The Complexity of Interactive Machine Learning

Steve Hanneke

May 2007

Machine Learning Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
shanneke@cs.cmu.edu

A Bound on the Label Complexity of Agnostic Active Learning

Steve Hanneke SHANNEKE@CS.CMU.EDU
Machine Learning Department, Carnegie Mellon University, Pittsburgh, PA 15213 USA

Abstract

We study the label complexity of pool-based active learning in the agnostic PAC model. Specifically, we derive general bounds on the number of label requests made by the A^2 algorithm proposed by Balcan, Beygelzimer & Langford (Balcan et al., 2006). This represents the first nontrivial general-purpose upper bound on label complexity in the agnostic PAC model.

1. Introduction

In active learning, a learning algorithm is given access to a large pool of unlabeled examples, and is allowed to request the label of any particular example from that pool. The objective is to learn an accurate classifier while requesting as few labels as possible. This contrasts with passive (semi)supervised learning, where the examples to be labeled are chosen randomly. In comparison, active learning can often significantly decrease the work load of human annotators by more carefully selecting which examples from the unlabeled pool should be labeled. This is of particular interest for learning tasks where unlabeled examples are available in abundance, but labeled examples require significant effort to obtain.

In the passive learning literature, there are well-known bounds on the number of training examples necessary and sufficient to learn a near-optimal classifier with high probability (i.e., the sample complexity) (Vapnik, 1998; Blumer et al., 1989; Kulkarni, 1989; Benedek & Itai, 1988; Long, 1995). This quantity depends largely on the VC dimension of the concept space being learned (in a distribution-independent analysis) or the metric entropy (in a distribution-dependent analysis). However, significantly less is presently known about the analogous quantity for active learning: namely, the *label complexity*, or number of label requests that are necessary and sufficient to learn. This knowledge gap is especially marked in the *agnostic* learning setting, where class labels can be noisy, and we have no assumption about the amount or type of noise. Building a thorough understanding of label complexity, along with the quantities on which it depends, seems essential to fully exploit the potential of active learning.

In the present paper, we study the label complexity by way of bounding the number of label requests made by a recently proposed active learning algorithm, A^2 (Balcan et al., 2006), which provably learns in the agnostic PAC model. The bound we find for this algorithm depends critically on a particular quantity, which we call the *disagreement coefficient*, depending on the concept space and example distribution. This quantity is often simple to calculate or bound for many concept spaces. Although we find that the bound we derive is not always tight for the label complexity, it represents a significant step forward, since it is the first nontrivial *general-purpose* bound on label complexity in the agnostic PAC model.

The rest of the paper is organized as follows. In Section 2, we briefly review some of the related literature, to place the present work in context. In Section 3, we continue with the introduction of definitions and notation. Section 4 discusses a variety of simple examples to help build intuition. Moving on in Section 5, we state and prove the main result of this paper: an upper bound on the number of label requests made by A^2, based on the disagreement coefficient. Following this, in Section 6, we prove a lower bound for A^2 with the same basic dependence on disagreement coefficient. We conclude in Section 7 with some open problems.

2. Background

The recent literature on the label complexity of active learning has been bringing us steadily closer to understanding the nature of this problem. Within that literature, there is a mix of positive and negative results, as well as a wealth of open problems.

Appearing in *Proceedings of the 24th International Conference on Machine Learning*, Corvallis, OR, 2007. Copyright 2007 by the author(s)/owner(s). Revised 04/2007.

While studying the noise-free (realizable) setting, Dasgupta defines a quantity ρ called the *splitting index* (Dasgupta, 2005). ρ is dependent on the concept space, data distribution, and a (new) parameter τ he defines, as well as the target function itself. It essentially quantifies how easy it is to reduce the diameter of the concept space. He finds that under the assumption that there is no noise, roughly $\tilde{O}(\frac{d}{\rho})$ label requests are sufficient (where d is VC dimension), and $\Omega(\frac{1}{\rho})$ are necessary for learning (for respectively appropriate τ values). Thus, it appears that something like splitting index may be an important quantity to consider when bounding the label complexity. However, at present the only published analysis using splitting index is restricted to the noise-free (realizable) case. Additionally, one can construct simple examples where the splitting index is $O(1)$ (for $\tau = O(\epsilon^2)$), but agnostic learning requires $\Omega\left(\frac{1}{\epsilon}\right)$ label requests (even when the noise rate is zero). See Appendix A for an example of this. Thus, agnostic active learning seems to be a fundamentally more difficult problem than realizable active learning.

In studying the possibility of active learning in the presence of arbitrary classification noise, Balcan, Beygelzimer, & Langford propose the A^2 algorithm (Balcan et al., 2006). The strategy behind A^2 is to induce confidence intervals for the error rates of all concepts, and remove any concepts whose estimated error rate is larger than the smallest estimate to a statistically significant extent. This guarantees that with high probability we do not remove the best classifier in the concept space. The key observation that sometimes leads to improvements over passive learning is that, since we are only interested in *comparing* the error estimates, we do not need to request the label of any example whose label is not in dispute among the remaining classifiers. Balcan et al. analyze the number of label requests A^2 makes for some example concept spaces and distributions (notably linear separators under the uniform distribution on the unit sphere). However, other than fallback guarantees, they do not derive a general bound on the number of label requests, applicable to *any* concept space and distribution. This is the focus of the present paper.

In addition to the above results, there are a number of known lower bounds, than which there cannot be a learning algorithm guarateeing a number of label requests smaller. In particular, Kulkarni proves that, even if we allow *arbitrary* binary-valued queries and there is no noise, any algorithm that learns to accuracy $1-\epsilon$ can guarantee no better than $\Omega(\log N(2\epsilon))$ queries (Kulkarni et al., 1993), where $N(2\epsilon)$ is the size of a minimal 2ϵ-cover (defined below). Another known lower bound is due to Kääriäinen, who proves that in agnostic active learning, for most nontrivial concept spaces and distributions, if the noise rate is ν, then any algorithm that with probability $1-\delta$ outputs a classifier with error at most $\nu+\epsilon$ can guarantee no better than $\Omega\left(\frac{\nu^2}{\epsilon^2}\log\frac{1}{\delta}\right)$ label requests (Kääriäinen, 2006). In particular, these lower bounds imply that we can reasonably expect even the tightest general upper bounds on the label complexity to have some term related to $\log N(\epsilon)$ and some term related to $\frac{\nu^2}{\epsilon^2}\log\frac{1}{\delta}$.

3. Notation and Definitions

Let $\mathcal{X}$ be an *instance space*, comprising all possible examples we may ever encounter. $\mathbb{C}$ is a set of measurable functions $h : \mathcal{X} \to \{-1, 1\}$, known as the *concept space*. $\mathcal{D}_{XY}$ is any probability distribution on $\mathcal{X} \times \{-1, 1\}$. In the active learning setting, we draw $(X, Y) \sim \mathcal{D}_{XY}$, but the Y value is hidden from the learning algorithm until requested. For convenience, we will abuse notation by saying $X \sim \mathcal{D}$, where $\mathcal{D}$ is the marginal distribution of $\mathcal{D}_{XY}$ over $\mathcal{X}$; we then say the learning algorithm (optionally) *requests* the label Y of X (which was implicitly sampled at the same time as X); we may sometimes denote this label Y by $Oracle(X)$. For any $h \in \mathbb{C}$ and distribution $\mathcal{D}'$ over $\mathcal{X} \times \{-1, 1\}$, let $er_{\mathcal{D}'}(h) = \mathcal{P}r_{(X,Y)\sim\mathcal{D}'}\{h(X) \neq Y\}$, and for $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\} \in (\mathcal{X} \times \{-1, 1\})^m$, $er_S(h) = \frac{1}{m}\sum_{i=1}^m |h(x_i) - y_i|/2$. When $\mathcal{D}' = \mathcal{D}_{XY}$ (the distribution we are learning with respect to), we abbreviate this by $er(h) = er_{\mathcal{D}_{XY}}(h)$. The *noise rate*, denoted ν, is defined as $\nu = \inf_{h\in\mathbb{C}} er(h)$. Our objective in agnostic active learning is to, with probability $\geq 1-\delta$, output a classifier h with $er(h) \leq \nu+\epsilon$ without making many label requests.

Let $\rho_{\mathcal{D}}(\cdot,\cdot)$ be the pseudo-metric on $\mathbb{C}$ induced by $\mathcal{D}$, s.t. $\forall h, h' \in \mathbb{C}, \rho_{\mathcal{D}}(h, h') = \mathcal{P}r_{X\sim\mathcal{D}}\{h(X) \neq h'(X)\}$. An ϵ-cover of $\mathbb{C}$ with respect to $\mathcal{D}$ is any set $V \subseteq \mathbb{C}$ such that $\forall h \in \mathbb{C}, \exists h' \in V : \rho_{\mathcal{D}}(h, h') \leq \epsilon$. We additionally let $N(\epsilon)$ denote the size of a minimal ϵ-cover of $\mathbb{C}$ with respect to $\mathcal{D}$. It is known that $N(\epsilon) < 2\left(\frac{2e}{\epsilon}\ln\frac{2e}{\epsilon}\right)^d$, where d is the VC dimension of $\mathbb{C}$ (Haussler, 1992). To focus on learnable cases, we assume $d < \infty$.

Definition 1. *For a set $V \subseteq \mathbb{C}$, define the* region of disagreement

$$DIS(V) = \{x \in \mathcal{X} | \exists h_1, h_2 \in V : h_1(x) \neq h_2(x)\}.$$

Definition 2. *The* disagreement rate $\Delta(V)$ *of a set $V \subseteq \mathbb{C}$ is defined as*

$$\Delta(V) = \mathcal{P}r_{X\sim\mathcal{D}}\{X \in DIS(V)\}.$$

Definition 3. *For $h \in \mathbb{C}$, $r > 0$, let*

$$B(h,r) = \{h' \in \mathbb{C} : \rho_{\mathcal{D}}(h',h) \leq r\}$$

and define the disagreement rate at radius r

$$\Delta_r = \sup_{h \in \mathbb{C}} \Delta(B(h,r)).$$

Definition 4. *The* disagreement coefficient *is the infimum value of $\theta > 0$ such that $\forall r > \nu + \epsilon$,*

$$\Delta_r \leq \theta r.$$

The disagreement coefficient plays a critical role in the bounds of the following sections, which are increasing in this θ. Roughly speaking, it quantifies how quickly the region of disagreement can grow as a function of the radius of the version space.

4. Examples

The canonical example of the potential improvements in label complexity of active over passive learning is the *thresholds* concept space. Specifically, consider the concept space of thresholds t_z on the interval $[0,1]$ (for $z \in [0,1]$), such that $t_z(x) = +1$ iff $x \geq z$. Furthermore, suppose $\mathcal{D}$ is uniform on $[0,1]$. In this case, it is clear that the disagreement coefficient is at most 2, since the region of disagreement of $B(t_z, r)$ is roughly $\{x \in [0,1] : |x - z| \leq r\}$. That is, since the disagreement region grows at rate 1 in two disjoint directions as r increases, the disagreement coefficient $\theta = 2$.

As a second example, consider the disagreement coefficient for *intervals* on $[0,1]$. As before, let $\mathcal{X} = [0,1]$ and $\mathcal{D}$ be uniform, but this time $\mathbb{C}$ is the set of intervals $I_{[a,b]}$ such that for $x \in [0,1]$, $I_{[a,b]}(x) = +1$ iff $x \in [a,b]$ (for $a, b \in [0,1]$, $a \leq b$). In contrast to thresholds, the space of intervals serves as a canonical example of situations where active learning *does not help* compared to passive learning. This fact clearly shows itself in the disagreement coefficient, which is $\frac{1}{\nu+\epsilon}$ here, since $\Delta_r = 1$ for all $r > \nu + \epsilon$. To see this, note that the set $B(I_{[0,0]}, r)$ contains all concepts of the form $I_{[a,a]}$. Note that $\frac{1}{\nu+\epsilon}$ is the largest possible value for θ.

An interesting extension of the intervals example is the space of *p-intervals*, or all intervals $I_{[a,b]}$ such that $b - a \geq p \in ((\nu+\epsilon)/2, 1/8)$. These spaces span the range of difficulty, with active learning becoming easier as p increases. This is reflected in the θ value, since here $\theta = \frac{1}{2p}$. When $r < 2p$, every interval in $B(I_{[a,b]}, r)$ has its lower and upper boundaries within r of a and b, respectively; thus, $\Delta_r \leq 4r$. However, when $r \geq 2p$, *every* interval of width p is in $B(I_{[0,p]}, r)$, so $\Delta_r = 1$.

As an example that takes a (small) step closer to realistic learning scenarios, consider the following theorem.

Theorem 1. *If $\mathcal{X}$ is the surface of the origin-centered unit sphere in $\mathbb{R}^d$ for $d > 2$, $\mathbb{C}$ is the space of homogeneous linear separators*[1]*, and $\mathcal{D}$ is the uniform distribution on $\mathcal{X}$, then the disagreement coefficient θ satisfies*

$$\frac{1}{4}\min\left\{\pi\sqrt{d}, \frac{1}{\nu+\epsilon}\right\} \leq \theta \leq \min\left\{\pi\sqrt{d}, \frac{1}{\nu+\epsilon}\right\}.$$

Proof. First we represent the concepts in $\mathbb{C}$ as weight vectors $w \in \mathbb{R}^d$ in the usual way. For $w_1, w_2 \in \mathbb{C}$, by examining the projection of $\mathcal{D}$ onto the subspace spanned by $\{w_1, w_2\}$, we see that $\rho_{\mathcal{D}}(w_1, w_2) = \frac{arccos(w_1 \cdot w_2)}{\pi}$. Thus, for any $w \in \mathbb{C}$ and $r \leq 1/2$, $B(w,r) = \{w' : w \cdot w' \geq cos(\pi r)\}$. Since the decision boundary corresponding to w' is orthogonal to the vector w', some simple trigonometry gives us that

$$DIS(B(w,r)) = \{x \in \mathcal{X} : |x \cdot w| \leq sin(\pi r)\}.$$

Letting $A(n,R) = \frac{2\pi^{n/2}R^{n-1}}{\Gamma\left(\frac{n}{2}\right)}$ denote the surface area of the radius-R sphere in $\mathbb{R}^n$, we can express the disagreement rate at radius r as

$$\begin{aligned}
\Delta_r &= \frac{1}{A(d,1)}\int_{-sin(\pi r)}^{sin(\pi r)} A\left(d-1, \sqrt{1-x^2}\right) dx \\
&= \frac{\Gamma\left(\frac{d}{2}\right)}{\sqrt{\pi}\,\Gamma\left(\frac{d-1}{2}\right)}\int_{-sin(\pi r)}^{sin(\pi r)} \left(1-x^2\right)^{\frac{d-2}{2}} dx \qquad (*) \\
&\leq \frac{\Gamma\left(\frac{d}{2}\right)}{\sqrt{\pi}\,\Gamma\left(\frac{d-1}{2}\right)} 2sin(\pi r) \\
&\leq \sqrt{d-2}\,sin(\pi r) \leq \sqrt{d}\pi r.
\end{aligned}$$

For the lower bound, note that $\Delta_{1/2} = 1$ so $\theta \geq \min\left\{2, \frac{1}{\nu+\epsilon}\right\}$, and thus we need only consider $\nu + \epsilon < \frac{1}{8}$. Supposing $\nu + \epsilon < r < \frac{1}{8}$, note that $(*)$ is at least

$$\begin{aligned}
&\geq \sqrt{\frac{d}{12}}\int_{-sin(\pi r)}^{sin(\pi r)} \left(1-x^2\right)^{\frac{d}{2}} dx \\
&\geq \sqrt{\frac{\pi}{12}}\int_{-sin(\pi r)}^{sin(\pi r)} \sqrt{\frac{d}{\pi}} e^{-d \cdot x^2} dx \\
&\geq \frac{1}{2}\min\left\{\frac{1}{2}, \sqrt{d}\,sin(\pi r)\right\} \geq \frac{1}{4}\min\left\{1, \pi\sqrt{d}r\right\} \qquad \square
\end{aligned}$$

Given knowledge of the disagreement coefficient for $\mathbb{C}$ under $\mathcal{D}$, the following lemma allows us to extend this to a bound for any $\mathcal{D}'$ λ-close to $\mathcal{D}$. The proof is straightforward, and left as an exercise.

---

[1] Homogeneous linear separators are those that pass through the origin.

```
Input: concept space $\mathbb{C}$, accuracy parameter $\epsilon \in (0,1)$, confidence parameter $\delta \in (0,1)$
Output: classifier $\hat{h} \in \mathbb{C}$
Let $\hat{n} = \log_2\left(\frac{64}{\epsilon^2}\left(d \ln \frac{8}{\epsilon} + \ln \frac{8}{\epsilon\delta}\right)\right) \log_2 \frac{4}{\epsilon}$, and let $\delta' = \delta/\hat{n}$
0. $V_0 \leftarrow \mathbb{C}$, $S_0 \leftarrow \varnothing$, $i \leftarrow 0$, $j_1 \leftarrow 0$, $k \leftarrow 1$
1. While $\Delta(V_i)\left(\min_{h \in V_i} UB(S_i, h, \delta') - \min_{h \in V_i} LB(S_i, h, \delta')\right) > \epsilon$
2.     $V_{i+1} \leftarrow \{h \in V_i : LB(S_i, h, \delta') \leq \min_{h' \in V_i} UB(S_i, h', \delta')\}$
3.     $i \leftarrow i + 1$
4.     If $\Delta(V_i) < \frac{1}{2}\Delta(V_{j_k})$
5.         $k \leftarrow k + 1$; $j_k \leftarrow i$
6.     $S'_i \leftarrow$ Rejection sample $2^{i - j_k}$ samples $x$ from $\mathcal{D}$ satisfying $x \in DIS(V_i)$
7.     $S_i \leftarrow \{(x, Oracle(x)) : x \in S'_i\}$
8. Return $\hat{h} = \arg\min_{h \in V_i} UB(S_i, h, \delta')$
```

Figure 1. The A^2 algorithm.

Lemma 1. *Suppose $\mathcal{D}'$ is such that, $\exists \lambda \in (0,1]$ s.t. for all measurable sets $A \subseteq \mathcal{X}$, $\lambda \mathcal{D}(A) \leq \mathcal{D}'(A) \leq \frac{1}{\lambda}\mathcal{D}(A)$. If Δ_r,θ,Δ'_r, and θ' are the disagreement rates at radius r and disagreement coefficients for $\mathcal{D}$ and $\mathcal{D}'$ respectively, then $\lambda \Delta_{\lambda r} \leq \Delta'_r \leq \frac{1}{\lambda}\Delta_{r/\lambda}$, and thus*

$$\lambda^2 \theta \leq \theta' \leq \frac{1}{\lambda^2}\theta.$$

5. Upper Bounds for the A^2 Algorithm

To prove bounds on the label complexity, we will additionally need to use some known results on finite sample rates of uniform convergence.

Definition 5. *Let d be the VC dimension of $\mathbb{C}$. For $m \in \mathbb{N}$, and $S \in (\mathcal{X} \times \{-1, 1\})^m$, define*

$$G(m, \delta) = \frac{1}{m} + \sqrt{\frac{\ln \frac{4}{\delta} + d \ln \frac{2em}{d}}{m}}.$$

$$UB(S, h, \delta) = \min\{er_S(h) + G(|S|, \delta), 1\},$$
$$LB(S, h, \delta) = \max\{er_S(h) - G(|S|, \delta), 0\}.$$

By convention, $G(0, \delta) = 1$. The following lemma is due to Vapnik (Vapnik, 1998).

Lemma 2. *For any distribution $\mathcal{D}_i$ over $\mathcal{X} \times \{-1, 1\}$, and any $m \in \mathbb{N}$, with probability at least $1 - \delta$ over the draw of $S \sim \mathcal{D}_i^m$, every $h \in \mathbb{C}$ satisfies*

$$|er_S(h) - er_{\mathcal{D}_i}(h)| \leq G(m, \delta).$$

In particular, this means

$$er_{\mathcal{D}_i}(h) - 2G(|S|, \delta) \leq LB(S, h, \delta) \leq$$
$$er_{\mathcal{D}_i}(h) \leq UB(S, h, \delta) \leq er_{\mathcal{D}_i}(h) + 2G(|S|, \delta).$$

Furthermore, for $\gamma > 0$, if $m \geq \frac{4}{\gamma^2}\left(2d \ln \frac{4}{\gamma} + \ln \frac{4}{\delta}\right)$, then $G(m, \delta) < \gamma$.

We use a (somewhat simplified) version of the A^2 algorithm, presented by Balcan et. al (Balcan et al., 2006). The algorithm is given in Figure 1.

The motivation behind the A^2 algorithm is to maintain a set of concepts V_i that we are confident contains any concepts with minimal error rate. If we can guarantee with statistical significance that a concept $h_1 \in V_i$ has error rate worse than another concept $h_2 \in V_i$, then we can safely remove the concept h_1 since it is suboptimal. To achieve such a statistical guarantee, the algorithm employs two-sided confidence intervals on the error rates of each classifier in the concept space; however, since we are only interested in the relative *differences* between error rates, on each iteration we obtain this confidence interval for the error rate when $\mathcal{D}$ is restricted to the *region of disagreement* $DIS(V_i)$. This restriction to the region of disagreement is the primary source of any improvements A^2 achieves over passive learning. We measure the progress of the algorithm by the reduction in the disagreement rate $\Delta(V_i)$; the key question in studying the number of label requests is bounding the number of random labeled examples from the region of disagreement that are sufficient to remove enough concepts from V_i to significantly reduce the measure of the region of disagreement.

Theorem 2. *If θ is the disagreement coefficient for $\mathbb{C}$, then with probability at least $1 - \delta$, given the inputs $\mathbb{C}$, ϵ, and δ, A^2 outputs $\hat{h} \in \mathbb{C}$ with $er(\hat{h}) \leq \nu + \epsilon$, and the number of label requests made by A^2 is at most*

$$O\left(\theta^2\left(\frac{\nu^2}{\epsilon^2} + 1\right)\left(d \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right) \log \frac{1}{\epsilon}\right).$$

Proof. Let κ be the value of k and ι be the value of i when the algorithm halts. By convention, let $j_{\kappa+1} = \iota + 1$. Let $\gamma_i = \max_{h \in V_i}(UB(S_i, h, \delta') - LB(S_i, h, \delta'))$. Since having $\gamma_i \leq \epsilon$ would break the loop at step 1, Lemma 2 implies we always have $|S_i| \leq \frac{16}{\epsilon^2}\left(2d\ln\frac{8}{\epsilon} + \ln\frac{4}{\delta'}\right)$, and thus $\iota \leq (\kappa + 1)\log_2\left(\frac{16}{\epsilon^2}\left(2d\ln\frac{8}{\epsilon} + \ln\frac{4}{\delta'}\right)\right)$. $\Delta(V_i) \leq \epsilon$ also suffices to break from the loop, so $\kappa \leq \log_2\frac{2}{\epsilon}$. Thus, $\iota \leq \hat{n}$. Lemma 2 and a union bound imply that, with probability $\geq 1-\delta$, for every i and every $h \in \mathbb{C}$, $|er_{S_i}(h) - er_{\mathcal{D}_i}(h)| \leq G(|S_i|, \delta')$, where $\mathcal{D}_i$ is the conditional distribution of $\mathcal{D}_{XY}$ given that $X \in DIS(V_i)$. For the remainder of this proof, we assume that these inequalities hold for all such S_i and $h \in \mathbb{C}$. In particular, this means we never remove the best classifier from V_i. Additionally, $\forall h_1, h_2 \in V_i$ we must have $\Delta(V_i)(er_{\mathcal{D}_i}(h_1) - er_{\mathcal{D}_i}(h_2)) = er(h_1) - er(h_2)$. Combined with the nature of the halting criterion, this implies that $er(\hat{h}) \leq \nu + \epsilon$, as desired.

