Active Learning with Connections to Confidence-rated Prediction

Chicheng Zhang

Department of Computer Science and Engineering
University of California, San Diego
chichengzhang@ucsd.edu

Abstract

In the problem of active learning, we are given a set of unlabelled examples and the ability to query the labels of a subset of them, in an adaptive manner. The goal is to find a classifier with a target excess error, while querying as few labels as possible. In this report, we review several existing solutions to this problem: generalized binary search, disagreement-based active learning and margin-based active learning. These solutions are based on different notions of uncertainty of unlabelled examples. Furthermore, we give a novel characterization of uncertainty based on confidence-rated prediction with guaranteed error. In light of this characterization, we propose a new aggressive agnostic active learning algorithm that has superior label complexity over disagreement-based approaches.

1 Introduction

Active learning, a model that lies in the middle of supervised learning and unsupervised learning, has been fruitful over the past two decades, both in theory and in practice. In supervised learning (aka passive learning), the learning algorithm receive labelled examples drawn at random. On the other hand, in active learning, the learning algorithm is first provided with a large set of unlabelled examples, then it is allowed to interact with labelling experts, i.e. asking the labels of an unlabelled example. A typical active learning algorithm may follow the paradigm of alternating between picking the "most informative" unlabelled examples to query, and updating its belief on the data distribution based on this feedback. It has been shown empirically that active learning can significantly reduce labelling effort (often done by human labellers), while achieve the same accuracy compared to passive learning [SC08].

There has been a variety of applications of active learning to different domains. For instance, in sequence tagging problem(e.g. Named Entity Recognition or Part-of-Speech tagging), a tagger is supposed to assign labels to each words in a given sentence. (e.g. in Part-of-Speech tagging, given the sentence "I see a car", a tagger is supposed to assign labels (Pronoun, Verb, Determiner, Noun) to the sentence.) In passive learning setting, a labelled dataset with (sentence, tag) pairs is given as input to an offline learning algorithm. In contrast, in active learning, the algorithm trains a tagger by starting with an untagged corpus, and keep querying the tags of carefully selected sentences. In particular, the active learning algorithm in [SC08] keeps a probabilistic model, e.g. a conditional random field, to model the distribution of tagging given a particular sentence, given the information so far. Then it picks the sentence that it is most uncertain about, based on its current model parameters and heuristic criteria such as entropy. As another example, in text classification, [TK01] proposes a support vector machine type algorithm that trains a linear classifier based on the current labelled dataset, and queries the label of a new example according to its distance to the current decision boundary. In the experiments it is shown that the trained classifier using carefully selected queries has better test accuracy than randomly selected queries, given a fixed budget of label requests.

In this report we address this question theoretically: what is the number of label queries needed, if we aim to learn a classifier that provably has low error? In particular, three lines of existing work will be discussed in Section 3, based on different notions of uncertainty, that is, generalized binary search, disagreement-based active learning, and margin-based active learning. Furthermore, we propose a new characterization of uncertainty in Section 4, based on a connection with confidence-rated prediction with guaranteed error. We show how to construct a provably consistent agnostic active learning algorithm using this new notion of uncertainty. The proposed algorithm can be shown to have superior label complexity over disagreement-based approaches.

2 The Problem and the Model

2.1 Active Learning and Passive Learning

We consider the problem of PAC learning for binary classification. That is, each example $Z = (X, Y)$, where $X \in \mathcal{X}$ is a vector of *feature* and $Y \in \mathcal{Y} = \{0, 1\}$ is its *label.* The examples Z_i's are drawn from a fixed data distribution D over $\mathcal{X} \times \mathcal{Y}$ independently. In passive learning, each example comes with a complete (X_i, Y_i) pair. In active learning, the algorithm has the freedom to pick whichever unlabelled example to ask for its label: for each example Z_i, initially the feature X_i is presented to the algorithm, and the label Y_i is revealed only if the algorithm makes an explicit query to the labelling oracle $\mathcal{O}$. Let $\eta(x)$ denote the conditional probability of Y given X, i.e. $\mathbb{P}(Y = 1|X = x)$ induced by D. That is, $\mathcal{O}(x)$ returns a Bernoulli random variable with mean $\eta(x)$.

We are also given a hypothesis class $\mathcal{H}$ of VC dimension d, where each hypothesis $h \in \mathcal{H}$ is a classification rule mapping from $\mathcal{X}$ to $\mathcal{Y}$. To evaluate the performance of a hypothesis h, we use its generalization error, defined as the fraction of examples drawn from D that h makes a mistake on: $\text{err}(h) = \mathbb{P}_{(x,y)\sim D}(h(x) \neq y)$. Denote by h^* the optimal classifier among $\mathcal{H}$: $h^* = \text{argmin}_{h\in\mathcal{H}}\text{err}(h)$. The error of the optimal classifier is defined as $\nu = \text{err}(h^*)$. Since there can be inherent label noise in the data, we do not expect the error of the classifier learned going to zero as the number of training examples tends to infinity; instead, the goal of the learning algorithm is, given a target excess error $\epsilon > 0$ and failure probability $\delta > 0$, output a hypothesis $\hat{h} \in \mathcal{H}$ such that its excess error, $\text{err}(\hat{h}) - \text{err}(h^*)$ is at most ϵ, with probability $1 - \delta$. In passive learning, the performance is measured by its *sample complexity*, the number of "passive" examples needed with respect to ϵ and δ. In contrast, an active learning algorithm's performance is measured by its *label complexity*, that is the number of queries to $\mathcal{O}$, in terms of ϵ and δ. Denote by $D_{\mathcal{X}}$ the marginal distribution of D over $\mathcal{X}$. The distribution $D_{\mathcal{X}}$ introduces a pseudometric ρ within hypothesis class $\mathcal{H}$: $\rho(h, h') = \mathbb{P}(h(X) \neq h'(X))$. We define the disagreement ball $\text{B}(h, r)$ as the subset of classifiers that are r-close to h, that is, $\text{B}(h, r) = \{h' \in \mathcal{H} : \rho(h, h') \leq r\}$.

For a sample $S = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, the empirical error of $h \in \mathcal{H}$ on S is defined as $\text{err}_S(h) = \frac{1}{n}\sum_{i=1}^{n} I(h(X_i) \neq Y_i)$. The empirical distance of two classifiers $h, h' \in \mathcal{H}$ on S is defined as $\rho_S(h, h') = \frac{1}{n}\sum_{i=1}^{n} I(h(X_i) \neq h'(X_i))$.

2.2 Query Models

There are three popular concrete query models of active learning in literature: stream-based active learning, pool-based active learning and membership queries.

Stream-based Active Learning. This model is sometimes referred to as selective sampling in literature [CAL94, FSST97, CGZ04, DHM07, CCG11, DGS12]. We are given a sequence of examples (X_1, Y_1), $\ldots$, (X_n, Y_n), $\ldots$, drawn iid from D. At time t initially the feature of example X_t is presented, the algorithm has to make a decision to query the label Y_t or not. The algorithm is not allowed to revisit examples it has seen before, that is to query $Y_s, s < t$ at time t. This "one-pass" nature of the model is somewhat restrictive compared to pool-based active learning, as will be discussed in the next paragraph.

Pool-based Active Learning. An alternative model is pool-based active learning [Das05, BBL09, BBZ07, Han07, Kol10] – the algorithm is first presented with a (possibly) huge set of unlabelled examples U, namely the pool, drawn iid from $D_{\mathcal{X}}$. Then the algorithm queries a subset of the unlabelled examples in an adaptive manner; for instance, at time t the algorithm is allowed to query the label of an arbitrary example in U, based on its current belief of the informativeness of labels for each example. This is more flexible compared to stream-based active learning, in the sense that the algorithm can "postpone" the decision of making label queries after seeing the whole set of unlabelled exmaples. On the other hand, designing stream-based active learning algorithm may have the advantage that it is conceptually clean and advocates easy implementation.

Membership Queries. A stronger model, namely membership query, has been considered in computational learning theory and information theory community [Ang87, KMT93, Ang04, CN07, Now11]. Instead of only allowed to query examples from U drawn iid from $D_{\mathcal{X}}$, the algorithm has the freedom to query the label of an arbitrary example in $\mathcal{X}$. This can be thought of as having the added power of synthesizing examples to query the labelling oracle $\mathcal{O}$.

2.3 Noise Models

Let us focus on subclasses of $(\mathcal{H}, D)$ that may be of interest. For each subclass, we will compare the sample complexity upper bounds achieved by active learning to that achieved by passive learning in the next two sections.

Realizable. A widely used assumption is realizability, that is, there exists an (unknown) classifier $h \in \mathcal{H}$ that perfectly classifies all examples drawn from D, formally, $\nu = 0$. An immediate corollary is that the labels of the examples are deterministic, that is Y is equal to $h^*(X)$ almost surely.

