Always Asking for Labels is Optimal (in a well-studied setting)

Siddhartha Banerjee SBANERJEE@CORNELL.EDU
Joseph Y. Halpern HALPERN@CS.CORNELL.EDU
Spencer Peters SP2473@CORNELL.EDU
Cornell University

Abstract

Selective sampling is the problem of learning from a stream of unlabeled data by requesting an (ideally small) subset of data labels. Much theoretical work has focused on the case where d-dimensional data x are distributed uniformly on the $(d-1)$-dimensional hypersphere, and the correct labels correspond to a linear threshold (homogeneous hyperplane); that is, there is some $w \in \mathbb{R}^n$ such that the correct label of x is 1 if $x \cdot w > 0$ and 0 otherwise (Freund et al., 1997; Long, 2003; Dasgupta et al., 2005; Balcan et al., 2007). We show that in this setting, the simple CAL algorithm of collecting labels when unsure (Cohn et al., 1994) achieves optimal cost $O(\log T)$ as a function of the number of decisions T. This implies optimal (expected) label complexity $O(\log 1/\epsilon)$ as a function of generalization error ϵ. Previously the best known label complexity was $O(\log(1/\epsilon) \log\log(1/\epsilon))$. In particular, we show CAL obtains cost $\Theta(d^{3/2} \log T)$ and thus label complexity $\Theta(d^{3/2} \log 1/\epsilon)$. Although previously proposed algorithms achieve better $\Theta(d)$ dependence on the dimension d, these algorithms necessarily make classification mistakes when run in an online context. CAL makes no mistakes (since it collects the labels of all ambiguous data points), which makes it potentially interesting for applications where mistakes are costly (e.g., medicine), or consistent decisions are especially important (e.g., enforcing content moderation policies). Moreover, its simple form makes it particularly appealing for applications. To support this claim, we demonstrate theoretically and experimentally that (in this setting) CAL can be implemented efficiently.

1. Introduction

Selective sampling is a learning model designed to capture the fact that labeled data is often much more expensive than unlabeled data (Freund et al., 1997; Dasgupta et al., 2005). In selective sampling, the learner is given access to a stream of unlabeled data $X_1, X_2, \ldots \in \mathcal{X}$ drawn i.i.d. from a known distribution $\mathcal{D}$ over the instance space $\mathcal{X}$. After viewing X_i, the learner may request its *label* $f(X_i) \in \{0, 1\}$, where f belongs to a known concept class $\mathcal{F}$. The typical goal is to output $g : \mathcal{X} \to \{0, 1\}$ such that the generalization error $\Pr_{X \sim \mathcal{D}}[g(X) \neq f(X)]$ is at most ϵ while requesting as few labels as possible. One natural and well-studied setting for selective sampling is that of learning linear thresholds against the uniform distribution on the hypersphere (Freund et al., 1997; Long, 2003; Dasgupta et al., 2005; Balcan et al., 2007). That is, $\mathcal{X} = \mathbb{R}^d$, $\mathcal{D} = \mathsf{Unif}\,\{x : \|x\|_2 = 1\}$, $\mathcal{F} = \{f_w : w \in \mathbb{R}^d\}$, where $f_w = 1$ iff $w \cdot x > 0$. This setting is both tractable (learning-theoretic analyses date back to the classic Perceptron algorithm (Rosenblatt, 1958)) and expressive ($\mathcal{F}$ can model many different decision boundaries via the kernel trick, and since $f_w(x)$ is linear, $\mathcal{D}$ may be replaced with a (possibly unknown) multivariate Gaussian without loss of generality.) However, until this work, it was unknown whether ϵ generalization error could be attained with optimal label complexity $O(\log 1/\epsilon)$ (that is, $O(h(d) \log 1/\epsilon)$ for some function $h(d)$).[1] Informally, let the CAL algorithm A_{CAL} (Cohn et al., 1994) be the simple algorithm that classifies a point whenever its label can be deduced with certainty from the previous points (and otherwise requests the label). Our main result shows that CAL achieves ϵ generalization error with optimal $O(\log(1/\epsilon))$ label complexity (in expectation). Alternatively, it shows that CAL achieves optimal expected cost $O(\log T)$, where the cost C_T is the number of label requests plus the number of incorrect classifications (mistakes). Of course, our asymptotic results also apply when label requests and mistakes have different fixed positive costs. We feel this latter point of view, although less standard, is more natural, since it corresponds to the cost incurred when running the algorithm in an online setting.

Theorem 1 *If $\mathcal{X} = \mathbb{R}^d$, the X_t are drawn uniformly at random from the unit hypersphere S^d, and the corresponding labels are consistent with some linear threshold f_w, A_{CAL} incurs expected cost $\Theta(d^{3/2} \log T)$. Alternatively, after requesting $\Theta(d^{3/2} \log(1/\epsilon))$ labels in expectation (over the training data $\{X_t\}_t$), A_{CAL} classifies $X \sim S^d$ (correctly) with probability at least $1 - \epsilon$.*

Moreover, we show that CAL is efficiently implementable in our setting.

Theorem 2 *In the setting of Theorem 1, there is an implementation of the CAL algorithm that processes each instance x_t in amortized expected time $O(d^{3.6})$ independent of t.*

This result is of interest not only because it is the first to show that optimal $O(\log 1/\epsilon)$ label complexity can be achieved, but also because the optimal algorithm turns out to be CAL. This algorithm is extremely simple to describe and has no parameters. Most importantly, when the CAL algorithm is run online, it makes no mistakes. That is, on input x_t, it either outputs a correct classification y_t, or requests a label, as opposed to more aggressive selective sampling policies that sometimes make incorrect classifications. (We mention that a margin-based algorithm given in (Balcan et al., 2007) also makes no mistakes with high probability, although at the cost of a $\log \log 1/\epsilon$ factor in the label complexity.) The practical benefits of no mistakes are obvious. Mistakes can be expensive, for example, in medical settings like automated radiology. Inconsistent classifications in themselves are also problematic. In settings like content moderation and policy enforcement systems more broadly, they can confuse affected individuals and lead to allegations of bias and discrimination. Referring every potential mistake to a labeling process (such as a human expert) eliminates all of these problems, so long as the labeling process is also consistent. Of course, in the real world, human experts are not consistent. Moreover, data is not always distributed via a multivariate Gaussian, and classifications of interest cannot always be expressed as linear thresholds. So we cannot claim that the CAL algorithm is a truly practical tool in these applications (e.g., compared to deep-learning based techniques); we are bound by the limitations of our simple theoretical setting. But our results suggest the CAL algorithm is more practical than previously believed, and that it is deserving of further study.

2. Results

To define A_{CAL} formally, we introduce some notation. Given $n > 0$ and *labeled instances* $\mathcal{S} \in (\mathcal{X} \times \{0,1\})^n$, let $\mathcal{H}_\mathcal{S} := \{h \in \mathcal{H} \mid \forall (x,y) \in \mathcal{S}, h(x) = y\}$ be the set of hypotheses consistent

[1] The fact that $O(\log 1/\epsilon)$ label complexity is optimal is a straightforward consequence of known Probably Approximately Correct (PAC) lower bounds, although for some reason we could not find a proof in the literature; see Proposition 6 for a proof.

with $\mathcal{S}$. Thus, $\mathcal{H}_\mathcal{S}$ consists of all the hypotheses that are consistent with the the labels on the instances in $\mathcal{S}$. A set of hypotheses $\mathcal{H}'$ *determines* x, written $\mathcal{H}' \twoheadrightarrow x$, if there is some $y \in \{0,1\}$ such that for all $h \in \mathcal{H}'$, $h(x) = y$. Intuitively, $\mathcal{H}'$ determines x if all the hypotheses in $\mathcal{H}'$ agree that the label of x should be y. The subset of $x \in \mathcal{X}$ such that $\mathcal{H}'$ does not determine x is often called the *region of uncertainty*. In this case, we call y the *inferred label* of x. Next, for an instance x, $\mathcal{S}$ *determines* x, written $\mathcal{S} \twoheadrightarrow x$, if $\mathcal{H}_\mathcal{S} \twoheadrightarrow x$. Finally, X_t *is determined* if $\mathcal{S}_t$ determines X_t, where $\mathcal{S}_t := \{(X_1, Y_1), \ldots, (X_{t-1}, Y_{t-1})\}$. Now we can define A_{CAL}: if X_t is determined, A_{CAL} classifies X_t according to the inferred label; otherwise A_{CAL} requests the label. This is allowed because at time t, A_{CAL} knows the labels of all X_s with $s < t$; this is because either A_{CAL} asked for advice at time s, or X_s is determined. In either case, A_{CAL} knows the label Y_s. Notice that A_{CAL}

1. makes no mistakes (and is therefore consistent), as discussed earlier,
2. is independent of the parameters of the data distribution,
3. and is independent of the time horizon T over which the cost is to be minimized.