The rest of the proof bounds the number of label requests made by A^2. Let $h^* \in V_i$ be such that $er(h^*) \leq \nu + \epsilon$. We consider two cases: large and small $\Delta(V_i)$. Informally, when $\Delta(V_i)$ is relatively large, the concepts far from h^* are responsible for most of the disagreements, and since these must have relatively large error rates, we need only a few examples to remove them. On the other hand, when $\Delta(V_i)$ is small, the halting condition is easy to satisfy.

We begin with the case where $\Delta(V_i)$ is large. Specifically, let $i' = \max\{i \leq \iota : \Delta(V_i) > 8\theta(\nu + \epsilon)\}$. (If no such i' exists, we can skip this case). Then $\forall i \leq i'$, let

$$V_i^{(\theta)} = \left\{h \in V_i : \rho_{\mathcal{D}}(h, h^*) > \frac{\Delta(V_i)}{2\theta}\right\}.$$

Since for $h \in V_i$, $\rho_{\mathcal{D}}(h,h^*)/\Delta(V_i) \leq er_{\mathcal{D}_i}(h) + er_{\mathcal{D}_i}(h^*) \leq er_{\mathcal{D}_i}(h) + \frac{\nu+\epsilon}{\Delta(V_i)}$, we have

$$\begin{aligned} V_i^{(\theta)} &\subseteq \left\{h \in V_i : er_{\mathcal{D}_i}(h) > \frac{1}{2\theta} - \frac{\nu+\epsilon}{\Delta(V_i)}\right\} \\ &\subseteq \left\{h \in V_i : er_{\mathcal{D}_i}(h) - \frac{1}{8\theta} > er_{\mathcal{D}_i}(h^*) + \frac{3}{8\theta} - 2\frac{\nu+\epsilon}{\Delta(V_i)}\right\} \\ &\subseteq \left\{h \in V_i : er_{\mathcal{D}_i}(h) - \frac{1}{8\theta} > er_{\mathcal{D}_i}(h^*) + \frac{1}{8\theta}\right\}. \end{aligned}$$

Let $\bar{V}_i$ denote the latter set. By Lemma 2, S_i of size $O\left(\theta^2\left(d\log\theta + \log\frac{1}{\delta'}\right)\right)$ suffices to guarantee every $h \in \bar{V}_i$ has $LB(S_i, h, \delta') > UB(S_i, h^*, \delta')$ in step 2. $V_i^{(\theta)} \subseteq \bar{V}_i$ and $\Delta(V_i \setminus V_i^{(\theta)}) \leq \Delta_{\frac{\Delta(V_i)}{2\theta}} \leq \frac{1}{2}\Delta(V_i)$, so in particular, any value of k for which $j_k \leq i' + 1$ satisfies $|S_{j_k - 1}| = O\left(\theta^2\left(d\log\theta + \log\frac{1}{\delta'}\right)\right)$.

To handle the remaining case, suppose $\Delta(V_i) \leq 8\theta(\nu + \epsilon)$. In this case, S_i of size $O\left(\theta^2\frac{(\nu+\epsilon)^2}{\epsilon^2}\left(d\log\frac{1}{\epsilon} + \log\frac{1}{\delta'}\right)\right)$ suffices to make $\gamma_i \leq \frac{\epsilon}{\Delta(V_i)}$, satisfying the halting condition. Therefore, every k for which $j_k > i' + 1$ satisfies $|S_{j_k - 1}| = O\left(\theta^2\frac{(\nu+\epsilon)^2}{\epsilon^2}\left(d\log\frac{1}{\epsilon} + \log\frac{1}{\delta'}\right)\right)$.

Since for $k > 1$, $\sum_{i=j_{(k-1)}}^{j_k - 1}|S_i| \leq 2|S_{j_k - 1}|$, we have that $\sum_{i=1}^{\iota}|S_i| = O\left(\theta^2\frac{(\nu+\epsilon)^2}{\epsilon^2}\left(d\log\frac{1}{\epsilon} + \log\frac{1}{\delta'}\right)\kappa\right)$. Noting that $\kappa = O(\log\frac{1}{\epsilon})$ and $\log\frac{1}{\delta'} = O\left(d\log\frac{1}{\epsilon} + \log\frac{1}{\delta}\right)$ completes the proof. □

Note that we can get an easy improvement to the bound by replacing $\mathbb{C}$ with an $\frac{\epsilon}{2}$-cover of $\mathbb{C}$, using bounds for a finite concept space instead of VC bounds, and running the algorithm with accuracy parameter $\frac{\epsilon}{2}$. This yields a similar, but sometimes much tighter, label complexity bound of

$$O\left(\theta^2\left(\frac{\nu^2}{\epsilon^2} + 1\right)\log\frac{N(\epsilon/2)\log\frac{1}{\epsilon}}{\delta}\log\frac{1}{\epsilon}\right).$$

6. Lower Bounds for the A^2 Algorithm

In this section, we prove a lower bound on the worst-case number of label requests made by A^2. As mentioned in Section 2, there are known lower bounds of $\Omega\left(\frac{\nu^2}{\epsilon^2}\log\frac{1}{\delta}\right)$ and $\Omega\left(\log N(2\epsilon)\right)$, than which *no* algorithm can guarantee better (Kulkarni et al., 1993; Kääriäinen, 2006). However, this leaves open the question of whether the θ^2 factor in the bound is necessary. The following theorem shows that it is for A^2.

Theorem 3. *For any $\mathbb{C}$ and $\mathcal{D}$, there exists an oracle with $\nu = 0$ such that, if θ is the disagreement coefficient, with probability $1-\delta$, the version of A^2 presented above makes a number of label requests at least*

$$\Omega\left(\theta^2\left(d\log\theta + \log\frac{1}{\delta}\right)\right).$$

Proof. The bound clearly holds if $\theta = 0$, so assume $\theta > 0$. By definition of disagreement coefficient, there is some $\alpha_0 > 0$ such that $\forall \alpha \in (0, \alpha_0)$, $\exists r_\alpha \in (\epsilon, 1], h_\alpha \in \mathbb{C}$ such that $\Delta(B(h_\alpha, r_\alpha)) \geq \Delta_{r_\alpha} - \alpha \geq \theta r_\alpha - 2\alpha > 0$. For some such α, let $Oracle(x) = h_\alpha(x)$ for all $x \in \mathcal{X}$. Clearly $\nu = 0$. As before, we assume all bound evaluations in the algorithm are valid, which occurs with probability $\geq 1 - \delta$. Since $LB(S_i, h_\alpha, \delta') = 0$ and $UB(S_i, h_\alpha, \delta') = G(|S_i|, \delta')$, if A^2 halts without removing any $h \in B(h_\alpha, r_\alpha)$, then $\exists i : UB(S_i, h_\alpha, \delta') \leq \frac{\epsilon}{\Delta(B(h_\alpha, r_\alpha))} \leq \frac{\epsilon}{\theta r_\alpha - 2\alpha} \leq \frac{r_\alpha}{\theta r_\alpha - 2\alpha}$. On the other hand, suppose A^2 removes some $h \in B(h_\alpha, r_\alpha)$ before halting, and in particular suppose the first time this happens is for some set S_i. In this case, $UB(S_i, h_\alpha, \delta') < LB(S_i, h, \delta') \leq er_{\mathcal{D}_i}(h) \leq \frac{er(h)}{\Delta(B(h_\alpha, r_\alpha))} \leq \frac{r_\alpha}{\theta r_\alpha - 2\alpha}$. In either case, by definition of $G(|S_i|, \delta')$, we must have $|S_i| = \Omega\left(\left(\theta - \frac{2\alpha}{r_\alpha}\right)^2\left(d\log\left(\theta - \frac{2\alpha}{r_\alpha}\right) + \log\frac{1}{\delta'}\right)\right)$. Since this is true for any such α, taking the limit as $\alpha \to 0$ proves the bound. □

Theorems 2 and 3 show that the variation in worst-case number of label requests made by A^2 for different $\mathbb{C}$ and $\mathcal{D}$ is largely determined by the disagreement coefficient (and VC dimension). Furthermore, they give us a good estimate of the number of label requests made by A^2. One natural question to ask is whether Theorem 2 is also tight for the *label complexity* of the learning problem. The following example indicates this is not the case. In particular, this means that A^2 can sometimes be suboptimal.

Suppose $\mathcal{X} = [0,1]^n$, and $\mathbb{C}$ is the space of axis-aligned rectangles on $\mathcal{X}$. That is, each $h \in \mathbb{C}$ can be expressed as n pairs $((a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n))$, such that $\forall x \in \mathcal{X}, h(x) = 1$ iff $\forall i, a_i \leq x_i \leq b_i$. Furthermore, suppose $\mathcal{D}$ is the uniform distribution on $\mathcal{X}$. We see immediately that $\theta = \frac{1}{\epsilon+\nu}$, since $\forall r > 0, \Delta_r = 1$. We will show the bound is not tight for the case when $\nu = 0$.[2] In this case, the bound value is $\Omega\left(\frac{1}{\epsilon^2}\left(n \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right)$.

Theorem 4. *When $\nu = 0$, the agnostic active learning label complexity of axis-aligned rectangles on $[0,1]^n$ with respect to the uniform distribution is at most*

$$O\left(n \log \frac{n}{\epsilon\delta} + \frac{1}{\epsilon} \log \frac{1}{\delta}\right).$$

A proof sketch for Theorem 4 is included in Appendix B. This clearly shows that the bound based on A^2 is sometimes not tight with respect to the true label complexity of learning problems. Furthermore, when $\epsilon < \frac{1}{en}$, this problem has $\log N(\epsilon/2) \geq n$, so the improvements offered by learning with an $\frac{\epsilon}{2}$-cover cannot reduce the slack by much here (see Lemma 3 in Appendix B).

7. Open Problems

Whether or not one can modify A^2 in a general way to improve this bound is an interesting open problem. One possible strategy would be to use Occam bounds, and adaptively set the prior for each iteration, while also maintaining several different types of bounds simultaneously. However, it seems that in order to obtain the dramatic improvements needed to close the gap demonstrated by Theorem 4, we need a more aggressive strategy than sampling randomly from $DIS(V_i)$. For example, Balcan, Broder & Zhang (Balcan et al., 2007) present an algorithm for linear separators which samples from a carefully chosen subregion of $DIS(V_i)$. Though their analysis is for a restricted noise model, we might hope a similar idea is possible in the agnostic model. The end of Appendix A contains another interesting example that highlights this issue.

One important aspect of active learning that has not been addressed here is the value of unlabeled examples. Specifically, given an overabundance of unlabeled examples, can we use them to decrease the number of label requests required, and by how much? The splitting index bounds of Dasgupta (Dasgupta, 2005) can be used to study these types of questions in the noise-free setting; however, we have yet to see a thorough exploration of the topic for agnostic learning, where the role of unlabeled examples appears fundamentally different (at least in A^2).

Acknowledgments

I am grateful to Nina Balcan for helpful discussions.

This research was sponsored through a generous grant from the Commonwealth of Pennsylvania. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the sponsoring body, or other institution or entity.

References

Balcan, M.-F., Beygelzimer, A., & Langford, J. (2006). Agnostic active learning. *Proc. of the 23rd International Conference on Machine Learning.*

Balcan, M.-F., Broder, A., & Zhang, T. (2007). Margin based active learning. *Proc. of the 20th Conference on Learning Theory.*

Benedek, G., & Itai, A. (1988). Learnability by fixed distributions. *Proc. of the First Workshop on Computational Learning Theory* (pp. 80–90).

Blumer, A., Ehrenfeucht, A., Haussler, D., & Warmuth, M. (1989). Learnability and the vapnik-chervonenkis dimension. *Journal of the Association for Computing Machinery*, *36*, 929–965.

Dasgupta, S. (2005). Coarse sample complexity bounds for active learning. *Advances in Neural Information Processing Systems 18.*

Haussler, D. (1992). Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, *100*, 78–150.

---

[2] In this particular case, the agnostic label complexity with $\nu = 0$ is within constant factors of the realizable complexity. However, in general, agnostic learning with $\nu = 0$ is *not* the same as realizable learning, since we are still interested in algorithms that would tolerate noise if it were present. See Appendix A for an interesting example.

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

Kulkarni, S. R. (1989). *On metric entropy, vapnik-chervonenkis dimension, and learnability for a class of distributions* (Technical Report CICS-P-160). Center for Intelligent Control Systems.

Kulkarni, S. R., Mitter, S. K., & Tsitsiklis, J. N. (1993). Active learning using arbitrary binary valued queries. *Machine Learning*, *11*, 23–35.

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

Vapnik, V. (1998). *Statistical learning theory.* John Wiley & Sons, Inc.

A. Realizable vs. Agnostic with $\nu = 0$

The following example indicates that agnostic active learning with $\nu = 0$ is sometimes fundamentally more difficult than realizable learning.

Let $\epsilon < 1/4$, $N = \left\lfloor \frac{1}{2\epsilon} \right\rfloor$. Let $\mathcal{X} = \mathbb{Z}$, and define $\mathcal{D}$ such that, for $x \in \mathcal{X} : 0 < x \leq N$, $\mathcal{D}(x) = \frac{\epsilon}{4N}$ and $\mathcal{D}(-x) = \frac{1-\epsilon/4}{N}$. $\mathcal{D}$ gives zero probability elsewhere. In particular, note that $\frac{3}{2}\epsilon < \mathcal{D}(-x) \leq 4\epsilon$ and $\frac{\epsilon^2}{2} \leq \mathcal{D}(x) \leq \epsilon^2$.

Define concept space $\mathbb{C} = \{h_1, h_2, \ldots\}$, where $\forall i, j \in \{1, 2, \ldots\}$, $h_i(0) = -1$ and

$$\begin{aligned} h_i(-j) &= \quad 2I[i = j] - 1 \\ h_i(j) &= \quad 2I[j \geq i] - 1. \end{aligned}$$

Note that this creates a learning problem where informative examples exist (the $x \in \{1, \ldots, N\}$ examples) but are rare.

Theorem 5. *For the learning problem described above, the realizable active learning label complexity is* $O\left(\log \frac{1}{\epsilon}\right)$.

Proof. By Chernoff and union bounds, drawing $\Theta\left(\frac{1}{\epsilon^2} \log \frac{1}{\epsilon\delta}\right)$ unlabeled examples suffices to guarantee, with probability at least $1 - \delta$, we have at least one unlabeled example of x, for all $x \in \{1, 2, \ldots, N\}$; suppose this happens. Suppose $f \in \mathbb{C}$ is the target function. If $f \notin \{h_1, h_2, \ldots, h_N\}$, querying the label of $x = N$ suffices to show $er(h_{N+1}) = 0$, so we output h_{N+1}. On the other hand, if we find $f(N) = +1$, we can perform binary search among the $\{1, 2, \ldots, N\}$ to find the smallest $i > 0$ such that $f(i) = +1$. In this case, we must have $h_i = f$, so we output h_i after $O(\log N)$ queries. ☐

Theorem 6. *For the learning problem described above, any agnostic active learning algorithm requires* $\Omega\left(\frac{1}{\epsilon}\right)$ *label requests, even if the oracle always agrees with some* $f \in \mathbb{C}$, *(i.e., even if* $\nu = 0$*).*

Proof. Suppose A is a correct agnostic learning algorithm. The idea of the proof is to assume A is guaranteed to make fewer than $(1 - 2\delta)N$ queries with probability $\geq 1 - \delta$ when the target function is some particular $f \in \mathbb{C}$, and then show that by adding noise we can force A to output a concept with error more than ϵ-worse than optimal with probability $> \delta$. Thus, either A cannot guarantee fewer than $(1 - 2\delta)N$ queries for that particular f, or A is not a correct agnostic learning algorithm.

Specifically, suppose that when the target function $f = h_{N+1}$, with probability $\geq 1 - \delta$ A returns an ϵ-good concept after making $\leq q < (1 - 2\delta)N$ label requests. If A is successful, then whatever concept it outputs labels all of $\{-1, -2, \ldots, -N\}$ as -1. So in particular, letting the random variable $R = (R_1, R_2, \ldots)$ denote the sequence of examples A requests the labels of when *Oracle* agrees with h_{N+1}, this implies that with probability at least $1 - \delta$, if $Oracle(R_i) = h_{N+1}(R_i)$ for $i \in \{1, 2, ldots, \min\{q, |R|\}\}$, then A outputs a concept labeling all of $\{-1, -2, \ldots, -N\}$ as -1.

Now suppose instead of h_{N+1}, we pick the target function f' as follows. Let f' be identical to h_{N+1} on all of $\mathcal{X}$ except a single $x \in \{-1, -2, \ldots, -N\}$ where $f'(x) = +1$; the value of x for which this happens is chosen uniformly at random from $\{-1, -2, \ldots, -N\}$. Note that $f' \notin \mathbb{C}$. Also note that any concept in $\mathbb{C}$ other than h_{-x} is $> \epsilon$-worse than h_{-x}.

Now consider the behavior of A when Oracle answers queries with this f' instead of h_{N+1}. Let $Q = (Q_1, Q_2, \ldots)$ denote the random sequence of examples A queries the labels of when *Oracle* agrees with f'. In particular, note that if $R_i \neq x$ for $i \leq \min\{q, |R|\}$, then $Q_i = R_i$ for $i \leq \min\{q, |Q|\}$.

$$\begin{aligned} &\mathbb{E}_{f'}\left[\mathcal{P}r\{A \text{ outputs } h_{-x}\}\right] \\ &\qquad \leq \mathbb{E}_R\left[\mathcal{P}r_x\{\exists i \leq q : R_i = x\}\right] + \delta < 1 - \delta. \end{aligned}$$

By the probabilistic method, we have proven that there exists some fixed oracle such that A fails with probability $> \delta$. This contradicts the premise that A is a correct agnostic learning algorithm. ☐

As an interesting aside, note that if we define $\mathbb{C}_\epsilon = \{h_1, h_2, \ldots, h_N\}$, dependent on ϵ, then the agnostic label complexity is $O\left(\log \frac{1}{\epsilon\delta}\right)$ when $\nu = 0$. This is because we can run the realizable learning algorithm to find $f = h_i$, and then sample $\Theta\left(\log \frac{1}{\delta}\right)$ labeled copies of the example $-i$; by observing that they are all labeled $+1$, we effectively *verify* that h_i is at most ϵ-worse than optimal. To make this a correct agnostic algorithm, we can simply be prepared to run A^2 if any of the $\Theta\left(\log \frac{1}{\delta}\right)$ samples of $-i$ are labeled -1 (which they won't be for $\nu = 0$). However, since the disagreement coefficient $\theta = \Theta\left(\frac{1}{\epsilon}\right)$, Theorem 3 implies A^2 does not achieve this improvement. See Appendix B for a similar example.

B. Axis-Aligned Rectangles

Proof Sketch of Theorem 4. To keep things simple, we omit the precise constants. Consider the following algorithm.[3]

```
0. Sample $\Theta\left(\frac{1}{\epsilon}\log\frac{1}{\delta}\right)$ labeled examples from $\mathcal{D}_{XY}$
1. If none of them are positive,
       return the "all negative" concept
2. Else let $x$ be one of the positive examples
3. For $i = 1, 2, \ldots, n$
4.     Rejection sample unlabeled set $\mathcal{U}_i$ of size
         $\Theta\left(\frac{n}{\epsilon\delta}\left(\log\frac{n}{\delta}\right)^2\right)$ from the conditional of $\mathcal{D}$ given
         $\forall j \neq i, x_j - O\left(\frac{\epsilon\delta}{n\log\frac{1}{\delta}}\right) \leq X_j \leq x_j + O\left(\frac{\epsilon\delta}{n\log\frac{1}{\delta}}\right)$
5.     Find $\hat{b}_i = \max\{z_i : z \in \mathcal{U}_i \cup \{x\}, Oracle(z) = +1\}$
         by binary search in $\{z_i : z \in \mathcal{U}_i \cup \{x\}, z_i \geq x_i\}$
6.     Find $\hat{a}_i = \min\{z_i : z \in \mathcal{U}_i \cup \{x\}, Oracle(z) = +1\}$
         by binary search in $\{z_i : z \in \mathcal{U}_i \cup \{x\}, z_i \leq x_i\}$
7. Let $\hat{h} = ((\hat{a}_1, \hat{b}_1), (\hat{a}_2, \hat{b}_2), \ldots, (\hat{a}_n, \hat{b}_n))$
8. Sample $\Theta\left(\frac{1}{\epsilon}\log\frac{1}{\delta}\right)$ labeled examples $T$ from $\mathcal{D}_{XY}$
9. If $er_T(\hat{h}) > 0$,
       run $A^2$ from the start and return its output
10.Else return $\hat{h}$
```

The correctness of the algorithm in the agnostic setting is clear from examining the three ways to exit the algorithm. First, any oracle with $\mathcal{P}r_{X\sim\mathcal{D}}\{Oracle(X) = +1\} > \epsilon$ will, with probability $\geq 1 - O(\delta)$ have a positive example in the initial $\Theta\left(\frac{1}{\epsilon}\log\frac{1}{\delta}\right)$ sample. So if the set has no positives, we can be confident the "all negative" concept has error $\leq \epsilon$. If we return in step 9, we know from Theorem 2 that A^2 will, with probability $1 - O(\delta)$, output a concept with error $\leq \nu + \epsilon$. The remaining possibility is to return in step 10. Any $\hat{h}$ with $er(\hat{h}) > \epsilon$ will, with probability $\geq 1 - O(\delta)$, have $er_T(\hat{h}) > 0$ in step 9. So we can be confident the $\hat{h}$ output in step 10 has $er(\hat{h}) \leq \epsilon$.

To bound the number of label requests, note that the two binary searches we perform for each i (steps 5 and 6) require only $O(\log|\mathcal{U}_i|)$ label requests each, so the entire For loop uses only $O\left(n\log\frac{n}{\epsilon\delta}\right)$ label requests. We additionally have the two labeled sets of size $O\left(\frac{1}{\epsilon}\log\frac{1}{\delta}\right)$, so if we do not return in step 9, the total number of label requests is at most $O\left(n\log\frac{n}{\epsilon\delta} + \frac{1}{\epsilon}\log\frac{1}{\delta}\right)$.