Agnostic with a Given Optimal Error Rate. In general the data may not be separable. Recall that $\nu = \mathrm{err}(h^*)$ is the error of the optimal classifier. There can be two sources of errors contributing to ν. One is the approximation error – the Bayes classifier lies outside $\mathcal{H}$, the other being inherent noise – $\eta(x)$ is bounded away from 0 and 1. As we shall see in Lemma 1, there is a smooth transition of the sample complexity upper bound from ν bounded away from zero to $\nu = 0$.

Random Classification Noise. The following noise model is also a natural generalization of realizability. Consider a distribution D_0 over $\mathcal{X} \times \mathcal{Y}$ and a hypothesis class $\mathcal{H}$ where we are in realizable case. Then for each example (X, Y) drawn from D_0, flip the label Y with a fixed probability γ, where $0 \leq \gamma \leq 1/2$, to get $\tilde{Y}$. We call the resulting distribution of $(X, \tilde{Y})$, D, satisfies random classification noise with parameter γ.

Tsybakov Noise Condition. Modern sample complexity analysis crucially depends on the variance of the random variable $I(h(X) \neq Y) - I(h^*(X) \neq Y)$, which is upper bounded by $\rho(h, h^*)$ [MT99, Tsy04]. We call a labelled distribution D and a hypothesis $\mathcal{H}$ satisfies (C, κ)-Tsybakov Noise Condition for $C > 0$ and $\kappa \geq 1$, if and only if for all $h \in \mathcal{H}$,

$$\rho(h, h^*) \leq C(\mathrm{err}(h) - \mathrm{err}(h^*))^{\frac{1}{\kappa}}$$

Note that random classification noise with parameter γ is a special case with $\kappa = 1$ and $C = \frac{1}{1-2\gamma}$.

2.4 Sample Complexity Results in Passive Learning

In this section, we provide a brief review of the number of examples sufficient or necessary to learn a classifier with excess error ϵ in passive learning. These results are intended to be compared with the results of active learning below as a baseline.

2.4.1 Upper Bounds on Sample Complexity

Agnostic with a Given Optimal Error Rate. The following is an immediate corollary of normalized VC bound [VC71]. This gives the number of examples needed to get a classifier with excess error ϵ, when a particular learning algorithm, namely empirical risk minimization is used. Denote by $\hat{h}_m$ the empirical risk minimizer $\text{argmin}_{h \in \mathcal{H}} \text{err}_S(h)$. For refinements of logarithmic factors, please refer to [Han13].

Lemma 1. *There is an absolute constant $c > 0$ such that for any $\epsilon, \delta, \nu \in (0, 1)$, for any distribution D satisfying $\inf_{h \in \mathcal{H}} err(h) = \nu$, any hypothesis class $\mathcal{H}$ of VC dimension d, if the sample size*

$$m \geq c \frac{\nu + \epsilon}{\epsilon^2} (d \ln \frac{1}{\epsilon} + \ln \frac{1}{\delta})$$

then with probability $1 - \delta$ over the random draw of a training sample $S \sim D^m$, $err(\hat{h}_m) - err(h^*) \leq \epsilon$.

Tsybakov Noise Condition. The following is also an immediate corollary of normalized VC bound.

Lemma 2. *There is an absolute constant $c > 0$ such that for any $\epsilon, \delta \in (0, 1)$, $C > 0$, $\kappa \geq 1$, for any distribution D and hypothesis class $\mathcal{H}$ with VC dimension d satisfying (C, κ)-Tsybakov Noise Condition, if the sample size*

$$m \geq c \max(C \epsilon^{\frac{1}{\kappa} - 2} (d \ln C \epsilon^{\frac{1}{\kappa} - 2} + \ln \frac{1}{\delta}), \epsilon^{-1} (d \ln \epsilon^{-1} + \ln \frac{1}{\delta}))$$

then with probability $1 - \delta$ over the random draw of a training sample $S \sim D^m$, $err(\hat{h}_m) - err(h^*) \leq \epsilon$.

Realizable. An immediate corollary of the two lemmas above is the sample complexity upper bound in realizable case. Plugging in $\nu = 0$ in Lemma 1, or $C = 1, \kappa = 1$ in Lemma 2, we get the following result.

Corollary 1. *There is an absolute constant $c > 0$ such that for any $\epsilon, \delta \in (0, 1)$, for any distribution D satisfying $\inf_{h \in \mathcal{H}} err(h) = 0$, if the sample size*

$$m \geq \frac{c}{\epsilon} (d \ln \frac{1}{\epsilon} + \ln \frac{1}{\delta})$$

then with probability $1 - \delta$ over the random draw of a training sample $S \sim D^m$, $err(\hat{h}_m) - err(h^*) \leq \epsilon$.

2.4.2 Lower Bounds on Sample Complexity

To complement the upper bounds above, let us review some results on lower bounds of the sample complexity. These results essentially shows that the empirical risk minimization approach is minimax optimal, modulo a logarithmic factor of $1/\epsilon$. In conjunction with label complexity upper bounds in active learning discussed in the next section, they show that active learning has asymptotically lower label complexity compared to passive learning under certain assumptions.

Agnostic with a Given Optimal Error Rate.

Lemma 3 (see e.g. [Han13]). *There is an absolute constant $c > 0$ such that for any $\epsilon, \delta, \nu \in (0, 1)$, there exists a distribution D satisfying $err(h^*) = \nu$, for any (randomized) learning algorithm $\mathcal{A}$ taking a sample as input and a classifier in $\mathcal{H}$ as output, if the sample size*

$$m \leq c \frac{\nu + \epsilon}{\epsilon^2} (d + \ln \frac{1}{\delta})$$

Then with probability at least δ over the draw of a training sample $S \sim D^m$ (and the randomness in $\mathcal{A}$), $err(\mathcal{A}(S)) - err(h^*) \geq \epsilon$.

Tsybakov Noise Condition.

Lemma 4 (see e.g. [Han13]). *There is an absolute constant $c > 0$ such that for any $\epsilon, \delta \in (0,1)$, $C > 0$, $\kappa \geq 1$, there exists a distribution D satisfying $\rho(h, h^*) \leq C(err(h) - err(h^*))^{\frac{1}{\kappa}}, \forall h \in \mathcal{H}$, for any (randomized) learning algorithm $\mathcal{A}$ taking a sample as input and a classifier in $\mathcal{H}$ as output, if the sample size*

$$m \leq c \cdot C\epsilon^{\frac{1}{\kappa}-2}(d + \ln\frac{1}{\delta})$$

Then with probability at least δ over the draw of a training sample $S \sim D^m$ (and the randomness in $\mathcal{A}$), $err(\mathcal{A}(S)) - err(h^*) \geq \epsilon$.

We summarize the results above in Table 1.

Table 1: A Comparison of the Sample Complexity Bounds in Passive Learning

Noise Model	Sample Complexity Upper Bound	Sample Complexity Lower Bound
Agnostic with Given ν	$O(\frac{\nu+\epsilon}{\epsilon^2} \cdot d\ln\frac{1}{\epsilon})$	$\Omega(\frac{\nu+\epsilon}{\epsilon^2} \cdot d)$
Tsybakov Noise Condition	$O(\max(\frac{C}{\epsilon^{2-\frac{1}{\kappa}}} \cdot d\ln\frac{C}{\epsilon^{2-\frac{1}{\kappa}}}, \frac{1}{\epsilon} \cdot d\ln\frac{1}{\epsilon}))$	$\Omega(\epsilon^{2-\frac{1}{\kappa}} \cdot d)$
Random Classification Noise	$O(\frac{1}{(1-2\gamma)\epsilon} \cdot d\ln\frac{1}{(1-2\gamma)\epsilon})$	$\Omega(\frac{1}{(1-2\gamma)\epsilon} \cdot d)$
Realizable	$O(\frac{1}{\epsilon} \cdot d\ln\frac{1}{\epsilon})$	$\Omega(\frac{1}{\epsilon} \cdot d)$

3 Existing Solutions

In this section, we review several previous approaches to active learning. An intuitive family of algorithm is a generalization of binary search, but such algorithm may not be consistent in general. To ensure consistency in agnostic setting, a second line of work, namely disagreement-based active learning, has been proposed. A third line of research, namely margin-based active learning, which has sharper label complexity than disagreement-based algorithms, works only when the hypothesis space $\mathcal{H}$ is the set of homogeneous linear classifiers and underlying distribution $D_{\mathcal{X}}$ is isotropic log-concave in $\mathbb{R}^d$.

3.1 Generalized Binary Search

Let us consider the following simple but illustrative example. Suppose we are in the realizable case. $D_{\mathcal{X}}$ is uniform over the interval $[0,1]$, and the hypothesis class contains all threshold classifiers, namely $\mathcal{H} = \{I(x > t), t \in (0,1)\}$. There is an unknown threshold $t^* \in (0,1)$ such that the examples lying in the right hand side of t is labelled 1 and vice versa. How many examples are needed for passive learning to get a classifier with error rate ϵ? Intuitively, if no examples are within, say 10ϵ of the threshold t^*, we might make a rather inaccurate guess of t^*, the generalization error thus exceeding ϵ. By making the argument rigorous, it can be shown that the sample complexity of passive learning is at least $\Omega(\frac{1}{\epsilon})$.