We also demonstrate, both theoretically and experimentally, that it is computationally efficient in the simple cases that we study. In particular, the update in round t will turn out to be independent of t, even though the set $\mathcal{S}_t$ is growing with t; see Section 4 for details.

As a warm-up, we study the one-dimensional case where the features X_t are bounded scalar values, and the hypotheses are thresholds. In this case, we need not assume the distribution $\mathcal{D}$ is uniform; CAL works for any continuous distribution. For $k \in [0,1]$, let $f_k = 1$ iff $x \geq k$.

Theorem 3 *If $\mathcal{X} = [0,1]$, the instances X_t are drawn from a continuous distribution $\mathcal{D}$ over $\mathcal{X}$, and the corresponding labels are consistent with some threshold f_k, then A_{CAL} has expected cost $O(\log T)$. Alternatively, after requesting $\Theta(\log(1/\epsilon))$ labels in expectation (over the training data $\{X_t\}_t$), A_{CAL} classifies $X \sim [0,1]$ (correctly) with probability at least $1-\epsilon$.*

Proof

Suppose that the target hypothesis corresponds to the threshold $k \in [0,1]$. Notice that at time t, the region of uncertainty is the interval (A_t^k, B_t^k) between the largest instance A_t^k among $X_1, \ldots, X_{t-1}$ with $A < k$, and the smallest instance B_t^k with $B_t^k \geq k$. (If $X_j \geq k$ for all j, define $A_t^k = 0$; if $X_j < k$ for all j, define $B_t^k = 1$.) Instances smaller than A_t^k are known to have label -1 if k is the correct threshold; those larger than B_t^k are known to have label 1.

Since (A_t^k, B_t^k) is the interval between two adjacent instances, and there are $t-1$ instances total, we expect the probability mass $Pr_{X\sim D}(X \in (A_t^k, B_t^k))$ of (A_t^k, B_t^k) to be of order $1/t$. Here is a proof: The event that X_t falls between A_t^k and B_t^k is the same as the event that either $X_t = A_{t+1}^k$ or $X_t = B_{t+1}^k$. But since the X_i, $i \in [t]$, are i.i.d., each X_i has equal probability of being A_{t+1}^k or B_{t+1}^k. Furthermore, there is some $i \leq t$ for which either $X_i = A_{t+1}^k$ or $X_i = B_{t+1}^k$. Hence, $1/t \leq Pr(X_t = A_{t+1}^k \text{ or } X_t = B_{t+1}^k) \leq 2/t$. It follows $1/t \leq Pr(X_t \in (A_t^k, B_t^k)) \leq 2/t$. However, the cost c_t at round t is 1 if A_{CAL} queries and 0 otherwise; and A_{CAL} queries iff $X_t \in (A_t^k, B_t^k)$. So we have

$$\mathbb{E}\, C_T = \sum_{t=1}^{T} \mathbb{E}\, c_t = \sum_{t=1}^{T} \mathbb{E}\, \mathbb{1}_{X_t \in (A_t^k, B_t^k)} = \sum_{t=1}^{T} Pr(X_t \in (A_t^k, B_t^k)).$$

It follows $\sum_{t=1}^{T} 1/t \leq \mathbb{E}\, C_T \leq \sum_{t=1}^{T} 2/t$. Standard bounds for these harmonic sums then give $\log T \leq \mathbb{E}\, C_T \leq 2\log T + 1$, as desired. For the "Alternatively" claim, observe that $\epsilon = \Pr(X_t \in (A_t^k, B_t^k)) \leq 2/T$ is achieved after an expected $O(\log T) = O(\log 1/\epsilon)$ label requests. ■

Theorem 3 is tight, as the following theorem shows.

Theorem 4 *In the setting of Theorem 3, if $\mathcal{D}$ is continuous, then for all selective sampling algorithms ALG, there exists a target hypothesis such that ALG has expected cost $\Omega(\log T)$.*

A proof of this theorem can be found in Appendix B in the supplementary material. To get a sense of why this statement is true, note that after dropping t instances in $[0, 1]$, even if all the labels are observed, the gap between the largest negatively labeled instance and smallest positively labeled instance is of order $1/t$. Any new instance that falls in, say, the middle half of this gap has a reasonable chance of being misclassified if the moderator does not ask for advice. Hence the expected cost cannot be better than $\sum_{t=1}^{T}(1/t) < \log T + 1$.

Combining these results, we see that so long as the X_t are drawn from a fixed continuous distribution on $[0, 1]$, then with high probability, A_{CAL} achieves optimal cost. It is, however, easy to see that if the X_t are chosen arbitrarily, no algorithm can beat the trivial $O(T)$ cost bound in the worst case (see Proposition 18).

Our analysis of the case $\mathcal{X} = [0, 1]$ can be extended to provide lower and upper bounds on cost in other cases; we later apply it to our main example. We can get a more general lower bound by applying PAC (Probably Approximately Correct) learning bounds Valiant (1984). This bound seems almost trivial (and may well be known by experts in the field, although we could not find a reference), but it is tight up to a factor polynomial in $\log\log 1/\epsilon$ in the cases we consider. To state this lower bound, we first recall the definition of *PAC learning sample complexity* Long (1995); Shalev-Shwartz and Ben-David (2014).[2]

Definition 5 *Given a set $\mathcal{X}$, a hypothesis class $\mathcal{H}$ consisting of hypotheses $h : \mathcal{X} \to \{-1, 1\}$, a distribution D over $\mathcal{X}$, and parameters $\epsilon > 0, \delta > 0$, the* PAC learning sample complexity *$m_{\mathcal{H},D}(\epsilon, \delta)$ is the minimum number m such that there exists an algorithm ALG such that for all $h \in \mathcal{H}$, given m labeled samples $(X_1, h(X_1)), \ldots, (X_m, h(X_m))$, ALG returns a function $f : \mathcal{X} \to \{-1, 1\}$, not necessarily in $\mathcal{H}$, such that with probability $1 - \delta$, $P_{x\sim D}(f(X) \neq h(X)) \leq \epsilon$.*

Our general lower bound is given in the following proposition.

Proposition 6 *If $m_{\mathcal{H},D}(\epsilon, \delta) \in \Omega(f(\epsilon, \delta))$, then there exist $\Delta > 0$ and $c > 0$ such that, defining $f'(\epsilon) = f(\epsilon, \Delta)$ and f'^{-1} to be the inverse of f', all selective sampling algorithms have*

$$\mathbb{E}\, C_T \in \Omega\left(\sum_{t=1}^{T} f'^{-1}(ct)\right).$$

Proof Consider a modified setting where the algorithm always observes Y_t, whether it asks for the label or not. Clearly, any algorithm for the selective sampling setting can be viewed as an algorithm for the modified setting, by simply ignoring the labels Y_t on rounds t where it does not ask for the

[2] There is a small difference between the definitions of PAC learning sample complexity given in Shalev-Shwartz and Ben-David (2014) and Long (1995); see Remark 3.2 of Shalev-Shwartz and Ben-David (2014).

label. Thus, it suffices to show an $\Omega(\log T)$ bound on expected cost in the modified setting. Fix an algorithm ALG. The bound $m_{\mathcal{H},D}(\epsilon,\delta) \in \Omega(f(\epsilon,\delta))$ implies the existence of $\mathcal{E} > 0$, $\Delta > 0$, and $c > 0$ such that for all $\epsilon \leq \mathcal{E}$ and all $\delta \leq \Delta$, there exists a target hypothesis h^* such that if ALG is given fewer than $cf(\epsilon,\delta)$ labeled samples, with probability at least δ, it will output a hypothesis h such that $P_{x\sim\mathcal{D}}(h(x) \neq h^*(x)) \geq \epsilon$. As in the theorem statement, let $f'(\epsilon) = f(\epsilon,\Delta)$ and f'^{-1} be the inverse of f. Then the above claim implies that if ALG is given fewer than t samples, with probability at least Δ, it will output a hypothesis h such that $P_{x\sim\mathcal{D}}(h(x) \neq h^*(x)) \geq f^{-1}(ct)$. Hence the expected cost incurred by ALG in round $t-1$ is at least $\Delta f^{-1}(ct)$. It follows that

$$\mathbb{E}\, C_t = \sum_{i=1}^{T} \mathbb{E}\, c_t = \sum_{i=1}^{T} \Delta f^{-1}(ct) = \Omega(\sum_{i=1}^{T} f^{-1}(ct)),$$

as claimed. ■

Intuitively, this bound holds because in the PAC setting, labels are free, so PAC is easier than selective sampling. Using the so-called "fundamental theorem of statistical learning" and the fact that the VC-dimension of the class of threshold hypotheses is 1, (see Shalev-Shwartz and Ben-David (2014), Section 6), this implies a weaker version of Theorem 4; namely, that the $\Omega(\log T)$ bound holds in the worst case over distributions.

