theorem 3 ([5, 16])**.** *Let $\mathcal{C}$ be a concept class and let $\mathcal{D}$ be a distribution. Let $\alpha, \beta > 0$, and $m \geq \frac{48}{\alpha}\left(10VC(\mathcal{C}) \log(\frac{48e}{\alpha}) + \log(\frac{5}{\beta})\right)$. Let S be a sample of m points drawn i.i.d. from $\mathcal{D}$. Then*

$$\Pr[\exists c, h \in \mathcal{C} \text{ s.t. } \mathrm{err}_S(c, h) \leq \alpha/10 \text{ and } \mathrm{err}_{\mathcal{D}}(c, h) \geq \alpha] \leq \beta.$$

[3] Commonly called ε-approximation. We use α-approximation as ε is used as a parameter of differential privacy.

A.2 Differential Privacy

Definition 5 (Differential Privacy [11]). *Let $\mathcal{X}$ be a data domain and $\mathcal{Y}$ be an output domain. A (randomized) mechanism M mapping $\mathcal{X}^n$ to $\mathcal{Y}$ is (ε, δ)-differentially private if for any pair of inputs $S, S' \in \mathcal{X}^n$ where S and S' differ on a single entry, and any event $E \subseteq \mathcal{Y}$, it holds that*

$$\Pr[M(S) \in E] \leq e^{\varepsilon} \cdot \Pr[M(S') \in E] + \delta,$$

where the probability is over the randomness of M.

Theorem 4 (Advanced composition [12]). *Let $M_1, \ldots, M_k : \mathcal{X}^n \to \mathcal{Y}$ be (ε, δ)-differentially private mechanisms. Then the algorithm that on input $S \in \mathcal{X}^n$ outputs $(M_1(S), \ldots, M_k(S))$ is $(\varepsilon', k\delta + \delta')$-differentially private, where $\varepsilon' = \sqrt{2k \ln(1/\delta')} \cdot \varepsilon + k\varepsilon \frac{e^{\varepsilon}-1}{e^{\varepsilon}+1}$ for every $\delta' > 0$.*

Specifically, when $\varepsilon < \frac{1}{\sqrt{k}}$, we have $\varepsilon' = \theta(\sqrt{k \ln(1/\delta')} \cdot \varepsilon)$.

A.2.1 Algorithm `BetweenThresholds` [7]

Algorithm 3: `BetweenThresholds` [7]

Input: Database $S \in X^n$.

Parameters: $\varepsilon, t_\ell, t_u \in (0,1)$ and $n, T \in \mathbb{N}$.

1. Sample $\mu \sim Lap(2/\varepsilon n)$ and initialize noisy thresholds $\hat{t}_\ell = t_\ell + \mu$ and $\hat{t}_u = t_u - \mu$.
2. For $j = 1, 2, \cdots, T$:
   (a) Receive query $q_j : X^n \to [0,1]$.
   (b) Set $c_j = q_j(S) + \nu_j$ where $\nu_j \sim Lap(6/\varepsilon n)$.
   (c) If $c_j < \hat{t}_\ell$, output L and continue.
   (d) If $c_j > \hat{t}_u$, output R and continue.
   (e) If $c_j \in [\hat{t}_\ell, \hat{t}_u]$, output $\top$ and halt.

Lemma 1 (Privacy for `BetweenThresholds` [7]). *Let $\varepsilon, \delta \in (0,1)$ and $n \in \mathbb{N}$. Then algorithm `BetweenThresholds` is (ε, δ)-differentially private for any adaptively-chosen sequence of queries as long as the gap between the thresholds t_ℓ, t_u satisfies*

$$t_u - t_\ell \geq \frac{12}{\varepsilon n} \left(\log(10/\varepsilon) + \log(1/\delta) + 1 \right).$$

