The Sample Complexity of Self-Verifying Bayesian Active Learning

Liu Yang
liuy@cs.cmu.edu
Machine Learning Department
Carnegie Mellon University

Steve Hanneke
shanneke@stat.cmu.edu
Department of Statistics
Carnegie Mellon University

Jaime Carbonell
jgc@cs.cmu.edu
Language Technologies Institute
Carnegie Mellon University

Abstract

We prove that access to a prior distribution over target functions can dramatically improve the sample complexity of self-terminating active learning algorithms, so that it is always better than the known results for prior-dependent passive learning. In particular, this is in stark contrast to the analysis of prior-independent algorithms, where there are simple known learning problems for which no self-terminating algorithm can provide this guarantee for all priors.

1 Introduction and Background

Active learning is a powerful form of supervised machine learning characterized by interaction between the learning algorithm and supervisor during the learning process. In this work, we consider a variant known as *pool-based* active learning, in which a learning algorithm is given access to a (typically very large) collection of unlabeled examples, and is able to select any of those examples, request the supervisor to label it (in agreement with the target concept), then after receiving the label, selects another example from the pool, etc. This sequential label-requesting process continues until some halting criterion is reached, at which point the algorithm outputs a function, and the objective is for this function to closely approximate the (unknown) target concept in the future. The primary motivation behind pool-based active learning is that, often, unlabeled examples are inexpensive and available in abundance, while annotating those examples can be costly or time-consuming; as such, we often wish to select only the informative examples to be labeled, thus reducing information-redundancy to some extent, compared to the baseline of selecting the examples to be labeled uniformly at random from the pool (passive learning).

Appearing in Proceedings of the 14^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2011, Fort Lauderdale, FL, USA. Volume 15 of JMLR: W&CP 15. Copyright 2011 by the authors.

There has recently been an explosion of fascinating theoretical results on the advantages of this type of active learning, compared to passive learning, in terms of the number of labels required to obtain a prescribed accuracy (called the *sample complexity*): e.g., [FSST97, Das04, DKM09, Das05, Han07b, BHV10, BBL09, Wan09, Kää06, Han07a, DHM07, Fri09, CN08, Now08, BBZ07, Han11, Kol10, Han09, BDL09]. In particular, [BHV10] show that in noise-free binary classifier learning, for any passive learning algorithm for a concept space of finite VC dimension, there exists an active learning algorithm with asymptotically much smaller sample complexity for any nontrivial target concept. In later work, [Han09] strengthens this result by removing a certain strong dependence on the distribution of the data in the learning algorithm. Thus, it appears there are profound advantages to active learning compared to passive learning.

However, the ability to rapidly converge to a good classifier using only a small number of labels is only one desirable quality of a machine learning method, and there are other qualities that may also be important in certain scenarios. In particular, the ability to *verify* the performance of a learning method is often a crucial part of machine learning applications, as (among other things) it helps us determine whether we have enough data to achieve a desired level of accuracy with the given method. In passive learning, one common practice for this verification is to hold out a random sample of labeled examples as a *validation sample* to evaluate the trained classifier (e.g., to determine when training is complete). It turns out this technique is not feasible in active learning, since in order to be really useful as an indicator of whether we have seen enough labels to guarantee the desired accuracy, the number of labeled examples in the random validation sample would need to be much larger than the number of labels requested by the active learning algorithm itself, thus (to some extent) canceling the savings obtained by performing active rather than passive learning. Another common practice in passive learning is to examine the training error rate of the returned classifier, which can serve as a reasonable indicator of performance (after adjusting for model complexity). However, again this measure of performance is not necessarily reasonable for active

learning, since the set of examples the algorithm requests the labels of is typically distributed very differently from the test examples the classifier will be applied to after training.

This reasoning indicates that performance verification is (at best) a far more subtle issue in active learning than in passive learning. Indeed, [BHV10] note that although the number of labels required to achieve good accuracy is significantly smaller than passive learning, it is often the case that the number of labels required to *verify* that the accuracy is good is not significantly improved. In particular, this phenomenon can dramatically increase the sample complexity of active learning algorithms that adaptively determine how many labels to request before terminating. In short, if we require the algorithm both to *learn* an accurate concept and to *know* that its concept is accurate, then the number of labels required by active learning is often not significantly smaller than the number required by passive learning.

We should note, however, that the above results were proven for a learning scenario in which the target concept is considered a constant, and no information about the process that generates this concept is known a priori. Alternatively, we can consider a modification of this problem, so that the target concept can be thought of as a random variable, a sample from a known distribution (called a *prior*) over the space of possible concepts. Such a setting has been studied in detail in the context of passive learning for noise-free binary classification. In particular, [HKS92] found that for any concept space of finite VC dimension d, for any prior and distribution over data points, $O(d/\varepsilon)$ random labeled examples are sufficient for the expected error rate of the Bayes classifier produced under the posterior distribution to be at most ε. Furthermore, it is easy to construct learning problems for which there is an $\Omega(1/\varepsilon)$ lower bound on the number of random labeled examples required to achieve expected error rate at most ε, by any passive learning algorithm; for instance, the problem of learning threshold classifiers on $[0, 1]$ under a uniform data distribution and uniform prior is one such scenario.