On the other hand, consider the following binary-search style algorithm. First we randomly draw $O(1/\epsilon)$ unlabelled examples. Then we start with querying the example farthest to the left and the one farthest to the right. If they are of the same label, stop. If their labels are different, it must be the case where the leftmost example is negative labelled and the rightmost one is positive labelled. Then we keep querying the label of the median of the remaining examples; the key observation is that once the label of the median is revealed, we can significantly refine our search space for the threshold t^*. If it is positive then we search within the left half, otherwise we search within the right half. By performing this search recursively, eventually we have two consequtive points in the dataset where their labels are different. Thus we only need $O(\ln \frac{1}{\epsilon})$ queries to the labelling oracle, achieving an *exponentially* label saving.

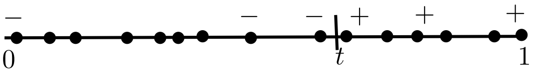

Figure 1: An example of binary search style active learning to (approximately) find the underlying threshold, which queries $O(\ln \frac{1}{\epsilon})$ examples compared to $O(\frac{1}{\epsilon})$ examples in passive learning.

Conceptually, the above procedure can be thought of as follows: first draw $n = O(\frac{1}{\epsilon})$ examples so that they split $\mathcal{H}$ into $n+1$ equivalence classes $C_1, \ldots, C_n \subseteq \mathcal{H}$, where within each class all the classifiers label the examples drawn unanimously. By realizability, there is exactly one equivalence class C_{i^*} that contains h^*. Meanwhile by standard VC theory, for all $h \in C_{i^*}$, $\text{err}(h) \leq \epsilon$. Our problem thus reduces to identifying i^* through actively querying the labels of the examples drawn. In particular, in the above example, the algorithm keeps track of all the "alive" equivalence classes with respect to queries it has made, and queries the example that "bisects" the set of alive equivalence classes as much as possible in a greedy fashion. Abstracting from this procedure, we get a binary search style algorithm on general hypothesis class $\mathcal{H}$ and data distribution D.

Algorithm 1 A Generalized Binary Search, based on ideas of [FSST97, Now11]

```
Inputs: Target excess error $\epsilon$, failure probability $\delta$.
Draw $m = O(\frac{d}{\epsilon}(\ln \frac{1}{\epsilon} + \ln \frac{1}{\delta}))$ unlabelled examples $x_1, \ldots, x_m$ from $D_{\mathcal{X}}$ to form a set $S$. $S$ partitions $\mathcal{H}$ into $n$ equivalence classes $\mathcal{C} = \{C_1, \ldots C_n\}$.
while $|\mathcal{C}| > 1$ do
    Find $j = \text{argmin}_{j=1}^{m} \max(|\{C \in \mathcal{C} : C(x_j) = 1\}|, |\{C \in \mathcal{C} : C(x_j) = 0\}|)$, i.e. an example $x_j \in S$ that maximally bisects $\mathcal{C}$.
    Query the label of $x_j$ to get $y_j$.
    Update the set of "alive" equivalence classes $\mathcal{C} \leftarrow \{C \in \mathcal{C} : C(x_j) = y_j\}$.
Return an arbitrary hypothesis in the only equivalence class in $\mathcal{C}$.
```

Although this algorithm is intuitive, it is not clear how to analyze it in general. However a conceptually similar algorithm, named *Splitting* in [Das05] that aims to reduce the number of disagreement "edges", achieves exponentially label savings in several canonical examples.

Theorem 1. *Let $D_{\mathcal{X}}$ be the uniform distribution over $[0,1]$, $\mathcal{H}$ be the class of threshold classifiers $\mathcal{H} = \{I(x > t), t \in (0,1)\}$. Suppose the Splitting Algorithm in [Das05] is run with $\mathcal{H}$ and D. Then with probability $1-\delta$, (1) err$(\hat{h}) \leq \epsilon$. (2) The number of queries is at most*

$$\tilde{O}(\ln \frac{1}{\epsilon})^1$$

Theorem 2. *Let $D_{\mathcal{X}}$ be the uniform distribution over $\mathbb{S}^{d-1}$, $\mathcal{H}$ be the class of homogeneous linear classifiers $\mathcal{H} = \{I(w \cdot x \geq 0), w \in \mathbb{S}^{d-1}\}$. Suppose the Splitting Algorithm in [Das05] is run with $\mathcal{H}$ and D. Then with probability $1-\delta$, (1) err$(\hat{h}) \leq \epsilon$. (2) The number of queries is at most*

$$\tilde{O}(d \ln \frac{1}{\epsilon})$$

3.1.1 Pros and Cons

On the positive side, the generalized binary search formulation provides sharp label complexity results. In the cases of homogeneous linear classifier with uniform distribution in $\mathbb{S}^{d-1}$, and threshold classifier

[1] $\tilde{O}(\cdot)$ hides multiplicative factors of $\ln\ln \frac{1}{\epsilon}$, $\ln \frac{1}{\delta}$

with uniform distribution in $[0, 1]$, matching lower bounds($\Omega(d \ln \frac{1}{\epsilon})$ and $\Omega(\ln \frac{1}{\epsilon})$, respectively) have been established [KMT93].

On the negative side, the algorithms often works only in realizable case. Although there has been attempts on extending the algorithm to noisy case [CN07, Kää06, BZ74], relative little progress has been made – it is not clear how to extend these results to the agnostic case (where the algorithm has no knowledge of data distribution D except for having access to the examples drawn from it). For instance, in [CN07], the analysis of active learning a threshold function under geometric Tsybakov noise condition crucially relies upon a reduction to random classification noise case and then applying the Burnashev-Zigangirov Algorithm [BZ74]. As another example, the algorithm of active learning a d-dimensional α-smoooth boundary fragment class under (C, κ)-Tsybakov noise condition in [CN07] reduces the problem to $O(n^{\frac{d-1}{\alpha(2\kappa-2)+d-1}})$ 1-dimensional noisy binary searches. Then a Lagrange interpolation is performed to get the learned decision boundary. Such algorithms are brittle, in the sense that: 1. they rely on strong assumptions on the geometry of the data. 2. the parameters of data distributions, e.g. the Tsybakov noise condition parameter κ, or the random classification noise parameter γ, have to be the input of the algorithms. It is not clear whether the algorithm will still be consistent, if the model parameters is misspecified(See [DH08] for a informal argument). We review a line of work providing algorithms that overcome these disadvantages in the next subsection.

3.2 Disagreement-Based Active Learning

Is there any active learning algorithm that works without any knowledge of underlying distribution, achieves consistency, and still provides label complexity superior to passive learning in certain scenarios? Below we review an important line of work, namely disagreement-based active learning, which gives a positive answer to this question. It maintains a set of hypotheses, and alternates between two steps: pruning the hypothesis set given the labelled examples we have queried so far, and querying the labels on a carefully sampled set of examples, based on a notion of *disagreement* of the candidate hypothesis set.

3.2.1 Realizable Case: CAL

It is instructive to consider the realizable case as a starting point. The following algorithm, CAL, named after its authors [CAL94], works on a stream-based setting. It keeps a set of hypotheses V_t which may potentially be h^* based the information we have so far, namely a *version space*. In particular, V_t is defined as all classifiers consistent with all examples $\{(x_s, y_s)\}_{s=1}^{t-1}$(See Figure 2 for an illustration). It is guaranteed that $h^* \in V_t$ for all t. For each new example x_t, if it lies on the disagreement region $\text{DIS}(V_{t-1}) = \{x : \exists h_1, h_2 \in V_{t-1}, h_1(x) \neq h_2(x)\}$, we simply query the label y_t and prune the version space accordingly. Otherwise, its label y_t can be directly "inferred", thus uninformative, since all the hypotheses in V_{t-1}(including h^*) classifies x_t unanimously.

Algorithm 2 The CAL Active Learning Algorithm

```
Inputs: Target excess error $\epsilon$, failure probability $\delta$.
$V_0 \leftarrow \mathcal{H}$, $t_0 \leftarrow \lceil \frac{96}{\epsilon}(d \ln \frac{96}{\epsilon} + \ln \frac{24}{\delta}) \rceil$.
for $t = 1, 2, \ldots, t_0$ do
    Get a new unlabelled example $x_t$.
    if $x_t \in \text{DIS}(V_{t-1})$ then
        Query the label $y_t$ of $x_t$ and set $Q_t \leftarrow 1$.
        Update version space $V_t \leftarrow \{h \in V_{t-1} : h(x_t) = y_t\}$.
    else
        Set $Q_t \leftarrow 0$.
Return an arbitrary classifer $\hat{h} \in V_{t_0}$.
```

It can be readily seen that the output of the algorithm will be equivalent to a classifier consistent on the t_0 examples as if we have seen the all the labels. Thus, the algorithm achieves the target error ϵ with

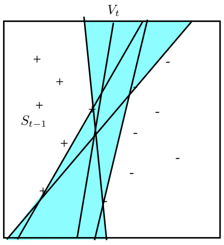

Figure 2: In linear classification and realizable case, the version space V_t is the set of hypotheses that are consistent with the examples in S_{t-1}.

probability $1 - \delta$, by the choice of t_0 and standard VC bounds.

