Agnostic Active Learning Without Constraints

Alina Beygelzimer
IBM Research
Hawthorne, NY
beygel@us.ibm.com

Daniel Hsu
Rutgers University &
University of Pennsylvania
djhsu@rci.rutgers.edu

John Langford
Yahoo! Research
New York, NY
jl@yahoo-inc.com

Tong Zhang
Rutgers University
Piscataway, NJ
tongz@rci.rutgers.edu

Abstract

We present and analyze an agnostic active learning algorithm that works without keeping a version space. This is unlike all previous approaches where a restricted set of candidate hypotheses is maintained throughout learning, and only hypotheses from this set are ever returned. By avoiding this version space approach, our algorithm sheds the computational burden and brittleness associated with maintaining version spaces, yet still allows for substantial improvements over supervised learning for classification.

1 Introduction

In active learning, a learner is given access to unlabeled data and is allowed to adaptively choose which ones to label. This learning model is motivated by applications in which the cost of labeling data is high relative to that of collecting the unlabeled data itself. Therefore, the hope is that the active learner only needs to query the labels of a small number of the unlabeled data, and otherwise perform as well as a fully supervised learner. In this work, we are interested in agnostic active learning algorithms for binary classification that are provably consistent, *i.e.* that converge to an optimal hypothesis in a given hypothesis class.

One technique that has proved theoretically profitable is to maintain a candidate set of hypotheses (sometimes called a version space), and to query the label of a point only if there is disagreement within this set about how to label the point. The criteria for membership in this candidate set needs to be carefully defined so that an optimal hypothesis is always included, but otherwise this set can be quickly whittled down as more labels are queried. This technique is perhaps most readily understood in the noise-free setting [1, 2], and it can be extended to noisy settings by using empirical confidence bounds [3, 4, 5, 6, 7].

The version space approach unfortunately has its share of significant drawbacks. The first is computational intractability: maintaining a version space and guaranteeing that *only* hypotheses from this set are returned is difficult for linear predictors and appears intractable for interesting nonlinear predictors such as neural nets and decision trees [1]. Another drawback of the approach is its brittleness: a single mishap (due to, say, modeling failures or computational approximations) might cause the learner to exclude the best hypothesis from the version space forever; this is an ungraceful failure mode that is not easy to correct. A third drawback is related to sample re-usability: if (labeled) data is collected using a version space-based active learning algorithm, and we later decide to use a different algorithm or hypothesis class, then the earlier data may not be freely re-used because its collection process is inherently biased.

Here, we develop a new strategy addressing all of the above problems given an oracle that returns an empirical risk minimizing (ERM) hypothesis. As this oracle matches our abstraction of many supervised learning algorithms, we believe active learning algorithms built in this way are immediately and widely applicable.

Our approach instantiates the importance weighted active learning framework of [5] using a rejection threshold similar to the algorithm of [4] which only accesses hypotheses via a supervised learning oracle. However, the oracle we require is simpler and avoids strict adherence to a candidate set of hypotheses. Moreover, our algorithm creates an importance weighted sample that allows for unbiased risk estimation, even for hypotheses from a class different from the one employed by the active learner. This is in sharp contrast to many previous algorithms (*e.g.*, [1, 3, 8, 4, 6, 7]) that create heavily biased data sets. We prove that our algorithm is always consistent and has an improved label complexity over passive learning in cases previously studied in the literature. We also describe a practical instantiation of our algorithm and report on some experimental results.

1.1 Related Work

As already mentioned, our work is closely related to the previous works of [4] and [5], both of which in turn draw heavily on the work of [1] and [3]. The algorithm from [4] extends the selective sampling method of [1] to the agnostic setting using generalization bounds in a manner similar to that first suggested in [3]. It accesses hypotheses only through a special ERM oracle that can enforce an arbitrary number of example-based constraints; these constraints define a version space, and the algorithm only ever returns hypotheses from this space, which can be undesirable as we previously argued. Other previous algorithms with comparable performance guarantees also require similar example-based constraints (*e.g.*, [3, 5, 6, 7]). Our algorithm differs from these in that (i) it never restricts its attention to a version space when selecting a hypothesis to return, and (ii) it only requires an ERM oracle that enforces at most one example-based constraint, and this constraint is only used for selective sampling. Our label complexity bounds are comparable to those proved in [5] (though somewhat worse that those in [3, 4, 6, 7]).