Lemma 2 (Accuracy for `BetweenThresholds` [7]). *Let $\alpha, \beta, \varepsilon, t_\ell, t_u \in (0,1)$ and $n, T \in \mathbb{N}$ satisfy*

$$n \geq \frac{8}{\alpha\varepsilon} \left(\log(T+1) + \log(1/\beta) \right).$$

Then, for any input $x \in X^n$ and any adaptively-chosen sequence of queries $q_1, q_2, \cdots, q_T$, the answers $a_1, a_2, \cdots a_{\leq T}$ produced by `BetweenThresholds` on input x satisfy the following with probability at least $1 - \beta$. For any $j \in [T]$ such that a_j is returned before `BetweenThresholds` halts,

- $a_j = L \implies q_j(x) \leq t_\ell + \alpha$,
- $a_j = R \implies q_j(x) \geq t_u - \alpha$, *and*
- $a_j = \top \implies t_\ell - \alpha \leq q_j(x) \leq t_u + \alpha$.

B Construction of the Predictor for Halfspaces with Adaptive Adversary

B.1 Additional Preliminaries

For $x = (x_1, \ldots, x_d)$ and $y = (y_1, \ldots, y_d)$, we denote by $\langle x, y \rangle = \sum_{i=1}^{d} x_i y_i$ the inner-product of x and y. For $a \in \mathbb{R}^d$ and $w \in \mathbb{R}$, we define the predicate $h_{a,w}(x) = (\langle a, x \rangle \geq w)$. Let $H_{a,w}$ be the halfspace $\{x \in \mathbb{R}^d \colon h_{a,w}(x) = 1\}$.

Definition 6 (Halfspaces and hyperplanes)**.** *For $a_1, \ldots, a_d, w \in \mathbb{R}$, the halfspace predicate $h_{a_1,\ldots,a_d,w} : \mathbb{R}^d \to \{\pm 1\}$ is defined as $h_{a_1,\ldots,a_d,w}(x_1, \ldots, x_d) = 1$ if and only if $\sum_{i=1}^{d} a_i x_i \geq w$. Define the concept class HALFSPACE$_d = \{h_{a_1,\ldots,a_d,w}\}_{a_1,\ldots,a_d,w \in \mathbb{R}}$.*

Linear Feasibility Problem [16]. In a linear feasibility problem, we are given a feasible collection $S \in (\mathbb{R}^{d+1})^n$ of n linear constraints over d variables $x_1, \ldots, x_d$. The target is to find a solution in $\mathbb{R}^d$ that satisfies all (or most) constraints. Each constraint has the form $h_{a,w}(x) = 1$ for some $a \in \mathbb{R}^d$ and $w \in \mathbb{R}$.

Definition 7 ((α, β)-solving (d, n)-linear feasibility [16][4])**.** *We say that an algorithm (α, β)-solves (d, n)-linear feasibility if for every feasible collection of n linear constraints over d variables, with probability $1 - \beta$ the algorithm finds a solution $(x_1, \ldots, x_d)$ that satisfies at least $(1 - \alpha)n$ constraints.*

Notice that there exists a reduction from PAC learning of halfspaces to solving linear feasibility. Each input labeled point $((x_1, \ldots, x_d), y) \in \mathbb{R}^d \times \{-1, 1\}$ is transformed to a linear constraint $h_{y \cdot (x_1,\ldots,x_d,-1),0}$. By Theorem 3, if we can (α, β)-solve $(d + 1, n)$ linear feasibility problems, then the reduction results is an $(O(\alpha), O(\beta))$-PAC learner for d dimensional halfspaces.