In the context of active learning (again, with access to the prior), [FSST97] analyze the *Query by Committee* algorithm, and find that if a certain information gain quantity for the points requested by the algorithm is lower-bounded by a value g, then the algorithm requires only $O((d/g)\log(1/\varepsilon))$ labels to achieve expected error rate at most ε. In particular, they show that this is satisfied for *constant* g for linear separators under a near-uniform prior, and a near-uniform data distribution over the unit sphere. This represents a marked improvement over the results of [HKS92] for passive learning, and since the Query by Committee algorithm is self-verifying, this result is highly relevant to the present discussion. However, the condition that the information gains be lower-bounded by a constant is quite restrictive, and many interesting learning problems are precluded by this requirement. Furthermore, there exist learning problems (with finite VC dimension) for which the Query by Committee algorithm makes an expected number of label requests exceeding $\Omega(1/\varepsilon)$. To date, there has not been a general analysis of how the value of g can behave as a function of ε, though such an analysis would likely be quite interesting.

In the present paper, we take a more general approach to the question of active learning with access to the prior. We are interested in the broad question of whether access to the prior bridges the gap between the sample complexity of *learning* and the sample complexity of learning *with verification*. Specifically, we ask the following question.

Can a prior-dependent self-terminating active learning algorithm for a concept class of finite VC dimension always achieve expected error rate at most ε using $o(1/\varepsilon)$ label requests?

After some basic definitions in Section 2, we begin in Section 3 with a concrete example, namely interval classifiers under a uniform data density but arbitrary prior, to illustrate the general idea, and convey some of the intuition as to why one might expect a positive answer to this question. In Section 4, we present a general proof that the answer is *always* "yes." As the known results for the sample complexity of passive learning with access to the prior are typically $\propto 1/\varepsilon$ [HKS92], and this is sometimes tight, this represents an improvement over passive learning. The proof is simple and accessible, yet represents an important step in understanding the problem of self-termination in active learning algorithms, and the general issue of the complexity of verification. Also, as this is a result that does *not* generally hold for prior-independent algorithms (even for their "average-case" behavior induced by the prior) for certain concept spaces, this also represents a significant step toward understanding the inherent value of having access to the prior.

2 Definitions and Preliminaries

First, we introduce some notation and formal definitions. We denote by $\mathcal{X}$ the *instance space*, representing the range of the unlabeled data points, and we suppose a distribution $\mathcal{D}$ on $\mathcal{X}$, which we will refer to as the *data distribution*. We also suppose the existence of a sequence $X_1, X_2, \ldots$ of i.i.d. random variables, each with distribution $\mathcal{D}$, referred to as the unlabeled data sequence. Though one could potentially analyze the achievable performance as a function of the number of unlabeled points made available to the learning algorithm (cf. [Das05]), for simplicity in the present work, we will suppose this unlabeled sequence is essentially inexhaustible, corresponding to the practical fact that unlabeled data are typically available in abundance as they are often relatively inexpensive to ob-

tain. Additionally, there is a set $\mathbb{C}$ of measurable classifiers $h : \mathcal{X} \to \{-1, +1\}$, referred to as the *concept space*. We denote by d the VC dimension of $\mathbb{C}$, and in our present context we will restrict ourselves to spaces $\mathbb{C}$ with $d < \infty$, referred to as a *VC class*. We also have a probability distribution π, called the *prior*, over $\mathbb{C}$, and a random variable $h^* \sim \pi$, called the *target function*; we suppose h^* is independent from the data sequence $X_1, X_2, \ldots$. We adopt the usual notation for conditional expectations and probabilities [ADD00]; for instance, $\mathbb{E}[A|B]$ can be thought of as an expectation of the value A, under the conditional distribution of A given the value of B (which itself is random), and thus the value of $\mathbb{E}[A|B]$ is essentially determined by the value of B. For any measurable $h : \mathcal{X} \to \{-1, +1\}$, define the *error rate* $\mathrm{er}(h) = \mathcal{D}(\{x : h(x) \neq h^*(x)\})$. So far, this setup is essentially identical to that of [HKS92, FSST97].