The use of importance weights to correct for sampling bias is a standard technique for many machine learning problems (*e.g.*, [9, 10, 11]) including active learning [12, 13, 5]. Our algorithm is based on the importance weighted active learning (IWAL) framework introduced by [5]. In that work, a rejection threshold procedure called *loss-weighting* is rigorously analyzed and shown to yield improved label complexity bounds in certain cases. Loss-weighting is more general than our technique in that it extends beyond zero-one loss to a certain subclass of loss functions such as logistic loss. On the other hand, the loss-weighting rejection threshold requires optimizing over a restricted version space, which is computationally undesirable. Moreover, the label complexity bound given in [5] only applies to hypotheses selected from this version space, and not when selected from the entire hypothesis class (as the general IWAL framework suggests). We avoid these deficiencies using a new rejection threshold procedure and a more subtle martingale analysis.

Many of the previously mentioned algorithms are analyzed in the agnostic learning model, where no assumption is made about the noise distribution (see also [14]). In this setting, the label complexity of active learning algorithms cannot generally improve over supervised learners by more than a constant factor [15, 5]. However, under a parameterization of the noise distribution related to Tsybakov's low-noise condition [16], active learning algorithms have been shown to have improved label complexity bounds over what is achievable in the purely agnostic setting [17, 8, 18, 6, 7]. We also consider this parameterization to obtain a tighter label complexity analysis.

2 Preliminaries

2.1 Learning Model

Let $\mathcal{D}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$ where $\mathcal{X}$ is the input space and $\mathcal{Y} = \{\pm 1\}$ are the labels. Let $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ be a pair of random variables with joint distribution $\mathcal{D}$. An active learner receives a sequence $(X_1, Y_1), (X_2, Y_2), \ldots$ of i.i.d. copies of (X, Y), with the label Y_i hidden unless it is explicitly queried. We use the shorthand $a_{1:k}$ to denote a sequence $(a_1, a_2, \ldots, a_k)$ (so $k = 0$ correspond to the empty sequence).

Let $\mathcal{H}$ be a set of hypotheses mapping from $\mathcal{X}$ to $\mathcal{Y}$. For simplicity, we assume $\mathcal{H}$ is finite but does not completely agree on any single $x \in \mathcal{X}$ (*i.e.*, $\forall x \in \mathcal{X}, \exists h, h' \in \mathcal{H}$ such that $h(x) \neq h'(x)$). This keeps the focus on the relevant aspects of active learning that differ from passive learning. The error of a hypothesis $h : \mathcal{X} \to \mathcal{Y}$ is $\mathrm{err}(h) := \Pr(h(X) \neq Y)$. Let $h^* := \arg\min\{\mathrm{err}(h) : h \in \mathcal{H}\}$ be a hypothesis of minimum error in $\mathcal{H}$. The goal of the active learner is to return a hypothesis $h \in \mathcal{H}$ with error $\mathrm{err}(h)$ not much more than $\mathrm{err}(h^*)$, using as few label queries as possible.

2.2 Importance Weighted Active Learning

In the importance weighted active learning (IWAL) framework of [5], an active learner looks at the unlabeled data $X_1, X_2, \ldots$ one at a time. After each new point X_i, the learner determines a probability $P_i \in [0, 1]$. Then a coin with bias P_i is flipped, and the label Y_i is queried if and only if the coin comes up heads. The query probability P_i can depend on all previous unlabeled examples $X_{1:i-1}$, any previously queried labels, any past coin flips, and the current unlabeled point X_i.

Formally, an IWAL algorithm specifies a *rejection threshold* function $p : (\mathcal{X} \times \mathcal{Y} \times \{0, 1\})^* \times \mathcal{X} \to [0, 1]$ for determining these query probabilities. Let $Q_i \in \{0, 1\}$ be a random variable conditionally independent of the current label Y_i,

$$Q_i \perp\!\!\!\perp Y_i \mid X_{1:i}, Y_{1:i-1}, Q_{1:i-1}$$

and with conditional expectation

$$\mathbb{E}[Q_i | Z_{1:i-1}, X_i] \;=\; P_i \;:=\; p(Z_{1:i-1}, X_i).$$

where $Z_j := (X_j, Y_j, Q_j)$. That is, Q_i indicates if the label Y_i is queried (the outcome of the coin toss). Although the notation does not explicitly suggest this, the query probability $P_i = p(Z_{1:i-1}, X_i)$ is allowed to explicitly depend on a label Y_j ($j < i$) if and only if it has been queried ($Q_j = 1$).