Definition 8 (Convexification of a function $f : (\mathbb{R}^{d+1})^* \times \mathbb{R}^d \to \mathbb{R}$ [16])**.** *For $S \in (\mathbb{R}^{d+1})^*$ and $y \in \mathbb{R}$, let $\mathcal{D}_S(y) = \{z \in \mathbb{R}^d : f(S, z) \geq y\}$. The convexification of f is the function $f_{Conv} : (\mathbb{R}^{d+1})^* \times \mathbb{R}^d \to \mathbb{R}$ defined by $f_{Conv}(S, x) = \max\{y \in \mathbb{R} : x \in ConvexHull(\mathcal{D}_{S(y)})\}$.*

I.e., if x is in the convex hull of points $Z \subset \mathbb{R}^d$ then $f_{Conv}(S, x)$ is at least $\min_{z \in Z}(f(S, z))$.

Definition 9 ($depth$ and $cdepth$ [16])**.** *Let S be a collection of predicates. Define $depth_S(x)$ to be the number of predicates $h_{a,w}$ in S such that $h_{a,w}(x) = 1$. Let $cdepth_S(x) = f_{Conv}(\mathcal{S}, x)$ for the function $f(S, x) = depth_{\mathcal{S}}(x)$.*

Fact 1 ([16])**.** *For any $S \in (\mathbb{R}^d \times \mathbb{R})^*$ and any $x \in \mathbb{R}^d$, it holds that*

$$depth_S(x) \geq (d + 1) \cdot cdepth_S(x) - d|S|.$$

By the above fact, if we can find a point with $cdepth_S \geq (1 - \alpha)|S|$ where $\alpha \ll 1/(d + 1)$, then this point has $depth_S \approx |S|$.

Lemma 3 (Approximation of cdepth [20])**.** *Let $S \subseteq (\mathbb{R}^d)^n$ and let $S' \subseteq S$ be a random subset of S with cardinality $m = |S'| \geq O\left(\frac{d \cdot \log(\frac{d}{\alpha}) + \log \frac{1}{\beta}}{\alpha^2}\right)$. Then, with probability at least $1 - \beta$, for all $p \in \mathbb{R}^d$, if $cdepth_{S'}(p) = \gamma' m$ and $cdepth_S(p) = \gamma n$ then $|\gamma - \gamma'| \leq \alpha$.*

B.2 Hypotheses Generator for Halfspaces Class

We adapt the approach of Nissim et al. [20] to design the hypothesis generator for halfspaces. The key step is to reduce learning a consistent halfspace to maintaining the feasibility of a suitable linear program (equivalently, a convex feasibility region in a dual parameter space). In this representation, a parameter point induces a hypothesis, and points of large $depth$ with respect to a labeled subset correspond to hypotheses with small empirical error on that subset. (See Figure 3 for illustration)

A technical complication is that updating according to $depth$ may incur an exponential increase of error, so we work with the convex version $cdepth$: at each round, we select a point of high $cdepth$ within the current feasible region. This choice has two crucial consequences. First, whenever `BetweenThresholds` outputs $\top$ (a hard query), the corresponding query induces a hyperplane constraint, and one can show that this hyperplane contains at least one point whose $cdepth$ remains large. Thus, we can safely shrink the approximate feasible region by intersecting it with this hyperplane without losing the existence of a high-$cdepth$ point. Second, high $cdepth$ implies high depth (by Fact 1), which in turn implies that the induced hypothesis has low empirical error on the block.

Finally, each hard query adds one hyperplane intersection to the feasible region. In $\mathbb{R}^{d+1}$, after at most $d+1$ such intersections (under the progress guarantee from [20]), the feasible region collapses to essentially a single point, at which point all remaining hypotheses agree on future queries and `BetweenThresholds` no longer outputs $\top$.

[4] In [16], the authors consider the coefficients are in a domain X. In this work, we let X to be $\mathbb{R}$.