The protocol in active learning is the following. An active learning algorithm $\mathcal{A}$ is given as input the prior π, the data distribution $\mathcal{D}$ (though see Section 5), and a value $\varepsilon \in (0, 1]$. It also (implicitly) depends on the data sequence $X_1, X_2, \ldots$, and has an indirect dependence on the target function h^* via the following type of interaction. The algorithm may inspect the values X_i for any initial segment of the data sequence, select an index $i \in \mathbb{N}$ to "request" the label of; after selecting such an index, the algorithm receives the value $h^*(X_i)$. The algorithm may then select another index, request the label, receive the value of h^* on that point, etc. This happens for a number of rounds, $N(\mathcal{A}, h^*, \varepsilon, \mathcal{D}, \pi)$, before eventually the algorithm halts and returns a classifier $\hat{h}$. An algorithm is said to be *correct* if $\mathbb{E}\left[\mathrm{er}\left(\hat{h}\right)\right] \leq \varepsilon$ for every $(\varepsilon, \mathcal{D}, \pi)$; that is, given direct access to the prior and the data distribution, and given a specified value ε, a correct algorithm must be guaranteed to have expected error rate at most ε. Define the *expected sample complexity* of $\mathcal{A}$ for $(\mathcal{X}, \mathbb{C}, \mathcal{D}, \pi)$ to be the function $SC(\varepsilon, \mathcal{D}, \pi) = \mathbb{E}[N(\mathcal{A}, h^*, \varepsilon, \mathcal{D}, \pi)]$: the expected number of label requests the algorithm makes.

We will be interested in proving that certain algorithms achieve a sample complexity $SC(\varepsilon, \mathcal{D}, \pi) = o(1/\varepsilon)$. For some $(\mathcal{X}, \mathbb{C}, \mathcal{D})$, it is known that there are π-independent algorithms (meaning the algorithm's behavior is independent of the π argument) $\mathcal{A}$ such that we always have $\mathbb{E}[N(\mathcal{A}, h^*, \varepsilon, \mathcal{D}, \pi)|h^*] = o(1/\varepsilon)$; for instance, threshold classifiers have this property under any $\mathcal{D}$, homogeneous linear separators have this property under a uniform $\mathcal{D}$ on the unit sphere in k dimensions, and intervals with positive width on $\mathcal{X} = [0, 1]$ have this property under $\mathcal{D} = \mathrm{Uniform}([0, 1])$ (see e.g., [Das05]). It is straightforward to show that any such $\mathcal{A}$ will also have $SC(\varepsilon, \mathcal{D}, \pi) = o(1/\varepsilon)$ for every π. In particular, the law of total expectation and the dominated convergence theorem imply

$$\lim_{\varepsilon \to 0} \varepsilon \cdot SC(\varepsilon, \mathcal{D}, \pi) = \lim_{\varepsilon \to 0} \varepsilon \cdot \mathbb{E}[\mathbb{E}[N(\mathcal{A}, h^*, \varepsilon, \mathcal{D}, \pi)|h^*]]$$
$$= \mathbb{E}\left[\lim_{\varepsilon \to 0} \varepsilon \cdot \mathbb{E}[N(\mathcal{A}, h^*, \varepsilon, \mathcal{D}, \pi)|h^*]\right] = 0.$$

In these cases, we can think of SC as a kind of "average-case" analysis of these algorithms. However, there are also many $(\mathcal{X}, \mathbb{C}, \mathcal{D})$ for which no such π-independent algorithm exists, achieving $o(1/\varepsilon)$ sample complexity for *all* priors. For instance, this is the case for $\mathbb{C}$ as the space of interval classifiers (including the empty interval) on $\mathcal{X} = [0, 1]$ under $\mathcal{D} = \mathrm{Uniform}([0, 1])$ (this essentially follows from a proof of [BHV10]). Thus, any general result on $o(1/\varepsilon)$ expected sample complexity for π-dependent algorithms would signify that there is a real advantage to having access to the prior.

3 An Example: Intervals

In this section, we walk through a simple and intuitive example, to illustrate how access to the prior makes a difference in the sample complexity. For simplicity, in this example (only) we will suppose the algorithm may request the label of any point in $\mathcal{X}$, not just those in the sequence $\{X_i\}$; the same ideas can easily be adapted to the setting where queries are restricted to $\{X_i\}$. Specifically, consider $\mathcal{X} = [0, 1]$, $\mathcal{D}$ uniform on $[0, 1]$, and the concept space of *interval classifiers*, where $\mathbb{C} = \{\mathbb{I}^{\pm}_{[a,b]} : 0 < a \leq b < 1\}$, where $\mathbb{I}^{\pm}_{[a,b]}(x) = +1$ if $x \in [a, b]$ and -1 otherwise. For each classifier $h \in \mathbb{C}$, let $w(h) = \mathbb{P}(h(x) = +1)$ (the width of the interval h).