2.3 Importance Weighted Estimators

We first review some standard facts about the importance weighting technique. For a function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, define the *importance weighted estimator* of $\mathbb{E}[f(X, Y)]$ from $Z_{1:n} \in (\mathcal{X} \times \mathcal{Y} \times \{0, 1\})^n$ to be

$$\widehat{f}(Z_{1:n}) \;:=\; \frac{1}{n} \sum_{i=1}^{n} \frac{Q_i}{P_i} \cdot f(X_i, Y_i).$$

Note that this quantity depends on a label Y_i only if it has been queried (*i.e.*, only if $Q_i = 1$; it also depends on X_i only if $Q_i = 1$). Our rejection threshold will be based on a specialization of this estimator, specifically the *importance weighted empirical error* of a hypothesis h

$$\mathrm{err}(h, Z_{1:n}) \;:=\; \frac{1}{n} \sum_{i=1}^{n} \frac{Q_i}{P_i} \cdot \mathbb{1}[h(X_i) \neq Y_i].$$

In the notation of Algorithm 1, this is equivalent to

$$\mathrm{err}(h, S_n) \;:=\; \frac{1}{n} \sum_{(X_i, Y_i, 1/P_i) \in S_n} (1/P_i) \cdot \mathbb{1}[h(X_i) \neq Y_i] \tag{1}$$

where $S_n \subseteq \mathcal{X} \times \mathcal{Y} \times \mathbb{R}$ is the importance weighted sample collected by the algorithm.

A basic property of these estimators is *unbiasedness*: $\mathbb{E}[\widehat{f}(Z_{1:n})] = (1/n) \sum_{i=1}^{n} \mathbb{E}[\mathbb{E}[(Q_i/P_i) \cdot f(X_i, Y_i) \mid X_{1:i}, Y_{1:i}, Q_{1:i-1}]] = (1/n) \sum_{i=1}^{n} \mathbb{E}[(P_i/P_i) \cdot f(X_i, Y_i)] = \mathbb{E}[f(X, Y)]$. So, for example, the importance weighted empirical error of a hypothesis h is an unbiased estimator of its true error $\mathrm{err}(h)$. This holds for *any* choice of the rejection threshold that guarantees $P_i > 0$.

3 A Deviation Bound for Importance Weighted Estimators

As mentioned before, the rejection threshold used by our algorithm is based on importance weighted error estimates $\mathrm{err}(h, Z_{1:n})$. Even though these estimates are unbiased, they are only reliable when

the variance is not too large. To get a handle on this, we need a deviation bound for importance weighted estimators. This is complicated by two factors that rules out straightforward applications of some standard bounds:

1. The importance weighted samples $(X_i, Y_i, 1/P_i)$ (or equivalently, the $Z_i = (X_i, Y_i, Q_i)$) are not i.i.d. This is because the query probability P_i (and thus the importance weight $1/P_i$) generally depends on $Z_{1:i-1}$ and X_i.
2. The effective range and variance of each term in the estimator are, themselves, random variables.

To address these issues, we develop a deviation bound using a martingale technique from [19].

Let $f : \mathcal{X} \times \mathcal{Y} \to [-1, 1]$ be a bounded function. Consider any rejection threshold function $p : (\mathcal{X} \times \mathcal{Y} \times \{0, 1\})^* \times \mathcal{X} \to (0, 1]$ for which $P_n = p(Z_{1:n-1}, X_n)$ is bounded below by some positive quantity (which may depend on n). Equivalently, the query probabilities P_n should have inverses $1/P_n$ bounded above by some deterministic quantity r_{max} (which, again, may depend on n). The *a priori* upper bound r_{max} on $1/P_n$ can be pessimistic, as the dependence on r_{max} in the final deviation bound will be very mild—it enters in as $\log \log r_{max}$. Our goal is to prove a bound on $|\widehat{f}(Z_{1:n}) - \mathbb{E}[f(X, Y)]|$ that holds with high probability over the joint distribution of $Z_{1:n}$.