What does the number of label requests result in? It turns out that a quantity that depends on the structure of the hypothesis space $\mathcal{H}$ and underlying distibution D, namely disagreement coefficient [Han07], is key to the analysis of CAL's label complexity.

Definition 1 ([Han07]). *Given a hypothesis space $\mathcal{H}$, data distribution D, the disagreement coefficient of h with respect to $\mathcal{H}$ and D is defined as:*

$$\theta_h(\epsilon) = \sup_{r > \epsilon} \frac{\mathbb{P}_D(DIS(B(h, r)))}{r}$$

See Figure 3 for an example illustrating the related concepts. Under nondegenerate assumptions, $1 \leq \theta_h(\epsilon) \leq 1/\epsilon$. Roughly speaking, the disagreement coefficient captures how fast the disagreement region shrinks with the radius of version space shrinking. If $\sup_{\epsilon > 0} = \theta_h(\epsilon) \leq C$ where C is a constant independent of ϵ, then we may expect the disagreement region to shrink relatively fast as the algorithm proceeds. Some canonical examples are presented below.

Fact 1. *Let $D_{\mathcal{X}}$ be the uniform distribution over $[0, 1]$, $\mathcal{H}$ be the class of threshold classifiers $\mathcal{H} = \{I(x > t), t \in (0, 1)\}$. The disagreement coefficient of any $h \in \mathcal{H}$ satisfies $\theta_h(\epsilon) \leq 2$.*

Fact 2. *Let $D_{\mathcal{X}}$ be the uniform distribution over $[0, 1]$, $\mathcal{H}$ be the class of interval classifiers $\mathcal{H} = \{I(a < x < b), a, b \in (0, 1)\}$. The disagreement coefficient of any $h = I(a < x < b) \in \mathcal{H}, a < b$ satisfies $\theta_h(\epsilon) \leq \max(4, \frac{1}{b-a})$.*

Fact 3. *Let $D_{\mathcal{X}}$ be the uniform distribution over unit sphere $\mathbb{S}^{d-1}$, $\mathcal{H}$ be the class of homogeneous linear classifiers $\mathcal{H} = \{I(w \cdot x \geq 0), w \in \mathbb{S}^{d-1}\}$. The disagreement coefficient of any $h \in \mathcal{H}$ satisfies $\theta_h(\epsilon) \leq \sqrt{d}$.*

However, there are certain settings of $(\mathcal{H}, D)$ where there exists classifier in $\mathcal{H}$ that has disagreement coefficient growing unbounded as $\epsilon \downarrow 0$:

Fact 4. *Let $D_{\mathcal{X}}$ be the uniform distribution on $[0, 1]$, $\mathcal{H}$ be the class of interval classifiers $I(a < x < b), a, b \in (0, 1)$. The disagreement coefficient of $h \in \mathcal{H}$, where $h = I(a < x < a) \equiv 0$ satisfies $\theta_h(\epsilon) = \frac{1}{\epsilon}$.*

To see how the disagreement coefficient affects the label complexity of CAL, let us review the following crucial result.

Theorem 3 ([Han09]). *Suppose Algorithm 2 is run for some certain hypothesis space $\mathcal{H}$ and distribution D, with target error ϵ and failure probability δ. Then with probability $1 - \delta$, (1) The output $\hat{h}$ satisfies $err(\hat{h}) \leq \epsilon$. (2) The number of labels queried is*

$$\tilde{O}(\theta_{h^*}(\epsilon) d \ln \frac{1}{\epsilon})$$

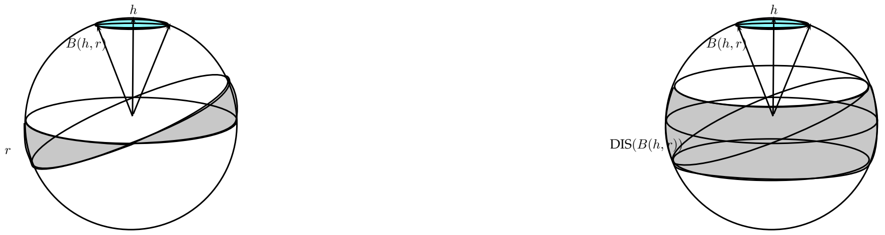

Figure 3: An illutration of the disagreement ball $\mathrm{B}(h, r)$ and its disagreement region in the case where $\mathcal{H}$ is the set of homogeneous linear classifiers and D is the uniform distribution over $\mathbb{S}^{d-1}$, where $d = 3$.

3.2.2 Agnostic Case

Note now in agnostic case, the definition of a version space is no longer valid: there might be no classifier consistent with the example so far due to the inherent noise of the model! Instead, a notion of candidiate set will be used(see [BBL09, Kol10] or implicitly [DHM07, Han09]). The idea is that we first want to guarantee with high probability, h^* is kept in the version space, other than this constraint, the version space is as "tight" as possible.

Consider Algorithm 3, which is a straightforward generalization of Algorithm 2. As first sight, it may not even be clear that it is implementable, since e.g. in time t, $\mathrm{err}_{S_t}(h)$ cannot be directly evaluated by the labelled and unlabelled examples we have. For those examples x_t's that lie in the agreement region of version space V_{t-1}, we do not query their labels, (y_t can not be inferred exactly, although $h^*(x_t)$ can be inferred exactly) but that whether y_t is -1 or $+1$ does not matter – what we are really interested in is the *difference* of empirical error between two classifiers lying in V_t. Meanwhile, since V_t's are nested, it is guaranteed that if we do not query the label of example x_s for $s < t$, then x_s is in the agreement region of V_s, hence the agreement region of V_{t-1}. That said, for $h \in V_{t-1}$, although $\mathrm{err}_{S_t}(h)$ is not observable, $\mathrm{err}_{S_t}(h) - \mathrm{err}_{S_t}(\hat{h}_t)$ is observable, and equals $\frac{1}{t}\sum_{s\le t, Q_s=1}(I(h(x_s) \neq y_s) - I(\hat{h}_t(x_s) \neq y_s))$. Also note that $\rho_{S_t}(h, h')$ is observable for any $h, h' \in \mathcal{H}$. Therefore, Algorithm 3 is well-defined.

To give a better illustration of what Algorithm 3 is doing, let us consider removing the active query step. The algorithm comes down to repeatedly applying the following update rule:

$$V_t \leftarrow \{h \in V_{t-1} : \mathrm{err}_{S_t}(h) - \mathrm{err}_{S_t}(\hat{h}_t) \le \sigma(t, \delta_t) + \sqrt{\sigma(t, \delta_t)\rho_{S_t}(h, \hat{h}_t)}\}$$

See Figure 4 for a pictorial description. By induction, it is ensured that $h^* \in V_t$.

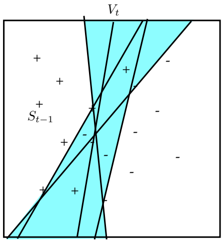

Figure 4: In linear classification and non-realizable case, the candidate V_t is the set of hypotheses that has relatively small empirical error with respect to S_{t-1}; see Algorithm 3 for its precise definition.

Same as in the realizable case, the number of label queries can be analyzed by bounding the individual query probability in time t in conjunction with a standard application of Freedman's Inequality. The probability of making a query in time t is related to two quantities. The first one is disagreement coefficient $\theta(\epsilon)$, and the second is the diameter of V_t. By assumptions of different noise models, the diameter of V_t shrinks differently, for instance $O(\sqrt{\frac{\nu d \ln t}{t}} + \frac{d \ln t}{t})$ if $\mathrm{err}(h^*) = \nu$, or $O(C(\frac{Cd \ln t}{t})^{\frac{1}{2\kappa-1}})$ under (C, κ)-Tsybakov noise condition. We address the label complexity of Algorithm 3 under two different noise models separately in the following two theorems.