On the other hand, PAC upper bounds do not in general imply any upper bound for the label complexity of the CAL algorithm. In particular, there are combinations of hypothesis classes and data distributions that can be PAC-learned efficiently, but for which CAL must always query *all* labels. Consider learning (arbitrary) linear separators in $\mathbb{R}^2$ against the following distribution for X_t: Z_t is drawn uniformly from $[-5,5]$, and $X_t = (Z_t, 1/Z_t)$. The true hypothesis is that instances with $x < 0$ are negative, and those with $x \geq 0$ are positive. It is easy to see that no X_t is ever determined, so the CAL algorithm will ask for every label; see Figure 1 for an illustration. However, the VC-dimension of arbitrary linear separators in $\mathbb{R}^2$ is 3, so the fundamental theorem of statistical learning (Shalev-Shwartz and Ben-David, 2014, Theorem 6.8) implies a PAC sample complexity of $O(\frac{d\log(1/\epsilon)+\log(1/\delta)}{\epsilon})$.

We can generalize our analysis of the upper bound as follows. Define

$$\mathcal{B}_t = \{X_i | i \in [t], (\{(X_1,Y_1),\dots,(X_t,Y_t)\} \setminus (X_i,Y_i)) \not\twoheadrightarrow X_i\}.$$

The points $X_t \in \mathcal{B}_t$ are called *undetermined observations*; they are the instances observed by the algorithm whose labels cannot be deduced from the other labeled instances that the algorithm has seen up until time t. In the proof of Theorem 3 above, $\mathcal{B}_t \subseteq \{A_t^k, B_t^k\}$.[3]

Proposition 7 *The cost incurred by A_{CAL} is $C_T = \sum_{i=1}^{T} Pr(X_t \in \mathcal{B}_t)$.*

It follows that

Corollary 8 *The expected cost incurred by A_{CAL} is $\mathbb{E}\, C_T = \sum_{i=1}^{T} \mathbb{E}\, |\mathcal{B}_t|/t$, where $|\mathcal{B}_t|$ denotes the cardinality of $\mathcal{B}_t$.*

[3] While it is typically the case that $\mathcal{B}_t = \{A_t^k, B_t^k\}$, if $X_i \geq k$ (resp. $X_i < k$) for all $i < t$, then $\mathcal{B}_t = \{B_t^k\}$ (resp. $\mathcal{B}_t = \{A_t^k\}$).

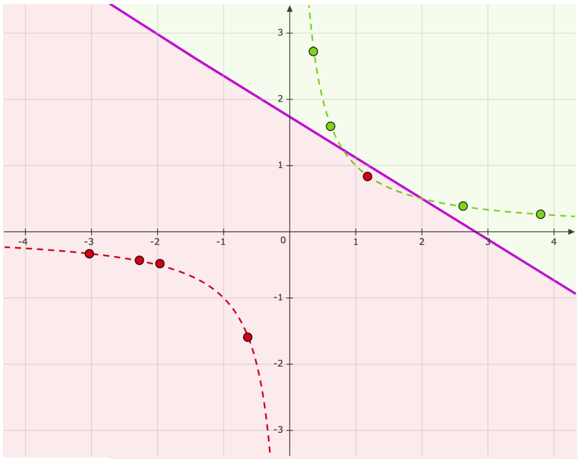


Figure 1: Separating X_t from other instances with the same label. The red (resp., green) dashed line is the support of the distribution of the negative (resp. positive) instances. The dots represent instances X_t. All instances on the green dashed line have positive labels, however, for any such instance, a linear separator can be found (the purple line) which classifies it as negative, but is consistent with the labels of all other instances.

The proofs of Proposition 7 and Corollary 8 are straightforward generalizations of the proof of Theorem 3.

These expressions relating expected cost and the number of undetermined observations are straightforward, and not entirely new (a similar approach can be found in the proof of Theorem 21 in (El-Yaniv and Wiener, 2012)). The difficult task is to find the probabilities $Pr(X_t \in \mathcal{B}_t)$ or the expectations $\mathbb{E}\,|\mathcal{B}_t|$. However, in the simple cases we study, the expectations $\mathbb{E}\,|\mathcal{B}_t|$ can be found. The proof sketch of Theorem 3 provides an example; for thresholds, $|\mathcal{B}_t| \leq 2$ for all t, so $\mathbb{E}\, C_T \sim 2 \log T$.

For the case of linear separators and the uniform distribution Unif S^d on the unit hypersphere, there is a well-known tight lower bound on PAC learning sample complexity; specifically (Long, 1995, Theorem 1) shows that $m_{\mathcal{H},\mathsf{Unif}\, S^d}(\epsilon, \delta) = \Omega(\frac{d}{\epsilon} + \frac{1}{\epsilon} \log \frac{1}{\delta})$.

Thus, we can take $f(\epsilon, \delta) = \frac{d}{\epsilon} + \frac{1}{\epsilon} \log \frac{1}{\delta}$ in our general lower bound (Proposition 6). Doing so yields the following lower bound.

Proposition 9 *If $\mathcal{X} = \mathbb{R}^d$ and the X_t are drawn i.i.d. from the uniform distribution on S^d, then all algorithms for selective sampling incur expected cost $\Omega(d \log T)$.*

Proof Taking $f(\epsilon, \delta) = \frac{d}{\epsilon} + \frac{1}{\epsilon} \log \frac{1}{\delta}$ in Proposition 6, we have that $f'(\epsilon) = \frac{d + \log 1/\Delta}{\epsilon}$. Hence $f'^{-1}(x) = \frac{d + \log 1/\Delta}{x}$, and Proposition 6 says that $\mathbb{E}\, C_T \in \Omega(\sum_{t=1}^{T} \frac{d + \log 1/\Delta}{ct})$. Simplifying,

$$\mathbb{E}\, C_T \in \Omega\left(\sum_{t=1}^{T} \frac{d + \log 1/\Delta}{ct}\right) = \Omega\left(\frac{d + \log 1/\Delta}{c} \log T\right) = \Omega(d \log T).$$

■

Our main result shows that A_{CAL} nearly matches this lower bound.

Theorem 1 *If $\mathcal{X} = \mathbb{R}^d$, the X_t are drawn uniformly at random from the unit hypersphere S^d, and the corresponding labels are consistent with some linear threshold f_w, A_{CAL} incurs expected cost $\Theta(d^{3/2} \log T)$. Alternatively, after requesting $\Theta(d^{3/2} \log(1/\epsilon))$ labels in expectation (over the training data $\{X_t\}_t$), A_{CAL} classifies $X \sim S^d$ (correctly) with probability at least $1 - \epsilon$.*

We defer the proof to Section 5.

3. Related Work

The simple algorithm A_{CAL} that we study is often called the CAL algorithm, after Cohn, Atlas, and Ladner, who first described it (Cohn et al., 1994). Note that the term "selective sampling" was coined by Cohn, Atlas, and Ladner specifically to refer to the CAL algorithm, but most later papers use the term as we do here (see, e.g., (Freund et al., 1997; Dasgupta et al., 2005; Balcan and Long, 2013)). Query By Committee (QBC), the first selective sampling algorithm to achieve a given error with exponentially fewer labels than its supervised counterpart, was analyzed in (Freund et al., 1997) for the same hypothesis class and instance distribution that we consider in this paper, namely, linear separators through the origin and the uniform distribution on the unit hypersphere. However, the computational complexity of QBC's t^{th} update step scales (polynomially) with t.