To start, we establish bounds on the range and variance of each term $W_i := (Q_i/P_i) \cdot f(X_i, Y_i)$ in the estimator, conditioned on $(X_{1:i}, Y_{1:i}, Q_{1:i-1})$. Let $\mathbb{E}_i[\,\cdot\,]$ denote $\mathbb{E}[\,\cdot\,|X_{1:i}, Y_{1:i}, Q_{1:i-1}]$. Note that $\mathbb{E}_i[W_i] = (\mathbb{E}_i[Q_i]/P_i) \cdot f(X_i, Y_i) = f(X_i, Y_i)$, so if $\mathbb{E}_i[W_i] = 0$, then $W_i = 0$. Therefore, the (conditional) range and variance are non-zero only if $\mathbb{E}_i[W_i] \neq 0$. For the range, we have $|W_i| = (Q_i/P_i) \cdot |f(X_i, Y_i)| \leq 1/P_i$, and for the variance, $\mathbb{E}_i[(W_i - \mathbb{E}_i[W_i])^2] \leq (\mathbb{E}_i[Q_i^2]/P_i^2) \cdot f(X_i, Y_i)^2 \leq 1/P_i$. These range and variance bounds indicate the form of the deviations we can expect, similar to that of other classical deviation bounds.

Theorem 1. *Pick any $t \geq 0$ and $n \geq 1$. Assume $1 \leq 1/P_i \leq r_{max}$ for all $1 \leq i \leq n$, and let $R_n := 1/\min(\{P_i : 1 \leq i \leq n \ \wedge \ f(X_i, Y_i) \neq 0\} \cup \{1\})$. With probability at least $1 - 2(3 + \log_2 r_{max})e^{-t/2}$,*

$$\left| \frac{1}{n} \sum_{i=1}^{n} \frac{Q_i}{P_i} \cdot f(X_i, Y_i) - \mathbb{E}[f(X, Y)] \right| \ \leq \ \sqrt{\frac{2R_n t}{n}} + \sqrt{\frac{2t}{n}} + \frac{R_n t}{3n}.$$

We defer all proofs to the appendices.

4 Algorithm

First, we state a deviation bound for the importance weighted error of hypotheses in a finite hypothesis class $\mathcal{H}$ that holds for all $n \geq 1$. It is a simple consequence of Theorem 1 and union bounds; the form of the bound motivates certain algorithmic choices to be described below.

Lemma 1. *Pick any $\delta \in (0, 1)$. For all $n \geq 1$, let*

$$\varepsilon_n := \frac{16 \log(2(3 + n \log_2 n) n(n+1)|\mathcal{H}|/\delta)}{n} = O\left(\frac{\log(n|\mathcal{H}|/\delta)}{n}\right). \tag{3}$$

Let $(Z_1, Z_2, \ldots) \in (\mathcal{X} \times \mathcal{Y} \times \{0, 1\})^*$ be the sequence of random variables specified in Section 2.2 using a rejection threshold $p : (\mathcal{X} \times \mathcal{Y} \times \{0, 1\})^* \times \mathcal{X} \to [0, 1]$ that satisfies $p(z_{1:n}, x) \geq 1/n^n$ for all $(z_{1:n}, x) \in (\mathcal{X} \times \mathcal{Y} \times \{0, 1\})^n \times \mathcal{X}$ and all $n \geq 1$.

The following holds with probability at least $1 - \delta$. For all $n \geq 1$ and all $h \in \mathcal{H}$,

$$|(\mathrm{err}(h, Z_{1:n}) - \mathrm{err}(h^*, Z_{1:n})) - (\mathrm{err}(h) - \mathrm{err}(h^*))| \leq \sqrt{\frac{\varepsilon_n}{P_{min,n}(h)}} + \frac{\varepsilon_n}{P_{min,n}(h)} \tag{4}$$

where $P_{min,n}(h) = \min\{P_i : 1 \leq i \leq n \ \wedge \ h(X_i) \neq h^*(X_i)\} \cup \{1\}$.

We let $C_0 = O(\log(|\mathcal{H}|/\delta)) \geq 2$ be a quantity such that ε_n (as defined in Eq. (3)) is bounded as $\varepsilon_n \leq C_0 \cdot \log(n+1)/n$. The following absolute constants are used in the description of the rejection

Algorithm 1
Notes: see Eq. (1) for the definition of err (importance weighted error), and Section 4 for the definitions of C_0, c_1, and c_2.
Initialize: $S_0 := \emptyset$.
For $k = 1, 2, \ldots, n$:

1. Obtain unlabeled data point X_k.
2. Let

   $h_k := \arg\min\{\mathrm{err}(h, S_{k-1}) : h \in \mathcal{H}\}$, and
   $h'_k := \arg\min\{\mathrm{err}(h, S_{k-1}) : h \in \mathcal{H} \wedge h(X_k) \neq h_k(X_k)\}$.