Algorithm 3 an Agnostic Active Learning Algorithm, based on [DHM07, Han09]

```
Inputs: Target excess error $\epsilon$, failure probability $\delta$.
Initialize version space $V_0 = \mathcal{H}$.
for $t = 1, 2, \ldots$ do
    Draw a new unlabelled example $x_t$ from $D_{\mathcal{X}}$.
    if $x_t \in \mathrm{DIS}(V_{t-1})$ then
        Query the label $y_t$ of $x_t$ and set $Q_t \leftarrow 1$.
    else
        Set $Q_t \leftarrow 0$.
    Get the empirical risk minimizer within $V_{t-1}$: $\hat{h}_t \leftarrow \arg\min_{h \in V_{t-1}} \mathrm{err}_{S_t}(h)$.
    Update candidate set of $h^*$: $V_t \leftarrow \{h \in V_{t-1} : \mathrm{err}_{S_t}(h) - \mathrm{err}_{S_t}(\hat{h}_t) \le \sigma(t, \delta_t) + \sqrt{\sigma(t, \delta_t)\rho_{S_t}(h, \hat{h}_t)}\}$.
    if $\sup_{h \in V_t} \sqrt{\sigma(t, \delta_t)\rho_{S_t}(h, \hat{h}_t)} + \sigma(t, \delta_t) \le \epsilon$ then
        $t_0 \leftarrow t$, break
Return $\hat{h}_{t_0}$.
```

Theorem 4. *Suppose Algorithm 3 is run for hypothesis space $\mathcal{H}$ and distribution D satisfying err$(h^*) = \nu$, with target excess error ϵ and failure probability δ. Then with probability $1 - \delta$, (1) The output $\hat{h}$ satisfies err$(\hat{h})$ − err$(h^*) \le \epsilon$. (2) The number of labels queried is at most*

$$\tilde{O}(\theta_{h^*}(2\nu + \epsilon) d \ln \frac{1}{\epsilon}(\ln \frac{1}{\epsilon} + \frac{\nu^2}{\epsilon^2}))$$

Theorem 5. *Suppose Algorithm 3 is run for hypothesis space $\mathcal{H}$ and distribution D satisfying (C, κ)-Tsybakov Noise Condition, with target excess error ϵ and failure probability δ. Then with probability $1 - \delta$, (1) The output $\hat{h}$ satisfies err$(\hat{h})$ − err$(h^*) \le \epsilon$. (2) The number of labels queried is at most*

$$\tilde{O}(\theta_{h^*}(C\epsilon^{\frac{1}{\kappa}}) d \ln^2 \frac{1}{\epsilon} \epsilon^{\frac{2}{\kappa} - 2})$$

Note that in realizable case, by letting $\nu = 0$ in Theorem 4 or letting $C = 1$ and $\kappa = 1$ in Theorem 5, we essentially recover the same label complexity bound as Theorem 3 by using a different algorithm. Note that since $\theta(\epsilon) \le 1/\epsilon$, the sample complexity bound achieved is never worse than passive learning, modulo logarithmic factors.

3.3 Pros and Cons

Compared with generalized binary search, the label query strategy in CAL is quite conservative – for any new example x, so long as it lies on the disagreement region of the version space, it will be queried. This "mellow" nature is somehow due to the proof technique – we reduce our active learning problem to passive learning over t examples without introducing any bias on the training examples, that is, h^* is optimal with respect to $D|_{\mathrm{DIS}(V_{t-1})}$, thus error minimization on $D|_{\mathrm{DIS}(V_{t-1})}$ is equivalent to error minimization on D. As a result, its label complexity is sometimes suboptimal – for example, in the case of homogeneous linear classifier with uniform distribution, CAL achieves $\tilde{O}(d^{\frac{3}{2}} \ln \frac{1}{\epsilon})$ in realizable case, while generalized binary search algorithms such as [Das05, FSST97] achieves a better bound of $\tilde{O}(d \ln \frac{1}{\epsilon})$.

On the positive side, Algorithm 3 is completely "agnostic", in the sense that it works for any hypothesis class of VC dimension d, nor does it need to know the parameters such as ν or the parameters of Tsybakov noise condition C, κ. Orthogonal to this, the notion of disagreement coefficient [Han07] is a generic tool for analyzing how much improvement in label complexity can be achieved, if a disagreement-based filter is employed.

3.4 Margin-Based Active Learning

A third line of work looks at the problem of learning a homogeneous linear classifier under isotropic log-concave distribution geometrically [BBZ07, BL13, ABL14, WS14]. It queries labels more aggressively than disagreement-based approaches. Judging from the algorithmic framework, it is not performing generalized binary search, in the sense that it does not attempt to reduce the "size" of the version space [FSST97, Now11], or disagreement "edges" [Das05]. The algorithm proceeds in epochs, where in epoch k a linear classifier w_k with excess error at most ϵ_k is trained. The name "Margin-Based" comes from the fact that in each epoch, it only queries the examples that lie within a distance b_k of the hyperplane $\{x : w_{k-1} \cdot x = 0\}$.

Algorithm 4 Margin Based Active Learning

```
Inputs: Target excess error $\epsilon$, failure probability $\delta$.
Initialization: Let $w_0$ be an arbitrary homogeneous linear classifier in $\mathcal{H}$.
for $k = 1, 2, \ldots, k_0 = \lceil \log \frac{1}{\epsilon} \rceil$ do
    Sample a labelled dataset $S_k$ of size $m_k$, from $D$ conditioned on the band $B_k = \{x : |w_{k-1} \cdot x| \le b_k\}$.
    Train a linear classifier $w_k$ that minimizes empirical error on $S_k$ over $\mathrm{B}(w_{k-1}, r_k)$.[2]
Return $w_{k_0}$.
```

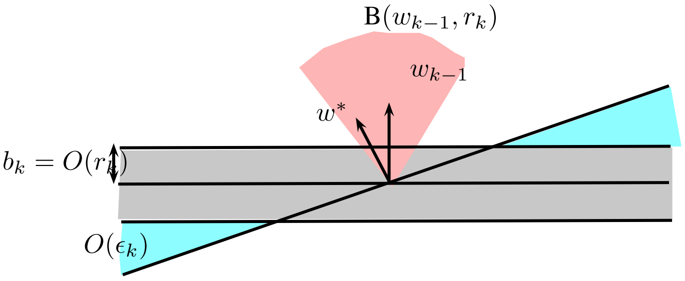

Figure 5: In margin-based active learning, on epoch k we only need to learn a classifier with excess error $O(\frac{\epsilon_k}{r_k})$ with respect to distribution $D|_{B_k}$.

The algorithm relies upon two ideas. The first idea is that under realizable assumptions or Tsybakov noise condition, the structure of the hypothesis space can be well characterized. Conceptually we can keep a "virtual" version space in a compact representation $\mathrm{B}(w_{k-1}, r_k)$(although it is not explicitly used anywhere in the algorithimic implemetation), for appropriate setting of r_k, without sacrificing much loss of tightness, if $\mathrm{err}(w_{k-1}) - \mathrm{err}(w^*) \le \epsilon_{k-1}$. A more important observation is that by a careful choice of b_k which makes $\mathbb{P}(B_k) = \tilde{O}(r_k)$(cf. $O(\theta_{h^*}(r_k) r_k)$, the probability of disagreement region of $\mathrm{B}(w_{k-1}, r_k)$), there is still hope that the algorithm is consistent when focusing on B_k. More formally, the following lemma guarantees that we only need to focus on learning a classifier with excess error at most $O(\frac{\epsilon_k}{\mathbb{P}(B_k)})$ with respect to distribution $D|_{B_k}$, if the goal is to learn a classifier w_k which has excess error at most ϵ_k with respect to D in epoch k.

Lemma 5 ([BL13]). *There exist absolute constants $c_1, c_2 > 0$ such that if two linear classifiers w and w' have angle α between them, then for any $\epsilon > 0$, we have*

$$\mathbb{P}(h_{w_1}(x) \neq h_{w_2}(x), |w_1 \cdot x| \ge c_1 \alpha \ln \frac{c_2 \alpha}{\epsilon}) \le \epsilon$$

We now state the label complexity results of the algorithm, under realizable and Tsybakov Noise Condition, with appropriate settings of parameters b_k and r_k respectively.

[2] If we replace $\mathrm{B}(w_{k-1}, r_k)$ with $\mathcal{H}$ we get a simpler algorithm that requires more involved analysis.