It only remains to show that when $\nu = 0$, we do not return in step 9. Let $f = ((a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n))$ be a rectangle with $er(f) = 0$. Note that $er(\hat{h}) \leq \sum_{i=1}^{n} |a_i - \hat{a}_i| + |b_i - \hat{b}_i|$. For each i, with probability $1 - O(\delta/n)$, none of the initial $\Theta\left(\frac{1}{\epsilon}\log\frac{1}{\delta}\right)$ examples w has $w_i \in [a_i, a_i + \gamma] \cup [b_i - \gamma, b_i]$, where $\gamma = O\left(\frac{\epsilon\delta}{n\log\frac{1}{\delta}}\right)$. In particular, if we do not return in step 1, with probability $1 - O(\delta)$, $\forall j, x_j \in [a_j + \gamma, b_j - \gamma]$. Suppose this happens. In particular, this means the oracle's labels for all $z \in \mathcal{U}_i$ are completely determined by whether $a_i \leq z_i \leq b_i$. We can essentially think of this as two "threshold" learning problems for each i: one above x_i and one below x_i. The binary searches find threshold values consistent with each $\mathcal{U}_i$. In particular, by standard passive sample complexity arguments, $|\mathcal{U}_i|$ is sufficient to guarantee with probability $1 - O(\delta/n)$, $|b_i - \hat{b}_i| \leq O\left(\frac{\epsilon\delta}{n\log\frac{1}{\delta}}\right)$ and $|a_i - \hat{a}_i| \leq O\left(\frac{\epsilon\delta}{n\log\frac{1}{\delta}}\right)$. Thus, with probability $1 - O(\delta)$, $er(\hat{h}) \leq O\left(\frac{\epsilon\delta}{\log\frac{1}{\delta}}\right)$. Therefore, the probability $\hat{h}$ makes a mistake on T of size $O\left(\frac{1}{\epsilon}\log\frac{1}{\delta}\right)$ is at most $O(\delta)$. Otherwise, we have $er_T(\hat{h}) = 0$ in step 9, so we return in step 10. □

Lemma 3. *If $\mathbb{C}$ is the space of axis-aligned rectangles on $[0,1]^n$, and $\mathcal{D}$ is the uniform distribution, then for $\epsilon < \frac{1}{en}$, $\log_2 N(\epsilon/2) \geq n$.*

Proof. Since $N(\epsilon/2)$ is at least the size of any ϵ-separated set, we can prove this lower bound by constructing an ϵ-separated set of size 2^n. In particular, consider the set of all rectangles $((a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n))$ satisfying $\forall i, a_i = 0, b_i \in \left\{1 - \frac{1}{n}, 1\right\}$. There are 2^n such rectangles.

For any two distinct such rectangles $((a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n))$ and $((a'_1, b'_1), (a'_2, b'_2), \ldots, (a'_n, b'_n))$, there is at least one i such that $b_i \neq b'_i$. So the region in which these two disagree contains $\left\{x \in \mathcal{X} : x_i \in \left(1 - \frac{1}{n}, 1\right], \forall j \neq i, x_j \in \left[0, 1 - \frac{1}{n}\right]\right\}$, which has measure $\left(1 - \frac{1}{n}\right)^{n-1}\frac{1}{n} \geq \frac{1}{en} > \epsilon$. □

---

[3] To keep the algorithm simple, we make little attempt to optimize the number of unlabeled examples. In particular, we could reduce $|\mathcal{U}_i|$ by using a nonzero cutoff in step 9, and could increase the window size in step 4 by using a noise-tolerant active threshold learner in steps 5 and 6.

Teaching Dimension and the Complexity of Active Learning

Steve Hanneke

Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213 USA
`shanneke@cs.cmu.edu`

Abstract. We study the label complexity of pool-based active learning in the PAC model with noise. Taking inspiration from extant literature on Exact learning with membership queries, we derive upper and lower bounds on the label complexity in terms of generalizations of *extended teaching dimension*. Among the contributions of this work is the first nontrivial general upper bound on label complexity in the presence of persistent classification noise.

1 Overview of Main Results

In supervised machine learning, it is becoming increasingly apparent that well-designed interactive learning algorithms can provide valuable improvements over passive algorithms in learning performance while reducing the amount of effort required of a human annotator. In particular, there is presently much interest in the pool-based active learning setting, in which a learner can request the label of any example in a large pool of unlabeled examples. In this case, one crucial quantity is the number of label requests required by a learning algorithm: the *label complexity*. This quantity is sometimes significantly smaller than the sample complexity of passive learning. A thorough theoretical understanding of these improvements seems essential to fully exploit the potential of active learning.

In particular, active learning is formalized in the PAC model as follows. The pool of m unlabeled examples are sampled i.i.d. according to some distribution $\mathcal{D}$. A binary label is assigned to each example by a (possibly randomized) oracle, but is hidden from the learner unless it requests the label. The *error rate* of a classifier h is defined as the probability of h disagreeing with the oracle on a fresh example $X \sim \mathcal{D}$. A learning algorithm outputs a classifier $\hat{h}$ from a *concept space* $\mathbb{C}$, and we refer to the infimum error rate over classifiers in $\mathbb{C}$ as the *noise rate*, denoted ν. For $\epsilon, \delta, \eta \in (0,1)$, we define the *label complexity*, denoted $\#LQ(\mathbb{C}, \mathcal{D}, \epsilon, \delta, \eta)$, as the smallest number q such that there is an algorithm that outputs a classifier $\hat{h} \in \mathbb{C}$, and for sufficiently large m, for any oracle with $\nu \leq \eta$, with probability at least $1 - \delta$ over the sample and internal randomness, the algorithm makes at most q label requests and $\hat{h}$ has error rate at most $\nu + \epsilon$.[1]

---

[1] Alternatively, if we know q ahead of time, we can have the algorithm halt if it ever tries to make more than q queries. The analysis is nearly identical in either case.

N. Bshouty and C. Gentile (Eds.): COLT 2007, LNAI 4539, pp. 66-81, 2007.
© Springer-Verlag Berlin Heidelberg 2007
Revised 04/2007.

The careful reader will note that this definition does not require the algorithm to be successful if $\nu > \eta$, distinguishing this from the fully agnostic setting [1]; we discuss possible methods to bridge this gap in later sections.

Kulkarni [2] has shown that if there is no noise, and one is allowed arbitrary binary valued queries, then $O\left(\log N(\epsilon)\right) \leq O\left(d \log \frac{1}{\epsilon}\right)$ queries suffice to PAC learn, where $N(\epsilon)$ denotes the size of a minimal ϵ-cover of $\mathbb{C}$ with respect to $\mathcal{D}$, and d is the VC dimension of $\mathbb{C}$. This bound often has exponentially better dependence on $\frac{1}{\epsilon}$, compared to the sample complexity of passive learning. However, many binary valued queries are unnatural and difficult to answer in practice. One of the driving motivations for research on the label complexity of active learning is identifying, in a general way, which concept spaces and distributions allow us to obtain this exponential improvement using only label requests for examples in the unlabeled sample. A further question is whether such improvements can be sustained in the presence of classification noise. In this paper, we investigate these questions from the perspective of a general analysis.

On the subject of learning through interaction, there is a rich literature concerning the complexity of Exact learning with membership queries [3, 4]. The interested reader should consult the limpid survey by Angluin [4]. The essential distinction between that setting and the setting we are presently concerned with is that, in Exact learning, the learning algorithm is required to *identify* the oracle's actual target function, rather than *approximating* it with high probability; on the other hand, in the Exact setting there is no classification noise and the algorithm can ask for the label of *any* example. In a sense, Exact learning with membership queries is a limiting case of PAC active learning. As such, we may hope to draw inspiration from the extant work on Exact learning when formulating an analysis for the PAC setting.

To quantify $\#MQ(\mathbb{C})$, the worst-case number of membership queries required for Exact learning with concept space $\mathbb{C}$, Hegedüs [3] defines a quantity called the *extended teaching dimension* of $\mathbb{C}$, based on the *teaching dimension* of Goldman & Kearns [5]. Letting t_0 denote this quantity, Hegedüs proves that

$$\max\{t_0, \log_2 |\mathbb{C}|\} \leq \#MQ(\mathbb{C}) \leq t_0 \log_2 |\mathbb{C}|,$$

where the upper bound is achieved by a version of the Halving algorithm.

Inspired by these results, we generalize the extended teaching dimension to the PAC setting, adding dependences on ϵ, δ, η, and $\mathcal{D}$. Specifically, we define two quantities, t and $\tilde{t}$, both of which have t_0 as a limiting case. We show that

$$\Omega\left(\max\left\{\frac{\eta^2}{\epsilon^2}, \tilde{t}, \log N(2\epsilon)\right\}\right) \leq \#LQ(\mathbb{C}, \mathcal{D}, \epsilon, \delta, \eta) \leq \tilde{O}\left(\left(\frac{\eta^2}{\epsilon^2}+1\right) t \log N(\epsilon/2)\right)$$

where $\tilde{O}$ hides factors logarithmic in $\frac{1}{\epsilon}, \frac{1}{\delta}$, and d. The upper bound is achieved by an active learning algorithm inspired by the Halving algorithm, which uses $\tilde{O}\left(d \frac{\eta+\epsilon}{\epsilon^2}\right)$ unlabeled examples. With these tools in hand, we analyze the label complexity of axis-aligned rectangles with respect to product distributions, showing improvements over known passive learning results in dependence on η when positive examples are not too rare.

The rest of the paper is organized as follows. In Section 2, we briefly survey the related literature on the label complexity of active learning. This is followed in Section 3 with the introduction of definitions and notation, and a brief discussion of known results for Exact learning in Section 4. In Section 5, we move into results for the PAC setting, beginning with the noise-free case for simplicity. Then, in Section 6, we describe the general setting, and prove an upper bound on the label complexity of active learning with noise; to the author's knowledge, this is the first general result of its kind, and along with lower bounds on label complexity presented in Section 7, represents the primary contribution of this work. We continue in Section 8, with an application of these bounds to describe the label complexity of axis-aligned rectangles with product distributions. We conclude with some enticing open problems in Section 9.

2 Context and Related Work

The recent literature studying general label complexity can be coarsely partitioned by the measure of progress used in the analysis. Specifically, there are at least three distinct ways to measure the progress of an active learning algorithm: *diameter* of the version space, *measure* of the region of disagreement, and *size* of the version space. By the *version space* at a time during the algorithm execution, we mean the set of concepts in $\mathbb{C}$ that have not yet been ruled out as a possible output. One approach to studying label complexity is to summarize in a single quantity how easy it is to make progress in terms of one of these progress metrics. This quantity, apart from itself being interesting, can then be used to derive upper and lower bounds on the label complexity.

To study the ease of reducing the diameter of the version space in active learning, Dasgupta [6] defines a quantity ρ he calls the *splitting index.* ρ is dependent on $\mathbb{C}$, $\mathcal{D}$, ϵ, and another parameter τ he defines, as well as the oracle itself. Dasgupta finds that when the noise rate is zero, roughly $\tilde{O}(\frac{d}{\rho})$ label requests are sufficient, and $\Omega(\frac{1}{\rho})$ are necessary for learning (for respectively appropriate τ values). However, Dasgupta's analysis is restricted to the noise-free case, and there are no known extensions addressing the noisy case.

In studying ways to enable active learning in the presence of noise, Balcan et al. [1] propose the A^2 algorithm. This algorithm is able to learn in the presence of arbitrary classification noise. The strategy behind A^2 is to induce confidence intervals for the differences of error rates of concepts in the version space. If an estimated difference is statistically significant, the algorithm removes the worst of the two concepts. The key observation is that, since the algorithm only estimates error *differences*, there is no need to request the label of any example that all remaining concepts agree on. Thus, the number of label requests made by A^2 is largely controlled by how quickly the *region of disagreement* collapses as the algorithm progresses. However, apart from fall-back guarantees and a few special cases, there is presently no published general analysis of the number of label requests made by A^2, and no general index of how easy it is to reduce the region of disagreement.

The third progress metric is reduction in the *size* of the version space. If the concept space is infinite, an ϵ'-cover of $\mathbb{C}$ can be substituted for $\mathbb{C}$, for some suitable ϵ'.[2] This paper presents the first general study of the ease of reducing the size of the version space. The corresponding index summarizing the potential for progress in this metric remains informative in the presence of noise, given access to an upper bound on the noise rate.

In addition to the above studies, Kääriäinen [7] presents an interesting analysis of active learning with various types of noise. Specifically, he proves that under noise that is not persistent (in that requesting the same label twice may yield different responses) and where the Bayes optimal classifier is in $\mathbb{C}$, any algorithm that is successful for the zero noise setting can be transformed into a successful algorithm for the noisy setting with only a small increase in the number of label requests. However, these positive results do not carry into our present setting (*arbitrary persistent* classification noise). In fact, in addition to these positive results, Kääriäinen [7] presents negative results in the form of a general lower bound on the label complexity of active learning with arbitrary (persistent) classification noise. Specifically, he finds that for most nontrivial distributions $\mathcal{D}$, one can force any algorithm to make $\Omega\left(\frac{\nu^2}{\epsilon^2}\right)$ label requests.

3 Notation

We begin by introducing some notation. Let $\mathcal{X}$ be a set, called the *instance space*, and $\mathcal{F}$ be a corresponding σ-algebra. Let $\mathcal{D}_{XY}$ be a probability measure on $\mathcal{X} \times \{-1, 1\}$. We use $\mathcal{D}$ to denote the marginal distribution of $\mathcal{D}_{XY}$ over $\mathcal{X}$. $\mathbb{C}_{\mathcal{F}}$ is the set of all $\mathcal{F}$-measurable $f : \mathcal{X} \to \{-1, 1\}$. $\mathbb{C} \subseteq \mathbb{C}_{\mathcal{F}}$ is a concept space on $\mathcal{X}$, and we use d to denote the VC dimension of $\mathbb{C}$; to focus on nontrivial learning, we assume $d > 0$. For any $h, h' \in \mathbb{C}_{\mathcal{F}}$, define $er_{\mathcal{D}}(h, h') = \mathcal{P}r_{X \sim \mathcal{D}}\{h(X) \neq h'(X)\}$. If $\mathcal{U} \in \mathcal{X}^m$, define $er_{\mathcal{U}}(h, h') = \frac{1}{m}\sum_{x \in \mathcal{U}} I[h(x) \neq h'(x)]$.[3] If $\mathcal{L} \in (\mathcal{X} \times \{-1, 1\})^m$, define $er_{\mathcal{L}}(h) = \frac{1}{m}\sum_{(x,y) \in \mathcal{L}} I[h(x) \neq y]$. For any $h \in \mathbb{C}_{\mathcal{F}}$, define $er(h) = \mathcal{P}r_{(X,Y) \sim \mathcal{D}_{XY}}\{h(X) \neq Y\}$. Define the *noise rate* $\nu = \inf_{h \in \mathbb{C}} er(h)$. An α-cover of $\mathbb{C}$ is any $V \subseteq \mathbb{C}$ s.t. $\forall h \in \mathbb{C}$, $\exists h' \in V$ with $er_{\mathcal{D}}(h, h') \leq \alpha$.

Generally, in this setting data is sampled i.i.d. according to $\mathcal{D}_{XY}$, but the labels are hidden from the learner unless it asks the oracle for them individually. In particular, requesting the same example's label twice gives the same label both times (though if the data sequence contains two identical examples, requesting both labels might give two different values). However, for notational simplicity, we often abuse this notation by stating that $X \sim \mathcal{D}$ and later stating that the algorithm requests the label of X, denoted $Oracle(X)$; by this, we implicitly mean that $(X, Y) \sim \mathcal{D}_{XY}$, and the oracle reveals the value of Y upon request. In particular, for $\mathcal{U} \sim \mathcal{D}^m$, $h \in \mathbb{C}_\mathcal{F}$, define $er_\mathcal{U}(h) = \frac{1}{m}\sum_{x\in\mathcal{U}} I[h(x) \neq Oracle(x)]$.

---

[2] An alternative, but very similar progress metric is the size of an ϵ-cover of the version space. The author suspects the analysis presented in this paper can be extended to describe that type of progress as well.

[3] We overload the standard set-theoretic notation to also apply to sequences. In particular, $\sum_{x \in \mathcal{U}}$ indicates a sum over entries of the sequence $\mathcal{U}$ (not necessarily all distinct). Similarly, we use $|\mathcal{U}|$ to denote length of the sequence $\mathcal{U}$, $S \subseteq \mathcal{U}$ to denote a subsequence of $\mathcal{U}$, $S \cup \mathcal{U}$ to denote concatenation of two sequences, and for any particular $x \in \mathcal{U}$, $\mathcal{U} \setminus \{x\}$ indicates the subsequence of $\mathcal{U}$ with all entries except the single occurrence of x that is implicitly referenced in the statement. It may help to think of each instance x in a sample as having a unique identifier.

Definition 1. *For $V \subseteq \mathbb{C}$ with finite $|V|$, the* majority vote *concept $h_{maj} \in \mathbb{C}_\mathcal{F}$ is defined by $h_{maj}(x) = 1$ iff $|\{h \in V : h(x) = 1\}| \geq \frac{1}{2}|V|$.*

Definition 2. *For $\mathcal{U} \in \mathcal{X}^m$, $h \in \mathbb{C}_\mathcal{F}$, we overload notation to define the sequence of labels $h(\mathcal{U}) = \{h(x)\}_{x\in\mathcal{U}}$ assigned to entries of $\mathcal{U}$ by h. For $V \subseteq \mathbb{C}_\mathcal{F}$, $V[\mathcal{U}]$ denotes any subset of V such that $\forall h \in V, |\{h' \in V[\mathcal{U}] : h'(\mathcal{U}) = h(\mathcal{U})\}| = 1$. $V[\mathcal{U}]$ represents the labelings of $\mathcal{U}$ realizable by V.*

4 Extended Teaching Dimension

Definition 3. *(Extended Teaching Dimension [3]) Let $V \subseteq \mathbb{C}$, $m \geq 0$, $\mathcal{U} \in \mathcal{X}^m$.*

$$\forall f \in \mathbb{C}_\mathcal{F},\ XTD(f, V, \mathcal{U}) = \inf\{t | \exists R \subseteq \mathcal{U} : |\{h \in V : h(R) = f(R)\}| \leq 1 \wedge |R| \leq t\}.$$

$$XTD(V, \mathcal{U}) = \sup_{f\in\mathbb{C}_\mathcal{F}} XTD(f, V, \mathcal{U}).$$

For a given f, we call any $R \subseteq \mathcal{U}$ such that $|\{h \in V : h(R) = f(R)\}| \leq 1$ a specifying set *for f on $\mathcal{U}$ with respect to V.*[4]

The goal of Exact learning with membership queries is to ask for the labels $f(x)$ of individual examples $x \in \mathcal{X}$ until the only concept in $\mathbb{C}$ consistent with the observed labels is the target $f \in \mathbb{C}$. Hegedüs [3] presents the following algorithm.

```
Algorithm: MembHalving
Output: The target concept $f \in \mathbb{C}$
0. $V \leftarrow \mathbb{C}$
1. Repeat until $|V| = 1$
2.     Let $h_{maj}$ be the majority vote of $V$
3.     Let $R \subseteq \mathcal{X}$ be a minimal specifying set for $h_{maj}$ on $\mathcal{X}$ with respect to $V$
4.     Ask for the label $f(x)$ of every $x \in R$
5.     Let $V \leftarrow \{h \in V | \forall x \in R, f(x) = h(x)\}$
6. Return the remaining element of $V$
```

Theorem 1. *(Exact Learning: Hegedüs [3]). Letting $\#MQ(\mathbb{C})$ denote the Exact learning query complexity of $\mathbb{C}$ with membership queries on any examples in $\mathcal{X}$, and $t_0 = XTD(\mathbb{C}, \mathcal{X})$, then the following inequalities are valid if $|\mathbb{C}| > 2$.*

$$\max\{t_0, \log_2 |\mathbb{C}|\} \leq \#MQ(\mathbb{C}) \leq t_0 \log_2 |\mathbb{C}|.$$

Furthermore, this upper bound is achieved by the MembHalving algorithm.[5]

---

[4] We also overload all of these definitions in the obvious way for *sets* $\mathcal{U} \subseteq \mathcal{X}$.

[5] By a slight alteration to choose queries in a particular greedy order, Hegedüs is able to reduce this upper bound to $2\frac{t_0}{\log_2 t_0}\log_2|\mathbb{C}|$. However, it is the simpler form of the algorithm (presented here) that we draw inspiration from in the following sections.

The upper bound of Theorem 1 is clear when we view MembHalving as a version of the Halving algorithm [8]. That is, querying all examples in a specifying set for h guarantees either h makes a mistake or we identify f. Thus, querying a specifying set for h_{maj} guarantees that we at least halve the version space.

The following definitions represent natural extensions of XTD to the PAC setting. The relation of these quantities to the complexity of active learning is our primary focus.

Definition 4. *(XTD Growth Function) For* $m \geq 0$*,* $V \subseteq \mathbb{C}$*,* $\delta \in [0, 1]$*,*

$$\begin{aligned} XTD(V, \mathcal{D}, m, \delta) = & \inf\{t | \forall f \in \mathbb{C}_{\mathcal{F}},\ \mathcal{P}r_{\mathcal{U} \sim \mathcal{D}^m}\{XTD(f, V[\mathcal{U}], \mathcal{U}) > t\} \leq \delta\}. \\ XTD(V, m) = & \sup_{\mathcal{U} \in \mathcal{X}^m} XTD(V[\mathcal{U}], \mathcal{U}). \end{aligned}$$

$XTD(\mathbb{C}, \mathcal{D}, m, \delta)$ plays an important role in distribution-dependent bounds on the label complexity, while $XTD(\mathbb{C}, m)$ plays an analogous role in distribution-free bounds. Clearly $0 \leq XTD(\mathbb{C}, \mathcal{D}, m, \delta) \leq XTD(\mathbb{C}, m) \leq m$.

As a simple example, consider the space of thresholds on the line. That is, suppose $\mathcal{X} = \mathbb{R}$ and $\mathbb{C} = \{h_\theta : \theta \in \mathbb{R}, h_\theta(x) = +1 \text{ iff } x \geq \theta\}$. In this case, $XTD(\mathbb{C}, m) = 2$, since for any set $\mathcal{U}$ of m points, and any $f \in \mathbb{C}_{\mathcal{F}}$, we can form a specifying set with the points $\min\{x \in \mathcal{U} : f(x) = +1\}$ and $\max\{x \in \mathcal{U} : f(x) = -1\}$, (if they exist).

5 The Complexity of Realizable Active Learning

Before discussing the general setting, we begin with realizable learning ($\eta = 0$), because the analysis is quite simple, and clearly highlights the relationship to the MembHalving algorithm. We handle noisy labels in the next section.