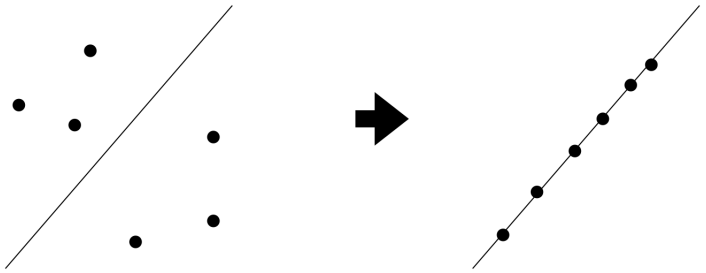


Figure 3: In the left figure, we illustrate that when a "hard" query occurs, the current hypotheses are roughly split across the two sides of the induced hyperplane. We then update the feasible set by restricting all hypotheses to lie on this hyperplane for subsequent rounds, while preserving the existence of a hypothesis (point) with high $cdepth$, as shown in the right figure.

Algorithm 4: $G_{halfspace}$: Hypotheses Generator for Halfspaces Class

Inputs: A labeled database S_i,
$transcript_j = ((x_1, y_1, \mathsf{Label}_1), \ldots, (x_{j-1}, y_{j-1}, \mathsf{Label}_{j-1}))$, where
$x_1, \ldots, x_{j-1} \in \mathbb{R}^d$, $y_1, \ldots, y_{j-1} \in \{\pm 1, \top\}$, $\mathsf{Label}_1, \ldots, \mathsf{Label}_{j-1} \in \{\pm 1\}$.

1. Let $HardQuery = \{(x, y, \mathsf{Label}) \in transcript_j | y = \top\}$. Let $\mathcal{C}|_{HardQuery} = \{c = (a_1, \ldots, a_d, w) : \langle a, x_{j^*} \rangle = w \text{ for all } (x_{j^*}, y_{j^*}, \mathsf{Label}_{j^*}) \in HardQuery\}$
   /* That is, the intersection space of all hyperplanes in $HardQuery$ and in the linear feasibility problem. */
2. Output a hypothesis $f_{i,j} = \arg\max_{h \in \mathcal{C}|_{HardQuery}} [cdepth_{S_i}(h)]$. (That is, selecting $f_{i,j}$ from $\mathcal{C}|_{HardQuery}$, such that $cdepth_{S_i}(f_{i,j}) = \max_{h \in \mathcal{C}|_{HardQuery}} [cdepth_{S_i}(h)]$)

B.2.1 Analysis

Here we recall that we partition the dataset S into k subsets $S_1, \ldots, S_k$, each with size $|S_i| = m$.

Claim 7. *If* $m \geq O\left(\frac{d \cdot \log \frac{d}{\alpha} + \log \frac{1}{\beta}}{\alpha^2}\right)$, *then with probability* $1 - k\beta$, *we have*

1. $\left|\frac{cdepth_S(h)}{N} - \frac{cdepth_{S_i}(h)}{m}\right| \leq \alpha$ *for all* $i \in [k]$ *and all* $h \in \mathcal{C}$.
2. $\left|\frac{cdepth_{S_j}(h)}{m} - \frac{cdepth_{S_i}(h)}{m}\right| \leq 2\alpha$ *for all* $i, j \in [k]$ *and all* $h \in \mathcal{C}$.

Proof. By Lemma 3, with probability $1 - \beta$, we have $\left|\frac{cdepth_S(h)}{N} - \frac{cdepth_{S_i}(h)}{m}\right| \leq \alpha$ for any fixed $i \in [k]$. The first statement can follow by the union bound. The second statement is the corollary of the first statement. □

Claim 8. *When* `BetweenThresholds` *runs under the guarantee of Lemma 2,* $\mathcal{C}|_{HardQuery}$ *is always non-empty.*

Proof. We prove it by induction on $|HardQuery|$. In the base case of $|HardQuery| = 0$, then $\mathcal{C}|_{HardQuery}$ and the claim is immediate.