Consider an active learning algorithm that makes label requests at the locations (in sequence) $1/2, 1/4, 3/4, 1/8, 3/8, 5/8, 7/8, 1/16, 3/16, \ldots$ until (case 1) it encounters an example x with $h^*(x) = +1$ or until (case 2) the set of classifiers $V \subseteq \mathbb{C}$ consistent with all observed labels so far satisfies $\mathbb{E}[w(h^*)|V] \leq \varepsilon$ (which ever comes first). In case 2, the algorithm simply halts and returns the constant classifier that always predicts -1: call it h_-; note that $\mathrm{er}(h_-) = w(h^*)$. In case 1, the algorithm enters a second phase, in which it performs a binary search (repeatedly querying the midpoint between the closest two -1 and $+1$ points, taking 0 and 1 as known negative points) to the left and right of the observed positive point, halting after $\log_2(2/\varepsilon)$ label requests on each side; this results in estimates of the target's endpoints up to $\pm\varepsilon/2$, so that returning any classifier among the set $V \subseteq \mathbb{C}$ consistent with these labels results in error rate at most ε; in particular, if $\tilde{h}$ is the classifier in V returned, then $\mathbb{E}[\mathrm{er}(\tilde{h})|V] \leq \varepsilon$.

Denoting this algorithm by $\mathcal{A}_{[]}$, and $\hat{h}$ the classifier it returns, we have

$$\mathbb{E}\left[\mathrm{er}\left(\hat{h}\right)\right] = \mathbb{E}\left[\mathbb{E}\left[\mathrm{er}\left(\hat{h}\right)\middle|V\right]\right] \leq \varepsilon,$$

so that the algorithm is definitely correct.

Note that case 2 will definitely be satisfied after at most $\frac{2}{\varepsilon}$ label requests, and case 1 will definitely be satisfied after at most $\frac{2}{w(h^*)}$ label requests, so that the algorithm never makes more than $\frac{2}{\max\{w(h^*),\varepsilon\}} + 2\log_2(2/\varepsilon)$ label requests. In particular, for any h^* with $w(h^*) > 0$, $N(\mathcal{A}_{[]}, h^*, \varepsilon, \mathcal{D}, \pi) = o(1/\varepsilon)$. Abbreviating $N(h^*) = N(\mathcal{A}_{[]}, h^*, \varepsilon, \mathcal{D}, \pi)$, we have

$$\mathbb{E}\left[N(h^*)\right] = \mathbb{E}\left[N(h^*)\middle| w(h^*) = 0\right]\mathbb{P}\left(w(h^*) = 0\right) + \mathbb{E}\left[N(h^*)\middle| w(h^*) > 0\right]\mathbb{P}\left(w(h^*) > 0\right). \quad (1)$$

Since $w(h^*) > 0 \Rightarrow N(h^*) = o(1/\varepsilon)$, the dominated convergence theorem implies

$$\lim_{\varepsilon\to 0} \varepsilon\mathbb{E}\left[N(h^*)\middle| w(h^*) > 0\right] = \mathbb{E}\left[\lim_{\varepsilon\to 0} \varepsilon N(h^*)\middle| w(h^*) > 0\right] = 0,$$

so that the second term in (1) is $o(1/\varepsilon)$. If $\mathbb{P}(w(h^*) = 0) = 0$, this completes the proof. We focus the rest of the proof on the first term in (1), in the case that $\mathbb{P}(w(h^*) = 0) > 0$: i.e. there is nonzero probability that the target h^* labels the space almost all negative. Letting V denote the subset of $\mathbb{C}$ consistent with all requested labels, note that on the event $w(h^*) = 0$, after n label requests (for n a power of 2) we have $\max_{h\in V} w(h) \le 1/n$. Thus, for any value $w_\varepsilon \in (0,1)$, after at most $\frac{2}{w_\varepsilon}$ label requests, on the event that $w(h^*) = 0$,

$$\begin{aligned}\mathbb{E}\left[w(h^*)\middle| V\right] &= \int w(h)\mathbb{I}[h \in V]\pi(\mathrm{d}h)/\pi(V) \\ &\le \int w(h)\mathbb{I}[w(h) \le w_\varepsilon]\pi(\mathrm{d}h)/\pi(V) \\ &= \mathbb{E}\left[w(h^*)\mathbb{I}\left[w(h^*) \le w_\varepsilon\right]\right]/\pi(V) \\ &\le \frac{\mathbb{E}\left[w(h^*)\mathbb{I}\left[w(h^*) \le w_\varepsilon\right]\right]}{\mathbb{P}\left(w(h^*) = 0\right)}. \end{aligned} \quad (2)$$

Now note that, by the dominated convergence theorem,

$$\lim_{w\to 0}\mathbb{E}\left[\frac{w(h^*)\mathbb{I}\left[w(h^*) \le w\right]}{w}\right] = \mathbb{E}\left[\lim_{w\to 0}\frac{w(h^*)\mathbb{I}\left[w(h^*) \le w\right]}{w}\right] = 0.$$