Based on Theorem 1, it should be clear that for $m \geq \Omega\left(\frac{1}{\epsilon}\left(d \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right)$, $\#LQ(\mathbb{C}, \mathcal{D}, \epsilon, \delta, 0) \leq XTD(\mathbb{C}, m) d \log_2 \frac{em}{d}$. Roughly speaking, this is achieved by drawing m unlabeled examples $\mathcal{U}$ and executing MembHalving with concept space $\mathbb{C}[\mathcal{U}]$ and instance space $\mathcal{U}$. This gives a *data-dependent* bound of $XTD(\mathbb{C}[\mathcal{U}], \mathcal{U}) \log_2 |\mathbb{C}[\mathcal{U}]| \leq XTD(\mathbb{C}, m) d \log_2 \frac{em}{d}$. We can also obtain a related *distribution-dependent* result as follows. Consider the following algorithm.

```
Algorithm: ActiveHalving
Input: $V \subseteq \mathbb{C}_{\mathcal{F}}$, values $\epsilon, \delta \in (0,1)$, $\mathcal{U} = \{x_1, x_2, \ldots, x_m\} \in \mathcal{X}^m$, constant $n \in \mathbb{N}$
Output: Concept $\hat{h} \in V$
0. Let $i \leftarrow 0$
1. Repeat
2.     $i \leftarrow i + 1$
3.     Let $\mathcal{U}_i = \{x_{1+n(i-1)}, x_{2+n(i-1)}, \ldots, x_{ni}\}$
4.     Let $h_{maj}$ be the majority vote of $V$
5.     Let $R \subseteq \mathcal{U}_i$ be a minimal specifying set for $h_{maj}$ on $\mathcal{U}_i$ w.r.t. $V[\mathcal{U}_i]$
6.     Ask for the label $f(x)$ of every $x \in R$
7.     Let $V \leftarrow \{h \in V | f(R) = h(R)\}$
8.     If $\exists h \in V$ s.t. $h_{maj}(\mathcal{U}_i) = h(\mathcal{U}_i)$, Return $\arg\min_{\hat{h} \in V} er_{\mathcal{U}}(\hat{h}, h_{maj})$
```

Theorem 2. *Let* $m = \left\lceil \frac{256d}{\epsilon}\left(\ln \frac{92d}{\epsilon\delta}\right)^2 \right\rceil$, *and* $n = \left\lceil \frac{4}{\epsilon}\ln\frac{12d\log_2\frac{4em}{\delta}}{\delta}\right\rceil$. *Let* $\hat{t} = XTD\left(\mathbb{C},\mathcal{D},n,\frac{\delta}{12d\log_2\frac{4em}{\delta}}\right)$. *If* $N(\delta/(2m))$ *is the size of a minimal* $\frac{\delta}{2m}$*-cover of* $\mathbb{C}$*, then*

$$\#LQ(\mathbb{C},\mathcal{D},\epsilon,\delta,0) \leq \hat{t}\log_2 N(\delta/(2m)) \leq O\left(\hat{t}d\log\frac{d}{\epsilon\delta}\right).$$

Proof. The bound is achieved by ActiveHalving$(V,\epsilon,\delta,\mathcal{U},n)$, where $\mathcal{U}\sim\mathcal{D}^m$, and V is a minimal $\frac{\delta}{2m}$-cover of $\mathbb{C}$. Let $f\in\mathbb{C}$ have $er(f)=0$. Let $\hat{f}=\arg\min_{h\in V} er(h)$. With probability $\geq 1-\delta/2$, $f(\mathcal{U}) = \hat{f}(\mathcal{U})$. Suppose this happens. In each iteration, if the condition in step 8 does not obtain, then either $\exists x\in R : h_{maj}(x) \neq f(x)$ or else $V[\mathcal{U}_i] = \{h\}$ for some $h\in V$ such that $\exists x\in\mathcal{U}_i : h_{maj}(x)\neq h(x) = f(x)$. Either way, we must have eliminated at least half of V in step 7, so the condition in step 8 fails at most $\log_2 N(\delta/(2m)) < 2d\log_2\frac{4em}{\delta} - 1$ times.

On the other hand, suppose the condition in step 8 obtains. This happens only when $h_{maj}(\mathcal{U}_i) = f(\mathcal{U}_i)$. $\mathcal{P}r_{\mathcal{U}_i}\left\{er_{\mathcal{U}_i}(h_{maj},f) = 0 \wedge er_{\mathcal{U}}(h_{maj},f) > \frac{\epsilon}{4}\right\} \leq \frac{\delta}{12d\log_2\frac{4em}{\delta}}$. By a union bound, the probability that an h_{maj} with $er_{\mathcal{U}}(h_{maj},f) > \frac{\epsilon}{4}$ satisfies the condition in step 8 on any iteration is at most $\frac{\delta}{6}$. If this does not happen, then the $\hat{h}\in V$ we return has $er_{\mathcal{U}}(\hat{h},f) \leq er_{\mathcal{U}}(\hat{h},h_{maj}) + er_{\mathcal{U}}(h_{maj},f) \leq er_{\mathcal{U}}(f,h_{maj}) + er_{\mathcal{U}}(h_{maj},f) \leq \frac{\epsilon}{2}$. By Chernoff and union bounds, m is large enough so that with probability at least $1-\frac{\delta}{6}$, $er_{\mathcal{U}}(\hat{h},f)\leq\frac{\epsilon}{2} \Rightarrow er_{\mathcal{D}}(\hat{h},f)\leq\epsilon$. So with probability $1-\frac{5\delta}{6}$, we return an $\hat{h}\in\mathbb{C}$ with $er_{\mathcal{D}}(\hat{h},f)\leq\epsilon$.

On the issue of number of queries, each iteration queries a minimal specifying set for h_{maj} on a set of size n. The probability the size of this set is larger than $\hat{t}$ for a particular set $\mathcal{U}_i$ is at most $\frac{\delta}{12d\log_2\frac{4em}{\delta}}$. By a union bound, the probability it is larger than $\hat{t}$ on any iteration is at most $\frac{\delta}{6}$. Thus, the total probability of success (in learning and obtaining the query bound) is at least $1-\delta$. □

Note that we can obtain a worst-case label bound for ActiveHalving by replacing $\hat{t}$ above with $XTD(\mathbb{C},n)$. Theorem 2 highlights the relationship to known results in Exact learning with membership queries [3]. In particular, if $\mathbb{C}$ and $\mathcal{X}$ are finite, and $\mathcal{D}$ has support everywhere on $\mathcal{X}$, then as $\epsilon\to 0$ and $\delta\to 0$, the bound converges to $XTD(\mathbb{C},\mathcal{X})\log_2|\mathbb{C}|$, the upper bound in Theorem 1.

6 The Complexity of Active Learning with Noise

The following algorithm can be viewed as a noise-tolerant version of ActiveHalving. Significant care is needed to ensure we do not discard the best concept, and that the final classifier is near-optimal. The main trick is to use subsamples of size $< \frac{1}{16\eta}$. Since the probability of such a subsample containing a noisy example is small, the specifying sets for h_{maj} will often be noise-free. Therefore, if $h\in V$ is contradicted in many such specifying sets, we can be confident h is suboptimal. Likewise, if for a particular unqueried x, there are many such subsamples containing x where h_{maj} is *not* contradicted, and where there is a consistent h, then more often than not, $h(x) = h^*(x)$, where $h^* = \arg\min_{h'\in V} er(h')$.

```
Algorithm: $ReduceAndLabel(V,\mathcal{U},\epsilon,\delta,\hat{\eta})$
Input: Finite $V \subseteq \mathbb{C}_{\mathcal{F}}$, $\mathcal{U} = \{x_1, x_2, \ldots, x_m\} \in \mathcal{X}^m$, values $\epsilon, \delta, \hat{\eta} \in (0,1]$.
Output: Concept $h \in V$.
Let $u = \lfloor |\mathcal{U}|/(5 \ln |V|) \rfloor$
Let $V_0 \leftarrow V$, $i \leftarrow 0$
Do
    $i \leftarrow i + 1$
    Let $\mathcal{U}_i = \{x_{1+u(i-1)}, x_{2+u(i-1)}, \ldots, x_{ui}\}$
    $V_i \leftarrow Reduce\left(V_{i-1}, \mathcal{U}_i, \frac{\delta}{48 \ln |V|}, \hat{\eta} + \frac{\epsilon}{2}\right)$
Until $|V_i| > \frac{3}{4}|V_{i-1}|$ or $|V_i| \leq 1$
Let $\bar{\mathcal{U}} = \{x_{ui+1}, x_{ui+2}, \ldots, x_{ui+\ell}\}$, where $\ell = \left\lceil 12 \frac{\hat{\eta}}{\epsilon^2} \ln \frac{12|V|}{\delta} \right\rceil$
$\mathcal{L} \leftarrow Label\left(V_{i-1}, \bar{\mathcal{U}}, \frac{\delta}{12}, \hat{\eta} + \frac{\epsilon}{2}\right)$
Return $h \in V_i$ having smallest $er_{\mathcal{L}}(h)$, (or any $h \in V$ if $V_i = \varnothing$)
```

```
Subroutine: $Reduce(V,\mathcal{U},\delta,\hat{\eta})$
Input: Finite $V \subseteq \mathbb{C}_{\mathcal{F}}$, unlabeled sequence $\mathcal{U}$, values $\delta, \hat{\eta} \in (0,1]$
Output: Concept space $V' \subseteq V$
Let $m = |\mathcal{U}|$, $n = \left\lfloor \frac{1}{16\hat{\eta}} \right\rfloor$, $r = \left\lceil 397 \ln \frac{2}{\delta} \right\rceil$, $\theta = \frac{27}{320}$
Let $h_{maj}$ be the majority vote of $V$
For $i \in \{1, 2, \ldots, r\}$
    Sample a subsequence $S_i$ of size $n$ uniformly without replacement from $\mathcal{U}$
    Let $R_i$ be a minimal specifying set for $h_{maj}$ in $S_i$ with respect to $V[S_i]$
    Ask for the label of every example in $R_i$
    Let $\bar{V}_i$ be the concepts $h \in V$ s.t. $h(R_i) \neq Oracle(R_i)$
Let $\bar{V}$ be the set of $h \in V$ that appear in $> \theta \cdot r$ of the sets $\bar{V}_i$
Return $V' = V \setminus \bar{V}$
```

```
Subroutine: $Label(V,\mathcal{U},\delta,\hat{\eta})$
Input: Finite $V \subseteq \mathbb{C}_{\mathcal{F}}$, unlabeled sequence $\mathcal{U}$, values $\delta, \hat{\eta} \in (0,1]$
Output: Labeled sequence $\mathcal{L}$
Let $\ell = |\mathcal{U}|$, $n = \left\lfloor \frac{1}{16\hat{\eta}} \right\rfloor$, $k = \left\lceil 167 \frac{\ell}{n} \ln \frac{3\ell}{\delta} \right\rceil$
Let $h_{maj}$ be the majority vote of $V$, and let $\mathcal{L} \leftarrow \{\}$
For $i \in \{1, 2, \ldots, k\}$
    Sample a subsequence $S_i$ of size $n$ uniformly without replacement from $\mathcal{U}$
    Let $R_i$ be a minimal specifying set for $h_{maj}$ in $S_i$ with respect to $V[S_i]$
    For each $x \in R_i$ not in $\mathcal{L}$, request its label $y_x$ and let $\mathcal{L} \leftarrow \mathcal{L} \cup \{(x, y_x)\}$
Let $\hat{\mathcal{U}} \subseteq \mathcal{U}$ be the subsequence of examples we did not ask for the label of
For each $x \in \hat{\mathcal{U}}$
    Let $\hat{I}_x = \{i : x \in S_i$ and $\exists h \in V$ s.t. $h(R_i) = h_{maj}(R_i) = Oracle(R_i)\}$
    For each $i \in \hat{I}_x$, let $h_i \in V$ be s.t. $h_i(R_i) = Oracle(R_i)$
    Let $y$ be the majority value of $\{h_i(x) : i \in \hat{I}_x\}$ (breaking ties arbitrarily)
    Let $\mathcal{L} \leftarrow \mathcal{L} \cup \{(x, y)\}$
Return $\mathcal{L}$
```

Lemma 1. *(Reduce) Suppose* $h^* \in V$ *is a concept such that* $er_{\mathcal{U}}(h^*) \leq \hat{\eta} < \frac{1}{32}$. *Let* V' *be the set returned by* $Reduce(V, \mathcal{U}, \epsilon, \delta, \hat{\eta})$. *With probability at least* $1 - \delta$, $h^* \in V'$, *and if* $er_{\mathcal{U}}(h_{maj}, h^*) \geq 10\hat{\eta}$ *then* $|V'| \leq \frac{3}{4}|V|$.

Proof. By a *noisy example*, in this context we mean any $x \in \mathcal{U}$ for which $h^*(x)$ disagrees with the oracle's label. Let $n = \left\lfloor \frac{1}{16\hat{\eta}} \right\rfloor$ and $r = \left\lceil 397 \ln \frac{2}{\delta} \right\rceil$, $\theta = \frac{27}{320}$. By a Chernoff bound, sampling r subsequences of size n, each without replacement from $\mathcal{U}$, guarantees with probability $\geq 1 - \frac{\delta}{2}$ that at most θr of the subsequences contain any noisy examples. In particular, this would imply $h^* \in V'$.

Now suppose $er_{\mathcal{U}}(h_{maj}, h^*) \geq 10\hat{\eta}$. For any particular subsampled sequence S_i, $\mathcal{P}r_{S_i \sim U_n(\mathcal{U})}\left\{h_{maj}(S_i) = h^*(S_i)\right\} \leq (1 - 10\hat{\eta})^n \leq 0.627$. So the probability there is some $x \in S_i$ with $h_{maj}(x) \neq h^*(x)$ is at least 0.373. By a Chernoff bound, with probability at least $1 - \frac{\delta}{2}$, at least $4\theta r$ of the r subsamples contain some $x \in \mathcal{U}$ such that $h_{maj}(x) \neq h^*(x)$.

By a union bound, the total probability the above two events succeed is at least $1 - \delta$. Suppose this happens. Any sequence S_i containing no noisy examples but $\exists x \in S_i$ such that $h_{maj}(x) \neq h^*(x)$ necessarily has $|\bar{V}_i| \geq \frac{1}{2}|V|$. Since there are at least $3\theta r$ such subsamples S_i, we have $|\bar{V}| \geq \left(3\theta r \cdot \frac{1}{2}|V| - \theta r \cdot |V|\right) / (2\theta r) = \frac{1}{4}|V|$, so that $|V'| \leq \frac{3}{4}|V|$. □

Lemma 2. *(Label) Let* $\mathcal{U} \in \mathcal{X}^\ell$, $\ell > n$. *Suppose* $h^* \in V$ *has* $er_{\mathcal{U}}(h^*) \leq \hat{\eta} < \frac{1}{32}$. *Let* h_{maj} *be the majority vote of* V, *and suppose* $er_{\mathcal{U}}(h_{maj}, h^*) \leq 12\hat{\eta}$. *Let* $\mathcal{L}$ *be the sequence returned by* $Label(V, \mathcal{U}, \delta, \hat{\eta})$. *With probability at least* $1 - \delta$, *for every* $(x, y) \in \mathcal{L}$, y *is either the oracle's label for* x *or* $y = h^*(x)$. *In any case,* $\forall x \in \mathcal{U}, |\{y : (x, y) \in \mathcal{L}\}| = 1$.

Proof. As above, a *noisy example* is any $x \in \mathcal{U}$ such that $h^*(x)$ disagrees with the oracle. For any x we ask for the label of, the entry $(x, y) \in \mathcal{L}$ has y equal to the oracle's label, so the focus of the proof is on $\hat{\mathcal{U}}$. For each $x \in \hat{\mathcal{U}}$, let $I_x = \{i : x \in S_i\}$, $A = \{i : \exists x' \in R_i, h^*(x') \neq Oracle(x')\}$, and $B = \{i : \exists x' \in R_i, h_{maj}(x') \neq h^*(x')\}$. $\forall x \in \hat{\mathcal{U}}$, if $|I_x \cap A| < |(I_x \setminus B) \setminus A|$, we have that $|\{i \in I_x : h^*(R_i) = h_{maj}(R_i) = Oracle(R_i)\}| > \frac{1}{2}|\hat{I}_x| > 0$. In particular, this means the majority value of $\{h_i(x) : i \in \hat{I}_x\}$ is $h^*(x)$. The remainder of the proof bounds the probability this fails to happen.

For $x \in \hat{\mathcal{U}}$, for $i \in \{1, 2, \ldots, k\}$ let $\bar{S}_{i,x}$ of size n be sampled uniformly without replacement from $\mathcal{U} \setminus \{x\}$, $\bar{A}_x = \{i : \exists x' \in \bar{S}_{i,x}, h^*(x') \neq Oracle(x')\}$, and $\bar{B}_x = \{i : \exists x' \in \bar{S}_{i,x}, h_{maj}(x') \neq h^*(x')\}$.

$$\begin{aligned}
&\mathcal{P}r\left\{\exists x \in \hat{\mathcal{U}} : |I_x \cap A| \geq |(I_x \setminus B) \setminus A|\right\} \\
&\leq \sum_{x \in \mathcal{U}} \mathcal{P}r\left\{|I_x| < \tfrac{nk}{2\ell}\right\} + \mathcal{P}r\left\{|I_x \cap \bar{A}_x| \geq \tfrac{\sqrt{96}-1}{80}|I_x| \wedge |I_x| \geq \tfrac{nk}{2\ell}\right\} + \\
&\qquad \mathcal{P}r\left\{|(I_x \setminus \bar{B}_x) \setminus \bar{A}_x| \leq \tfrac{\sqrt{96}-1}{80}|I_x| \wedge |I_x| \geq \tfrac{nk}{2\ell}\right\} \leq \ell\left[e^{-\frac{kn}{8\ell}} + 2e^{-\frac{nk}{167\ell}}\right] \leq \delta.
\end{aligned}$$

The second inequality is due to Chernoff and Hoeffding bounds. □

Lemma 3. *Suppose* $\nu = \inf_{h \in \mathbb{C}} er(h) \leq \eta$ *and* $\eta + \frac{3}{4}\epsilon < \frac{1}{32}$. *Let* V *be an* $\frac{\epsilon}{2}$*-cover of* $\mathbb{C}$. *Let* $\mathcal{U} \sim \mathcal{D}^m$, *with* $m = \left\lceil 224 \frac{\eta+\epsilon/2}{\epsilon^2} \ln \frac{48 \ln |V|}{\delta} \right\rceil \lceil 5 \ln |V| \rceil$. *Let* $n = \left\lfloor \frac{1}{16(\eta+3\epsilon/4)} \right\rfloor$, $\ell = \left\lceil 48 \frac{\eta+\epsilon/2}{\epsilon^2} \ln \frac{12|V|}{\delta} \right\rceil$, $s = \left\lceil 397 \ln \frac{96 \ln |V|}{\delta} \right\rceil (4 \ln |V|) + \left\lceil 167 \frac{\ell}{n} \ln \frac{36\ell}{\delta} \right\rceil$, *and* $t = XTD\left(V, \mathcal{D}, n, \frac{\delta}{2s}\right)$. *With probability* $\geq 1 - \delta$, *ReduceAndLabel* $\left(V, \mathcal{U}, \frac{\epsilon}{2}, \delta, \eta + \frac{\epsilon}{2}\right)$ *makes at most ts label queries and returns a concept h with* $er(h) \leq \nu + \epsilon$.

Proof. Let $h^* \in V$ have $er(h^*) \leq \nu + \frac{\epsilon}{2}$. Suppose the value of i is ι when we reach step 7. Clearly $\iota \leq \log_{4/3} |V| \leq 4 \ln |V|$. Let h^i_{maj} denote the majority vote of V_i. We proceed by bounding the probability that any of six specific events fail to happen. The first event is

$$\left[\forall i \in \{1, 2, \ldots, \iota\}, er_{\mathcal{U}_i}(h^*) \leq \eta + \tfrac{3}{4}\epsilon\right].$$

The probability this fails is $\leq (4 \ln |V|) e^{-\left\lfloor \frac{m}{5 \ln |V|} \right\rfloor \frac{\epsilon^2}{\eta+\epsilon/2} \frac{1}{48}} \leq \frac{\delta}{12}$ (by Chernoff and union bounds). The next event we consider is

$$\left[\forall i \in \{1, 2, \ldots, \iota\}, h^* \in V_i \text{ and (if } |V_\iota| > 1\text{) } er_{\mathcal{U}_\iota}\left(h^{\iota-1}_{maj}, h^*\right) < 10\left(\eta + \tfrac{3}{4}\epsilon\right)\right].$$

By Lemma 1 and a union bound, the previous event succeeds but this one fails with probability $\leq \frac{\delta}{12}$. Next, note that the event

$$\left[\forall i \in \{1, 2, ..., \iota\}, er_{\mathcal{U}_i}\left(h^{i-1}_{maj}, h^*\right) < 10\left(\eta + \tfrac{3}{4}\epsilon\right) \Rightarrow er_{\mathcal{D}}\left(h^{i-1}_{maj}, h^*\right) \leq \tfrac{21}{2}\left(\eta + \tfrac{3}{4}\epsilon\right)\right]$$

fails with probability $\leq (4 \ln |V|) e^{-\left\lfloor \frac{m}{5 \ln |V|} \right\rfloor \left(\eta + \frac{3}{4}\epsilon\right) \frac{1}{84}} \leq \frac{\delta}{12}$. The fourth event is

$$\left[er_{\bar{\mathcal{U}}}\left(h^{\iota-1}_{maj}, h^*\right) \leq 12\left(\eta + \tfrac{3}{4}\epsilon\right)\right].$$

By a Chernoff bound, the probability this fails when the previous three events succeed is $\leq e^{-\frac{\ell}{14}\left(\eta + \frac{3}{4}\epsilon\right)} \leq \frac{\delta}{12}$. The fifth event is

$$[er_{\bar{\mathcal{U}}}(h^*) \leq er(h^*) + \tfrac{\epsilon}{4} \text{ and } \forall h \in V_{\iota-1}, er(h) > er(h^*) + \tfrac{\epsilon}{2} \Rightarrow er_{\bar{\mathcal{U}}}(h) > er_{\bar{\mathcal{U}}}(h^*)].$$

By Chernoff and union bounds, the probability the previous events succeed but this fails is $\leq |V| e^{-\frac{\ell}{48} \frac{\epsilon^2}{\eta+\epsilon/2}} \leq \frac{\delta}{12}$. Finally, consider the event

$$[\forall (x, y) \in \mathcal{L}, y = h^*(x) \text{ or } y = Oracle(x)].$$

By Lemma 2, this fails when the other five succeed with probability $\leq \frac{\delta}{12}$. Thus the probability all of these events succeed is $\geq 1 - \frac{\delta}{2}$. If they succeed, then any $h' \in V_\iota$ with $er(h') > \nu + \epsilon \geq er(h^*) + \frac{\epsilon}{2}$ has $er_{\mathcal{L}}(h') > er_{\mathcal{L}}(h^*) \geq \min_{h \in V_\iota} er_{\mathcal{L}}(h)$. Thus the h we return has $er(h) \leq \nu + \epsilon$.