Assume the claim holds for the cases $|HardQuery| \leq a - 1$, then we prove it also holds for $|HardQuery| = a$. Let $(x_{a^*}, \top, \mathsf{Label}_{a^*})$ be the a-th element of $HardQuery$. On the a^*-th query, `BetweenThresholds` output $\top$ in Step 2c. It happens when there exists $f_{i,a^*}(x_{a^*}) = \mathsf{Label}_{a^*}$ and $f_{i',a^*}(x_{a^*}) = -\mathsf{Label}_{a^*}$ in query $q_{x_{a^*}}$ for some $i, i' \in [k]$.

Viewing hypotheses as feasible points in the associated linear-feasibility (dual) space, the condition $f(x_{a^*}) = \pm \mathsf{Label}_{a^*}$ corresponds to lying on opposite sides of the hyperplane induced by x_{a^*}. Hence

the segment joining these two feasible points intersects that hyperplane. Let p be an intersection point. Then p satisfies all previous hard-query constraints and also satisfies the new constraint induced by x_{a^*}. Therefore $p \in \mathcal{C}|_{HardQuery}$, proving that the restricted class remains non-empty.

□

Claim 9. *When* `BetweenThresholds` *runs under the guarantee of Lemma 2, with probability* $1-\beta$*,* `BetweenThresholds` *outputs* $\top$ *at most* $d+1$ *times.*

Proof. Notice that we maintain the feasible set as the intersection of the hyperplanes induced by the hard queries. Consider a new query x. If the hyperplane induced by x is redundant with the existing constraints (e.g., it is parallel to, or coincides with, one of the previously imposed hyperplanes in the relevant feasibility representation), then the feasible region lies entirely on one side of the decision boundary for x. In this case, all feasible hypotheses agree on the label of x, so the ensemble vote is far from the threshold and `BetweenThresholds` will not output $\top$.

More generally, `BetweenThresholds` can output $\top$ only when the new hard query induces a *non-redundant* hyperplane constraint, i.e., one that strictly shrinks the feasible region. Each such non-redundant constraint decreases the affine dimension of the feasible region by at least one. Since the feasible region lives in $\mathbb{R}^{d+1}$, this can happen at most $d+1$ times. Once the affine dimension drops to 0, the feasible region collapses to (at most) a single point, and hence all remaining hypotheses agree on every future query; therefore `BetweenThresholds` will never output $\top$ again.

□

Let $v = d+1$, we have the privacy guarantee of the predictor for halfspace class with an adaptive adversary .

Corollary 3 (Privacy Guarantee)**.** *Using* $G_{halfspace}$*, Algorithm 1 is* $(O(\sqrt{d \cdot \ln(1/\delta)} \cdot \varepsilon), O(d \cdot \delta))$*-differentially private*

Claim 10. *Under the guarantee of Claim 7 and Claim 9, all* $f_{i,j}$ *satisfies* $\frac{cdepth_S(f_{i,j})}{N} \geq 1-(2d+3)\alpha$

Proof. We prove the stronger inductive statement: if $|HardQuery| = c$, then for every $i \in [k]$, we have $\frac{\max_{h \in \mathcal{C}|_{HardQuery}}[cdepth_{S_i}(h)]}{m} \geq 1 - 2c\alpha$. Since $G_{halfspace}$ outputs a maximizer of $cdepth_{S_i}$ over the restricted set, this implies $\frac{cdepth_{S_i}(f_{i,j})}{m} \geq 1-2c\alpha$, and then Claim 7 transfers the bound from S_i to S.

In the base case of $c = 0$, we optimize over $\mathcal{C}$, and there exists a feasible hypothesis achieving $cdepth_{S_i}(h) = m$. Hence the claim holds.

Assume the statement holds for $c-1$, and consider the first time $|HardQuery| = c$. Let $(x_{t^\star}, \top, \mathsf{Label}_{t^\star})$ be the c-th hard query added. At round $t^\star$, `BetweenThresholds` outputs $\top$. Under Lemma 2, this implies that the ensemble is uncertain on $x_{t^\star}$, and in particular there exist two generated hypotheses $f_{i^\star,t^\star}$ and $f_{i^{\star\star},t^\star}$ such that $f_{i^\star,t^\star}(x_{t^\star}) = \mathsf{Label}_{t^\star}$ and $f_{i^{\star\star},t^\star}(x_{t^\star}) = -\mathsf{Label}_{t^\star}$.