Therefore, $\mathbb{E}\left[w(h^*)\mathbb{I}\left[w(h^*) \le w\right]\right] = o(w)$. If we define w_ε as the largest value of w for which $\mathbb{E}\left[w(h^*)\mathbb{I}\left[w(h^*) \le w\right]\right] \le \varepsilon\mathbb{P}(w(h^*) = 0)$ (or, say, half the supremum if the maximum is not achieved), then we have $w_\varepsilon = \omega(\varepsilon)$. Combined with (2), this implies

$$\mathbb{E}\left[N(h^*)\middle| w(h^*) = 0\right] \le \frac{2}{w_\varepsilon} = o(1/\varepsilon).$$

Thus, all of the terms in (1) are $o(1/\varepsilon)$, so that in total $\mathbb{E}[N(h^*)] = o(1/\varepsilon)$.

In conclusion, for this concept space $\mathbb{C}$ and data distribution $\mathcal{D}$, we have a correct active learning algorithm achieving a sample complexity $SC(\varepsilon, \mathcal{D}, \pi) = o(1/\varepsilon)$ for all priors π on $\mathbb{C}$.

4 Main Result

In this section, we present our main result: a general result stating that $o(1/\varepsilon)$ expected sample complexity is always achievable by some correct active learning algorithm, for *any* $(\mathcal{X}, \mathbb{C}, \mathcal{D}, \pi)$ for which $\mathbb{C}$ has finite VC dimension. Since the known results for the sample complexity of passive learning with access to the prior are typically $\Theta(1/\varepsilon)$, and since there are known learning problems $(\mathcal{X}, \mathbb{C}, \mathcal{D}, \pi)$ for which every passive learning algorithm requires $\Omega(1/\varepsilon)$ samples, this $o(1/\varepsilon)$ result for active learning represents an improvement over passive learning. Additionally, as mentioned, this type of result is often not possible for algorithms lacking access to the prior π, as there are well-known problems $(\mathcal{X}, \mathbb{C}, \mathcal{D})$ for which no prior-independent correct algorithm (of the self-terminating type studied here) can achieve $o(1/\varepsilon)$ sample complexity for every prior π [BHV10]; in particular, the intervals problem studied above is one such example.

First, we have a small lemma.

Lemma 1. *For any sequence of functions $\phi_n : \mathbb{C} \to [0,\infty)$ such that, $\forall f \in \mathbb{C}$, $\phi_n(f) = o(1/n)$ and $\forall n \in \mathbb{N}$, $\phi_n(f) \le c/n$ (for an f-independent constant $c \in (0,\infty)$), there exists a sequence $\bar{\phi}_n$ in $[0,\infty)$ such that*

$$\bar{\phi}_n = o(1/n) \quad \text{and} \quad \lim_{n\to\infty}\mathbb{P}\left(\phi_n(h^*) > \bar{\phi}_n\right) = 0.$$

Proof. For any constant $\delta \in (0,\infty)$, we have (by Markov's inequality and the dominated convergence theorem)

$$\lim_{n\to\infty}\mathbb{P}\left(n\phi_n(h^*) > \delta\right) \le \frac{1}{\delta}\lim_{n\to\infty}\mathbb{E}\left[n\phi_n(h^*)\right] = \frac{1}{\delta}\mathbb{E}\left[\lim_{n\to\infty} n\phi_n(h^*)\right] = 0.$$

Therefore (by induction), there exists a diverging sequence n_i in $\mathbb{N}$ such that $\lim_{i\to\infty}\sup_{n\ge n_i}\mathbb{P}\left(n\phi_n(h^*) > 2^{-i}\right) = 0$. Inverting this, let $i_n = \max\{i \in \mathbb{N} : n_i \le n\}$, and define $\bar{\phi}_n(h^*) = (1/n)\cdot 2^{-i_n}$. By construction, $\mathbb{P}\left(\phi_n(h^*) > \bar{\phi}_n\right) \to 0$. Furthermore, $n_i \to \infty \implies i_n \to \infty$, so that we have

$$\lim_{n\to\infty} n\bar{\phi}_n = \lim_{n\to\infty} 2^{-i_n} = 0,$$

implying $\bar{\phi}_n = o(1/n)$. □

Theorem 1. *For any VC class $\mathbb{C}$, there is a correct active learning algorithm that, for every data distribution $\mathcal{D}$ and prior π, achieves expected sample complexity SC for $(\mathcal{X}, \mathbb{C}, \mathcal{D}, \pi)$ such that*

$$SC(\varepsilon, \mathcal{D}, \pi) = o(1/\varepsilon).$$

Our approach to proving Theorem 1 is via a reduction to established results about active learning algorithms that are *not* self-verifying. Specifically, consider a slightly different type of active learning algorithm than that defined above: namely, an algorithm $\mathcal{A}_a$ that takes as input a *budget* $n \in \mathbb{N}$ on the number of label requests it is allowed to make, and that after making at most n label requests returns as output a classifier $\hat{h}_n$. Let us refer to any such algorithm as a *budget-based* active learning algorithm. Note that budget-based active learning algorithms are prior-independent (have no direct access to the prior). The following result was proven by [Han09] (see also the related earlier work of [BHV10]).