In each call to *Reduce*, we ask for the labels of a minimal specifying set for $r = \left\lceil 397 \ln \frac{96 \ln |V|}{\delta} \right\rceil$ sequences of length n. For each, we make at most t label requests with probability $\geq 1 - \frac{\delta}{2s}$, so the probability any call to *Reduce* makes more than tr label requests is $\leq \frac{4\delta r \ln |V|}{2s}$. Similarly, in *Label* we request the labels of a minimal specifying set for $\leq k = \left\lceil 167 \frac{\ell}{n} \ln \frac{36\ell}{\delta} \right\rceil$ sequences of length n. So we make at most tk queries in *Label* with probability $\geq 1 - \frac{\delta k}{2s}$. Thus, the total probability we make more than $t(k + 4r \ln |V|) = ts$ queries is $\leq \frac{4\delta r \ln |V|}{2s} + \frac{\delta k}{2s} = \frac{\delta}{2}$. The total probability either the query or error bound is violated is at most δ. $\square$

Theorem 3. *Let* $n = \left\lfloor \frac{1}{16(\eta+3\epsilon/4)} \right\rfloor$*, and let* N *be the size of a minimal* $\frac{\epsilon}{2}$*-cover of* $\mathbb{C}$*. Let* $\ell = \left\lceil 48 \frac{\eta+\epsilon/2}{\epsilon^2} \ln \frac{12N}{\delta} \right\rceil$*. Let* $s = \left\lceil (397 \ln \frac{96 \ln N}{\delta}) \right\rceil (4 \ln N) + \left\lceil 167 \frac{\ell}{n} \ln \frac{36\ell}{\delta} \right\rceil$*, and* $t = XTD\left(\mathbb{C}, \mathcal{D}, n, \frac{\delta}{2s}\right)$*.*

$$\#LQ(\mathbb{C}, \mathcal{D}, \epsilon, \delta, \eta) \leq ts = O\left(t\left(\frac{\eta^2}{\epsilon^2}+1\right)\left(d \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\left(\log \frac{d}{\epsilon\delta}\right)\right).$$

Proof. It is known that $N < 2\left(\frac{4e}{\epsilon} \ln \frac{4e}{\epsilon}\right)^d$ [9]. If $\eta + \frac{3}{4}\epsilon \geq \frac{1}{32}$, the bound exceeds the passive sample complexity, so it clearly holds. Otherwise, the result follows from Lemma 3 and the fact that $XTD\left(V, \mathcal{D}, n, \frac{\delta}{2s}\right) \leq XTD\left(\mathbb{C}, \mathcal{D}, n, \frac{\delta}{2s}\right)$. $\square$

Generally, if we do not know an upper bound η on the noise rate ν, then we can perform a guess-and-double procedure using a labeled validation set, which grows to size at most $\tilde{O}\left(\frac{\nu+\epsilon}{\epsilon^2}\right)$. See Section 9 for more discussion of this matter.

We can create a general algorithm, independent of $\mathcal{D}$, by using unlabeled examples to (with probability $\geq 1-\delta/2$) construct the $\frac{\epsilon}{2}$-cover. It is possible to do this while maintaining $|V| \leq N' = 2\left(\frac{16e}{\epsilon} \ln \frac{16e}{\epsilon}\right)^d$ using $O\left(\frac{1}{\epsilon^2}\left(d \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right)$ unlabeled examples. Thus, replacing t in Theorem 3 with $XTD(\mathbb{C}, n)$ and increasing N to N' gives an upper bound on the *distribution-free* label complexity.

7 Lower Bounds

In this section, we prove lower bounds on the label complexity.

Definition 5. *(Extended Partial Teaching Dimension) Let* $V \subseteq \mathbb{C}$*,* $m \geq 0$*,* $\delta \geq 0$*.* $\forall f \in \mathbb{C}_{\mathcal{F}}, \mathcal{U} \in \mathcal{X}^{\lceil m \rceil}$*,*

$$XPTD(f, V, \mathcal{U}, \delta) = \inf\{t \mid \exists R \subseteq \mathcal{U} : |\{h \in V : h(R) = f(R)\}| \leq \delta|V|+1 \wedge |R| \leq t\}.$$
$$XPTD(V, \mathcal{D}, \delta) \quad = \inf\{t \mid \forall f \in \mathbb{C}_{\mathcal{F}}, \lim_{n \to \infty} \mathcal{P}r_{\mathcal{U} \sim \mathcal{D}^n}\{XPTD(f, V, \mathcal{U}, \delta) > t\} = 0\}.$$
$$XPTD(V, m, \delta) \quad = \sup_{f \in \mathbb{C}_{\mathcal{F}}} \sup_{\mathcal{U} \in \mathcal{X}^{\lceil m \rceil}} XPTD(f, V[\mathcal{U}], \mathcal{U}, \delta).$$

Theorem 4. *Let* $\epsilon \in [0, 1/2)$*,* $\delta \in [0, 1)$*. For any* 2ϵ*-separated set* $V \subseteq \mathbb{C}$ *with respect to* $\mathcal{D}$*,*

$$\max\{\log\left[(1-\delta)|V|\right], XPTD(V, \mathcal{D}, \delta)\} \leq \#LQ(\mathbb{C}, \mathcal{D}, \epsilon, \delta, 0).$$

If $0 < \delta < 1/16$ *and* $0 < \epsilon/2 \leq \eta < 1/2$*, and there are* $h_1, h_2 \in \mathbb{C}$ *such that* $er_{\mathcal{D}}(h_1, h_2) > 2(\eta+\epsilon)$*, then*

$$\Omega\left(\left(\frac{\eta^2}{\epsilon^2}+1\right)\log \frac{1}{\delta}\right) \leq \#LQ(\mathbb{C}, \mathcal{D}, \epsilon, \delta, \eta).$$

Also, the following distribution-free lower bound applies. If $\forall x \in \mathcal{X}, \{x\} \in \mathcal{F}$*,*[6] *then letting* $\mathfrak{D}$ *denote the set of all probability distributions on* $\mathcal{X}$*, for* any $V \subseteq \mathbb{C}$*,*

$$XPTD(V, (1-\epsilon)/\epsilon, \delta) \leq \sup_{\mathcal{D}' \in \mathfrak{D}} \#LQ(\mathbb{C}, \mathcal{D}', \epsilon, \delta, 0).$$

---

[6] This condition is not necessary, but simplifies the proof.

Proof. The $\log[(1-\delta)|V|]$ lower bound is due to Kulkarni [2].

We prove the $XPTD(V, \mathcal{D}, \delta)$ lower bound by the probabilistic method as follows. If $\delta|V| + 1 \geq |V|$, the bound is trivially true, so assume $\delta|V| + 1 < |V|$ (and in particular, $|V| < \infty$). Let $m \geq 0$, $\tilde{t} = XPTD(V, \mathcal{D}, \delta)$. By definition of $\tilde{t}$, $\exists f' \in \mathbb{C}_{\mathcal{F}}$ such that $\lim_{n\to\infty} \mathcal{P}r_{\mathcal{U}\sim\mathcal{D}^n}\{XPTD(f', V, \mathcal{U}, \delta) \geq \tilde{t}\} > 0$. By the Dominated Convergence Theorem and Kolmogorov's Zero-One Law, this implies $\lim_{n\to\infty} \mathcal{P}r_{\mathcal{U}\sim\mathcal{D}^n}\{XPTD(f', V, \mathcal{U}, \delta) \geq \tilde{t}\} = 1$. Since this probability is nonincreasing in n, this means $\mathcal{P}r_{\mathcal{U}\sim\mathcal{D}^m}\{XPTD(f', V, \mathcal{U}, \delta) \geq \tilde{t}\} = 1$. Suppose $\mathcal{A}$ is a learning algorithm. For $\mathcal{U} \in \mathcal{X}^m$, $f \in \mathbb{C}_{\mathcal{F}}$, define random quantities $R_{\mathcal{U},f} \subseteq \mathcal{U}$ and $h_{\mathcal{U},f} \in \mathbb{C}$, denoting the examples queried and classifier returned by $\mathcal{A}$, respectively, when the oracle answers consistent with f and the input unlabeled sequence is $\mathcal{U} \sim \mathcal{D}^m$. If we sample f uniformly at random from V,

$$\begin{aligned}
&\mathbb{E}_f\left[\mathcal{P}r_{\mathcal{U},R_{\mathcal{U},f},h_{\mathcal{U},f}}\left\{er_{\mathcal{D}}(f, h_{\mathcal{U},f}) > \epsilon \vee |R_{\mathcal{U},f}| \geq \tilde{t}\right\}\right] \\
&\geq \mathcal{P}r_{f,\mathcal{U},R_{\mathcal{U},f},h_{\mathcal{U},f}}\left\{f(R_{\mathcal{U},f}) = f'(R_{\mathcal{U},f}) \wedge er_{\mathcal{D}}(f, h_{\mathcal{U},f}) > \epsilon \vee |R_{\mathcal{U},f}| \geq \tilde{t}\right\} \\
&\geq \mathbb{E}_{\mathcal{U}}\left[\inf_{h\in\mathbb{C}, R\subseteq\mathcal{U}:|R|<\tilde{t}} \mathcal{P}r_f\left\{f(R) = f'(R) \wedge er_{\mathcal{D}}(h, f) > \epsilon\right\}\right] > \delta.
\end{aligned}$$

Therefore, there must be some fixed target $f \in \mathbb{C}$ such that the probability that $er_{\mathcal{D}}(f, h_{\mathcal{U},f}) > \epsilon$ or $|R_{\mathcal{U},f}| \geq XPTD(V, \mathcal{D}, \delta)$ is $> \delta$, proving the lower bound.

Kääriäinen [7] proves a distribution-free version of the $\Omega\left(\left(\frac{\eta^2}{\epsilon^2} + 1\right)\log\frac{1}{\delta}\right)$ bound, and also mentions its extendibility to the distribution-dependent setting. Since the distribution-dependent claim and proof thereof are only implicit in that reference, for completeness we present a brief proof here. Let $\Delta = \{x : h_1(x) \neq h_2(x)\}$. Suppose h^* is chosen from $\{h_1, h_2\}$ by an adversary. Given $\mathcal{D}$, we construct a distribution $\mathcal{D}_{XY}$ with the following property[7]. $\forall A \in \mathcal{F}, \mathcal{P}r_{(X,Y)\sim\mathcal{D}_{XY}}\{Y = h^*(X) | X \in A \cap \Delta\} = \frac{1}{2} + \frac{\epsilon}{2(\eta+\epsilon)}$, and $\mathcal{P}r_{(X,Y)\sim\mathcal{D}_{XY}}\{Y = h_1(X) | X \in A \setminus \Delta\} = 1$. Any concept $h \in \mathbb{C}$ with $er(h) \leq \eta + \epsilon$ has $\mathcal{P}r\{h(X) = h^*(X) | h_1(X) \neq h_2(X)\} > \frac{1}{2}$. Since this probability can be estimated to arbitrary precision with arbitrarily high probability using unlabeled examples, we have a reduction to active learning from the task of determining with probability $\geq 1 - \delta$ whether h_1 or h_2 is h^*. Examining the latter task, since every subset of Δ in $\mathcal{F}$ yields the same conditional distribution, any optimal strategy is based on samples from this distribution. It is known (e.g., [10, 11]) that this requires expected number of samples at least

$$\frac{(1-8\delta)\log\frac{1}{8\delta}}{8D_{KL}\left(\frac{1}{2}+\frac{\epsilon}{2(\eta+\epsilon)} || \frac{1}{2}-\frac{\epsilon}{2(\eta+\epsilon)}\right)} > \frac{1}{40}\frac{(\eta+\epsilon)^2}{\epsilon^2}\ln\frac{1}{8\delta},$$

where $D_{KL}(p||q) = p\log\frac{p}{q} + (1-p)\log\frac{1-p}{1-q}$.

We prove the $XPTD(V, (1-\epsilon)/\epsilon, \delta)$ bound as follows. Let $n = \frac{1-\epsilon}{\epsilon}$. For $S \in \mathcal{X}^{\lceil n\rceil}$, let $\mathcal{D}_S$ be the uniform distribution on the entries of S. Let $f'' = \arg\max_{f\in\mathbb{C}_{\mathcal{F}}} XPTD(f, V[S], S, \delta)$, and define $t'' = XPTD(f'', V[S], S, \delta)$. Let

---

[7] Although this proof relies on stochasticity of the oracle, with additional assumptions on $\mathcal{D}$ and Δ similar to Kääriäinen's [7], a similar result holds for deterministic oracles.

$m \geq 0$. Let $R_{\mathcal{U},f}$ and $h_{\mathcal{U},f}$ be defined as above, for $\mathcal{U} \sim \mathcal{D}_S^m$. As above, we use the probabilistic method, this time by sampling the target function f uniformly from $V[S]$.

$$\begin{aligned}
&\mathbb{E}_f \left[\mathcal{P}r_{\mathcal{U},R_{\mathcal{U},f},h_{\mathcal{U},f}} \left\{er_{\mathcal{D}_S}(h_{\mathcal{U},f}, f) > \epsilon \vee |R_{\mathcal{U},f}| \geq t''\right\}\right] \\
&\geq \mathbb{E}_{\mathcal{U}} \left[\mathcal{P}r_{f,R_{\mathcal{U},f},h_{\mathcal{U},f}} \left\{f(R_{\mathcal{U},f}) = f''(R_{\mathcal{U},f}) \wedge h_{\mathcal{U},f}(S) \neq f(S) \vee |R_{\mathcal{U},f}| \geq t''\right\}\right] \\
&\geq \min_{h \in \mathbb{C}, R \subseteq S : |R| < t''} \mathcal{P}r_f \left\{f(R) = f''(R) \wedge h(S) \neq f(S)\right\} > \delta.
\end{aligned}$$

Taking the supremum over $S \in \mathcal{X}^{\lceil n \rceil}$ completes the proof. □

8 Example: Axis-Aligned Rectangles

As an application, we analyze axis-aligned rectangles, when $\mathcal{D}$ is a product density. An axis-aligned rectangle in $\mathbb{R}^n$ is defined by a sequence $\{(a_i, b_i)\}_{i=1}^n$, such that $a_i \leq b_i$, and the examples labeled $+1$ are $\{x \in \mathcal{X} : \forall i, a_i \leq x \leq b_i\}$. Throughout this section, we assume $\mathcal{F}$ is the standard Borel σ-algebra on $\mathbb{R}^n$.

Lemma 4. *(Balanced Axis-Aligned Rectangles) If $\mathcal{D}$ is a product distribution on $\mathbb{R}^n$ with continuous CDF, and $\mathbb{C}$ is the set of axis-aligned rectangles such that $\forall h \in \mathbb{C}, \mathcal{P}r_{X \sim \mathcal{D}}\{h(X) = +1\} \geq \lambda$, then*

$$XTD(\mathbb{C}, \mathcal{D}, m, \delta) \leq O\left(\frac{n^2}{\lambda} \log \frac{nm}{\delta}\right).$$

Proof. If G_i is the CDF of X_i for $X \sim \mathcal{D}$, then $G_i(X_i)$ is uniform in $(0, 1)$, and for any $h \in \mathbb{C}$, the function $h'(x) = h(\{\min\{y : x_i = G_i(y)\}\}_{i=1}^n)$ (for $x \in (0, 1)^n$) is an axis-aligned rectangle. This mapping of the problem into $(0, 1)^n$ is equivalent to the original, so for the rest of this proof, we assume $\mathcal{D}$ is uniform on $(0, 1)^n$.

If m is smaller than the bound, the result clearly holds, so assume $m \geq 2n + \frac{4n}{\lambda}\left(\ln \frac{8n}{\delta} + 2n \ln \frac{2nm^2}{\delta}\right)$. Our first step is to discretize the concept space. Let S be the set of concepts h such that the region $\{x : h(x) = +1\}$ is specified by the interior of some rectangle $\{(a_i, b_i)\}_{i=1}^n$ with $a_i, b_i \in \left\{0, \frac{\delta}{2nm^2}, 2\frac{\delta}{2nm^2}, \ldots, \left\lceil \frac{2nm^2}{\delta} \right\rceil \frac{\delta}{2nm^2}\right\}$, $a_i < b_i$. By a union bound, with probability $\geq 1 - \delta/2$ over the draw of $\mathcal{U} \sim \mathcal{D}^m$, $\forall x, y \in \mathcal{U}$, $\forall i \in \{1, 2, \ldots, n\}$, $|x_i - y_i| > \frac{\delta}{2nm^2}$. In particular, this would imply there are valid choices of $S[\mathcal{U}]$ and $\mathbb{C}[\mathcal{U}]$ so that $\mathbb{C}[\mathcal{U}] \subseteq S[\mathcal{U}]$. As such, $XTD(\mathbb{C}, \mathcal{D}, m, \delta) \leq XTD(S \cap \mathbb{C}, \mathcal{D}, m, \delta/2)$.

Let $f \in \mathbb{C}_{\mathcal{F}}$. If $\mathcal{P}r_{X \sim \mathcal{D}}\{f(X) = +1\} < \frac{3}{4}\lambda$, then with probability $\geq 1 - \delta/2$, for each $h \in S \cap \mathbb{C}$, there is some x within the first $\frac{4}{\lambda} \ln \frac{2|S|}{\delta}$ examples in $\mathcal{U}$ s.t. $h(x) = +1 \neq f(x)$. Thus $\mathcal{P}r_{\mathcal{U} \sim \mathcal{D}^m}\left\{XTD(f, (\mathbb{C} \cap S)[\mathcal{U}], \mathcal{U}) > \frac{4}{\lambda} \ln \frac{2|S|}{\delta}\right\} \leq \delta/2$.

For any set of examples R, let $CLOS(R)$ be the smallest axis-aligned rectangle $h \in S$ that labels all of R as $+1$. This is known as the *closure* of R. Additionally, let $A \subseteq R$ be a smallest set such that $CLOS(A) = CLOS(R)$. This is known as a minimal spanning set of R. Clearly $|A| \leq 2n$, since the extreme points in each direction form a spanning set.

Let $h \in S$ be such that $\mathcal{P}r_{X \sim \mathcal{D}}\{h(X) = +1\} \geq \frac{\lambda}{2}$. Let $\{(a_i, b_i)\}_{i=1}^n$ define the rectangle. Let $x^{(ai)}$ be the example in $\mathcal{U}$ with largest $x_i^{(ai)}$ component such that $x_i^{(ai)} < a_i$ and $\forall j \neq i, a_j \leq x_j^{(ai)} \leq b_j$, or if no such example exists, $x^{(ai)}$ is defined as the $x \in \mathcal{U}$ with smallest x_i. Let $x^{(bi)}$ be defined similarly, except having the smallest $x_i^{(bi)}$ component with $x_i^{(bi)} > b_i$, and again $\forall j \neq i, a_j \leq x_j^{(bi)} \leq b_j$. If no such example exists, then $x^{(bi)}$ is defined as the $x \in \mathcal{U}$ with largest x_i. Let $A_{h,\mathcal{U}} \subseteq \mathcal{U}$ be the subsequence of all examples $x \in \mathcal{U}$ such that $\exists i \in \{1, 2, \ldots, n\}$ with $x_i^{(ai)} \leq x_i < a_i$ or $b_i < x_i \leq x_i^{(bi)}$. The surface volume of each face of the rectangle is at least $\lambda/2$. By a union bound over the $2n$ faces of the rectangle, with probability at least $1 - \delta/(4|S|)$, $|A_{h,\mathcal{U}}| \leq \frac{4n}{\lambda} \ln \frac{8n|S|}{\delta}$. With probability $\geq 1 - \delta/4$, this is satisfied for every $h \in S$ with $\mathcal{P}r_{X \sim \mathcal{D}}\{h(X) = +1\} \geq \frac{\lambda}{2}$.

Now suppose $f \in \mathbb{C}_{\mathcal{F}}$ satisfies $\mathcal{P}r_{X \sim \mathcal{D}}\{f(X) = +1\} \geq \frac{3\lambda}{4}$. Let $\mathcal{U}_+ = \{x \in \mathcal{U} : f(x) = +1\}$, $h_{clos} = CLOS(\mathcal{U}_+)$. If any $x \in \mathcal{U} \setminus \mathcal{U}_+$ has $h_{clos}(x) = +1$, we can form a specifying set for f on $\mathcal{U}$ with respect to $S[\mathcal{U}]$ using a minimal spanning set for $\mathcal{U}_+$ along with this x. If there is no such x, then $h_{clos}(\mathcal{U}) = f(\mathcal{U})$, and we use a minimal specifying set for h_{clos}. With probability $\geq 1 - \delta/4$, for every $h \in S$ such that $\mathcal{P}r_{X \sim \mathcal{D}}\{h(X) = +1\} < \frac{\lambda}{2}$, there is some $x \in \mathcal{U}_+$ such that $h(x) = -1$. If this happens, since $h_{clos} \in S$, this implies $\mathcal{P}r_{X \sim \mathcal{D}}\{h_{clos}(X) = +1\} \geq \frac{\lambda}{2}$. In this case, for a specifying set, we use $A_{h_{clos},\mathcal{U}}$ along with a minimal spanning set for $\mathcal{U}_+$. So $\mathcal{P}r_{\mathcal{U} \sim \mathcal{D}^m}\left\{XTD(f, (\mathbb{C} \cap S)[\mathcal{U}], \mathcal{U}) > 2n + \frac{4n}{\lambda} \ln \frac{8n|S|}{\delta}\right\} \leq \delta/2$. Noting that $|S| \leq \left(\frac{2nm^2}{\delta}\right)^{2n}$ completes the proof. $\square$

Note that we can obtain an estimate $\hat{p}$ of $p = \mathcal{P}r_{(X,Y) \sim \mathcal{D}_{XY}}\{Y = +1\}$ that, with probability $\geq 1 - \delta/2$, satisfies $p/2 \leq \hat{p} \leq 2p$, using at most $O\left(\frac{1}{p} \log \frac{1}{p\delta}\right)$ labeled examples (by guess-and-halve). Since clearly $\mathcal{P}r_{X \sim \mathcal{D}}\{h^*(X) = +1\} \geq p - \eta$, we can take $\lambda = (\hat{p}/2) - \eta$, giving the following oracle-dependent bound.

Theorem 5. *If $\mathcal{D}$ is as in Lemma 4 and $\mathbb{C}$ is the set of* all *axis-aligned rectangles, then if $p = \mathcal{P}r_{(X,Y) \sim \mathcal{D}_{XY}}\{Y = +1\} > 4\eta$, we can, with probability $\geq 1 - \delta$, find an $h \in \mathbb{C}$ with $er(h) \leq \nu + \epsilon$ without the number of label requests exceeding*

$$\tilde{O}\left(\frac{n^3}{(p/4) - \eta}\left(\frac{\eta^2}{\epsilon^2} + 1\right)\right).$$

This result is somewhat encouraging, since if $\eta < \epsilon$ and p is not too small, the label bound represents an exponential improvement in $\frac{1}{\epsilon}$ compared to known results for passive learning, while maintaining polylog dependence on $\frac{1}{\delta}$ and polynomial dependence on n, though the degree increases from 1 to 3. We might wonder whether the property of being balanced is sufficient for these improvements. However, as the following theorem shows, balancedness alone is insufficient for guaranteeing polylog dependence on $\frac{1}{\epsilon}$. The proof is omitted for brevity.