   Let $G_k := \mathrm{err}(h'_k, S_{k-1}) - \mathrm{err}(h_k, S_{k-1})$, and

   $$P_k := \begin{cases} 1 & \text{if } G_k \leq \sqrt{\frac{C_0 \log k}{k-1}} + \frac{C_0 \log k}{k-1} \\ s & \text{otherwise} \end{cases} \quad \left(= \min\left\{1,\ O\left(\frac{1}{G_k^2} + \frac{1}{G_k}\right) \cdot \frac{C_0 \log k}{k-1}\right\}\right)$$

   where $s \in (0, 1)$ is the positive solution to the equation

   $$G_k = \left(\frac{c_1}{\sqrt{s}} - c_1 + 1\right) \cdot \sqrt{\frac{C_0 \log k}{k-1}} + \left(\frac{c_2}{s} - c_2 + 1\right) \cdot \frac{C_0 \log k}{k-1}. \qquad (2)$$

3. Toss a biased coin with $\Pr(\text{heads}) = P_k$.

   If heads, then query Y_k, and let $S_k := S_{k-1} \cup \{(X_k, Y_k, 1/P_k)\}$.
   Else, let $S_k := S_{k-1}$.

Return: $h_{n+1} := \arg\min\{\mathrm{err}(h, S_n) : h \in \mathcal{H}\}$.

Figure 1: Algorithm for importance weighted active learning with an error minimization oracle.

threshold and the subsequent analysis: $c_1 := 5 + 2\sqrt{2}$, $c_2 := 5$, $c_3 := ((c_1 + \sqrt{2})/(c_1 - 2))^2$, $c_4 := (c_1 + \sqrt{c_3})^2$, $c_5 := c_2 + c_3$.

Our proposed algorithm is shown in Figure 1. The rejection threshold (Step 2) is based on the deviation bound from Lemma 1. First, the importance weighted error minimizing hypothesis h_k and the "alternative" hypothesis h'_k are found. Note that both optimizations are over the entire hypothesis class $\mathcal{H}$ (with h'_k only being required to disagree with h_k on x_k)—this is a key aspect where our algorithm differs from previous approaches. The difference in importance weighted errors G_k of the two hypotheses is then computed. If $G_k \leq \sqrt{(C_0 \log k)/(k-1)} + (C_0 \log k)/(k-1)$, then the query probability P_k is set to 1. Otherwise, P_k is set to the positive solution s to the quadratic equation in Eq. (2). The functional form of P_k is roughly $\min\{1,\ (1/G_k^2 + 1/G_k) \cdot (C_0 \log k)/(k-1)\}$. It can be checked that $P_k \in (0, 1]$ and that P_k is non-increasing with G_k. It is also useful to note that $(\log k)/(k-1)$ is monotonically decreasing with $k \geq 1$ (we use the convention $\log(1)/0 = \infty$).

In order to apply Lemma 1 with our rejection threshold, we need to establish the (very crude) bound $P_k \geq 1/k^k$ for all k.

Lemma 2. *The rejection threshold of Algorithm 1 satisfies* $p(z_{1:n-1}, x) \geq 1/n^n$ *for all* $n \geq 1$ *and all* $(z_{1:n-1}, x) \in (\mathcal{X} \times \mathcal{Y} \times \{0, 1\})^{n-1} \times \mathcal{X}$.

Note that this is a worst-case bound; our analysis shows that the probabilities P_k are more like $1/\mathrm{poly}(k)$ in the typical case.

5 Analysis

5.1 Correctness

We first prove a consistency guarantee for Algorithm 1 that bounds the generalization error of the importance weighted empirical error minimizer. The proof actually establishes a lower bound on

the query probabilities $P_i \geq 1/2$ for X_i such that $h_n(X_i) \neq h^*(X_i)$. This offers an intuitive characterization of the weighting landscape induced by the importance weights $1/P_i$.

Theorem 2. *The following holds with probability at least* $1-\delta$*. For any* $n \geq 1$*,*

$$0 \;\leq\; \mathrm{err}(h_n) - \mathrm{err}(h^*) \;\leq\; \mathrm{err}(h_n, Z_{1:n-1}) - \mathrm{err}(h^*, Z_{1:n-1}) + \sqrt{\frac{2C_0 \log n}{n-1}} + \frac{2C_0 \log n}{n-1}.$$

This implies, for all $n \geq 1$*,*

$$\mathrm{err}(h_n) \;\leq\; \mathrm{err}(h^*) + \sqrt{\frac{2C_0 \log n}{n-1}} + \frac{2C_0 \log n}{n-1}.$$

Therefore, the final hypothesis returned by Algorithm 1 after seeing n unlabeled data has roughly the same error bound as a hypothesis returned by a standard passive learner with n labeled data. A variant of this result under certain noise conditions is given in the appendix.