Theorem 6 ([BL13]). *Suppose Algorithm 4 is run on hypothesis space $\mathcal{H}$ consisting all homogeneous linear classifiers and distribution D satisfying $D_{\mathcal{X}}$ is isotropic log-concave and err$(h^*) = 0$, with target error ϵ and failure probability δ. Then with settings of parameters $\epsilon_k = 2^{-k}$, $r_k = \epsilon_{k-1}$, and $b_k = c_1 r_k \ln \frac{c_2 r_k}{\epsilon_k}$, with probability $1 - \delta$, (1) The output $\hat{h}$ satisfies err$(\hat{h}) \leq \epsilon$. (2) The number of labels queried is*

$$\tilde{O}(d \ln^2 \frac{1}{\epsilon})$$

Theorem 7 ([BL13]). *Suppose Algorithm 4 is run on hypothesis space $\mathcal{H}$ consisting all homogeneous linear classifiers and distribution D satisfying $D_{\mathcal{X}}$ is isotropic log-concave and $(\mathcal{H}, D)$ satisfies (C, κ) Tsybakov noise condition, with target excess error ϵ and failure probability δ. Then with settings of parameters $\epsilon_k = 2^{-k}$, $r_k = C\epsilon_{k-1}^{\frac{1}{\kappa}}$, and $b_k = c_1 r_k \ln \frac{c_2 r_k}{\epsilon_k}$, with probability $1 - \delta$, (1) The output $\hat{h}$ satisfies err$(\hat{h})$ − err$(h^*) \leq \epsilon$. (2) The number of labels queried is*

$$\tilde{O}(d \ln^2 \frac{1}{\epsilon} \epsilon^{\frac{2}{\kappa} - 2})$$

3.5 Pros and Cons

Algorithm 4 is particularly specific – it works only under the assumption that $D_{\mathcal{X}}$ is isotropic log-concave(or more weakly, *admissible* distribution defined in [BL13]). Meanwhile, κ and C must be inputs of the algorithm under Tsybakov noise condition, if we hope to get the tightest label complexity bounds. Some attempts has been made to make κ and C not being inputs to the algorithm [WS14, BBZ07]. But still, in general agnostic settings where there can be arbitrary noise, it is not clear how to choose parameters b_k and r_k to ensure consistency and enjoy superior label complexity than passive learning.[3]

Nevertheless, in the setting of homogeneous linear classifier and isotropic log-concave distribution, this algorithm achieves near optimal label complexity $\tilde{O}(d \ln \frac{1}{\epsilon})$, as is achieved by generalized binary search style algorithms. Compared to generalized binary search, the algorithm can tolerate a broader family of noise.

Table 2: A comparison of the existing approach and our proposed Algorithm

Algorithms	Noise Tolerance?	Works under any $(\mathcal{H}, D)$?	Label Query Strategy
Generalized Binary Search	Limited(RCN)	Yes	Aggressive
Disagreement Based Active Learning	Any Noise	Yes	Not Aggressive
Margin Based Active Learning	Limited(RCN, TNC)	No	Somewhat Aggressive
Our Algorithm	Any Noise	Yes	Somewhat Aggressive

4 Confidence-Based Active Learning

In this section we propose a new active learning algorithm that has the following properties: it is consistent, works under general $(\mathcal{H}, D)$, and has better label complexity bound than known upper bounds of disagreement-based algorithms. Our algorithm is based on a reduction from active learning to confidence-rated prediction with guaranteed error.

[3] Note that if we are in agnostic case and ν is part of input of the algorithm, it is also possible to develop a version of Algorithm 4 achieving consistency and having a label complexity of $\tilde{O}(d(\frac{\nu^2}{\epsilon^2} + \ln \frac{1}{\epsilon}))$.

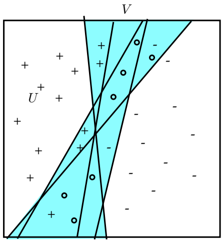

Figure 6: A typical output of a confidence-rated predictor P, where a version space V is computed beforehand. We use $+,-,\circ$ to represent P predicting 1, 0, $\perp$, respectively. Note that in general, a confidence-rated predictor may randomize rather than deterministically output a label for a given $x \in U$.

4.1 Confidence-Rated Prediction

We give a brief introduction to confidence-rated prediction. Informally, a confidence-rated predictor is a randomized classifier that not only has the ability to output a label in $\{0, 1\}$, but is also allowed to output "Don't know"($\perp$), for each test example. Intuitively, when a confidence-rated predictor outputs "Don't know", the test example is "difficult" to classify, based on the information we have so far. We use $V \subseteq \mathcal{H}$, a version space or candidate set of h^*, as a proxy for our current knowledge about h^*. That is, we know h^* is in V, and any classifier in V may potentially be h^*. V can be obtained by performing calculations of error confidence bounds, as is done in e.g. Algorithm 3. A confidence-rated predictor P is a randomized mapping from $\mathcal{X}$ to $\{0, 1, \perp\}$. Alternatively, it can be represented as a 3-tuple of functions $(\zeta(x), \xi(x), \gamma(x))$, where each coordinate represents the probability of outputting 0, 1 or $\perp$ given x(thus $\zeta(x) + \xi(x) + \gamma(x) \equiv 1$ and $\zeta(x), \xi(x), \gamma(x) \geq 0$). For a given P, we focus on two performance metrics. The first one is its error, the probability that P outputs a label in $\{0, 1\}$ that is different from y. Formally,

$$\mathrm{err}(P) = \mathbb{P}(P(x) \neq y, P(x) \neq \perp)$$

Where the probability is taken over the randomness of the predictor P, and $(x, y) \sim D$. Of course, we can trivially minimize the error of a confidence-rated predictor by outputting $\perp$ on all the test examples x. To avoid such degeneracy, a notion of abstention, the probability that P outputs $\perp$(which is 1 minus its *coverage* in the notation of [EYW10]) is also considered. Formally,

$$\mathrm{abs}(P) = \mathbb{P}(P(x) = \perp)$$

Typically, there is a tradeoff between the two metrics. When more errors are allowed to be made, the abstention may decrease. Equivalently, if we are forced to output a small fraction of "Don't know"'s, the error is likely to be high. We would like to get a confidence-rated predictor with guaranteed error, that is $\mathbb{P}(P(x) \neq y, P(x) \neq \perp) \leq \eta$ for a given $\eta \geq 0$. However, as also noted by [EYW11], this problem is difficult. Instead, we focus on those confidence-rated predictors with guaranteed disagreement with h^*, in the sense that the probability of $P(x)$ outputting a different label from $h^*(x)$ is controlled. That is, $\mathbb{P}(P(x) \neq h^*(x), P(x) \neq \perp) \leq \eta$ for a given $\eta \geq 0$. Note in realizable case, this exactly corresponds to error guarantee η. In agnostic case, this suffice to ensure that $\mathbb{P}(P(x) \neq y, P(x) \neq \perp) - \mathbb{P}(h^*(x) \neq y, P(x) \neq \perp) \leq \eta$, which is related to the *weakly optimal* condition proposed by [EYW11]. Finally, we remark that if we predict $\perp$ if $x \in \mathrm{DIS}(V)$, and predict $h(x)$ if $x \notin \mathrm{DIS}(V)$ using an arbitrary classifier $h \in V$, then we essentially recover the *Consistent Selective Strategy* of [EYW10, EYW11]. Such strategy guarantees that the probability of disagreement with h^* is 0.

Consider the following problem: suppose an adversary is allowed to choose an arbitrary h^* inside V to label all the examples in $\mathcal{X}$, can we design a confidence-rated predictor guaranteeing disagreement with h^* at most η, with small abstention probability? This viewpoint turns out to be crucial in the design of our new active learning algorithm, as we will see shortly.

Let us look at the following optimization problem:

$$\begin{aligned} &\min \mathbb{E}_D \gamma(x) && && (1)\\ &\text{s.t. } \mathbb{E}_D[I(h(x)=1)\zeta(x) + I(h(x)=0)\xi(x)] \leq \eta, && \text{for all } h \in V \\ &\quad \gamma(x) + \xi(x) + \zeta(x) \equiv 1 \\ &\quad \gamma(x), \xi(x), \zeta(x) \geq 0 \end{aligned}$$

It can be seen that the first group of constraints ensure that $\mathbb{P}(P(x) \neq h^*(x), P(x) \neq \perp)$ is at most η, while the second and the third constraints ensure that the predictor outputs labels according to a proper distribution on $\{0, 1, \perp\}$. Suppose $(\zeta^*(x), \xi^*(x), \gamma^*(x))$ is an optimal solution, we define a confidence-rated predictor P^* as the one that correspond to $(\zeta^*(x), \xi^*(x), \gamma^*(x))$. We can see from the construction that $P^*(x)$ is optimal, i.e. for any other confidence-rated predictors $P'(x)$ guaranteeing disagreement with h^* at most η(Recall that h^* can be chosen adversarially from V), $\text{abs}(P^*) \leq \text{abs}(P')$. This optimal abstention probability is denoted by $\Phi(V, \eta)$.