Theorem 6. *If $n \geq 2$, there is a distribution $\mathcal{D}'$ on $\mathbb{R}^n$ such that, if $\mathbb{C}$ is the set of axis-aligned rectangles h with $\mathcal{P}r_{X \sim \mathcal{D}'}\{h(X) = +1\} \geq \lambda$, then there is a $V \subset \mathbb{C}$ 2ϵ-separated with respect to $\mathcal{D}'$ such that $\Omega\left(\frac{(1-\delta)(1-\lambda)}{\epsilon}\right) \leq XPTD(V, \mathcal{D}', \delta)$.*

9 Open Problems

There are a number of possibilities for tightening these bounds. The upper bound of Theorem 3 contains a $O\left(\log \frac{d}{\epsilon\delta}\right)$ factor, which does not appear in any known lower bounds. In the worst case, when $XTD(\mathbb{C}, \mathcal{D}, n, \delta) = O(n)$, this factor clearly does not belong, since the bound exceeds the passive learning sample complexity in that case. It may be possible to reduce or remove this factor. On a related note, Hegedüs [3] introduces a modified MembHalving algorithm, which makes queries in a particular greedy order. By doing so, the bound decreases to $2\frac{t_0}{\log t_0} \log_2 |\mathbb{C}|$ instead of $t_0 \log_2 |\mathbb{C}|$. A similar technique might be possible here, though the effect seems more difficult to quantify. Additionally, a more careful treatment of the constants in these bounds may yield significant improvements.

The present analysis requires access to an upper bound η on the noise rate. As mentioned, it is possible to remove this assumption by a guess-and-double procedure, using a labeled validation set of size $\Omega(1/\epsilon)$. In practice, this may not be too severe, since we often use a validation set to tune parameters or estimate the final error rate anyway. Nonetheless, it would be nice to remove this requirement without sacrificing anything in dependence on $\frac{1}{\epsilon}$. In particular, it may sometimes be possible to determine whether a classifier is near-optimal using only a few carefully chosen queries.

As a final remark, exploring the connections between the present analysis and the related approaches discussed in Section 2 could prove fruitful. Thorough study of these approaches and their interrelations seems essential for a complete understanding of the label complexity of active learning.

References

1. Balcan, M.-F., Beygelzimer, A., Langford, J.: Agnostic active learning. In: Proc. of the 23rd International Conference on Machine Learning. (2006)
2. Kulkarni, S.R., Mitter, S.K., Tsitsiklis, J.N.: Active learning using arbitrary binary valued queries. Machine Learning **11** (1993) 23–35
3. Hegedüs, T.: Generalized teaching dimension and the query complexity of learning. In: Proc. of the 8th Annual Conference on Computational Learning Theory. (1995)
4. Angluin, D.: Queries revisited. Theoretical Computer Science **313** (2004) 175–194
5. Goldman, S.A., Kearns, M.J.: On the complexity of teaching. Journal of Computer and System Sciences **50** (1995) 20–31
6. Dasgupta, S.: Coarse sample complexity bounds for active learning. In: Advances in Neural Information Processing Systems 18. (2005)
7. Kääriäinen, M.: Active learning in the non-realizable case. In: Proc. of the 17th International Conference on Algorithmic Learning Theory. (2006)
8. Littlestone, N.: Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. Machine Learning **2** (1988) 285–318
9. Haussler, D.: Decision theoretic generalizations of the PAC model for neural net and other learning applications. Information and Computation **100** (1992) 78–150
10. Wald, A.: Sequential tests of statistical hypotheses. The Annals of Mathematical Statistics **16** (1945) 117–186
11. Bar-Yossef, Z.: Sampling lower bounds via information theory. In: Proc. of the 35th Annual ACM Symposium on the Theory of Computing. (2003) 335–344

The Cost Complexity of Interactive Learning

Steve Hanneke
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213 USA
`shanneke@cs.cmu.edu`

June 2006

Abstract

In this paper, I describe a general framework in which a learning algorithm is tasked with learning some concept from a known class by interacting with a teacher via questions. Each question has an arbitrary known cost associated with it, which the learner is required to pay in order to have the question answered. Exploring the information-theoretic limits of this framework, I define a notion called the *cost complexity* of learning, analogous to traditional notions of sample complexity. I discuss this topic for the Exact Learning setting as well as PAC Learning with a pool of unlabeled examples. In the former case, the learner is allowed to ask *any* question, while in the latter case, all questions must concern the target concept's behavior on a set of unlabeled examples. In both settings, I derive upper and lower bounds on the cost complexity of learning, based on a combinatorial quantity I call the *General Identification Cost*.

1 Introduction

The ability to ask questions to a knowledgeable teacher can make learning easier. This fact is no secret to any elementary school student. But how much easier? Some questions are more difficult for the teacher to answer than others. How much inconvenience must even the most conscientious learner cause to a teacher in order to learn a concept? This paper explores these and related questions about the fundamental advantages and limitations of learning by interaction.

In machine learning research, it is becoming increasingly apparent that well-designed interactive learning algorithms can provide valuable improvements in learning performance while reducing the amount of effort required of a human annotator. This research has mainly focused on two formal settings of learning: Exact Learning by queries and pool-based Active PAC Learning. Informally, the objective in the setting of Exact Learning by queries is to perfectly identify a target concept (classifier) by asking questions. In contrast, the pool-based Active PAC setting is concerned only with approximating the concept with high probability with respect to an unknown distribution on the set of possible instances. In this latter setting, the learning algorithm is restricted to asking only questions that relate to the concept's behavior on a particular set of unannotated instances drawn independently from the unknown distribution.

In this paper, I study both of these active learning settings under a broad definition. Specifically, I consider a learning protocol in which the learner can ask *any* question, but each possible question has an associated *cost*. For example, a query of the form "what is the label of example x" might cost \$1, while a query of the form "show me a positive example" might cost \$10. The objective is to learn the concept while minimizing the total *cost* of queries made. One would like to know how much cost even the most clever learner might be required to pay to learn a concept from a particular concept space in the worst case. This can be viewed as a generalization of notions of *sample complexity* or *query complexity* found in the learning theory literature. I refer to this best worst case cost as the *cost complexity* of learning. This quantity is defined without reference to computational feasibility, focusing instead on the information-theoretic boundaries of this setting (in the limit of unbounded computation). Below, I derive bounds on the cost complexity of learning, as a function of the concept space and cost function, for both Exact Learning from queries and pool-based Active PAC Learning.

Section 2 formally introduces the setting of Exact Learning from queries, describes some related work, and defines cost complexity for that setting. It also serves to introduce the notation and fundamental definitions used throughout this paper. The section closely parallels the work of Balcázar et al. [1]. The primary contribution of Section 2 is a derivation of upper and lower bounds on the cost complexity of Exact Learning from queries. This is followed, in Section 3, by a formal definition of pool-base Active PAC Learning and extension of the notion of cost complexity to that setting. The primary contributions of Section 3 include a derivation of upper and lower bounds on the cost complexity of learning in that general setting, as well as an interesting corollary for intersection-closed concept spaces. I know of no previous work giving general results of this type.

2 Active Exact Learning

In this setting, there is an *instance space* $\mathcal{X}$ and *concept space* $\mathcal{C}$ on $\mathcal{X}$ such that any $h \in \mathcal{C}$ is a distinct function $h : \mathcal{X} \to \{0, 1\}$.[1] Additionally, define $\mathcal{C}^* = \{h : \mathcal{X} \to \{0, 1\}\}$. That is, $\mathcal{C}^*$ is the *most general* concept space, containing all possible labelings of $\mathcal{X}$. In particular, any concept space $\mathcal{C}$ is a subset of $\mathcal{C}^*$. For a particular learning problem, there is an unknown *target concept* $f \in \mathcal{C}$, and the task is to identify f using a teacher's answers to queries made by the learning algorithm. Formally, an *actual query* is any function in $\tilde{Q} = \{\tilde{q} : \mathcal{C}^* \to 2^{\mathcal{A}^*} \setminus \{\varnothing\}\}$,[2] for some *answer set* $\mathcal{A}^*$. By a learning algorithm "making an actual query", I mean that it selects a function $\tilde{q} \in \tilde{Q}$, passes it to the teacher, and the teacher returns a single *answer* $\tilde{a} \in \tilde{q}(f)$ where f is the target concept. A concept $h \in \mathcal{C}^*$ is *consistent* with an answer $\tilde{a}$ to an actual query $\tilde{q}$ if $\tilde{a} \in \tilde{q}(h)$. Thus, I assume the teacher always returns an answer that the target concept is consistent with; however, when there are multiple such answers, the teacher may arbitrarily select from amongst them.

Traditionally, the subject of active learning has been studied with respect to specific restricted query types, such as membership queries, and the learning algorithm's objective has been to minimize the *number* of queries used to learn. However, it is often the case that learning with these simple types of queries is difficult, but if the learning algorithm is allowed just a few *special* queries, learning becomes significantly easier. The reason we are initially reluctant to allow the learner to ask certain types of queries is that these queries are difficult, expensive, or sometimes impossible to answer. However, we can incorporate this difficulty level into the framework by assigning each query type a specific *cost*, and then allowing the learning algorithm to explicitly optimize the *cost* needed to learn, rather than the *number* of queries. In addition to allowing the algorithm to trade off between different types of queries, this also gives us the added flexibility to specify different costs within the same family (e.g., perhaps some membership queries are more expensive than others).

Formally, in this framework there is a *cost function*. Let $\alpha > 0$ be a constant. A cost function is any $c : \tilde{Q} \to (\alpha, \infty]$. In practice, c would typically be defined by the user responsible for answering the queries, and could be based on the time, resources, or operating expenses necessary to obtain the answer. Note that if a particular type of query is unanswerable for a particular application, or if the user wishes to work with a reduced set of possible queries, one can always define the costs of those undesirable query types to be ∞, so that any reasonable learning algorithm ignores them if possible.

While the notion of *actual query* closely corresponds to the actual mechanism of querying in practice, it will be more convenient to work with the information-theoretic implications of these queries. Define the set of *effective queries* $\mathcal{Q} = \{q : \mathcal{C}^* \to 2^{2^{\mathcal{C}^*}} \setminus \{\varnothing\} | \forall f \in \mathcal{C}^*, a \in q(f) \Rightarrow [f \in a \wedge \forall h \in a, a \in q(h)]\}$. Each effective query corresponds to an equivalence class of actual queries, defined by mapping any answer to the set of concepts consistent with it. We can thus define the mapping

---

[1] All of the main results easily generalize to multiclass as well.

[2] The restriction that $\tilde{q}(f) \neq \{\}$ is a bit like an assumption that every valid question has at least one answer for any target concept. However, we can always define some particular answer to mean "there is no answer," so this restriction is really more of a notational convenience than an assumption.

$$\mathcal{E}(q) = \{\tilde{q} | \tilde{q} \in \tilde{Q}, \forall f \in \mathcal{C}^*, [\exists \tilde{a} \in \tilde{q}(f) \text{ with } a = \{h | h \in \mathcal{C}^*, \tilde{a} \in \tilde{q}(h)\}] \Leftrightarrow a \in q(f)\}.$$

By an algorithm "making an effective query q," I mean that it makes an actual query in $\mathcal{E}(q)$,[3] (a good algorithm will pick a cheaper actual query). For the purpose of this best-worst-case analysis, the following definition is appropriate. For a cost function c, define a corresponding *effective cost function* (overloading notation) $c : \mathcal{Q} \rightarrow [\alpha, \infty]$, such that $\forall q \in \mathcal{Q}, c(q) = \inf_{\tilde{q} \in \mathcal{E}(q)} c(\tilde{q})$. The following definitions illustrate how query types can be defined using effective queries.

A *positive example query* is any $\tilde{q} \in \mathcal{E}(q_S)$ for some $S \subseteq \mathcal{X}$, such that $q_S \in \mathcal{Q}$ is defined by $\forall f \in \mathcal{C}^*$ s.t. $[\exists x \in S : f(x) = 1], q_S(f) = \{\{h | h \in \mathcal{C}^*, h(x) = 1\} | x \in S : f(x) = 1\}$, and $\forall f \in \mathcal{C}^*$ s.t. $[\forall x \in S, f(x) = 0], q_S(f) = \{\{h | h \in \mathcal{C}^* : \forall x \in S, h(x) = 0\}\}$.

A *membership query* is any $\tilde{q} \in \mathcal{E}(q_{\{x\}})$ for some $x \in \mathcal{X}$. This special case of a positive example query can equivalently be defined by $\forall f \in \mathcal{C}^*, q_{\{x\}}(f) = \{\{h | h \in \mathcal{C}^*, h(x) = f(x)\}\}$.

These effectively correspond to asking for any example labeled 1 in S or an indication that there are none (positive example query), and asking for the label of a particular example in $\mathcal{X}$ (membership query). I will refer to these two query types in subsequent examples, but the reader should keep in mind that the theorems below apply to *all* types of queries.

Additionally, it will be useful to have a notion of an *effective oracle*, which is an unknown function defining how the teacher will answer the various queries. Formally, an effective oracle T is any function in $\mathcal{T} = \{T : \mathcal{Q} \rightarrow 2^{\mathcal{C}^*} | \forall q \in \mathcal{Q}, T(q) \in \cup_{f \in \mathcal{C}^*} q(f)\}$.[4] For convenience, I also overload this notation, defining for a set of queries $R \subseteq \mathcal{Q}$, $T(R) = \cap_{q \in R} T(q)$.

Definition 2.1. *A* learning algorithm *$\mathcal{A}$ for $\mathcal{C}$ using cost function c is any algorithm which, for any (unknown) target concept $f \in \mathcal{C}$, by a finite number of finite cost actual queries, is guaranteed to reduce the set of concepts in $\mathcal{C}$ consistent with the answers to precisely $\{f\}$. A concept space $\mathcal{C}$ is* learnable *with cost function c using total cost t if there exists a learning algorithm for $\mathcal{C}$ using c guaranteed to have the sum of costs of the queries it makes at most t.*[5]

Definition 2.2. *For any instance space $\mathcal{X}$, concept space $\mathcal{C}$ on $\mathcal{X}$, and cost function c, define the* cost complexity*, denoted CostComplexity$(\mathcal{C}, c)$, as the infimum $t \geq 0$ such that $\mathcal{C}$ is learnable with cost function c using total cost no greater than t.*

Equivalently, we can define cost complexity using the following recurrence. If $|\mathcal{C}| = 1$, *CostComplexity*$(\mathcal{C}, c) = 0$. Otherwise,

$$\mathit{CostComplexity}(\mathcal{C}, c) = \inf_{\tilde{q} \in \tilde{Q}} c(\tilde{q}) + \max_{f \in \mathcal{C}, \tilde{a} \in \tilde{q}(f)} \mathit{CostComplexity}(\{h | h \in \mathcal{C}, \tilde{a} \in \tilde{q}(h)\}, c)$$

Since

$$\begin{aligned} &\inf_{\tilde{q} \in \tilde{Q}} c(\tilde{q}) + \max_{f \in \mathcal{C}, \tilde{a} \in \tilde{q}(f)} \mathit{CostComplexity}(\{h | h \in \mathcal{C}, \tilde{a} \in \tilde{q}(h)\}, c) \\ &\quad = \inf_{q \in \mathcal{Q}} \inf_{\tilde{q} \in \mathcal{E}(q)} c(\tilde{q}) + \max_{f \in \mathcal{C}, \tilde{a} \in \tilde{q}(f)} \mathit{CostComplexity}(\mathcal{C} \cap \{h | h \in \mathcal{C}^*, \tilde{a} \in \tilde{q}(h)\}, c) \\ &\qquad = \inf_{q \in \mathcal{Q}} c(q) + \max_{f \in \mathcal{C}, a \in q(f)} \mathit{CostComplexity}(\mathcal{C} \cap a, c), \end{aligned}$$

we can equivalently define cost complexity in terms of *effective queries* and *effective cost*. That is, *CostComplexity*$(\mathcal{C}, c)$ is the infimum $t \geq 0$ such that there is an algorithm guaranteed to identify any $f \in \mathcal{C}$ using *effective* queries with total of *effective* costs no greater than t.

---

[3] I assume $\mathcal{A}^*$ is sufficiently expressive so that $\forall q \in \mathcal{Q}, \mathcal{E}(q) \neq \varnothing$; alternatively, we could define $\mathcal{E}(q) = \varnothing \Rightarrow c(q) = \infty$ without sacrificing the main theorems. Additionally, I will assume that it is possible to find an actual query in $\mathcal{E}(q)$ with cost arbitrarily close to $\inf_{\tilde{q} \in \mathcal{E}(q)} c(\tilde{q})$ for any $q \in \mathcal{Q}$ using finite computation.

[4] An effective oracle corresponds to a deterministic stateless teacher, which gives up as little information as possible. It is also possible to analyze a setting in which asking two queries from the same equivalence class, or asking the same question twice, can possibly lead to two different answers. However, the worst case in both settings is identical, so the worst case results obtained for this setting also apply to the more general case.

[5] I have made the dependence of $\mathcal{A}$ on the teacher implicit. To be formally correct, $\mathcal{A}$ should have the teacher's effective oracle T as input, and is guaranteed to output f for any $T \in \mathcal{T}$ s.t. $\forall q \in \mathcal{Q}, T(q) \in q(f)$. Cost is then a book-keeping device recording how $\mathcal{A}$ uses T during execution.

2.1 Related Work

There have been a relatively large number of contributions to the study of Exact Learning from queries. In particular, much interest has been given to settings in which the learning algorithm is restricted to a few specific types of queries (e.g. membership queries and equivalence queries). However, these contributions focus entirely on the *number* of queries needed, rather than *cost*. The most relevant work in this area is by Balcázar, Castro, and Guijarro [1]. Prior to publication of [2], there were a variety of publications in which the learning algorithm could use some specific set of queries, and which derived bounds on the number of queries any algorithm might be required to make in the worst case in order to learn. For example, [3] analyzed the combination of membership and proper equivalence queries, [4] additionally analyzed learning from membership queries alone, while [5] considered learning from just proper equivalence queries. Amidst these various special case analyses, somewhat surprisingly, Balcázar et al. [2] discovered that the query complexity bounds derived in these works were all special cases of a single general theorem, applying to the broad class of *sample-based queries*. They further generalized this result in [1], giving results that apply to any combination of *any* query types. That work defines an abstract combinatorial quantity, which they call the *General Dimension*, which provides a lower bound on the query complexity, and is within a log factor of it. Furthermore, the General Dimension can actually be computed for a variety of interesting combinations of query types. Until now there has not been any analysis I know of that considers learning with *all* query types, but giving each query a cost, and bounding the worst-case *cost* that a learning algorithm might be required to incur. In particular, the analysis of the next subsection can be viewed as a generalization of [1] to add this notion of cost, such that [1] represents the special case of cost that is uniformly 1 on a particular set of queries and ∞ on all other queries.

2.2 Cost Complexity Bounds

I now turn to the subject of exploring the fundamental limits of interactive learning in terms of cost. This discussion closely parallels that of Balcázar, Castro, and Guijarro [1].

Definition 2.3. *For any instance space $\mathcal{X}$, concept space $\mathcal{C}$ on $\mathcal{X}$, and cost function c, define the* General Identification Cost*, denoted $GIC(\mathcal{C}, c)$, as follows.*

$$GIC(\mathcal{C}, c) = \inf\{t | t \geq 0, \forall T \in \mathcal{T}, \exists R \subseteq \mathcal{Q}, s.t. [\textstyle\sum_{q \in R} c(q) \leq t] \wedge [|\mathcal{C} \cap T(R)| \leq 1]\}$$

We can also express this as $GIC(\mathcal{C}, c) = \sup_{T \in \mathcal{T}} \inf_{R \subseteq \mathcal{Q} : |\mathcal{C} \cap T(R)| \leq 1} \sum_{q \in R} c(q)$. Note that calculating this corresponds to a much simpler optimization problem than calculating the cost complexity. The General Identification Cost is a direct generalization of the General Dimension of [1], which itself generalizes quantities such as Extended Teaching Dimension [4], Strong Consistency Dimension [5], and the Certificate Sizes of [3]. It can be interpreted as a sort of game. This game is similar to the usual setting, except that the teacher's answers are not restricted to be consistent with a concept. Imagine there is a helpful spy who knows precisely how the teacher will respond to every query. The spy is able to suggest queries to the learner, and wishes to cause the learner to pay as little as possible. If the spy is sufficiently clever at suggesting queries, and the learner follows every suggestion by the spy, then after asking some minimal cost set of queries the learner can narrow the set of concepts in $\mathcal{C}$ consistent with the answers down to at most one. The General Identification Cost is precisely the worst case limiting cost the learner might be forced to pay during this process, no matter how clever the spy is at suggesting queries.

Lemma 2.1. *For any instance space $\mathcal{X}$, concept space $\mathcal{C}$ on $\mathcal{X}$, and cost function c, if $V \subseteq \mathcal{C}$, then $GIC(V, c) \leq GIC(\mathcal{C}, c)$.*

Proof. It clearly holds if $GIC(\mathcal{C}, c) = \infty$. If $GIC(\mathcal{C}, c) < k$, then $\forall T \in \mathcal{T}$, $\exists R \subseteq \mathcal{Q}$ s.t. $\sum_{q \in R} c(q) < k$ and $1 \geq |\mathcal{C} \cap T(R)| \geq |V \cap T(R)|$, and therefore $GIC(V, c) < k$. The limit as $k \to GIC(\mathcal{C}, c)$ gives the result. □

Lemma 2.2. *For any $\gamma > 0$, instance space $\mathcal{X}$, finite concept space $\mathcal{C}$ on $\mathcal{X}$ with $|\mathcal{C}| > 1$, and cost function c such that $GIC(\mathcal{C}, c) < \infty$, $\exists q \in \mathcal{Q}$ such that $\forall T \in \mathcal{T}$,*

$$|\mathcal{C} \setminus T(q)| \geq c(q) \frac{|\mathcal{C}| - 1}{GIC(\mathcal{C}, c) + \gamma}.$$

That is, regardless of which answer the teacher picks, there are at least $c(q)\frac{|\mathcal{C}|-1}{GIC(\mathcal{C},c)+\gamma}$ *concepts in* $\mathcal{C}$ *inconsistent with the answer.*

Proof. Suppose $\forall q \in \mathcal{Q}, \exists T_q \in \mathcal{T}$ such that $|\mathcal{C} \setminus T_q(q)| < c(q)\frac{|\mathcal{C}|-1}{GIC(\mathcal{C},c)+\gamma}$. Then define an effective oracle T with the property that $\forall q \in \mathcal{Q}, T(q) = T_q(q)$. We have thus defined an oracle such that $\forall R \subseteq \mathcal{Q}, \sum_{q\in R} c(q) \leq GIC(\mathcal{C},c) + \gamma \Rightarrow$

$$\begin{aligned}|\mathcal{C} \cap T(R)| = |\mathcal{C}| - |\mathcal{C} \setminus T(R)| \geq |\mathcal{C}| - \sum_{q\in R} |\mathcal{C} \setminus T_q(q)|\\ > |\mathcal{C}| - \sum_{q\in R} c(q)\frac{|\mathcal{C}|-1}{GIC(\mathcal{C},c)+\gamma} \geq |\mathcal{C}| - (GIC(\mathcal{C},c)+\gamma)\frac{|\mathcal{C}|-1}{GIC(\mathcal{C},c)+\gamma} = 1.\end{aligned}$$

In particular, this contradicts the definition of $GIC(\mathcal{C},c)$. □

This brings us to the main theorem of this section.