Several more recent algorithms improve on QBC in the sense that the computational complexity of these algorithms' t^{th} update step is independent of t. The Active Modified Perceptron (AMP) algorithm (Dasgupta et al., 2005) is a variation on the classical Perceptron algorithm (Rosenblatt, 1958). It maintains a hypothesis V_t and, if a label is inconsistent with V_t, updates V_t in the direction of $Y_t X_t$. It maintains and adaptively shrinks a margin threshold S_t and requests a label when $|X_t \cdot V_t| \leq S_t$. Another set of "margin-based" algorithms were presented in (Balcan et al., 2007). Like the AMP algorithm, these algorithms request labels when a margin condition $|X \cdot V_k| < b_k$ is met. However, the thresholds b_k in each phase k are fixed up front. It is worth noting that one of these algorithms (Procedure 2 with parameters as in Theorem 1) actually makes no mistakes with high probability. However, this algorithm does sometimes query points whose labels are determined by those of previously queried points, so it is not an implementation of CAL. All of these algorithms have cost bounds that depend on T as $\log T \log\log T$; see Appendix A in the supplementary material for details. These bounds typically have optimal linear or log-linear dependence on d. In contrast, we show that A_{CAL} has expected label complexity that depends optimally on T, specifically, $O(d^{3/2} \log T)$. If the dimension is a constant, our results show that A_{CAL} will eventually have the best performance. A_{CAL} has other advantages as well; most importantly, it is extremely simple to describe, and its description is independent of the details of the instance space, hypothesis class, and distribution. All the other algorithms discussed rely on the notion of "margin", which depends on the instance space. This does not mean that A_{CAL} will have similar guarantees in other settings; only that it is clear how to generalize to these settings. It is worth noting that the margin-based algorithms in (Balcan et al., 2007) have also been shown to apply to learning linear separators through the origin for a more general class of distributions, namely isotropic log-concave distributions. It would be interesting to see if we could extend our analysis of CAL to these distributions.

Removing the assumption that $Y_t = h^*(X_t)$ for some $h \in \mathcal{H}$ leads to the more general "agnostic" case. This case is also well studied; see (Dasgupta et al., 2007; Balcan et al., 2009; Hanneke, 2007). The strategy A^2 described in (Balcan et al., 2009) is conceptually very similar to A_{CAL}. It labels an instance whenever two surviving hypotheses disagree on that instance. However, rather than discarding a hypothesis whenever a label inconsistent with it is observed (which could result in discarding the best hypotheses), A^2 discards a hypothesis when it is confident that it is worse than some other hypothesis. This is a nice alternative to A_{CAL} if labels are imperfect or noisy. In principle, our techniques could be used to analyze A^2 and other algorithms for the agnostic case, although this seems much harder than in the realizable case.

4. Computational Complexity

It is evident that A_{CAL} runs in $O(1)$ time per instance X_t, independent of t, in the setting of Theorem 3. A_{CAL} simply maintains A_t^k and B_t^k as in the proof of Theorem 3, and asks for a label if $X_t \in (A_t^k, B_t^k)$.

By contrast, it is not at all obvious that A_{CAL} runs in expected per-instance time independent of t in the setting of Theorem 1. Nevertheless, A_{CAL} can be implemented to run in amortized expected per-instance time $O(poly(d))$ independent of t, as Proposition 13 below shows. Our implementation is given in Algorithm 2 (CAL). CAL is a bona fide *implementation of* A_{CAL}; that is, if X_t is determined, then CAL outputs the inferred label; otherwise CAL asks for the label. It depends on checks of the form $x \in \mathsf{cone}(S)$, where $x \in \mathbb{R}^d$, S is a finite set of points in $\mathbb{R}^d$, and $\mathsf{cone}(S)$ denotes the set of all nonnegative linear combinations of points in S, or alternatively, the cone spanned by S. These checks can be implemented efficiently using linear programming. Specifically, the linear program P given by

$$\begin{aligned} &\min c^T y \text{ s.t.} \\ &Ay = x, \\ &y \geq 0, \end{aligned} \tag{1}$$

where c is the zero vector and A is the matrix whose columns are the points in S, is feasible if and only if $x \in \mathsf{cone}(S)$. Whether P is feasible can be checked in time polynomial in d and $|S|$.

To explain CAL, we need to define the notion of *equivalent positive sample*. For $i \in [t]$, define $Z_i = Y_i X_i$. It is easy to see that, for all $h \in \mathcal{H}$ and $i \in [t]$, $h(X_i) = Y_i$ iff $h(Z_i) = 1$. Thus, if we define the *equivalent positive sample* $\mathcal{Z}_t = \{(Z_1, 1), \ldots, (Z_{t-1}, 1)\}$, we have $\mathcal{H}_{\mathcal{S}_t} = \mathcal{H}_{\mathcal{Z}_t}$; that is, $\mathcal{S}_t$ and $\mathcal{Z}_t$ are compatible with the same set of hypotheses. It follows that

Claim 1 *$\mathcal{S}_t \twoheadrightarrow X_t$ iff $\mathcal{Z}_t \twoheadrightarrow Z_t$.*

This fact permits us to think of $\mathcal{Z}_t$ as equivalent to the original labeled data $\mathcal{S}_t$; hence the name. The point of introducing the equivalent positive sample is that there is a simple characterization of when $\mathcal{Z}_t \twoheadrightarrow X_t$.

Lemma 10 *$\mathcal{Z}_t \twoheadrightarrow X_t$ iff either (1) $X_t \in \mathsf{cone}(\{Z_1, \ldots, Z_{t-1}\})$ or (2) $-X_t \in \mathsf{cone}(\{Z_1, \ldots, Z_{t-1}\})$. Moreover, if (1) holds, then the inferred label is $Y_t = 1$; if (2) holds, then it is $Y_t = -1$.*

Proof Suppose $\mathcal{Z}_t \twoheadrightarrow X_t$. Since either $Z_t = X_t$ or $Z_t = -X_t$, by Lemma 17, $\mathcal{Z}_t \twoheadrightarrow Z_t$. Hence $Z_t \in \mathsf{cone}(\{Z_1, \ldots, Z_{t-1}\})$ (with inferred label 1). If $X_t = Z_t$, then X_t is also in the cone, and its inferred label is 1. If $-X_t = Z_t$, then $-X_t$ is in the cone, and its inferred label is 1; so the inferred label of X_t is -1. Conversely, suppose $X_t \in \mathsf{cone}(\{Z_1, \ldots, Z_{t-1}\})$. (The case $-X_t \in \mathsf{cone}(\{Z_1, \ldots, Z_{t-1}\})$ is similar.) Then $\mathcal{Z}_t \twoheadrightarrow X_t$, and the inferred label of X_t is 1, by the argument for the "only if" direction of Lemma 17. ■

CAL works by maintaining a set $\mathcal{B}$ such that $\mathsf{cone}(\mathcal{B}) = \mathsf{cone}(\{Z_1, \dots, Z_{t-1}\})$, as the following lemma shows.

Lemma 11 *At the beginning of each iteration t of CAL,* $\mathsf{cone}(\mathcal{B}) = \mathsf{cone}(\{Z_1, \dots, Z_{t-1}\})$.

Proof We proceed by induction on t. Clearly the result holds for $t = 1$. Suppose it is true for $t = k$. Let $\mathcal{B}_k$ and $\mathcal{B}_{k+1}$ be the values of $\mathcal{B}$ at the beginning of iterations k and $k+1$ respectively. There are three cases corresponding to the three conditional branches. For the "if" and "else if" cases, if $X_k \in \mathsf{cone}(\mathcal{B}_k)$ or $-X_k \in \mathsf{cone}(\mathcal{B}_k)$, then $Z_k \in \mathsf{cone}(\mathcal{B}) = \mathsf{cone}(\{Z_1, \dots, Z_{k-1}\})$. Since $\mathcal{B}$ is not changed in the "if" or "else if" cases, we have $\mathcal{B}_{k+1} = \mathcal{B}_k = \mathsf{cone}(\{Z_1, \dots, Z_{k-1}\}) = \mathsf{cone}(\{Z_1, \dots, Z_k\})$. For the "else" case, let R be the set of vectors removed by RRP. Since R only contains vectors that are in the cone spanned by the remaining vectors, we have $\mathsf{cone}(\mathcal{B}_k \setminus R) = \mathsf{cone}(\mathcal{B}_k)$. Then we have $\mathcal{B}_{k+1} = \mathcal{B}_k \setminus R \cup \{Z_k\}$, so

$$\mathsf{cone}(\mathcal{B}_{k+1}) = \mathsf{cone}(\mathsf{cone}((\mathcal{B}_k \backslash R) \cup Z_k)) = \mathsf{cone}(\mathsf{cone}(\{Z_1, \dots, Z_{k-1}\} \cup Z_k))) = \mathsf{cone}(\{Z_1, \dots, Z_k\}).$$

■

Since $\mathsf{cone}(\mathcal{B}) = \mathsf{cone}(\{Z_1, \dots, Z_{t-1}\})$, it is immediate from Lemma 10 that the action taken by CAL in each iteration corresponds to the action chosen by A_{CAL}, so CAL is an implementation of A_{CAL}. This is formalized in the following proposition.

Proposition 12 *CAL is an implementation of A_{CAL}.*

Moreover, CAL runs in amortized expected time per instance independent of t, as the following proposition shows. This is a consequence of Lemma 15, which implies that the expected size of $\mathcal{B}$ is bounded by a constant independent of t; see Section 5 for details.