In the associated feasibility space, these correspond to two feasible points lying on opposite sides of the hyperplane $H_{x_{t^\star}}$ induced by the query. Let p_{good} be an intersection point of $H_{x_{t^\star}}$ with the segment joining these two feasible points. Then p_{good} satisfies all constraints in $\mathsf{HardQuery} \setminus \{(x_{t^\star}, \top, \mathsf{Label}_{t^\star})\}$ and also lies on $H_{x_{t^\star}}$, hence $p_{\mathsf{good}} \in \mathcal{C}|_{HardQuery}$.

By the convexity property of $cdepth$ in the feasibility representation, we have

$$\begin{aligned}\frac{cdepth_S(p_{good})}{N} &\geq \min\{\frac{cdepth_S(f_{i^\star,t^\star})}{N}, \frac{cdepth_S(f_{i^{\star\star},t^\star})}{N}\} \\ &\geq \min\{\frac{cdepth_{S_{i^\star}}(f_{i^\star,t^\star})}{m} - \alpha, \frac{cdepth_{S_{i^{\star\star}}}(f_{i^{\star\star},t^\star})}{m} - \alpha\} \\ &\geq 1-(2c-1)\alpha\end{aligned}$$

It implies $\frac{\max_{h \in \mathcal{C}|_{HardQuery}}[cdepth_{S_i}(h)]}{m} \geq \frac{cdepth_S(p_{good})}{N} - \alpha \geq 1 - 2c\alpha$ for all $i \in [k]$. At the end, this claim can result from $c \leq d+1$.

□

Using Fact 1, we have the following corollary:

Corollary 4. *Under the guarantee of Claim 7 and Claim 9, all $f_{i,j}$ satisfies $depth_S(f_{i,j}) \geq (1 - O(d^2\alpha))N$*

By Theorem 3, it implies

Claim 11. *Under the guarantee of Claim 7 and Claim 9, all $f_{i,j}$ are $O(d^2 \cdot \alpha)$-good with respect to the distribution $\mathcal{D}$.*

Using Claim 2, we have the accuracy guarantee:

Corollary 5 (Accuracy Guarantee)**.** *Using $G_{halfspace}$, with probability $1 - O((d+k) \cdot \beta)$, all queries are predicted by a predictor that is $O(d^2 \cdot \alpha)$-good with respect to the distribution $\mathcal{D}$.*

Select the appropriate parameters, and we can have the main result. We put the details in Appendix C.3

Theorem 5. *For halfspaces class $\mathcal{C}$ over domain $\mathbb{R}^d$, there exists an (ε, δ)-differentially private algorithm that can (α, β) predict T queries if the given dataset has size*

$$N \quad = \tilde{O}_{\alpha,\beta,\varepsilon,\delta}\left(d^{5.5} \cdot \log T\right)$$

C Appendix

C.1 Claims on Generic Predictor

Claim 12 (Privacy of Generic Predictor [19], restate of Claim 1)**.** *If `BetweenThresholds` in Step 2c outputs $\top$ at most v times, then Algorithm 1 is $\left(O(\sqrt{v \cdot \ln(1/\delta)} \cdot \varepsilon), O(v\delta)\right)$-differentially private.*

Proof. Notice that one different point makes one $f_{i,j}$ different. This claim can result from advanced composition (Theorem 4). □

Claim 13 (Accuracy of Generic Predictor [19], restate of Claim 2)**.** *If all $f_{i,j}$ are α-good with respect to the underlying distribution $\mathcal{D}$, then with probability $1 - v\beta$, all predictors defined above are 4α-good with respect to the distribution $\mathcal{D}$.*

Proof. Let $\mathcal{G}$ be the event that every invocation of `BetweenThresholds` satisfies the accuracy guarantee of Lemma 2. By a union bound over at most v invocations, $\Pr[\mathcal{G}] \geq 1 - v\beta$. Condition on $\mathcal{G}$.