5.2 Label Complexity Analysis

We now bound the number of labels requested by Algorithm 1 after n iterations. The following lemma bounds the probability of querying the label Y_n; this is subsequently used to establish the final bound on the expected number of labels queried. The key to the proof is in relating empirical error differences and their deviations to the probability of querying a label. This is mediated through the *disagreement coefficient*, a quantity first used by [14] for analyzing the label complexity of the A^2 algorithm of [3]. The disagreement coefficient $\theta := \theta(h^*, \mathcal{H}, \mathcal{D})$ is defined as

$$\theta(h^*, \mathcal{H}, \mathcal{D}) := \sup\left\{ \frac{\Pr(X \in \mathrm{DIS}(h^*, r))}{r} : r > 0 \right\}$$

where

$$\mathrm{DIS}(h^*, r) := \{x \in \mathcal{X} : \exists h' \in \mathcal{H} \text{ such that } \Pr(h^*(X) \neq h'(X)) \leq r \text{ and } h^*(x) \neq h'(x)\}$$

(the disagreement region around h^* at radius r). This quantity is bounded for many learning problems studied in the literature; see [14, 6, 20, 21] for more discussion. Note that the supremum can instead be taken over $r > \epsilon$ if the target excess error is ϵ, which allows for a more detailed analysis.

Lemma 3. *Assume the bounds from Eq.* (4) *holds for all* $h \in \mathcal{H}$ *and* $n \geq 1$*. For any* $n \geq 1$*,*

$$\mathbb{E}[Q_n] \leq \theta \cdot 2\,\mathrm{err}(h^*) + O\left(\theta \cdot \sqrt{\frac{C_0 \log n}{n-1}} + \theta \cdot \frac{C_0 \log^2 n}{n-1}\right).$$

Theorem 3. *With probability at least* $1-\delta$*, the expected number of labels queried by Algorithm 1 after* n *iterations is at most*

$$1 + \theta \cdot 2\,\mathrm{err}(h^*) \cdot (n-1) + O\left(\theta \cdot \sqrt{C_0 n \log n} + \theta \cdot C_0 \log^3 n\right).$$

The bound is dominated by a linear term scaled by $\mathrm{err}(h^*)$, plus a sublinear term. The linear term $\mathrm{err}(h^*) \cdot n$ is unavoidable in the worst case, as evident from label complexity lower bounds [15, 5]. When $\mathrm{err}(h^*)$ is negligible (*e.g.*, the data is separable) and θ is bounded (as is the case for many problems studied in the literature [14]), then the bound represents a polynomial label complexity improvement over supervised learning, similar to that achieved by the version space algorithm from [5].

5.3 Analysis under Low Noise Conditions

Some recent work on active learning has focused on improved label complexity under certain noise conditions [17, 8, 18, 6, 7]. Specifically, it is assumed that there exists constants $\kappa > 0$ and $0 < \alpha \leq 1$ such that

$$\Pr(h(X) \neq h^*(X)) \leq \kappa \cdot (\mathrm{err}(h) - \mathrm{err}(h^*))^{\alpha} \tag{5}$$

for all $h \in \mathcal{H}$. This is related to Tsybakov's low noise condition [16]. Essentially, this condition requires that low error hypotheses not be too far from the optimal hypothesis h^* under the disagreement metric $\Pr(h^*(X) \neq h(X))$. Under this condition, Lemma 3 can be improved, which in turn yields the following theorem.

Theorem 4. *Assume that for some value of $\kappa > 0$ and $0 < \alpha \leq 1$, the condition in Eq.* (5) *holds for all $h \in \mathcal{H}$. There is a constant $c_\alpha > 0$ depending only on α such that the following holds. With probability at least $1 - \delta$, the expected number of labels queried by Algorithm 1 after n iterations is at most*

$$\theta \cdot \kappa \cdot c_\alpha \cdot (C_0 \log n)^{\alpha/2} \cdot n^{1-\alpha/2}.$$

Note that the bound is sublinear in n for all $0 < \alpha \leq 1$, which implies label complexity improvements whenever θ is bounded (an improved analogue of Theorem 2 under these conditions can be established using similar techniques). The previous algorithms of [6, 7] obtain even better rates under these noise conditions using specialized data dependent generalization bounds, but these algorithms also required optimizations over restricted version spaces, even for the bound computation.