Proposition 13 *CAL runs in amortized expected time $O(d^{3.6})$ per instance X_t.*

Proof First we need a lemma concerning the subroutine RRP.

Lemma 14 *If RRP is called in iteration t of CAL, then immediately after RRP returns, $\mathcal{B} = \{Z_i \mid i \in [t], \mathcal{Z}_i \not\twoheadrightarrow Z_i\}$.*

Proof Lemma 11 states that at the start of iteration t, $\mathsf{cone}(\mathcal{B}) = \mathsf{cone}(\{Z_1, \dots, Z_{t-1}\})$. It is easy to see that just before RRP is called, since Z_t has been added if it is not in the cone, $\mathsf{cone}(\mathcal{B}) = \mathsf{cone}(\{Z_1, \dots, Z_t\})$. Each iteration of RRP preserves this property. Hence after RRP, $\mathcal{B} \supseteq \{Z_i \mid i \in [t], \mathcal{Z}_i \not\twoheadrightarrow Z_i\}$; that is, $\mathcal{B}$ is a superset of the edges of the cone. However, after RRP, all points Z_i that are determined by the other points (that is, $\mathcal{Z}_i \twoheadrightarrow Z_i$) have been removed. Hence $\mathcal{B} = \{Z_i \mid i \in [t], \mathcal{Z}_i \not\twoheadrightarrow Z_i\}$, as claimed. ■

As in the proof of Theorem 1, without loss of generality, suppose that the correct hypothesis is $h^* = [1, 0, \dots, 0]$. Hence, as in that theorem, the Z_i are i.i.d. with distribution Unif S^d_+.

Let $T_1, T_2, \dots$ be the iterations of CAL where RRP is called. By Lemma 14, at the end of each iteration T_i, we have $\mathcal{B} = \{Z_i \mid i \in [T_i], \mathcal{Z}_i \not\twoheadrightarrow i\}$. Furthermore, RRP is run every time $\mathcal{B}$ doubles in size. So for all iterations $T_i < t \leq T_{i+1}$, letting $\mathcal{B}_t$ be the value of $\mathcal{B}$ at the beginning of iteration t, we have $|\mathcal{B}_t| \leq 2|\{Z_i \mid i \in [T_i], \mathcal{Z}_i \not\twoheadrightarrow i\}|$. Now $\{Z_i \mid i \in [T_i], \mathcal{Z}_i \not\twoheadrightarrow i\}$ are the points corresponding to edges of $\mathsf{cone}(\{Z_1, \dots, Z_{T_i}\})$. So Lemma 15 (see Section 5) implies that $\lim_{d\to\infty} \lim_{t\to\infty} \mathbb{E}\,|\mathcal{B}_t| = 2kd^{3/2}$ for an absolute constant k. Thus for sufficiently large d, for all $T_i < t \leq T_{i+1}$, $\mathbb{E}\,|\mathcal{B}_t| = \Theta(d^{3/2})$.

Each iteration t, $T_i < t < T_{i+1}$, consists of two checks of the form $x \in \mathsf{cone}(\mathcal{B})$, plus some operations that are obviously $O(1)$. The iteration T_{i+1} consists of these two checks, along with a call to RRP, which consists of $|\mathcal{B}_{T_i}|$ checks, plus some $O(1)$ operations. Suppose that $T_{i+1} - T_i = m$. Then the expected running time $\mathbb{E}\,\tau$ of the m iterations $t = T_i{+}1, \dots, T_{i+1}$ is $O(m){+}2mc{+}|\mathcal{B}_{T_i}|c$, where c is the maximum expected running time of any of the checks. Since $\mathcal{B}$ doubles in size from T_i to T_{i+1}, but increases by at most 1 in any iteration, $|\mathcal{B}_{t_i}| < 2m$. Hence $\mathbb{E}\,\tau \leq O(m) + 4mc$.

The checks $x \in \mathsf{cone}(\mathcal{B})$ are implemented as checking feasibility of the linear program (1). This linear program has d constraints and $|\mathcal{B}_t|$ variables. It is known that linear programs of the form 1 with no redundant constraints can be solved in time $O(n^\omega \log(n))$, where n is the number of variables, and $\omega < 2.39$ is the exponent of matrix multiplication (van den Brand, 2020). The constraints of the linear program (1) are almost surely not redundant, because each constraint is a random equality constraint. Hence (1) can be solved (and its feasibility checked) in $O((d^{3/2})^{2.39}) \subset O(d^{3.6})$ time. This means $c \in O(d^{3.6})$. Hence $\mathbb{E}\,\tau = mO(d^{3.6})$. Dividing both sides by m, this implies that the amortized expected running time per iteration $\mathbb{E}\,\tau/m$ is $O(d^{3.6})$, as claimed. ■

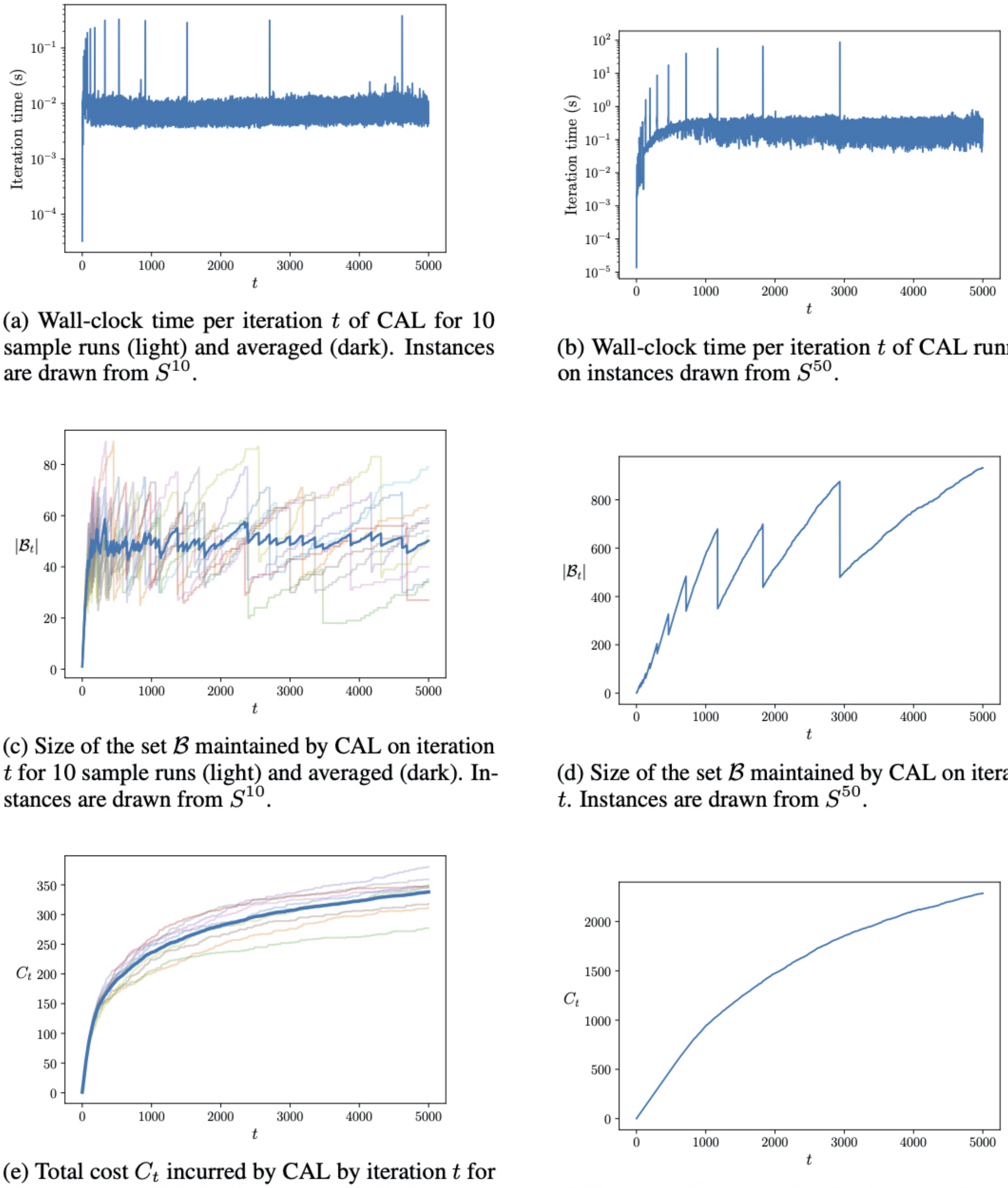
To demonstrate that CAL has not only good asymptotic runtime and cost bounds, but also good runtime and cost performance for realistic time horizons and values of d, we implemented CAL and tested it on simulated instances. See the supplementary material for the code and collected data. Figure 2 shows data from running CAL on instances drawn from Unif S^{10} (left) and Unif S^{50} (right). The top column shows the time per instance in seconds. For $d = 10$, the average time per instance is 0.0082 seconds; for $d = 50$, it is 0.32 seconds. Importantly, the time per instance is roughly constant in t for $t > 100$ ($d = 10$) and $t > 1000$ ($d = 50$). The "spikes" are calls to RRP. The middle column shows the size of $\mathcal{B}_t$, which levels off in lockstep with the time per instance, and does not vary too much between runs. The last column shows the cost C_t, which is just the total number of label requests.

```
Input: $\mathcal{B}$, a finite set of points in $\mathbb{R}^d$.
for $Z \in \mathcal{B}$ do
    if $Z \in \mathsf{cone}(\mathcal{B} - \{Z\})$ then
        $\mathcal{B} \leftarrow \mathcal{B} \setminus \{Z\}$
    end
end
return $\mathcal{B}$
```