Fix a round j. Note that $q_x(F_j)$ is exactly the fraction of hypotheses in F_j that output $+1$ on x.

Run `BetweenThresholds` on $q_x(F_j)$ with thresholds $t_\ell = 3/8$ and $t_u = 5/8$ (and let $\tau = 1/8$ denote the accuracy parameter appearing in Lemma 2). Under $\mathcal{G}$, Lemma 2 implies:

- if the output is L, then $q_x(F_j) \leq t_\ell + \tau$;
- if the output is R, then $q_x(F_j) \geq t_u - \tau$;
- if the output is $\top$, then $q_x(F_j) \in [t_\ell - \tau,\, t_u + \tau]$.

We claim that whenever the algorithm outputs an *incorrect* label on x, at least a $1/4$-fraction of the hypotheses $\{f_{i,j}\}_{i=1}^k$ are incorrect on x (with respect to c), provided $\tau \leq 1/8$. Indeed:

- If the algorithm outputs -1 (output L) but $c(x) = +1$, then the fraction of -1 votes is $1 - q_x(F_j) \geq 1 - (t_\ell + \tau) = 5/8 - \tau \geq 1/4$.
- If the algorithm outputs $+1$ (output R) but $c(x) = -1$, then the fraction of $+1$ votes is $q_x(F_j) \geq t_u - \tau = 5/8 - \tau \geq 1/4$.

- If the output is $\top$, the algorithm outputs a uniformly random label; if this label is wrong, then either $c(x) = +1$ and the wrong label is -1, or vice versa. In either case, since $q_x(F_j) \in [t_\ell - \tau,\ t_u + \tau]$, the fraction of votes for the wrong label is at least $\min\{q_x(F_j), 1 - q_x(F_j)\} \geq 3/8 - \tau \geq 1/4$.

Therefore, for every x,

$$\mathbf{1}[\text{algorithm errs on } x] \ \leq\ 4 \cdot \frac{1}{k} \sum_{i=1}^{k} \mathbf{1}[f_{i,j}(x) \neq c(x)].$$

Taking expectation over $x \sim \mathcal{D}$ and using $\mathrm{err}_{\mathcal{D}}(c, f_{i,j}) \leq \alpha$ for all i yields

$$\mathrm{err}_{\mathcal{D}}(c, h_j) \ \leq\ 4 \cdot \frac{1}{k} \sum_{i=1}^{k} \mathrm{err}_{\mathcal{D}}(c, f_{i,j}) \ \leq\ 4\alpha.$$

Since this holds for every $j \in [T]$ under $\mathcal{G}$, the claim follows. □

C.2 Parameter Setting for the Predictor with Oblivious Adversary

Let $VC(\mathcal{C}) = d$. Set parameter

- $\beta_{BT} = \frac{\beta\varepsilon}{(d\log T + \log(d\log T/\alpha\beta\varepsilon\delta))\sqrt{(d\log T + \log(d\log T/\alpha\beta\varepsilon\delta))\cdot\log(d\log T/\alpha\beta\varepsilon\delta)}\cdot\log(d\log T/\alpha\beta\varepsilon\delta)}$
- $\varepsilon_{BT} = \frac{\varepsilon}{\sqrt{(d\cdot\log T + \log(d\log T/\alpha\beta\varepsilon\delta))\cdot\ln(d\log T/\alpha\beta\varepsilon\delta)}}$
- $\delta_{BT} = \frac{\delta}{d\cdot\log T + \log(d\log T/\alpha\beta\varepsilon\delta)}$
- $\alpha_{BT} = \frac{\alpha}{d\cdot\log T + \log(d\log T/\alpha\beta\varepsilon\delta)}$