Lemma 2. *[Han09] For any VC class $\mathbb{C}$, there exists a constant $c \in (0,\infty)$, a function $R(n; f, \mathcal{D})$, and a (prior-independent) budget-based active learning algorithm $\mathcal{A}_a$ such that*

$$\forall \mathcal{D}, \forall f \in \mathbb{C}, R(n; f, \mathcal{D}) \leq c/n \text{ and } R(n; f, \mathcal{D}) = o(1/n),$$

and $\mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right) \middle| h^*\right] \leq R(n; h^*, \mathcal{D})$ *(always), where $\hat{h}_n$ is the classifier returned by $\mathcal{A}_a$.*[1]

That is, equivalently, for any fixed value for the target function, the expected error rate is $o(1/n)$, where the random variable in the expectation is only the data sequence $X_1, X_2, \ldots$. Our task in the proof of Theorem 1 is to convert such a budget-based algorithm into one of the form defined in Section 1: that is, a self-terminating prior-dependent algorithm, taking ε as input.

Proof of Theorem 1. Consider $\mathcal{A}_a$, $\hat{h}_n$, R, and c as in Lemma 2, and define

$$n_\varepsilon = \min\left\{n \in \mathbb{N} : \mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right)\right] \leq \varepsilon\right\}.$$

This value is accessible based purely on access to π and $\mathcal{D}$. Furthermore, we clearly have (by construction) $\mathbb{E}\left[\mathrm{er}\left(\hat{h}_{n_\varepsilon}\right)\right] \leq \varepsilon$. Thus, denoting by $\mathcal{A}'_a$ the active learning algorithm, taking $(\mathcal{D}, \pi, \varepsilon)$ as input, which runs $\mathcal{A}_a(n_\varepsilon)$ and then returns $\hat{h}_{n_\varepsilon}$, we have that $\mathcal{A}'_a$ is a *correct* algorithm (i.e., its expected error rate is at most ε).

As for the expected sample complexity $SC(\varepsilon, \mathcal{D}, \pi)$ achieved by $\mathcal{A}'_a$, we have $SC(\varepsilon, \mathcal{D}, \pi) \leq n_\varepsilon$, so that it remains only to bound n_ε. By Lemma 1, there is a π-dependent function $R(n; \pi, \mathcal{D})$ such that

$$\forall \pi, \qquad \pi\left(\{f \in \mathbb{C} : R(n; f, \mathcal{D}) > R(n; \pi, \mathcal{D})\}\right) \to 0$$
$$\text{and } R(n; \pi, \mathcal{D}) = o(1/n).$$

[1] Furthermore, it is not difficult to see that we can take this R to be measurable in the h^* argument.

Therefore, by the law of total expectation,

$$\begin{aligned}
&\mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right)\right] = \mathbb{E}\left[\mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right)\middle| h^*\right]\right] \leq \mathbb{E}\left[R(n; h^*, \mathcal{D})\right] \\
&\leq \frac{c}{n}\pi(\{f \in \mathbb{C} : R(n; f, \mathcal{D}) > R(n; \pi, \mathcal{D})\}) + R(n; \pi, \mathcal{D}) \\
&= o(1/n).
\end{aligned}$$

If $n_\varepsilon = O(1)$, then clearly $n_\varepsilon = o(1/\varepsilon)$ as needed. Otherwise, since n_ε is monotonic in ε, we must have $n_\varepsilon \uparrow \infty$ as $\varepsilon \downarrow 0$. In particular, in this latter case we have

$$\begin{aligned}
&\lim_{\varepsilon\to 0} \varepsilon \cdot n_\varepsilon \\
&\leq \lim_{\varepsilon\to 0} \varepsilon \cdot \left(1 + \max\left\{n \geq n_\varepsilon - 1 : \mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right)\right] > \varepsilon\right\}\right) \\
&= \lim_{\varepsilon\to 0} \varepsilon \cdot \max_{n \geq n_\varepsilon - 1} n\mathbb{I}\left[\mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right)\right]/\varepsilon > 1\right] \\
&\leq \lim_{\varepsilon\to 0} \varepsilon \cdot \max_{n \geq n_\varepsilon - 1} n\mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right)\right]/\varepsilon \\
&= \lim_{\varepsilon\to 0} \max_{n \geq n_\varepsilon - 1} n\mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right)\right] = \limsup_{n\to\infty} n\mathbb{E}\left[\mathrm{er}\left(\hat{h}_n\right)\right] = 0,
\end{aligned}$$