Algorithm 1: RRP

5. Proof of Theorem 1

To prove Theorem 1, we need the following lemma, which is essentially implicit in (Kabluchko, 2020). Let S^d_+ be the upper half-sphere $S^d_+ = \{x \in S^d \mid x[1] > 0\}$.



(a) Wall-clock time per iteration t of CAL for 10 sample runs (light) and averaged (dark). Instances are drawn from S^{10}.

(b) Wall-clock time per iteration t of CAL running on instances drawn from S^{50}.

(c) Size of the set $\mathcal{B}$ maintained by CAL on iteration t for 10 sample runs (light) and averaged (dark). Instances are drawn from S^{10}.

(d) Size of the set $\mathcal{B}$ maintained by CAL on iteration t. Instances are drawn from S^{50}.

(e) Total cost C_t incurred by CAL by iteration t for 10 sample runs (light) and averaged (dark). Instances are drawn from S^{10}.

(f) Total cost C_t incurred by CAL by iteration t. Instances are drawn from S^{50}.

Figure 2: Data from runs of CAL on instances from Unif S^d.

Lemma 15 *Let $Z_1, \ldots, Z_t$ be drawn i.i.d. from the uniform distribution on S^d_+. Let $C_t = \mathsf{cone}(\{Z_1, \ldots, Z_t\})$. Let N_t be the number of edges of C_t. Then*

$$\lim_{d\to\infty} \lim_{t\to\infty} \mathbb{E}\, N_t/d^{3/2} = \sqrt{\frac{2\pi}{3}}.$$

```
$\mathcal{B} \leftarrow \emptyset$
$N = 0$
for $t = 1, 2, \ldots$ do
    if $X_t \in \mathsf{cone}(\mathcal{B})$ then
        Classify $X_t$ as positive.
    else if $-X_t \in \mathsf{cone}(\mathcal{B})$ then
        Classify $X_t$ as negative.
    else
        Ask for the label $Y_t$.
        Compute $Z_t = Y_t X_t$.
        $\mathcal{B} \leftarrow \mathcal{B} \cup \{Z_t\}$
    if $|\mathcal{B}| \geq 2N$ then
        $\mathcal{B} \leftarrow$ RRP$(\mathcal{B})$  $N \leftarrow |\mathcal{B}|$.
    end
end
```

Algorithm 2: CAL

Proof Equivalently, N_t is the number of vertices of the random *spherical polytope* $C_t \cap S^d_+$ (see Section 2 of (Kabluchko, 2020)). Several recent papers have investigated the properties of this polytope (Bárány et al., 2017; Kabluchko et al., 2019; Kabluchko, 2020). In particular, the following theorem was shown regarding $f_k(C_t \cap S^d_+)$, the number of k-faces of $C_t \cap S^d_+$.

Before stating the theorem, we need some notation. Define $A[n, m]$ as follows: Given a polynomial $P(x)$ (a formal power series in positive and negative powers of x), let $[x^k]P(x)$ be the coefficient of x^k in $P(x)$. Let

$$Q_n(x) = \Pi_{j \in \{1,\ldots,n-1\}, j \not\equiv n (\text{mod } 2)} (1 + j^2 x^2).$$

Then

$$A[n, m] = \begin{cases} [x^m] Q_n(x) & \text{if m is even,} \\ [x^m](\tanh(\frac{\pi}{2x}) \cdot Q_n(x)), & \text{if m is odd and n is even,} \\ [x^m](\coth(\frac{\pi}{2x}) \cdot Q_n(x)), & \text{if m is odd and n is odd.} \end{cases}$$

We remark that although the appearance of $\tanh$ and $\coth$ in these formulas is a little surprising, the proof does not use any special properties of these functions. Rather, the second and the third cases follow from the first via the Dehn-Sommerville equations relating the numbers of faces of different dimensions of a simplicial polytope, and the $\tanh$ and $\coth$ factors conveniently summarize the coefficients that arise when these equations are solved. See (Kabluchko, 2020) for more details.

Proposition 16 (Theorem 2.1 of (Kabluchko, 2020))

$$\lim_{t \to \infty} \mathbb{E} f_k(C_t \cap S^d_+) = \frac{\pi^{k+1}}{(k+1)!} A[d, k+1].$$

We are interested in the special case of Proposition 16 where $k = 0$, corresponding to vertices of $C_t \cap S^d_+$. Recall that $\mathbb{E} N_t = \mathbb{E} f_0(C^t \cap S^d_+)$. By Proposition 16 with $k = 0$, we have

$$\lim_{t \to \infty} N_t = \lim_{t \to \infty} \mathbb{E} f_0(C^t \cap S^d_+) = \pi A[d, 1]. \tag{2}$$

Zakhar Kabluchko has proved the following claim about the asymptotic behavior of the $A[n, m]$ (private correspondence, Jan 31, 2021). [4]

Claim 2 *For all $m \geq 1$,*

$$\lim_{n\to\infty} \frac{A[n,m]}{n^{3m/2}} = \frac{1}{6^{m/2}\Gamma((m/2)+1)},$$

where $\Gamma(x) = \int_0^\infty y^{x-1}e^{-y}dy$ denotes the gamma function. In the special case of $m = 1$, Claim 2 becomes

$$\lim_{n\to\infty} \frac{A[n,1]}{n^{3/2}} = \frac{1}{\sqrt{6}\sqrt{\pi}/2} = \sqrt{\frac{2}{3\pi}}. \tag{3}$$

Hence dividing both sides of Equation (2) by $d^{3/2}$ and taking the limit as $d \to \infty$ yields

$$\lim_{d\to\infty}\lim_{t\to\infty} \mathbb{E}\, N_t/d^{3/2} = \pi\sqrt{\frac{2}{3\pi}} = \sqrt{\frac{2\pi}{3}},$$

as claimed. ■

Theorem 1 *If $\mathcal{X} = \mathbb{R}^d$, the X_t are drawn uniformly at random from the unit hypersphere S^d, and the corresponding labels are consistent with some linear threshold f_w, A_{CAL} incurs expected cost $\Theta(d^{3/2}\log T)$. Alternatively, after requesting $\Theta(d^{3/2}\log(1/\epsilon))$ labels in expectation (over the training data $\{X_t\}_t$), A_{CAL} classifies $X \sim S^d$ (correctly) with probability at least $1-\epsilon$.*

Proof We first prove the claim for a different distribution, namely $\mathsf{Unif}\, S^d$. Since $\mathsf{Unif}\, S^d$ is spherically symmetric, the expected cost is the same for all hypotheses. Without loss of generality, suppose the correct hypothesis is $h^* = [1, 0, \ldots, 0]$. That is, $Y_t = 1$ if $X_t[1] > 0$ and $Y_t = 0$ if $X_t[1] < 0$. (We will neglect the zero-probability case $X_t[1] = 0$.)

Since A_{CAL} guesses only when X_t is determined, it does not make any wrong classifications. Thus, the cost C_T it incurs is simply the number of times it queries:

$$\mathbb{E}\, C_T = \sum_{i=1}^{T} \mathbb{E}\, c_t = \sum_{i=1}^{T} \mathbb{E}\, \mathbb{1}_{\{\hat{Y}_t=0\}} = \sum_{i=1}^{T} P(\hat{Y}_t = 0).$$

Thus, the goal is to bound $P(\hat{Y}_t = 0)$, the probability that the naive algorithm queries X_t. Recall that the naive algorithm queries X_t exactly when X_t is not determined, that is, when $\mathcal{S}_t := \{(X_1, Y_1), \ldots, (X_{t-1}, Y_{t-1})\} \not\twoheadrightarrow X_t$, and that $\mathcal{S}_t \twoheadrightarrow X_t$ iff $\mathcal{Z}_t \twoheadrightarrow Z_t$. The following lemma characterizes when $\mathcal{Z}_t \twoheadrightarrow Z_t$.