It is not even clear if the optimization problem above is solvable using finite samples drawn from D. Interestingly, in *transductive* setting, given an unlabelled set U, it is possible to design a confidence-rated predictor on U with error guarantee η.[4] In particular, upon test, we are first given unlabelled examples $U = \{x_1, \dots, x_m\}$ drawn iid from $D_{\mathcal{X}}$ and we are asked to predict the labels of all the examples in U all at once. Since we only need to make decisions on examples in U, P_U is defined as a randomized mapping from U to $\{0, 1, \perp\}$. Specifically $P_U(x_i)$ corresponds to a 3-tuple $(\zeta_i, \xi_i, \gamma_i)$, for each i. In this scenario, the error and abstention are evaluated with respect to the uniform distribution of U. Formally, $\text{err}_U(P_U) = \mathbb{P}_{(x,y)\sim U}(P_U(x) \neq y, P_U(x) \neq \perp) = \frac{1}{m}\sum_{i=1}^m [I(h^*(x_i) = 1)\zeta_i + I(h^*(x_i) = 0)\xi_i]$(where we slightly abuse the notation to denote $(x, y) \sim U$ as (x, y) drawn from uniform distribution on U), and $\text{abs}_U(P_U) = \mathbb{P}_{(x,y)\sim U}(P_U(x) \neq \perp) = \frac{1}{m}\sum_{i=1}^m \gamma_i$. Then, optimization problem (1) simplifies to a linear program, described in the following algorithm:

Algorithm 5 Proposed Confidence-Rated Predictor

```
**Inputs:** Unlabelled test examples $U = \{x_1, \dots x_m\}$, candidate set $V$.
Solve the linear program:
```

$$\begin{aligned} &\min \frac{1}{m}\sum_{i=1}^m \gamma_i && && (2)\\ &\text{s.t. } \frac{1}{m}\sum_{i=1}^m [I(h(x_i)=1)\zeta_i + I(h(x_i)=0)\xi_i] \leq \eta, && \text{for all } h \in V \\ &\quad \gamma_i + \xi_i + \zeta_i = 1, && \text{for all } i \\ &\quad \gamma_i, \xi_i, \zeta_i \geq 0 && \text{for all } i \end{aligned}$$

```
Return the confidence-rated predictor corresponding to $(\zeta_i, \xi_i, \gamma_i)_{i=1}^m$.
```

Although the number of constraints in the first group is infinite, since V is a hypothesis class with VC dimension d, there are at most $(em/d)^d$ distinguishable constraints. Therefore the linear program has essentially finite constraints, thus can be solved in principle. Suppose $\{(\xi_i^*, \zeta_i^*, \gamma_i^*)\}_{i=1}^m$ is an optimal solution of the linear program, denote P_U^* as the confidence-rated predictor over U corresponding to this solution, and the optimal objective value as $\Phi_U(V, \eta)$. Standard VC inequalities implies that if a sufficiently large number of examples are drawn from D to form U, then the absention of P_U^* over U with error guarantee η will be close to the abstention of P^* over D with slightly larger error guarantee. Essentially, this implies that the error-abstension tradeoff behavior under uniform distribution over U is close to that of D. The fact above is summarized in the following lemma.

[4] This is different from [EYW10], where only one single test example drawn from D is considered. This subtle difference may result in significant reduction of abstension probability as we will see in the next subsections.

Lemma 6. *Given a hypothesis set $V \subseteq \mathcal{H}$. Suppose $m \geq O(\frac{1}{\lambda^2}(d \ln \frac{1}{\lambda} + \ln \frac{1}{\delta}))$ unlabelled examples are drawn from $D_{\mathcal{X}}$. Then with probability $1 - \delta$ over the random draw of the unlabelled dataset $U \sim D^m$,*

$$\Phi_U(V, \eta + \lambda) \leq \Phi(V, \eta) + \lambda$$

As we will see in our proposed active learning algorithm (Algorithm 6), in each epoch k, it is enough to first draw a large pool of unlabelled examples U as a proxy for the true underlying distribution $D_{\mathcal{X}}$, then reduce our problem to learn a candidate set V_k with excess error ϵ_k, with respect to the uniform distribution over U in conjunction with $D_{Y|X}$.

4.2 Active Learning via Perfect Confidence-rated Prediction

In the realizable case, it is shown by [EYW12] that active learning can be reduced to perfect confidence-rated prediction, i.e. confidence-rated predicton with error guarantee 0. Consider running CAL on a stream of training examples. Then $\mathbb{P}(\text{DIS}(V_{t-1}))$, the probability of querying the t th example given information up to time $t-1$, is at least $\Phi(V_{t-1}, 0)$. This can be checked by first pick an arbitrary h in V_{t-1} and define $\zeta(x) = I(x \notin \text{DIS}(V_{t-1}) \wedge h(x) = 0)$, $\xi(x) = I(x \notin \text{DIS}(V_{t-1}) \wedge h(x) = 1)$, $\gamma(x) = I(x \in \text{DIS}(V_{t-1}))$, and let P be the confidence-rated predictor corresponding to $(\zeta(x), \xi(x), \gamma(x))$.

By standard VC inequalities, if $\theta(\epsilon) \leq \text{polylog}(\frac{1}{\epsilon})$, then $\mathbb{P}(\text{DIS}(V_{t-1})) = O(\frac{\text{polylog}(t)}{t})$. It is noted in [EYW12, Han12] that the converse is also true. This fact are used by [EYW12] to prove examples of the disagreement coefficient being $\text{polylog}(1/\epsilon)$, using a new technique called version space compression. Note that under such conditions the label complexity of CAL is $O(\text{polylog}(\frac{1}{\epsilon}))$.

The work of [EYW10, EYW12] makes it transparent that in realizable case, if a finite sample upper bound of abstention probability can be established by a confidence-rated predictor with error guarantee 0(either by analyzing the geometry of the disagreement ball, or by version space compression), then a label complexity upper bound for CAL can be obtained. This raises two questions: 1. can the same analysis be carried out for designing agnostic active learning algorithms? 2. can some confidence-rated predictor with nonzero error guarantee help designing active learning algorithms, possibly with better label complexity? The first question has been partially addressed by the large body of literature on disagreement-based active learning. For the second question, we provide an answer in the next subsection.

4.3 Active Learning via Imperfect Confidence-rated Prediction

We introduce our active learning algorithm, namely Algorithm 6 in this subsection. The algorithm proceeds in epochs. Our claim is that the algorithm makes progress as the number of iterations increases, and h^* is always kept in the candidate set, therefore ensuring consistency.

Algorithm 6 An Agnostic Active Learning Algorithm based on Confidence-rated Prediction

```
Inputs: Excess error guaratee $\epsilon$, failure probability $\delta$.
Initialize $V_0 = \mathcal{H}$.
for $k = 1, 2, \ldots, k_0 = \lceil \log \frac{1}{\epsilon} \rceil$ do
    Draw $n_k$ unlabelled examples $U_k = \{x_1, \ldots, x_{n_k}\}$ from $D_{\mathcal{X}}$.
    Run Algorithm 5 with respect to $V_{k-1}$ over $U_k$ with error guarantee $\epsilon_k/32$, to get $\{\xi_i, \zeta_i, \gamma_i\}_{i=1}^{n_k}$. $\phi_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \gamma_i$.
    Denote as $\Gamma_k$ the normalized distribution with weight $\{\gamma_i\}_{i=1}^{n_k}$. Define $\tilde{\Gamma}_k$ as the joint distribution over $\mathcal{X} \times \mathcal{Y}$ which has marginal $\Gamma_k$ over $\mathcal{X}$ and conditonal probability $D_{Y|X=x}$.
    Call Algorithm 7 with input distribution $\tilde{\Gamma}_k$, candidate set $V_{k-1}$, target excess error $\epsilon_k/8\phi_k$, failure probability $\delta_k$, to get a new candidate set $V_k$.
Return an arbitrary hypothesis $h$ in $V_{k_0}$.
```

Lemma 7. *Suppose Algorithm 6 is run with inputs example oracle $\mathcal{U}$, labelling oracle $\mathcal{O}$, hypothesis class $\mathcal{H}$, confidence-rated predictor P, target excess error ϵ and failure probability δ. Then with probability $1-\delta$, for all $k = 1, 2, \dots, k_0$,*

(1) for all $h \in V_k$, $err(h) - err(h^*) \leq \epsilon_k$.

(2) $h^* \in V_k$.

In particular, the $\hat{h}$ returned at the end of the algorithm satisfies $err(\hat{h}) - err(h^*) \leq \epsilon$.

In epoch k, we would like to sample over D conditioned on a subset of $\mathcal{X}$. It turns out that the smaller this subset is(with respect to the measure $D_{\mathcal{X}}$), the sharper label complexity bound we may hope to achieve. One particular choice of the regionis $\text{DIS}(V_{k-1})$, which makes this procedure a disagreement-based active learning algorithm. Perhaps not surprisingly, the approach achieves the same label complexity upper bounds given by Algorithm 3, as in Theorems 4 and 5, up to logarithmic factors.

Recall that as shown by [BBZ07, BL13], in the case of $\mathcal{H}$ being the class of homogeneous linear classifiers and $D_{\mathcal{X}}$ isotropic log-concave on $\mathbb{R}^d$, in order to output a classifier h_k in epoch k such that $\text{err}(h_k) - \text{err}(h^*) \leq \epsilon_k$, it is possible to subsample and query on a set different from $\text{DIS}(B(h_{k-1}, r_k))$ that has considerably smaller probability measure with respect to D. Our main contribution can be seen as generalizing this idea to arbitrary $(\mathcal{H}, D)$ by employing a confidence-rated predictor in transductive setting.