6 Experiments

Although agnostic learning is typically intractable in the worst case, empirical risk minimization can serve as a useful abstraction for many practical supervised learning algorithms in non-worst case scenarios. With this in mind, we conducted a preliminary experimental evaluation of Algorithm 1, implemented using a popular algorithm for learning decision trees in place of the required ERM oracle. Specifically, we use the `J48` algorithm from Weka v3.6.2 (with default parameters) to select the hypothesis h_k in each round k; to produce the "alternative" hypothesis h'_k, we just modify the decision tree h_k by changing the label of the node used for predicting on x_k. Both of these procedures are clearly heuristic, but they are similar in spirit to the required optimizations. We set $C_0 = 8$ and $c_1 = c_2 = 1$—these can be regarded as tuning parameters, with C_0 controlling the aggressiveness of the rejection threshold. We did not perform parameter tuning with active learning although the importance weighting approach developed here could potentially be used for that. Rather, the goal of these experiments is to assess the compatibility of Algorithm 1 with an existing, practical supervised learning procedure.

6.1 Data Sets

We constructed two binary classification tasks using MNIST and KDDCUP99 data sets. For MNIST, we randomly chose 4000 training 3s and 5s for training (using the 3s as the positive class), and used all of the 1902 testing 3s and 5s for testing. For KDDCUP99, we randomly chose 5000 examples for training, and another 5000 for testing. In both cases, we reduced the dimension of the data to 25 using PCA.

To demonstrate the versatility of our algorithm, we also conducted a multi-class classification experiment using the entire MNIST data set (all ten digits, so 60000 training data and 10000 testing data). This required modifying how h'_k is selected: we force $h'_k(x_k) \neq h_k(x_k)$ by changing the label of the prediction node for x_k to the next best label. We used PCA to reduce the dimension to 40.

6.2 Results

We examined the test error as a function of (i) the number of unlabeled data seen, and (ii) the number of labels queried. We compared the performance of the active learner described above to a passive learner (one that queries every label, so (i) and (ii) are the same) using `J48` with default parameters.

In all three cases, the test errors as a function of the number of unlabeled data were roughly the same for both the active and passive learners. This agrees with the consistency guarantee from Theorem 2. We note that this is a basic property *not* satisfied by many active learning algorithms (this issue is discussed further in [22]).

In terms of test error as a function of the number of labels queried (Figure 2), the active learner had minimal improvement over the passive learner on the binary MNIST task, but a substantial improvement over the passive learner on the KDDCUP99 task (even at small numbers of label queries). For the multi-class MNIST task, the active learner had a moderate improvement over the passive learner. Note that KDDCUP99 is far less noisy (more separable) than MNIST 3s vs 5s task, so the results are in line with the label complexity behavior suggested by Theorem 3, which states that the label complexity improvement may scale with the error of the optimal hypothesis. Also,