Lemma 17 *$\mathcal{Z}_t \twoheadrightarrow Z_t$ iff $Z_t \in \mathsf{cone}(\{Z_1, \ldots, Z_{t-1}\})$.*

Proof The "only if" direction is straightforward. Indeed, suppose there are nonnegative numbers a_j such that $Z_i = a_1 Z_1 + \cdots + a_{t-1}Z_{t-1}$. It suffices to show that for all $h = \mathbb{1}_{\{x\cdot w \geq 0\}} \in \mathcal{H}_{\mathcal{Z}_i}$, we have $h(Z_i) = 1$, that is, $Z_i \cdot w \geq 0$. We have $Z_i \cdot w = a_1(Z_1\dot{w}) + \cdots + a_{t-1}(Z_{t-1}\dot{w})$. But since $h \in \mathcal{H}_{\mathcal{Z}_i}$, $Z_i\dot{w} \geq 0$ for all $i \in [t-1]$. Hence $Z_t \cdot w \geq 0$. The "if" direction follows from the well-known Farkas Lemma (see, for example, (Rader, 2010, Lemma 6.1)), which states that given a $m \times n$ real matrix A and $c \in R^n$, exactly one of the following two systems has a solution:

[4] Z. Kabluchko will make a preprint with the proof of this claim available before the end of the ALT'22 review period.

1. $Ad \leq 0$ and $c^T d > 0$
2. $A^T y = c$ and $y \geq 0$.

Let A be the $d \times (t-1)$ matrix whose columns are $Z_1, \ldots, Z_{t-1}$, and let $c = Z_i$. The Farkas Lemma then implies that if Z_t is not a nonnegative linear combination of $Z_1, \ldots, Z_{t-1}$ (that is, the system 2 has no solutions) then there is a vector d such that $Ad \leq 0$ and $c^T d > 0$. But then the hypothesis $h = \mathbb{1}_{\{x \cdot (-d) \geq 0\}}$ satisfies $h(Z_1) = \cdots = h(Z_{t-1}) = 1$, but $h(Z_t) = 0$. Since $h^*(Z_1) = \cdots = h^*(Z_t) = 1$, $\mathcal{Z}_i$ does not determine Z_i. ■

Since the Z_i for $i \in [t]$ are i.i.d., this lemma implies the following claim.

Claim 3 *For all $i \in [t]$ $\mathcal{Z}_i \twoheadrightarrow Z_i$ iff $\mathcal{Z}_i \in \mathsf{cone}(\{Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_t\})$.*

It follows from our general upper bound (Corollary 8) that

$$\begin{aligned}
\mathbb{E}\, C_T &= \sum_{t=1}^{T} \mathbb{E}\, |\mathcal{B}_t|/t \\
&= \sum_{t=1}^{T} \mathbb{E}\, |\{S_i \mid i \in [t], \mathcal{S}_i \not\twoheadrightarrow S_i\}| \\
&= \sum_{t=1}^{T} \mathbb{E}\, |\{Z_i \mid i \in [t], \mathcal{Z}_i \not\twoheadrightarrow Z_i\}|,
\end{aligned}$$

where the last equality follows from Claim 1.

It is easy to see that the Z_i are i.i.d. with distribution $\mathsf{Unif}\, S^d_+$. Let $C_t = \mathsf{cone}(\{Z_1, \ldots, Z_t\})$. By Claim 3, the random variable $N_t := |\{Z_i \mid i \in [t], \mathcal{Z}_i \not\twoheadrightarrow Z_i\}|$ is the number of edges of C_t (each edge corresponds to an instance Z_i not contained in the cone of the other instances, and vice versa). By Lemma 15, we have

$$\lim_{d \to \infty} \lim_{t \to \infty} \mathbb{E}\, N_t / d^{3/2} = \pi \sqrt{\frac{2}{3\pi}} = \sqrt{\frac{2\pi}{3}}.$$

That is, for sufficiently large d, we have $\mathbb{E}\, N_T \in \Theta(d^{3/2})$. It follows that

$$\mathbb{E}\, C_T = \sum_{t=1}^{T} \mathbb{E}\, N_t / T \in \Theta(d^{3/2} \log T),$$

as claimed. The “Alternatively” claim follows in exactly the same way as for Theorem 3.

Finally, we generalize to the case of multivariate Gaussians. Suppose $\mathcal{D}$ is a multivariate Gaussian with mean 0 and covariance Σ. For simplicity, consider the case where Σ is invertible; if Σ is not invertible (so that $\mathcal{D}$ is supported on only a subspace of $\mathbb{R}^d$), the argument is similar, except that the definition of $T(x)$ must be modified to map the X_t to the unit hypersphere in the subspace corresponding to the support of $\mathcal{D}$.

Let $T(x) = \frac{\Sigma^{-1} X_t}{\|\Sigma^{-1} X_t\|}$. Take $X'_t = T(X_t)$. Further take the target hypothesis to be $h' = T(h)$. It is easy to see that for all $x \in X$ and all $g \in \mathcal{H}$, $T(g)(x) = g(T(x))$. Hence the following are equivalent:

- $\{(X_1, h(X_1)), \ldots, (X_{t-1}, h(X_{t-1}))\} \twoheadrightarrow X_t$,
- $\{(X'_1, h'(X_1)), \ldots, (X'_{t-1}, h'(X'_{t-1}))\} \twoheadrightarrow X'_t$.

Thus, A_{CAL} asks for the label of X_t given target hypothesis h iff A_{CAL} asks for the label of X'_t given target hypothesis h'. Since the cost incurred by A_{CAL} by time t is simply the number of labels requested on or before round t, the cost C_t that A_{CAL} incurs on hypothesis h and arrivals X_t is the same as the cost that A_{CAL} incurs on hypothesis h' and arrivals X'_t. But it is easy to see that the X'_t are i.i.d. with distribution $\mathsf{Unif}\, S^d$. Thus the previous argument applies and $EC_t = \Theta(d^{3/2} \log T)$. ■

References

M.-F. Balcan and P. Long. Active and passive learning of linear separators under log-concave distributions. In *26 Annual Conference on Learning Theory (COLT 2013)*, pages 288–316, 2013.

M.-F. Balcan, A. Broder, and T. Zhang. Margin based active learning. In *20th Annual Conference on Learning Theory (COLT 2007)*, pages 35–50, 2007.

M.-F. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. *Journal of Computer and System Sciences*, 75(1):78–89, 2009.

I. Bárány, D. Hug, M. Reitzner, and R. Schneider. Random points in halfspheres. *Random Structures & Algorithms*, 50(1):3–22, 2017.

D. Cohn, L. Atlas, and R. Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.

S. Dasgupta, A. T. Kalai, and C. Monteleoni. Analysis of perceptron-based active learning. In *International Conference on Computational Learning Theory*, pages 249–263, 2005.

S. Dasgupta, D. J. Hsu, and C. Monteleoni. A general agnostic active learning algorithm. In *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, pages 353–360, 2007.

R. El-Yaniv and Y. Wiener. Active learning via perfect selective classification. *Journal of Machine Learning Research*, 13(9), 2012.

Y. Freund, H. S. Seung, E. Shamir, and N. Tishby. Selective sampling using the query by committee algorithm. *Machine Learning*, 28(2):133–168, 1997.

S. Hanneke. A bound on the label complexity of agnostic active learning. In *Proc. 24th International Conference on Machine learning*, pages 353–360, 2007.

Z. Kabluchko, A. Marynych, D. Temesvari, and C. Thäle. Cones generated by random points on half-spheres and convex hulls of poisson point processes. *Probability Theory and Related Fields*, 175(3):1021–1061, 2019.

Zakhar Kabluchko. Expected f-vector of the poisson zero polytope and random convex hulls in the half-sphere. *Mathematika*, 66(4):1028–1053, 2020.

P. M Long. On the sample complexity of PAC learning half-spaces against the uniform distribution. *IEEE Transactions on Neural Networks*, 6(6):1556–1559, 1995.

P. M. Long. An upper bound on the sample complexity of PAC learning halfspaces with respect to the uniform distribution. *Information Processing Letters*, 87(5):229–234, 2003.

D. J. Rader. *Deterministic operations research: models and methods in linear optimization*. John Wiley & Sons, 2010.

F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, New York, NY, 2014.

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

J. van den Brand. A deterministic linear program solver in current matrix multiplication time. In *Proc. qtth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 259–278, 2020.