Then we have the number of subsets

$$\begin{aligned} k &= \tfrac{64}{\varepsilon_{BT}}\left(\log(T+1) + \log(1/\beta_{BT})\right) \\ &= O\Big(\frac{\sqrt{(d\cdot\log T + \log(d\log T/\alpha\beta\varepsilon\delta))\cdot\ln(d\log T/\alpha\beta\varepsilon\delta)}\cdot(\log T + \log(\frac{d\log\log T\log\log(1/\alpha\delta)}{\beta\varepsilon})}{\varepsilon}\Big) \\ &= \tilde{O}_{\alpha,\beta,\varepsilon,\delta}(d^{0.5}\cdot\log^{1.5} T) \end{aligned}$$

and the size of each subset

$$\begin{aligned} m &= O\left(\frac{d\cdot\log\frac{d}{\alpha_{BT}} + \log\frac{1}{\beta_{BT}}}{\alpha_{BT}^2}\right) \\ &= O\left(\frac{(d\cdot\log T + \log(d\log T/\alpha\beta\varepsilon\delta))^2\cdot(d\log\frac{d}{\alpha\beta\varepsilon\delta} + \log\frac{d\log\log T\log\log(1/\alpha\delta)}{\beta\varepsilon})}{\alpha^2}\right) \\ &= \tilde{O}_{\alpha,\beta,\varepsilon,\delta}(d^3\cdot\log^2 T) \end{aligned}$$

Then we have $N = mk = \tilde{O}_{\alpha,\beta,\varepsilon,\delta}\left(d^{3.5}\cdot\log^{3.5} T\right)$

C.3 Parameter Setting for the Predictor of halfspace class

Set parameter

- $\beta_{BT} = \frac{\beta\varepsilon}{d\log T\sqrt{d\log(d\log T/\delta)}\cdot(\log d + \log\log T + \log\log(1/\delta) + \log(1/\varepsilon))}$
- $\varepsilon_{BT} = \frac{\varepsilon}{\sqrt{d\cdot\ln(d/\delta)}}$
- $\delta_{BT} = \frac{\delta}{d}$
- $\alpha_{BT} = \frac{\alpha}{d^2}$

Then we have the number of subsets

$$\begin{aligned} k &= \tfrac{64}{\varepsilon_{BT}}\left(\log(T+1) + \log(1/\beta_{BT})\right) \\ &= O\Big(\frac{\sqrt{d\ln(d/\delta)}\cdot(\log T + \log(\frac{d\log\log T\log\log(1/\delta)}{\beta\delta})}{\varepsilon}\Big) \\ &= \tilde{O}_{\alpha,\beta,\varepsilon,\delta}(d^{0.5}\cdot\log T) \end{aligned}$$

and the size of each subset (we ignore the $O(\log d)$ and $O(\log\log T)$ part)

$$\begin{aligned} m &= O\left(\frac{d\cdot\log\frac{d}{\alpha_{BT}}+\log\frac{1}{\beta_{BT}}}{\alpha_{BT}^2}\right) \\ &= O\left(\frac{d^4\cdot(d\log\frac{d}{\alpha}+\log(\frac{d\log\log T\log\log(1/\delta)}{\beta\delta}))}{\alpha^2}\right) \\ &= \tilde{O}_{\alpha,\beta,\varepsilon,\delta}(d^5) \end{aligned}$$

Then we have $N = mk = \tilde{O}_{\alpha,\beta,\varepsilon,\delta}\left(d^{5.5}\cdot\log T\right)$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [N/A]

   Justification: This is theoretical research, and there is no experiment.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [N/A]

   Justification: This is theoretical research, and there is no experiment or code.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [N/A]

   Justification: This is theoretical research, and there is no experiment.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: This is theoretical research, and there is no experiment.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [N/A]

   Justification: This is theoretical research, and there is no experiment.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

Justification:

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.