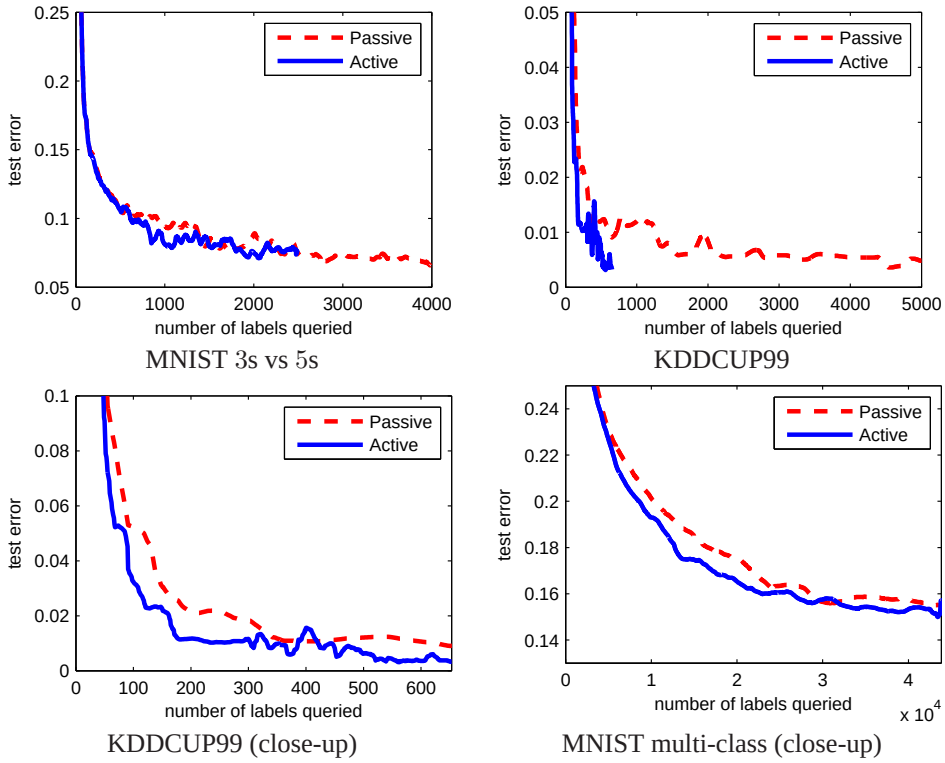

Figure 2: Test errors as a function of the number of labels queried.

the results from MNIST tasks suggest that the active learner may require an initial random sampling phase during which it is equivalent to the passive learner, and the advantage manifests itself after this phase. This again is consistent with the analysis (also see [14]), as the disagreement coefficient can be large at initial scales, yet much smaller as the number of (unlabeled) data increases and the scale becomes finer.

7 Conclusion

This paper provides a new active learning algorithm based on error minimization oracles, a departure from the version space approach adopted by previous works. The algorithm we introduce here motivates computationally tractable and effective methods for active learning with many classifier training algorithms. The overall algorithmic template applies to any training algorithm that (i) operates by approximate error minimization and (ii) for which the cost of switching a class prediction (as measured by example errors) can be estimated. Furthermore, although these properties might only hold in an approximate or heuristic sense, the created active learning algorithm will be "safe" in the sense that it will eventually converge to the same solution as a passive supervised learning algorithm. Consequently, we believe this approach can be widely used to reduce the cost of labeling in situations where labeling is expensive.

Recent theoretical work on active learning has focused on improving rates of convergence. However, in some applications, it may be desirable to improve performance at much smaller sample sizes, perhaps even at the cost of improved rates as long as consistency is ensured. Importance sampling and weighting techniques like those analyzed in this work may be useful for developing more aggressive strategies with such properties.

Acknowledgments

This work was completed while DH was at Yahoo! Research and UC San Diego.

References

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

[2] S. Dasgupta. Coarse sample complexity bounds for active learning. In *Advances in Neural Information Processing Systems 18*, 2005.

[3] M.-F. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. In *Twenty-Third International Conference on Machine Learning*, 2006.

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

[5] A. Beygelzimer, S. Dasgupta, and J. Langford. Importance weighted active learning. In *Twenty-Sixth International Conference on Machine Learning*, 2009.

[6] S. Hanneke. Adaptive rates of convergence in active learning. In *Twenty-Second Annual Conference on Learning Theory*, 2009.

[7] V. Koltchinskii. Rademacher complexities and bounding the excess risk in active learning. Manuscript, 2009.

[8] M.-F. Balcan, A. Broder, and T. Zhang. Margin based active learning. In *Twentieth Annual Conference on Learning Theory*, 2007.

[9] R. .S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[10] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal of Computing*, 32:48–77, 2002.

[11] M. Sugiyama, M. Krauledat, and K.-R. Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, 2007.

[12] M. Sugiyama. Active learning for misspecified models. In *Advances in Neural Information Processing Systems 18*, 2005.

[13] F. Bach. Active learning for misspecified generalized linear models. In *Advances in Neural Information Processing Systems 19*, 2006.

[14] S. Hanneke. A bound on the label complexity of agnostic active learning. In *Twenty-Fourth International Conference on Machine Learning*, 2007.

[15] M. Kääriäinen. Active learning in the non-realizable case. In *Seventeenth International Conference on Algorithmic Learning Theory*, 2006.

[16] A. B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *Annals of Statistics*, 32(1):135–166, 2004.

[17] R. Castro and R. Nowak. Upper and lower bounds for active learning. In *Allerton Conference on Communication, Control and Computing*, 2006.

[18] R. Castro and R. Nowak. Minimax bounds for active learning. In *Twentieth Annual Conference on Learning Theory*, 2007.

[19] T. Zhang. Data dependent concentration bounds for sequential prediction algorithms. In *Eighteenth Annual Conference on Learning Theory*, 2005.

[20] E. Friedman. Active learning for smooth problems. In *Twenty-Second Annual Conference on Learning Theory*, 2009.

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

[22] S. Dasgupta and D. Hsu. Hierarchical sampling for active learning. In *Twenty-Fifth International Conference on Machine Learning*, 2008.