so that $n_\varepsilon = o(1/\varepsilon)$, as required. □

5 Dependence on $\mathcal{D}$ in the Learning Algorithm

The dependence on $\mathcal{D}$ in the algorithm described in the proof is fairly weak, and we can eliminate any direct dependence on $\mathcal{D}$ by replacing $\mathrm{er}\left(\hat{h}_n\right)$ by a $1 - \varepsilon/2$ confidence upper bound based on $m_\varepsilon = \Omega\left(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon}\right)$ i.i.d. unlabeled examples $X'_1, X'_2, \ldots, X'_{m_\varepsilon}$ independent from the examples used by the algorithm: for instance, set aside in a pre-processing step, where the bound is derived based on Hoeffding's inequality and a union bound over the values of n that we check, of which there are at most $O(1/\varepsilon)$. Then we simply increase the value of n (starting at some constant, such as 1) until

$$\frac{1}{m_\varepsilon}\sum_{i=1}^{m_\varepsilon} \mathbb{P}\left(h^*(X'_i) \neq \hat{h}_n(X'_i) \middle| \{X_j\}_j, \{X'_j\}_j\right) \leq \varepsilon/2.$$

The expected value of the smallest value of n for which this occurs is $o(1/\varepsilon)$. Note that the probability only requires access to the prior π, not the data distribution $\mathcal{D}$ (the budget-based algorithm $\mathcal{A}_a$ of [Han09] has no direct dependence on $\mathcal{D}$); if desired for computational efficiency, this probability may also be estimated by a $1 - \varepsilon/4$ confidence upper bound based on $\Omega\left(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon}\right)$ independent samples of h^* values with distribution π, where for each sample we simulate the execution of $\mathcal{A}_a(n)$ for that (simulated) target function in order to obtain the returned classifier. In particular, note that no actual label requests to the oracle are required during this process of estimating the appropriate label budget n_ε, as all executions of $\mathcal{A}_a$ are *simulated*.

6 Inherent Dependence on π in the Sample Complexity

We have shown that for every prior π, the sample complexity is bounded by a function that is $o(1/\varepsilon)$. One might wonder whether it is possible that the asymptotic dependence on ε in the sample complexity can be prior-independent, while still being $o(1/\varepsilon)$. That is, we can ask whether there exists a (π-independent) function $s(\varepsilon) = o(1/\varepsilon)$ such that, for all priors π, there is a correct π-dependent algorithm achieving a sample complexity $SC(\varepsilon, \mathcal{D}, \pi) = O(s(\varepsilon))$, possibly involving π-dependent constants. Certainly in some cases, such as threshold classifiers, this is true. However, it seems this is not generally the case, and in particular it fails to hold for the space of interval classifiers.

For instance, consider a prior π on the space $\mathbb{C}$ of interval classifiers, constructed as follows. We are given an arbitrary monotonic $g(\varepsilon) = o(1/\varepsilon)$; since $g(\varepsilon) = o(1/\varepsilon)$, there must exist (nonzero) functions $q_1(i)$ and $q_2(i)$ such that $\lim_{i\to\infty} q_1(i) = \lim_{i\to\infty} q_2(i) = 0$ and $\forall i \in \mathbb{N}, g(q_1(i)/2^{i+1}) \leq q_2(i) \cdot 2^i$; furthermore, letting $q(i) = \max\{q_1(i), q_2(i)\}$, by monotonicity of g we also have $\forall i \in \mathbb{N}, g(q(i)/2^{i+1}) \leq q(i) \cdot 2^i$, and $\lim_{i\to\infty} q(i) = 0$. Then define a function $p(i)$ with $\sum_{i\in\mathbb{N}} p(i) = 1$ such that $p(i) \geq q(i)$ for infinitely many $i \in \mathbb{N}$; for instance, this can be done inductively as follows. Let $\alpha_0 = 1/2$; for each $i \in \mathbb{N}$, if $q(i) > \alpha_{i-1}$, set $p(i) = 0$ and $\alpha_i = \alpha_{i-1}$; otherwise, set $p(i) = \alpha_{i-1}$ and $\alpha_i = \alpha_{i-1}/2$. Finally, for each $i \in \mathbb{N}$, and each $j \in \{0, 1, \ldots, 2^i - 1\}$, define $\pi\left(\left\{\mathbb{I}^{\pm}_{[j\cdot 2^{-i},(j+1)\cdot 2^{-i}]}\right\}\right) = p(i)/2^i$.