Theorem 2.1. *For any instance space* $\mathcal{X}$*, concept space* $\mathcal{C}$ *on* $\mathcal{X}$*, and cost function c,*

$$GIC(\mathcal{C},c) \leq \mathit{CostComplexity}(\mathcal{C},c) \leq GIC(\mathcal{C},c)\log_2 |\mathcal{C}|$$

Proof. I begin with the lower bound. Let $k < GIC(\mathcal{C},c)$. By definition of GIC, $\exists T \in \mathcal{T}$, such that $\forall R \subseteq \mathcal{Q}, \sum_{q\in R} c(q) \leq k \Rightarrow |\mathcal{C} \cap T(R)| > 1$. In particular, this implies that an adversarial teacher can answer any sequence of queries with cost no greater than k in a way that leaves at least 2 concepts in $\mathcal{C}$ consistent with the answers, either of which could be the target concept f. This implies *CostComplexity*$(\mathcal{C},c) > k$. The limit as $k \to GIC(\mathcal{C},c)$ gives the bound.

Next I prove the upper bound. If $GIC(\mathcal{C},c) = \infty$ or $|\mathcal{C}| = \infty$, the bound holds vacuously, so let us assume these are finite. Say the teacher's answers correspond to some effective oracle $T \in \mathcal{T}$. Consider a recursive algorithm $\mathcal{A}_\gamma$ that makes effective queries from $\mathcal{Q}$.[6] If $|\mathcal{C}| = 1$, then $\mathcal{A}_\gamma$ halts and outputs the single remaining concept. Otherwise, let q be an effective query having the property guaranteed by Lemma 2.2. That is, $|\mathcal{C} \setminus T(q)| \geq c(q)\frac{|\mathcal{C}|-1}{GIC(\mathcal{C},c)+\gamma}$. Defining $V = \mathcal{C} \cap T(q)$ (a generalized notion of *version space*), this implies that $c(q) \leq (GIC(\mathcal{C},c)+\gamma)\frac{|\mathcal{C}|-|V|}{|\mathcal{C}|-1}$ and $|V| < |\mathcal{C}|$. Say $\mathcal{A}_\gamma$ makes effective query q, and then recurses on V. In particular, we can immediately see that this algorithm identifies f using no more than $|\mathcal{C}| - 1$ queries.

I now prove by induction on $|\mathcal{C}|$ that *CostComplexity*$(\mathcal{C},c) \leq (GIC(\mathcal{C},c)+\gamma)H_{|\mathcal{C}|-1}$, where $H_n = \sum_{i=1}^{n} \frac{1}{i}$ is the n^{th} harmonic number. If $|\mathcal{C}| = 1$, then the cost complexity is 0. For $|\mathcal{C}| > 1$,

CostComplexity$(\mathcal{C},c)$

$$\begin{aligned}&\leq c(q) + \mathit{CostComplexity}(V,c)\\ &\leq (GIC(\mathcal{C},c)+\gamma)\frac{|\mathcal{C}|-|V|}{|\mathcal{C}|-1} + (GIC(V,c)+\gamma)H_{|V|-1}\\ &\leq (GIC(\mathcal{C},c)+\gamma)\left(\frac{|\mathcal{C}|-|V|}{|\mathcal{C}|-1} + H_{|V|-1}\right)\\ &\leq (GIC(\mathcal{C},c)+\gamma)H_{|\mathcal{C}|-1}\end{aligned}$$

where the second inequality uses the inductive hypothesis along with the properties of q guaranteed by Lemma 2.2, and the third inequality uses Lemma 2.1. Finally, noting that $H_{|\mathcal{C}|-1} \leq \log_2 |\mathcal{C}|$ and taking the limit as $\gamma \to 0$ proves the theorem. □

---

[6] I use the definition of cost complexity in terms of effective cost, so that we need not concern ourselves with how $\mathcal{A}_\gamma$ chooses its *actual queries*. However, we could define $\mathcal{A}_\gamma$ to make actual queries with cost within γ of the effective query cost, so that the result still holds as $\gamma \to 0$.

2.3 An Example: Discrete Intervals

As a simple example of cost complexity, consider $\mathcal{X} = \{1, 2, \ldots, N\}$, for $N \geq 4$, $\mathcal{C} = \{h_{a,b} : \mathcal{X} \to \{0,1\} | a, b \in \mathcal{X}, a \leq b, \forall x \in \mathcal{X}, [a \leq x \leq b \Leftrightarrow h_{a,b}(x) = 1]\}$, and define an effective cost function c that is 1 for membership queries $q_{\{x\}}$ for any $x \in \mathcal{X}$, k for the positive example query $q_{\mathcal{X}}$ where $3 \leq k \leq N - 1$, and ∞ for any other queries. In this case, $GIC(\mathcal{C}, c) = k + 1$. In the spy game, say the teacher answers effective queries with an effective oracle T. Let $\mathcal{X}_+ = \{x | x \in \mathcal{X}, T(q_{\{x\}}) = \{h | h \in \mathcal{C}^*, h(x) = 1\}\}$. If $\mathcal{X}_+ \neq \varnothing$, then let $a = \min \mathcal{X}_+$ and $b = \max \mathcal{X}_+$. The spy tells the learner to make queries $q_{\{a\}}, q_{\{b\}}, q_{\{a-1\}}$ (if $a > 1$), and $q_{\{b+1\}}$ (if $b < N$). This narrows the version space to $\{h_{a,b}\}$, at a worst-case effective cost of 4. If $\mathcal{X}_+ = \varnothing$, then the spy suggests query $q_{\mathcal{X}}$. If $T(q_{\mathcal{X}}) = \{f_-\}$, the "all 0" concept, then no concepts in $\mathcal{C}$ are consistent. Otherwise, $T(q_{\mathcal{X}}) = \{h | h \in \mathcal{C}^*, h(x) = 1\}$ for some $x \in \mathcal{X}$, and the spy suggests membership query $q_{\{x\}}$. In this case, $T(q_{\{x\}}) \cap T(q_{\mathcal{X}}) = \varnothing$, so the worst-case cost is $k + 1$ (without $q_{\mathcal{X}}$, it would cost $N - 1$). These are the only cases to consider, so $GIC(\mathcal{C}, c) = k + 1$. By Theorem 2.1, this implies $k + 1 \leq \mathit{CostComplexity}(\mathcal{C}, c) \leq 2(k + 1)\log_2 N$.

We can slightly improve this by noting that we only use $q_{\mathcal{X}}$ once. Specifically, if a learning algorithm begins (in the regular setting) by asking $q_{\mathcal{X}}$, revealing that $f(x) = 1$ for some $x \in \mathcal{X}$, then we can reduce to two disjoint learning problems, with concept spaces $\mathcal{C}_1' = \{h_{x,b} | b \in \{x, \ldots, N\}\}$, and $\mathcal{C}_2' = \{h_{a,x} | a \in \{1, 2, \ldots, x\}\}$, with cost functions $c_1(q) = c(q)$ for $q \in \{q_{\{x\}}, q_{\{x+1\}}, \ldots, q_{\{N\}}\}$ and ∞ otherwise, and $c_2(q) = c(q)$ for $q \in \{q_{\{1\}}, q_{\{2\}}, \ldots, q_{\{x\}}\}$ and ∞ otherwise, and corresponding $GIC(\mathcal{C}_1', c) \leq 2$, $GIC(\mathcal{C}_2', c) \leq 2$. So we can say that $\mathit{CostComplexity}(\mathcal{C}, c) \leq k + \mathit{CostComplexity}(\mathcal{C}_1', c_1) + \mathit{CostComplexity}(\mathcal{C}_2', c_2) \leq k + 4\log_2 N$. One algorithm that achieves this begins by making the positive example query, and then performs binary search above and below the indicated positive example to find the boundaries.

3 Pool-Based Active PAC Learning

In many scenarios, a more realistic definition of learning is that supplied by the Probably Approximately Correct (PAC) model. In this case, unlike the previous section, we are interested only in discovering with high probability a function with behavior very *similar* to the target concept on examples sampled from some distribution. Formally, as above there is an instance space $\mathcal{X}$, and a concept space $\mathcal{C} \subseteq \mathcal{C}^*$ on $\mathcal{X}$; unlike above, there is also a distribution $\mathcal{D}$ over $\mathcal{X}$, and I assume $\mathcal{C}$ is well-behaved in a measure-theoretic sense[7]. As with Exact Learning, the learning algorithm interacts with a teacher by making queries. However, in this setting the learning algorithm is given as input a finite sequence[8] of unlabeled examples $\mathcal{U}$, each drawn independently according to $\mathcal{D}$, and *all queries* made by the algorithm must concern only the behavior of the target concept on examples in $\mathcal{U}$.Formally, a *data-dependent cost function* is any function $c : \tilde{Q} \times 2^{\mathcal{X}} \to (\alpha, \infty]$. For a given set of unlabeled examples $\mathcal{U}$, and data-dependent cost function c, define $c_{\mathcal{U}}(\cdot) = c(\cdot, \mathcal{U})$. Thus, $c_{\mathcal{U}}$ is a cost function in the sense of the previous section. For a given $c_{\mathcal{U}}$, the corresponding effective cost function $c_{\mathcal{U}} : \mathcal{Q} \to [\alpha, \infty]$ is defined as in the previous section.

Definition 3.1. *Let $\mathcal{X}$ be an instance space, $\mathcal{C}$ a concept space on $\mathcal{X}$, and $\mathcal{U} = (x_1, x_2, \ldots, x_{|\mathcal{U}|})$ a finite sequence of unlabeled examples. Define $\forall h \in \mathcal{C}, h(\mathcal{U}) = (h(x_1), h(x_2), \ldots, h(x_{|\mathcal{U}|}))$. Define[9] $\mathcal{C}[\mathcal{U}] \subseteq \mathcal{C}$ as any concept space such that $\forall h \in \mathcal{C}, |\{h' | h' \in \mathcal{C}[\mathcal{U}], h'(\mathcal{U}) = h(\mathcal{U})\}| = 1$.*

Definition 3.2. *A* sample-based cost function *is any data-dependent cost function c such that for all finite $\mathcal{U} \subseteq \mathcal{X}, \forall q \in \mathcal{Q}$,*

$$c_{\mathcal{U}}(q) < \infty \Rightarrow \forall f \in \mathcal{C}^*, \forall a \in q(f), \forall h \in \mathcal{C}^*, [h(\mathcal{U}) = f(\mathcal{U}) \Rightarrow h \in a].$$

This corresponds to queries that are about *the target concept's labels on some subset of $\mathcal{U}$. Additionally, $\forall \mathcal{U} \subseteq \mathcal{X}, x \in \mathcal{X}$, and $q \in \mathcal{Q}$, $c(q, \mathcal{U} \cup \{x\}) \leq c(q, \mathcal{U})$. That is, in addition to the above property, adding extra examples to which q's answers do not refer does not increase its cost.*

---

[7] This mild assumption has almost no practical impact. See [6] for a full description.

[8] I will implicitly overload all notation for sets and sequences, so that if a set is used where a sequence is required, then an arbitrary ordering of the set is implied (though this ordering should be used consistently), and if a sequence is used where a set is required, then the set of distinct elements of the sequence is implied.

[9] The choice of which concept from each equivalence class to include in $\mathcal{C}[\mathcal{U}]$ can be made arbitrarily.

For example, membership queries on $x \in \mathcal{U}$ and positive examples queries on $S \subseteq \mathcal{U}$ could have finite costs under a sample-based cost function. As in the previous section, there is a target concept $f \in \mathcal{C}$, but unlike that section, we do not try to *identify* f, but instead attempt to *approximate* it with high probability.

Definition 3.3. *For instance space $\mathcal{X}$, concept space $\mathcal{C}$ on $\mathcal{X}$, distribution $\mathcal{D}$ on $\mathcal{X}$, target concept $f \in \mathcal{C}$, and concept $h \in \mathcal{C}$, define the* error rate *of h, denoted $error_{\mathcal{D}}(h, f)$, as*

$$error_{\mathcal{D}}(h, f) = \mathcal{P}r_{X \sim \mathcal{D}} \{h(X) \neq f(X)\}$$

Definition 3.4. *For $(\epsilon, \delta) \in (0, 1)^2$, an (ϵ, δ)-*learning algorithm *for $\mathcal{C}$ using sample-based cost function c is any algorithm $\mathcal{A}$ taking as input a finite sequence of unlabeled examples, such that for any target concept $f \in \mathcal{C}$ and finite sequence $\mathcal{U}$, $\mathcal{A}(\mathcal{U})$ outputs a concept in $\mathcal{C}$ after making a finite number of actual queries with finite costs under $c_{\mathcal{U}}$. Additionally, any (ϵ, δ)-*learning algorithm *$\mathcal{A}$ has the property that $\exists m \in [0, \infty)$ such that, for any target concept $f \in \mathcal{C}$ and distribution $\mathcal{D}$ on $\mathcal{X}$,*

$$\mathcal{P}r_{\mathcal{U} \sim \mathcal{D}^m} \{error_{\mathcal{D}}(\mathcal{A}(\mathcal{U}), f) > \epsilon\} \leq \delta.$$

*A concept space $\mathcal{C}$ is (ϵ, δ)-*learnable *given sample-based cost function c using total cost t if there exists an (ϵ, δ)-learning algorithm $\mathcal{A}$ for $\mathcal{C}$ using c such that for all finite example sequences $\mathcal{U}$, $\mathcal{A}(\mathcal{U})$ is guaranteed to have the sum of costs of the queries it makes at most t under $c_{\mathcal{U}}$.*

Definition 3.5. *For any instance space $\mathcal{X}$, concept space $\mathcal{C}$ on $\mathcal{X}$, sample-based cost function c, and $(\epsilon, \delta) \in (0, 1)^2$, define the (ϵ, δ)-*cost complexity*, denoted CostComplexity$(\mathcal{C}, c, \epsilon, \delta)$, as the infimum $t \geq 0$ such that $\mathcal{C}$ is (ϵ, δ)-learnable given c using total cost no greater than t.*

As in the previous section, because it is the *limiting* case, we can equivalently define the (ϵ, δ)-cost complexity as the infimum $t \geq 0$ such that there is an (ϵ, δ)-learning algorithm guaranteed to have the sum of *effective* costs of the *effective* queries it makes at most t.

The main results from this section include a new combinatorial quantity $GPIC(\mathcal{C}, c, m, \tau)$ such that if d is the VC-dimension of $\mathcal{C}$, then

$$GPIC(\mathcal{C}, c, \Theta(\tfrac{1}{\epsilon}), \delta) \leq \textit{CostComplexity}(\mathcal{C}, c, \epsilon, \delta) \leq GPIC(\mathcal{C}, c, \tilde{\Theta}\left(\tfrac{d}{\epsilon}\right), 0)\tilde{\Theta}(d).$$

3.1 Related Work

Previous work on pool-based active learning in the PAC model has been restricted almost exclusively to uniform-cost membership queries on examples in the unlabeled set $\mathcal{U}$. There has been some recent progress on query complexity bounds for that restricted setting. Specifically, Dasgupta [7] analyzes a greedy active learning scheme and derives bounds for the number of membership queries in $\mathcal{U}$ it uses under an *average case* setting, in which the target concept is selected randomly from a known distribution. A similar type of analysis was previously given by Freund et al. [8] to prove positive results for the Query by Committee algorithm. In a subsequent paper, Dasgupta [9] derives upper and lower bounds on the number of membership queries in $\mathcal{U}$ required for active learning for any particular distribution $\mathcal{D}$, under the assumption that $\mathcal{D}$ is known. The results I derive in this section imply *worst-case* results (over both $\mathcal{D}$ and f) for this as a special case of more general bounds applying to *any* sample-based cost function.

3.2 Cost Complexity Upper Bounds

I now derive bounds on the cost complexity of pool-based Active PAC Learning.

Definition 3.6. *For an instance space $\mathcal{X}$, concept space $\mathcal{C}$ on $\mathcal{X}$, sample-based cost function c, and nonnegative integer m, define the* General Identification Cost Growth Function*, denoted $GIC(\mathcal{C}, c, m)$, as follows.*

$$GIC(\mathcal{C}, c, m) = \sup_{\mathcal{U} \in \mathcal{X}^m} GIC(\mathcal{C}[\mathcal{U}], c_{\mathcal{U}})$$

Definition 3.7. *For any instance space $\mathcal{X}$, concept space $\mathcal{C}$ on $\mathcal{X}$, and $(\epsilon, \delta) \in (0, 1)^2$, let $M(\mathcal{C}, \epsilon, \delta)$ denote the* sample complexity *of $\mathcal{C}$ (in the classic* passive learning *sense), or the smallest m such that there is an algorithm $\mathcal{A}$ taking as input a set of examples $\mathcal{L}$ and labels, and outputting a classifier* (without making any queries)*, such that for any $\mathcal{D}$ and $f \in \mathcal{C}$,*

$$\mathcal{P}r_{\mathcal{L} \sim \mathcal{D}^m} \{error_{\mathcal{D}}(\mathcal{A}(\mathcal{L}, f(\mathcal{L})), f) > \epsilon\} \leq \delta.$$

It is known (e.g., [10]) that

$$\max\{\frac{d-1}{32\epsilon}, \frac{1}{2\epsilon}\ln\frac{1}{\delta}\} \leq M(\mathcal{C}, \epsilon, \delta) \leq \frac{4d}{\epsilon}\ln\frac{12}{\epsilon} + \frac{4}{\epsilon}\ln\frac{2}{\delta}$$

for $0 < \epsilon < 1/8$*,* $0 < \delta < .01$*, and* $d \geq 2$*, where* d *is the VC-dimension of* $\mathcal{C}$*. Furthermore, Warmuth has conjectured [11] that* $M(\mathcal{C}, \epsilon, \delta) = \Theta(\frac{1}{\epsilon}(d + \log\frac{1}{\delta}))$*.*

With these definitions in mind, we have the following novel theorem.

Theorem 3.1. *For any instance space* $\mathcal{X}$*, concept space* $\mathcal{C}$ *on* $\mathcal{X}$ *with VC-dimension* $d \in (0, \infty)$*, sample-based cost function* c*,* $\epsilon \in (0, 1)$*, and* $\delta \in (0, \frac{1}{2})$*, if* $m = M(\mathcal{C}, \epsilon, \delta)$*, then*

$$\mathit{CostComplexity}(\mathcal{C}, c, \epsilon, \delta) \leq GIC(\mathcal{C}, c, m)d\log_2\frac{em}{d}$$

Proof. For the unlabeled sequence, sample $\mathcal{U} \sim \mathcal{D}^m$. If $GIC(\mathcal{C}, c, m) = \infty$, then the upper bound holds vacuously, so let us assume this is finite. Also, $d \in (0, \infty)$ implies $|\mathcal{U}| \in (0, \infty)$ [10]. By definition of $M(\mathcal{C}, \epsilon, \delta)$, there exists a (passive learning) algorithm $\mathcal{A}$ such that $\forall f \in \mathcal{C}, \forall \mathcal{D}, \mathcal{P}r_{\mathcal{U}\sim\mathcal{D}^m}\{error_{\mathcal{D}}(\mathcal{A}(\mathcal{U}, f(\mathcal{U})), f) > \epsilon\} \leq \delta$. Therefore any algorithm that, by a finite sequence of effective queries with finite cost under $c_{\mathcal{U}}$, identifies $f(\mathcal{U})$ and then outputs $\mathcal{A}(\mathcal{U}, f(\mathcal{U}))$, is an (ϵ, δ)-learning algorithm for $\mathcal{C}$ using c.

Suppose now that there is a *ghost teacher*, who knows the teacher's target concept $f \in \mathcal{C}$. The ghost teacher uses the $h \in \mathcal{C}[\mathcal{U}]$ with $h(\mathcal{U}) = f(\mathcal{U})$ as its target concept. In order to answer any actual queries $\tilde{q} \in \tilde{Q}$ with $c_{\mathcal{U}}(\tilde{q}) < \infty$, the ghost teacher simply passes the query to the real teacher and then answers the query using the real teacher's answer. This answer is guaranteed to be valid because $c_{\mathcal{U}}$ is a sample-based cost function. Thus, identifying $f(\mathcal{U})$ can be accomplished by identifying $h(\mathcal{U})$, which can be accomplished by identifying h. The task of identifying h can be reduced to an *Exact Learning* task with concept space $\mathcal{C}[\mathcal{U}]$ and cost function $c_{\mathcal{U}}$, where the teacher for the Exact Learning task is the ghost teacher. Therefore, by Theorem 2.1, the total cost required to identify $f(\mathcal{U})$ with a finite sequence of queries is no greater than

$$\mathit{CostComplexity}(\mathcal{C}[\mathcal{U}], c_{\mathcal{U}}) \leq GIC(\mathcal{C}[\mathcal{U}], c_{\mathcal{U}})\log_2|\mathcal{C}[\mathcal{U}]| \leq GIC(\mathcal{C}[\mathcal{U}], c_{\mathcal{U}})d\log_2\frac{|\mathcal{U}|e}{d}, \quad (1)$$

where the last inequality is due to Sauer's Lemma (e.g., [10]). Finally, taking the worst case (supremum) over all $\mathcal{U} \in \mathcal{X}^m$ completes the proof. □

Note that (1) also implies a data-dependent bound, which could potentially be useful for practical applications in which the unlabeled examples are available when bounding the cost. It can also be used to state a distribution-dependent bound.

3.3 An Example: Intersection-Closed Concept Spaces

As an example application, we can use the above theorem to prove new results for any intersection-closed concept space[10] as follows.

Lemma 3.1. *For any instance space* $\mathcal{X}$*, intersection-closed concept space* $\mathcal{C}$ *with VC-dimension* $d \geq 1$*, sample-based cost function* c *such that membership queries in* $\mathcal{U}$ *have cost* $\leq \mu$ *(i.e.,* $\forall \mathcal{U} \subseteq \mathcal{X}, x \in \mathcal{U}, c_{\mathcal{U}}(q_{\{x\}}) \leq \mu$*) and positive example queries in* $\mathcal{U}$ *have cost* $\leq \kappa$ *(i.e.,* $\forall \mathcal{U} \subseteq \mathcal{X}, S \subseteq \mathcal{U}, c_{\mathcal{U}}(q_S) \leq \kappa$*), and integer* $m \geq 0$*,*

$$GIC(\mathcal{C}, c, m) \leq \kappa + \mu d$$

Proof. Say we have some set of unlabeled examples $\mathcal{U}$, and consider bounding the value of $GIC(\mathcal{C}[\mathcal{U}], c_{\mathcal{U}})$. In the spy game, suppose the teacher is answering with effective oracle $T \in \mathcal{T}$. Let $\mathcal{U}_+ = \{x | x \in \mathcal{U}, T(q_{\{x\}}) = \{h | h \in \mathcal{C}^*, h(x) = 1\}\}$. The spy first tells the learner to make the $q_{\mathcal{U}\setminus\mathcal{U}_+}$ query (if $\mathcal{U} \setminus \mathcal{U}_+ \neq \varnothing$). If $\exists x \in \mathcal{U} \setminus \mathcal{U}_+$ s.t. $T(q_{\mathcal{U}\setminus\mathcal{U}_+}) = \{h | h \in \mathcal{C}^*, h(x) = 1\}$, then the spy tells the learner to make effective query $q_{\{x\}}$ for this x, and there are no concepts in $\mathcal{C}[\mathcal{U}]$ consistent with the answers to these two queries; the total effective cost for this case is $\kappa + \mu$. If this is not the case, but $|\mathcal{U}_+| = 0$, then there is at most one concept in $\mathcal{C}[\mathcal{U}]$ consistent with the answer to $q_{\mathcal{U}\setminus\mathcal{U}_+}$: namely, the $h \in \mathcal{C}[\mathcal{U}]$ with $h(x) = 0$ for all $x \in \mathcal{U}$, if there is such an h. In this case, the cost is just κ.