Appendix A. Details of label complexities of previously proposed algorithms

In (Dasgupta et al., 2005, Theorem 3) it was proved that if the AMP algorithm is presented with

$$O(\frac{d}{\epsilon}\log(\frac{1}{\epsilon\delta})(\log d/\delta + \log\log\frac{1}{\epsilon}))$$

samples, with probability $1-\delta$, it will misclassify at most an ϵ fraction of the samples, and the same for labels requested. That is, the number of misclassifications and the number labels requested are both bounded by

$$O(d\log(\frac{1}{\epsilon\delta})(\log d/\delta + \log\log\frac{1}{\epsilon})).$$

It is easy to see that this implies that the cost C_T incurred by the AMP algorithm is

$$O(d\log T\log\frac{1}{\delta}(\log\frac{d}{\delta} + \log\log T).$$

This is almost the optimal $O(d\log T)$ cost, but is higher by multiplicative $\log\log T$ and $\log d$ factors. It is the $\log\log T$ factor that we show A_{CAL} improves on.

To compare things another way, after processing $T = O(d^{3/2}/\epsilon)$ instances, A_{CAL} makes zero mistakes, makes $\Theta(d^{3/2}\log\frac{d^{3/2}}{\epsilon})$ label requests in expectation, and incurs expected cost ϵ.

Moving to the "margin-based" algorithms presented in (Balcan et al., 2007), one of these algorithms (Margin-Based Active Learning; Procedure 2 of (Balcan et al., 2007) with parameters as in Theorem 2 of (Balcan et al., 2007)) makes just as few label queries, up to factors of $\log\log 1/\epsilon$. More precisely, the label query bound given in Theorem 2 simplifies to

$$O(\log(1/\epsilon)\sqrt{\log\log(1/\epsilon)})(d\log\log\log 1/\epsilon + \log\log\epsilon - \log\delta)/$$

This algorithm does make mistakes, and no mistake bound is given, so it is not possible to make a direct comparison with the AMP algorithm. Another algorithm of (Balcan et al., 2007) (Margin-Based Active Learning with parameters as in Theorem 1 of (Balcan et al., 2007)) makes no mistakes, but its label query bound depends on dimension as $d^{3/2}$ (that is, it has a $d^{3/2}$ factor). The label query bound given in Theorem 1 of (Balcan et al., 2007) simplifies to $O(\log(1/\epsilon)d^{1/2}(d\log d + \log(\log 1/\epsilon) - \log(1/\delta)))$. This is consistent with the analysis of this paper, where the algorithm which is *guaranteed* to make no mistakes with probability 1 has a label complexity depending on $\Theta(d^{3/2})$. The question of whether every algorithm which makes no mistakes with high probability in this setting has label complexity depending on d as $\Omega(d^{3/2})$ is still open.

Indeed, for X_t drawn from any isotropic log-concave distribution D, there are parameters (see Theorem 5 of (Balcan and Long, 2013)) such that Margin-based Active Learning has label complexity $O((d + \log(1/\delta) + \log\log 1/\epsilon)\log(1/\epsilon))$. Again, the expected label complexity of A_{CAL} improves on this by a $\log\log 1/\epsilon$ factor, although its dependence on d is worse.

Appendix B. Additional Proofs

Theorem 4 *In the setting of Theorem 3, if $\mathcal{D}$ is continuous, then for all selective sampling algorithms ALG, there exists a target hypothesis such that ALG has expected cost* $\Omega(\log T)$.

Proof It suffices to show an $\Omega(\log T)$ bound on expected cost in the modified setting of Proposition 6. It is convenient to prove a slightly stronger statement; rather than proving that all selective sampling algorithms have $\Omega(\log T)$ expected cost in the worst case over hypotheses, we prove that this is true if the hypothesis is drawn from the distribution $\mathcal{D}$. This is a stronger statement, because if the expectation over hypotheses is $\Omega(\log T)$, there must be some hypothesis that achieves $\Omega(\log T)$. Since the hypothesis and the instances are all drawn from $\mathcal{D}$, and the cost only depends on the relative ordering of instances versus the hypothesis, we can assume without loss of generality that $\mathcal{D} = \mathsf{Unif}\,[0,1]$.

In the modified setting, the optimal algorithm is easy to characterize. In the modified setting, no optimal algorithm ever asks for a label, since this is guaranteed to incur cost 1, and the algorithm will see the label anyway. Define A_t^k, B_t^k as in the proof of Theorem 3(1). It is easy to check that after seeing $X_1, \dots, X_{t-1}$, the posterior over hypotheses h is uniform on (A_t^k, B_t^k). Hence, the optimal algorithm classifies X_t as negative if $X_t \le \frac{A_t^k + B_t^k}{2}$ and classifies X_t as positive otherwise. It is also easy to check that the expected cost incurred by this strategy at time t is

$$\begin{aligned}
2\int_{A_t^k}^{(A_t^k+B_t^k)/2} Pr(x>h)dx &= 2\int_{A_t^k}^{(A_t^k+B_t^k)/2} (x-A_t^k)/(B_t^k-A_t^k)dx \\
&= \frac{2}{(B_t^k-A_t^k)}\frac{((B_t^k-A_t^k)/2)^2}{2} \\
&= (B_t^k-A_t^k)/4.
\end{aligned}$$

Define $M_t = B_t^k - A_t^k$. It remains to lower bound $\mathbb{E}\, M_t$. By the law of iterated expectations, $\mathbb{E}\, M_t \ge \inf_{x_1,x_2,\dots,x_{t-1}} E[M_t \mid X_1 = x_1, \dots, X_t = x_{t-1}]$. Fix arbitrary values $x_1, \dots, x_{t-1} \in [0,1]$. Let $x_{-1} = 0$ and $x_0 = 1$. Let $x^{(-1)} \ge x^{(0)} \ge \dots \ge x^{(t-1)}$ be the points $x_{-1}, x_0, \dots, x_{t-1}$ sorted in nondecreasing order. We have

$$\begin{aligned}
&\mathbb{E}\left[M_t \mid X_1 = x_1, \dots, X_{t-1} = x_{t-1}\right] \\
&= \sum_{-1\le i\le t-1} (x^{(i+1)} - x^{(i)}) P(x^{(i-1)} > B > x^{(i)}) \\
&= \sum_{-1\le i\le t-1} (x^{(i-1)} - x^{(i)})^2 \\
&\ge (t+1)(1/t)^2 \\
&\ge 1/t,
\end{aligned}$$

where the last inequality follows because $\sum_{-1\le i\le t-1}(x^{(i+1)} - x^{(i)}) = 1$ and the function x^2 is convex, so $\sum_{-1\le i\le t-1}(x^{(i+1)} - x^{(i)})^2$ is minimized when $x^{(i+1)} - x^{(i)} = 1/t$ for all $-1 \le i \le t-1$. It follows that

$$\mathbb{E}\, C_T = \sum_{i=1}^{T} \mathbb{E}\, c_t = \sum_{i=1}^{T} \mathbb{E}\, \frac{M_t}{4} \ge \sum_{i=1}^{T} \frac{1}{4t} \ge \frac{1}{4}\log T.$$

So $\mathbb{E}\, C_T \in \Omega(\log T)$ as desired. ■

Proposition 18 *If $\mathcal{X} = [a,b]$, the arrivals X_t are arbitrary, and $\mathcal{H} = \{\mathbb{1}_{x>t} \mid t \in [a,b]\}$, all algorithms incur worst-case cost $C_T = T$.*

Proof Fix an algorithm ALG. Consider the modified setting in the proof of Proposition 6. For $t \geq 1$, define A_t^k, B_t^k as in the proof of Theorem 3, Claim 1. Recall that (A_t^k, B_t^k) is the region of uncertainty; at time 1, since (intuitively) no labels have been observed, we have $A_1 = 0$, $B_1 = 1$. For each round t, let $X_t = \frac{A_t^k + B_t^k}{2}$. Clearly, (A_t^k, B_t^k) is nonempty for all $t \geq 1$, and $B_t^k - A_t^k = 1/2^{t-1}$. If ALG classifies X_t as positive, label X_t as negative, and vice versa. If ALG queries, label X_t as positive. Clearly, if this labeling is valid, that is, if all these labels are consistent with some hypothesis $h \in [0, 1]$, then for this hypothesis, $C_T = T$ for all $T \geq 1$. Let h be the number with binary representation $0.\omega_1\omega_2\dots$, where $\omega_t = 0$ if ALG classifies X_t as negative, and $\omega_t = 1$ if ALG classifies X_t as positive or queries. It is clear that h is contained in all the intervals (A_t^k, B_t^k), hence consistent with all labels. ■