We let $\mathcal{D}$ be uniform on $\mathcal{X} = [0, 1]$. Then for each $i \in \mathbb{N}$ s.t. $p(i) \geq q(i)$, there is a $p(i)$ probability the target interval has width 2^{-i}, and given this any algorithm requires $\propto 2^i$ expected number of requests to determine which of these 2^i intervals is the target, failing which the error rate is at least 2^{-i}. In particular, letting $\varepsilon_i = p(i)/2^{i+1}$, any correct algorithm has sample complexity at least $\propto p(i) \cdot 2^i$ for $\varepsilon = \varepsilon_i$. Noting $p(i)\cdot 2^i \geq q(i)\cdot 2^i \geq g(q(i)/2^{i+1}) \geq g(\varepsilon_i)$, this implies there exist arbitrarily small values of $\varepsilon > 0$ for which the optimal sample complexity is at least $\propto g(\varepsilon)$, so that the sample complexity is *not* $o(g(\varepsilon))$.

For any $s(\varepsilon) = o(1/\varepsilon)$, there exists a monotonic $g(\varepsilon) = o(1/\varepsilon)$ such that $s(\varepsilon) = o(g(\varepsilon))$. Thus, constructing π as above for this g, we have that the sample complexity is not $o(g(\varepsilon))$, and therefore not $O(s(\varepsilon))$. So at least for the space of interval classifiers, the specific $o(1/\varepsilon)$ asymptotic dependence on ε is inherently π-dependent.

References

[ADD00] R. B. Ash and C. A. Doléans-Dade. *Probability & Measure Theory*. Academic Press, 2000.

[BBL09] M.-F. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. *Journal of Computer and System Sciences*, 75(1):78–89, 2009.

[BBZ07] M.-F. Balcan, A. Broder, and T. Zhang. Margin based active learning. In *Proceedings of the 20^{th} Conference on Learning Theory*, 2007.

[BDL09] A. Beygelzimer, S. Dasgupta, and J. Langford. Importance weighted active learning. In *Proceedings of the International Conference on Machine Learning*, 2009.

[BHV10] M.-F. Balcan, S. Hanneke, and J. Wortman Vaughan. The true sample complexity of active learning. *Machine Learning*, 80(2–3):111–139, September 2010.

[CN08] R. Castro and R. Nowak. Minimax bounds for active learning. *IEEE Transactions on Information Theory*, 54(5):2339–2353, July 2008.

[Das04] S. Dasgupta. Analysis of a greedy active learning strategy. In *Advances in Neural Information Processing Systems*, pages 337–344. MIT Press, 2004.

[Das05] S. Dasgupta. Coarse sample complexity bounds for active learning. In *Proc. of Neural Information Processing Systems (NIPS)*, 2005.

[DHM07] S. Dasgupta, D. Hsu, and C. Monteleoni. A general agnostic active learning algorithm. In *Advances in Neural Information Processing Systems 20*, 2007.

[DKM09] S. Dasgupta, A. T. Kalai, and C. Monteleoni. Analysis of perceptron-based active learning. *Journal of Machine Learning Research*, 10:281–299, 2009.

[Fri09] E. Friedman. Active learning for smooth problems. In *Proceedings of the 22^{nd} Conference on Learning Theory*, 2009.

[FSST97] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby. Selective sampling using the query by committee algorithm. In *Machine Learning*, pages 133–168, 1997.

[Han07a] S. Hanneke. A bound on the label complexity of agnostic active learning. In *Proc. of the 24th International Conference on Machine Learning*, 2007.

[Han07b] S. Hanneke. Teaching dimension and the complexity of active learning. In *Proc. of the 20th Annual Conference on Learning Theory (COLT)*, 2007.

[Han09] S. Hanneke. *Theoretical Foundations of Active Learning*. PhD thesis, Carnegie Mellon University, 2009.

[Han11] S. Hanneke. Rates of convergence in active learning. *The Annals of Statistics*, 39(1):333–361, 2011.

[HKS92] D. Haussler, M. Kearns, and R. Schapire. Bounds on the sample complexity of bayesian learning using information theory and the vc dimension. In *Machine Learning*, pages 61–74. Morgan Kaufmann, 1992.

[Kää06] M. Kääriäinen. Active learning in the non-realizable case. In *Proc. of the 17th International Conference on Algorithmic Learning Theory*, 2006.

[Kol10] V. Koltchinskii. Rademacher complexities and bounding the excess risk in active learning. *Journal of Machine Learning Research*, To Appear, 2010.

[Now08] R. D. Nowak. Generalized binary search. In *Proceedings of the 46$^{\text{th}}$ Annual Allerton Conference on Communication, Control, and Computing*, 2008.

[Wan09] L. Wang. Sufficient conditions for agnostic active learnable. In *Advances in Neural Information Processing Systems 22*, 2009.