---

[10] An intersection-closed concept space $\mathcal{C}$ has the property that for any $h_1, h_2 \in \mathcal{C}$, there is a concept $h_3 \in \mathcal{C}$ such that $\forall x \in \mathcal{X}, [h_1(x) = h_2(x) = 1 \Leftrightarrow h_3(x) = 1]$. For example, conjunctions and axis-aligned rectangles are intersection-closed.

Otherwise, let $\bar{S}$ be a largest subset of $\mathcal{U}_+$ such that $\exists h \in \mathcal{C}$ with $\forall x \in \bar{S}, h(x) = 1$. If $\bar{S} = \varnothing$, then making any membership query in $\mathcal{U}_+$ leaves all concepts in $\mathcal{C}[\mathcal{U}]$ inconsistent (at cost μ), so let us assume $\bar{S} \neq \varnothing$. For any $S \subseteq \mathcal{X}$, define

$$CLOS(S) = \{x | x \in \mathcal{X}, \forall h \in \mathcal{C}, [\forall y \in S, h(y) = 1] \Rightarrow h(x) = 1\}$$

the *closure* of S. Let $\bar{S}'$ be a smallest subset of $\bar{S}$ such that $CLOS(\bar{S}') = CLOS(\bar{S})$, known as a *minimal spanning set* of $\bar{S}$ [12]. The spy now tells the learner to make queries $q_{\{x\}}$ for all $x \in \bar{S}'$.

Any concept in $\mathcal{C}$ consistent with the answer to $q_{\mathcal{U}\setminus\mathcal{U}_+}$ must label every $x \in \mathcal{U} \setminus \mathcal{U}_+$ as 0. Any concept in $\mathcal{C}$ consistent with the answers to the membership queries on $\bar{S}'$ must label every $x \in CLOS(\bar{S}') = CLOS(\bar{S}) \supseteq \bar{S}$ as 1. Additionally, every concept in $\mathcal{C}$ that labels every $x \in \bar{S}$ as 1 must label every $x \in \mathcal{U}_+ \setminus \bar{S}$ as 0, since $\bar{S}$ is defined to be maximal. This labeling of these three sets completely defines a labeling of $\mathcal{U}$, and as such there is at most one $h \in \mathcal{C}[\mathcal{U}]$ consistent with the answers to all queries made by the learner. Helmbold, Sloan, and Warmuth [12] proved that, for an intersection-closed concept space with VC-dimension d, for any set $\bar{S}$, all minimal spanning sets of $\bar{S}$ have size at most d. This implies the learner makes at most d membership queries in $\mathcal{U}$, and thus has a total cost of at most $\kappa + \mu d$. □

Corollary 3.1. *Under the conditions of Lemma 3.1, if* $d \geq 10$, *then for* $0 < \epsilon < 1$, *and* $0 < \delta < \frac{1}{2}$,

$$\textit{CostComplexity}(\mathcal{C}, c, \epsilon, \delta) \leq (\kappa + \mu d) d \log_2 \left(\frac{e}{d} \max \left\{ \frac{16d}{\epsilon} \ln d, \frac{6}{\epsilon} \ln \frac{28}{\delta} \right\} \right)$$

Proof. This follows from Theorem 3.1, Lemma 3.1, and Auer & Ortner's result [13] that for intersection-closed concept spaces with $d \geq 10$, $M(\mathcal{C}, \epsilon, \delta) \leq \max\left\{\frac{16d}{\epsilon} \ln d, \frac{6}{\epsilon} \ln \frac{28}{\delta}\right\}$. □

For example, consider the concept space of axis-parallel hyper-rectangles in $\mathcal{X} = \mathbb{R}^n$, $\mathcal{C} = \{h : \mathcal{X} \to \{0,1\} | \exists ((a_1, b_1), (a_2, b_2), \dots, (a_n, b_n)) : \forall x \in \mathbb{R}^n, h(x) = 1 \Leftrightarrow \forall i \in \{1, 2, \dots, n\}, a_i \leq x_i \leq b_i\}$. One can show that this is an intersection-closed concept space with VC-dimension $2n$. For a sample-based cost function c of the form stated in Lemma 3.1, we have that $\textit{CostComplexity}(\mathcal{C}, c, \epsilon, \delta) \leq \tilde{O}\left((\kappa + n\mu)n\right)$. Unlike the example in the previous section, if all other query types have infinite cost, then for $n \geq 2$ there are distributions that force any algorithm achieving this bound for small ϵ and δ to use multiple positive example queries q_S with $|S| > 1$. In particular, for finite constant κ, this is an exponential improvement over the cost complexity of PAC active learning with only uniform cost membership queries on $\mathcal{U}$.

3.4 A Cost Complexity Lower Bound

At first glance, it might seem that $GIC(\mathcal{C}, c, \left\lceil\frac{1-\epsilon}{\epsilon}\right\rceil)$ could be a lower bound on $\textit{CostComplexity}(\mathcal{C}, c, \epsilon, \delta)$. In fact, one can show this is true for $\delta < (\frac{\epsilon d}{e})^d$. However, there are simple examples for which this is not a lower bound for general ϵ and δ.[11] We therefore require a slight modification of GIC to introduce dependence on δ.

Definition 3.8. *For an instance space* $\mathcal{X}$, *finite concept space* $\mathcal{C}$ *on* $\mathcal{X}$, *cost function c, and* $\delta \in [0, 1)$, *define the* General Partial Identification Cost, *denoted* $GPIC(\mathcal{C}, c, \delta)$ *as follows.*

$$GPIC(\mathcal{C}, c, \delta) = \inf\{t | t \geq 0, \forall T \in \mathcal{T}, \exists R \subseteq \mathcal{Q}, \text{ s.t. } [\textstyle\sum_{q \in R} c(q) \leq t] \wedge [|\mathcal{C} \cap T(R)| \leq \delta|\mathcal{C}| + 1]\}$$

Definition 3.9. *For an instance space* $\mathcal{X}$, *concept space* $\mathcal{C}$ *on* $\mathcal{X}$, *sample-based cost function c, non-negative integer m, and* $\delta \in [0, 1)$, *define the* General Partial Identification Cost Growth Function, *denoted* $GPIC(\mathcal{C}, c, m, \delta)$, *as follows.*

$$GPIC(\mathcal{C}, c, m, \delta) = \sup_{\mathcal{U} \in \mathcal{X}^m} GPIC(\mathcal{C}[\mathcal{U}], c_{\mathcal{U}}, \delta)$$

---

[11] The infamous "Monty Hall" problem is an interesting example of this. For another example, consider $\mathcal{X} = \{1, 2, \dots, N\}$, $\mathcal{C} = \{h_x | x \in \mathcal{X}, \forall y \in \mathcal{X}, h_x(y) = I[x = y]\}$, and cost that is 1 for membership queries in $\mathcal{U}$ and infinite for other queries. Although $GIC(\mathcal{C}, c, N) = N - 1$, it is possible to achieve better than $\epsilon = \frac{1}{N+1}$ with probability close to $\frac{N-2}{N-1}$ using cost no greater than $N - 2$.

It is easy to see that $GIC(\mathcal{C}, c) = GPIC(\mathcal{C}, c, 0)$ and $GIC(\mathcal{C}, c, m) = GPIC(\mathcal{C}, c, m, 0)$, so that all of the above results could be stated in terms of $GPIC$.

Theorem 3.2. *For any instance space $\mathcal{X}$, concept space $\mathcal{C}$ on $\mathcal{X}$, sample-based cost function c, $(\epsilon, \delta) \in (0,1)^2$, and any $V \subseteq \mathcal{C}$,*

$$GPIC(V, c, \left\lceil \tfrac{1-\epsilon}{\epsilon} \right\rceil, \delta) \leq \mathit{CostComplexity}(\mathcal{C}, c, \epsilon, \delta)$$

Proof. Let $S \subseteq \mathcal{X}$ be a set with $1 \leq |S| \leq \left\lceil \frac{1-\epsilon}{\epsilon} \right\rceil$, and let $\mathcal{D}_S$ be the uniform distribution on S. Thus, $error_{\mathcal{D}_S}(h, f) \leq \epsilon \Leftrightarrow h(S) = f(S)$. I will show that any algorithm $\mathcal{A}$ guaranteeing $\mathcal{P}r_{\mathcal{U} \sim \mathcal{D}_S^m}\{error_{\mathcal{D}_S}(\mathcal{A}(\mathcal{U}), f) > \epsilon\} \leq \delta$ cannot also guarantee cost strictly less than $GPIC(V[S], c_S, \delta)$. If $\delta|V[S]| \geq |V[S]| - 1$, the result is clear since no algorithm guarantees cost less than 0, so assume $\delta|V[S]| < |V[S]| - 1$. Suppose $\mathcal{A}$ is an algorithm that guarantees, for every finite sequence $\mathcal{U}$ of elements from S, $\mathcal{A}(\mathcal{U})$ incurs total cost strictly less than $GPIC(V[S], c_S, \delta)$ under $c_{\mathcal{U}}$ (and therefore also under c_S). By definition of $GPIC$, $\exists \hat{T} \in \mathcal{T}$ such that for any set of queries R that $\mathcal{A}(\mathcal{U})$ makes, $|V[S] \cap \hat{T}(R)| > \delta|V[S]| + 1$. I now proceed by the probabilistic method. Say the teacher draws the target concept f uniformly at random from $V[S]$, and $\forall q \in \mathcal{Q}$ s.t. $f \in \hat{T}(q)$, answers with $\hat{T}(q)$. Any $q \in \mathcal{Q}$ such that $f \notin \hat{T}(q)$ can be answered with an arbitrary $a \in q(f)$. Let $h_{\mathcal{U}} = \mathcal{A}(\mathcal{U})$; let $R_{\mathcal{U}}$ denote the set of queries $\mathcal{A}(\mathcal{U})$ would make if *all* queries were answered with $\hat{T}$.

$$\begin{aligned}
&\mathbb{E}_f[\mathcal{P}r_{\mathcal{U} \sim \mathcal{D}_S^m}\{error_{\mathcal{D}_S}(\mathcal{A}(\mathcal{U}), f) > \epsilon\}] \\
&\qquad = \mathbb{E}_{\mathcal{U} \sim \mathcal{D}_S^m}[\mathcal{P}r_f\{h_{\mathcal{U}}(S) \neq f(S)\}] \\
&\qquad \geq \mathbb{E}_{\mathcal{U} \sim \mathcal{D}_S^m}[\mathcal{P}r_f\{h_{\mathcal{U}}(S) \neq f(S) \wedge f \in \hat{T}(R_{\mathcal{U}})\}] \\
&\qquad \geq \min_{\mathcal{U} \in S^m} \frac{|V[S] \cap \hat{T}(R_{\mathcal{U}})| - 1}{|V[S]|} > \delta.
\end{aligned}$$

Therefore, there exists a deterministic method for selecting f and answering queries such that $\mathcal{P}r_{\mathcal{U} \sim \mathcal{D}_S^m}\{error_{\mathcal{D}_S}(\mathcal{A}(\mathcal{U}), f) > \epsilon\} > \delta$. In particular, this proves that there are no (ϵ, δ)-learning algorithms that guarantee cost strictly less than $GPIC(V[S], c_S, \delta)$. Taking the supremum over sets S completes the proof. □

Corollary 3.2. *Under the conditions of Theorem 3.2,*

$$GPIC(\mathcal{C}, c, \left\lceil \tfrac{1-\epsilon}{\epsilon} \right\rceil, \delta) \leq \mathit{CostComplexity}(\mathcal{C}, c, \epsilon, \delta).$$

Equipped with Theorem 3.2, it is straightforward to prove the claim made in Section 3.3 that there are distributions forcing any (ϵ, δ)-learning algorithm for Axis-parallel rectangles using only membership queries (at cost μ) to pay $\Omega(\frac{\mu(1-\delta)}{\epsilon})$. The details are left as an exercise.

4 Discussion and Open Problems

Note that the usual "query counting" analysis done for Active Learning is a special case of cost complexity (uniform cost 1 on the allowed queries, infinite cost on the others). In particular, Theorem 3.1 can easily be specialized to give a worst-case bound on the query complexity for the widely studied setting in which the learner can make any *membership queries* on examples in $\mathcal{U}$ [9, 14]. However, for this special case, one can derive a slightly tighter bound. Following the proof technique of Hegedüs [4], one can show that for any sample-based cost function c such that $\forall \mathcal{U} \subseteq \mathcal{X}, q \in \mathcal{Q}, c_{\mathcal{U}}(q) < \infty \Rightarrow [c_{\mathcal{U}}(q) = 1 \wedge \forall f \in \mathcal{C}^*, |q(f)| = 1]$, $\mathit{CostComplexity}(\mathcal{C}, c_{\mathcal{X}}) \leq 2\frac{GIC(\mathcal{C}, c_{\mathcal{X}}) \log_2 |\mathcal{C}|}{\log_2 GIC(\mathcal{C}, c_{\mathcal{X}})}$. This implies for the PAC setting that $\mathit{CostComplexity}(\mathcal{C}, c, \epsilon, \delta) \leq 2\frac{GIC(\mathcal{C}, c, m) d \log_2 m}{\log_2 GIC(\mathcal{C}, c, m)}$, for VC-dimension $d \geq 3$ and $m = M(\mathcal{C}, \epsilon, \delta)$. This includes the cost function assigning 1 to membership queries on $\mathcal{U}$ and ∞ to all others.

Active Learning in the PAC model is closely related to the topic of *Semi-Supervised Learning*. Balcan & Blum [15] have recently derived a variety of sample complexity bounds for Semi-Supervised Learning. Many of the techniques can be transfered to the pool-based Active Learning setting in a fairly natural way. Specifically, suppose there is a quantitative notion of "compatibility" between a concept and a distribution, which can be estimated from a finite unlabeled sample. If we know the target concept is highly compatible with the data distribution, we can draw enough unlabeled examples to estimate compatibility, then identify and discard those concepts that are probably highly incompatible. The set of highly compatible concepts may be significantly less expressive, therefore reducing *both* the number of examples for which an algorithm must learn the labels to guarantee generalization *and* the number of labelings of those examples the algorithm must distinguish between, thereby also reducing the cost complexity.

There are a variety of interesting extensions of this framework worth pursuing. Perhaps the most natural direction is to move into the agnostic PAC framework, which has thus far been quite elusive for active learning except for a few results [16, 17]. Another possibility is to derive cost complexity bounds when the cost c is a function of not only the query, but also the target concept. Then every time the learning algorithm makes a query q, it is charged $c(q, f)$, but does not necessarily know what this value is. However, it can always upper bound the total cost so far by the worst case over concepts in the version space. Can anything interesting be said about this setting (or variants), perhaps under some benign smoothness constraints on $c(q, \cdot)$? This is of some practical importance since, for example, it is often more difficult to label examples that occur near a decision boundary.

References

[1] Balcázar, J.L., Castro, J., Guijarro, D.: A general dimension for exact learning. In: 14th Annual Conference on Learning Theory. (2001)

[2] Balcázar, J.L., Castro, J.: A new abstract combinatorial dimension for exact learning via queries. Journal of Computer and System Sciences **64** (2002) 2–21

[3] Hellerstein, L., Pillaipakkamnatt, K., Raghavan, V., Wilkins, D.: How many queries are needed to learn? Journal of the Association for Computing Machinery **43** (1996) 840–862

[4] Hegedüs, T.: Generalized teaching dimension and the query complexity of learning. In: 8th Annual Conference on Computational Learning Theory. (1995)

[5] Balcázar, J.L., Castro, J., Guijarro, D., Simon, H.U.: The consistency dimension and distribution-dependent learning from queries. In: Algorithmic Learning Theory. (1999)

[6] Blumer, A., Ehrenfeucht, A., Haussler, D., Warmuth, M.: Learnability and the vapnik-chervonenkis dimension. Journal of the Association for Computing Machinery **36** (1989) 929–965

[7] Dasgupta, S.: Analysis of a greedy active learning strategy. In: Advances in Neural Information Processing Systems (NIPS). (2004)

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

[9] Dasgupta, S.: Coarse sample complexity bounds for active learning. In: Advances in Neural Information Processing Systems (NIPS). (2005)

[10] Anthony, M., Bartlett, P.L.: Neural Network Learning: Theoretical Foundations. Cambridge University Press (1999)

[11] Warmuth, M.: The optimal pac algorithm. In: Conference on Learning Theory. (2004)

[12] Helmbold, D., Sloan, R., Warmuth, M.: Learning nested differences of intersection-closed concept classes. Machine Learning **5** (1990) 165–196

[13] Auer, P., Ortner, R.: A new PAC bound for intersection-closed concept classes. In: 17^{th} Annual Conference on Learning Theory (COLT). (2004)

[14] Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. Journal of Machine Learning Research **2** (2001)

[15] Balcan, M.F., Blum, A.: A PAC-style model for learning from labeled and unlabeled data. In: Conference on Learning Theory. (2005)

[16] Balcan, M.F., Beygelzimer, A., Langford, J.: Agnostic active learning. In: 23rd International Conference on Machine Learning (ICML). (2006)

[17] Kääriäinen, M.: On active learning in the non-realizable case. In: NIPS Workshop on Foundations of Active Learning. (2005)

Some Open Problems on the Complexity of Interactive Machine Learning

Steve Hanneke

Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213 USA
`shanneke@cs.cmu.edu`

May 2007

1 The Cost Complexity of Interactive Learning

In the chapter on *Teaching Dimension and the Complexity of Active Learning*, the distribution-free definition of XTD is basically a special case of the distribution-free definition of GIC. We were able to show that, in the realizable case the label complexity is at most

$$XTD\left(\mathbb{C}, \tilde{O}\left(\frac{1}{\epsilon}\right)\right)\tilde{O}(d),$$

and when the noise rate is ν, the label complexity is at most

$$XTD\left(\mathbb{C}, O\left(\frac{1}{\epsilon+\nu}\right)\right)\tilde{O}\left(d\left(\frac{\epsilon+\nu}{\epsilon}\right)^2\right).$$

However, for the general cost complexity, we were only able to show the cost complexity of realizable learning is at most

$$GIC\left(\mathbb{C}, c, \tilde{O}\left(\frac{d}{\epsilon}\right)\right)\tilde{O}(d).$$

We therefore have the following open problems.

(1) Is it true that the cost complexity of realizable learning is at most

$$GIC\left(\mathbb{C}, c, \tilde{O}\left(\frac{1}{\epsilon}\right)\right)\tilde{O}(d)?$$

(2) Is it true that the cost complexity of learning with noise rate ν is at most

$$GIC\left(\mathbb{C}, c, O\left(\frac{1}{\epsilon+\nu}\right)\right)\tilde{O}\left(d\left(\frac{\epsilon+\nu}{\epsilon}\right)^2\right)?$$

(3) Do these bounds still hold for the distribution-dependent GIC generalization of the distribution-dependent XTD definition?

There are some sample-based cost functions for which these results are easily verified, since we can simulate a Halving-like algorithm as we did with label queries in the teaching dimension analysis. However, for other sample-based cost functions it is less obvious. For example, one type of query involves asking whether example x_1 and example x_2 have the same label. It is possible that they have the same label for almost every concept in $\mathbb{C}$, yet h_{maj} labels them differently. Thus, even though the teacher's answer may contradict h_{maj}, it does not necessarily mean it contradicts a constant fraction of the concepts in $\mathbb{C}$.

2 Nonparametric Active Learning

It seems the formulation of active learning with label queries from the previous chapters isn't quite the proper way to think about the learning problem. Indeed, what we care about is not finding a concept in $\mathbb{C}$ with $er(h) \leq \epsilon + \nu$, but rather finding a concept in $\mathbb{C}_{\mathcal{F}}$ with $er(h) \leq \epsilon + \beta$, where β is the *Bayes optimal error rate* (the error rate of the best classifier in the set of *all* measurable classifiers $\mathbb{C}_{\mathcal{F}}$, not just a specific concept space $\mathbb{C}$). So in this formulation, we can basically eliminate the notion of *concept space* entirely, and just focus on search through the space of *all* classifiers. This is sometimes refered to as *nonparametric* learning.

So now the question becomes, "how many label requests are necessary and sufficient to guarantee with high probability we will find an h with $er(h) \leq \epsilon + \beta$?" The answer will, of course, depend on what kind of classifier is the Bayes optimal, how big the Bayes optimal error rate is (and how benign is the noise), and possibly other factors.

One general quality we might want from an algorithm solving this problem is that, the more queries it makes, the smaller our guarantee on the error rate of the proposed classifier becomes (like an "anytime" algorithm)[1]. In this view, it makes sense to talk about the *rate* of convergence of the guaranteed error rate toward the Bayes optimal error rate, as the number of label requests increases.

At this point, the open problem is simply to give *any* active learning algorithm for this nonparametric setting, such that there are no distributions (on $\mathcal{X} \times \{-1, 1\}$) where the rates of convergence are much worse than for passive learning, and such that there exist interesting families of distributions where we achieve faster rates of convergence toward the Bayes optimal error rate compared with passive learning. No algorithms of this type have yet been proposed by anyone. Once we have such an algorithm, the task becomes figuring out a general bound on the number of label requests it makes, whether there is a notion of optimality of an algorithm for this task, whether there are lower bounds on label complexity, what the trade-offs are between number of unlabeled examples and number of label requests, when the algorithm can be made efficient, etc.

---

[1] In this view, we're implicitly assuming we have an endless stream of unlabeled data, so the unlabeled data is not a fundamentally limiting factor on the error rate (though we may also want an algorithm that has a good convergence rate in terms of number of unlabeled examples it looks at too).

3 Disagreement Coefficient

In the chapter on agnostic active learning, we proved a bound on the label complexity of

$$\tilde{O}\left(\theta^2 d \left(\frac{\epsilon+\nu}{\epsilon}\right)^2\right),$$

where θ is the disagreement coefficient.

For an (unpublished) slight modification of the A^2 algorithm, which focuses more directly on pairwise comparisons, we can obtain a bound of the form[2]

$$\tilde{O}\left(\theta d^2 \left(\frac{\epsilon+\nu}{\epsilon}\right)^2\right).$$

This bound is often worse than the previous, but for some problems it is better (e.g., the p-intervals example).

It remains an enticing open problem to determine whether we can obtain a bound of the form

$$\tilde{O}\left(\theta d \left(\frac{\epsilon+\nu}{\epsilon}\right)^2\right).$$

Proving this would improve the best known bounds on the agnostic learning label complexity for several concept spaces and distributions. For example, in the example of linear separators under uniform distribution on a unit sphere, this bound would improve the current best known bound by a factor of $\sqrt{d}$.

---

[2] Technically, the current proof for this bound uses a slightly different definition of θ, given by $\inf\{\theta > 0 | \forall r > \epsilon, \Delta_r \leq \theta r\}$. In my experience, this modification typically only makes a difference when both definitions of θ are large enough to make the bounds vacuous anyway.