The algorithm has two important components: a *selection* procedure based on confidence-rated predictor by Algorithm 5 and a *label query* procedure, namely Algorithm 7, to determine the number of examples needed to get a version space that has target excess error ϵ_k in epoch k, for Algorithm 6 to proceed. We discuss each component in turn.

First, to make the idea of selection implementable, in epoch k we first draw a fresh pool of unlabelled examples U_k as a proxy of the underlying distribution D. By standard VC inequalities, for all $h \in \mathcal{H}$, $\text{err}_{\tilde{U}_k}(h)$ is essentially the same as $\text{err}(h)$ when n_k is sufficiently large. Thus it suffice to get a candidate set V_k tight enough with respect to $\tilde{U}_k$, the distribution on $\mathcal{X} \times \mathcal{Y}$ such that its marginal over $\mathcal{X}$ and conditional distribution of Y given X are U_k and $D_{Y|X}$, respectively. Then a confidence-rated predictor with guaranteed disagreement with h^* with probability η_k on the unlabelled set U_k is run, outputting the "abstention" probability $\gamma_1, \dots, \gamma_{n_k}$. Intuitively the weighted set $\{(\gamma_1, x_1), \dots, (\gamma_{n_k}, x_{n_k})\}$ represent the "hard-core" of our learning goal in epoch k, that is, outside the abstention region, any pairs of classifier $h, h' \in V_k$ are approximately indistinguishable(cf. exactly indistinguishable, if $\text{DIS}(V_{k-1}) \cap U_k$ were selected). $\tilde{\Gamma}_k$ is the normailized distribution over U_k with weight $\gamma_i, i = 1, \dots, n_k$. Note that $\tilde{\Gamma}_k$ can be significantly different from the uniform distribution over $\text{DIS}(V_{k-1}) \cap U_k$(which is usually considered in disagreement-based algorithms). Therefore, some bias are introduced – the optimal classifier with respect to $\tilde{\Gamma}_k$ may not necessarily be h^*! The bias comes from two sources. One is that we use the distribution $\tilde{U}_k$ to approximate D, which we have addressed before. The other is that we use a confidence-rated predictor to focus on learning with respect to distribution $\tilde{\Gamma}_k$. On one hand η_k should not be too large, otherwise a large bias will be introduced. This may result in either V_k excluding h^*, or V_k being not tight enough for the algorithm to proceed. On the other hand, η_k should not be small, otherwise there will be no significant improvement over disagreement-based active learning. In fact, the choice of parameters $n_k = O(\frac{1}{\epsilon_k^2}(d \ln \frac{1}{\epsilon_k} + \ln \frac{1}{\delta_k}))$ and $\eta = \epsilon_k/32$ suffice for our purposes. In this case, h^* will still be approximately optimal, in the sense that $\text{err}_{\tilde{\Gamma}_k}(h^*) \leq \inf_{h \in V_{k-1}} \text{err}_{\tilde{\Gamma}_k}(h) + \frac{\epsilon_k}{16\phi_k}$.

The second important component is Algorithm 7, which is an empirical Bernstein stopping-like procedure. It will be called by Algorithm 6 in each epoch k, based on subsampling on $\tilde{\Gamma}_k$ to prune the version space from V_{k-1} to V_k such that $\{h \in V_{k-1} : \text{err}_{\tilde{\Gamma}_k}(h) - \inf_{h \in V_{k-1}} \text{err}_{\tilde{\Gamma}_k}(h) \leq \epsilon_k/16\phi_k\} \subseteq V_k \subseteq \{h \in V_{k-1} : \text{err}_{\tilde{\Gamma}_k}(h) - \inf_{h \in V_{k-1}} \text{err}_{\tilde{\Gamma}_k}(h) \leq \epsilon_k/8\phi_k\}$. This guarantees that $h^* \in V_k \subseteq \{h \in \mathcal{H} : \text{err}(h) - \text{err}(h^*) \leq \epsilon_k\}$, which completes the induction in the proof of Lemma 7. An important property of Algorithm 7 is that the number of label queries it results in is *adaptive* to different noise conditions of Π, which is crucial to the label complexity analysis.[5] The number of label queries are characterized by the following two lemmas under two separate assumptions of $(V_{k-1}, \tilde{\Gamma}_k)$ that are of particular interest in our analysis.

[5] A much more general idea has appeared in [Kol10], where not only VC class but also a broader class satisfying entropy conditon are taken into account.

Algorithm 7 An Adaptive Procedure to get a Candidate Set with Target Excess Error

```
**Inputs:** Data distribution $\Pi$, candidate set $V$, excess error guaratee $\tilde{\epsilon}$, failure probability $\tilde{\delta}$.
**Outputs:** a candidate set $V'$ such that $\{h \in V : \text{err}_\Pi(h) - \inf_{h\in V} \text{err}_\Pi(h) \leq \tilde{\epsilon}/2\} \subseteq V' \subseteq \{h \in V : \text{err}_\Pi(h) - \inf_{h\in V} \text{err}_\Pi(h) \leq \tilde{\epsilon}\}$
**for** $k = 1, 2, \dots$ **do**
    Draw $2^k$ labelled examples $S_k$ from $\Pi$.
    Train an ERM classifier $\hat{h}_k = \text{argmin}_{h\in V}\text{err}_{S_k}(h)$.
    Define $V_k = \{h \in V : \text{err}_{S_k}(h) - \text{err}_{S_k}(\hat{h}_k) \leq \sigma(n_k, \tilde{\delta}_k) + \sqrt{\sigma(n_k, \tilde{\delta}_k)\rho_{S_k}(h, h_k)}\}$.
    **if** $\sup_{h\in V_k} \sigma(n_k, \tilde{\delta}_k) + \sqrt{\sigma(n_k, \tilde{\delta}_k)\rho_{S_k}(h, h_k)} \leq \tilde{\epsilon}/2$ **then**
        **break**
Return $V_k$.
```

Lemma 8. *Suppose Algorithm 7 is run under distribution* Π*, candidate set* V*, excess error guarantee* $\tilde{\epsilon}$*, failure probability* $\tilde{\delta}$*, where* (V, Π) *satisfies* $\inf_{h\in V} err_\Pi(h) = \nu^*(\Pi)$*. Then with probability* $1 - \tilde{\delta}$*,*

(1) The new candidate set V' *satisfies*

$$\{h \in V : err_\Pi(h) - \inf_{h\in V} err_\Pi(h) \leq \tilde{\epsilon}/2\} \subseteq V' \subseteq \{h \in V : err_\Pi(h) - \inf_{h\in V} err_\Pi(h) \leq \tilde{\epsilon}\}$$

(2) The total number of labelled examples drawn is at most

$$\tilde{O}(d \ln \frac{1}{\tilde{\epsilon}} \cdot \frac{\nu^*(\Pi) + \tilde{\epsilon}}{\tilde{\epsilon}^2})$$

Lemma 9. *Suppose Algorithm 7 is run under distribution* Π*, candidate set* V*, excess error guarantee* $\tilde{\epsilon}$*, failure probability* $\tilde{\delta}$*, where* (V, Π) *satisfies* $(C, \kappa, \tilde{\epsilon})$*-Tsybakov noise condition, that is, there is a hypothesis* $\tilde{h}$ *in* V *such that* $\rho(h, \tilde{h}) \leq C \max(\tilde{\epsilon}, err(h) - err(\tilde{h}))^{\frac{1}{\kappa}}$*. Then with probability* $1 - \tilde{\delta}$*,*

(1) The new candidate set V' *satisfies*

$$\{h \in V : err_\Pi(h) - \inf_{h\in V} err_\Pi(h) \leq \tilde{\epsilon}/2\} \subseteq V' \subseteq \{h \in V : err_\Pi(h) - \inf_{h\in V} err_\Pi(h) \leq \tilde{\epsilon}\}$$

(2) The total number of labelled examples drawn is at most

$$\tilde{O}(\max(d\tilde{\epsilon}^{-1} \ln(\tilde{\epsilon}^{-1}), dC\tilde{\epsilon}^{\frac{1}{\kappa}-2} \ln(C\tilde{\epsilon}^{\frac{1}{\kappa}-2})))$$

In light of these two components, the label complexity of Algorithm 6 can be argued as follows. First, in epoch k, by Lemma 6, $\phi_k \leq \Phi(V_{k-1}, \epsilon_k/64)$. This ensures that the target error $\epsilon_k/8\phi_k$ is not too small. Second, we establish some properties of the subpopulation $\tilde{\Gamma}_k$. If the error of h^* with respect to D is ν, then the error of the optimal classifier within V_{k-1} with respect to $\tilde{\Gamma}_k$ is $O(\frac{\nu+\epsilon}{\phi_k})$. If $(\mathcal{H}, D)$ satisfies (C, κ)-Tsybakov noise condition, then $(V_{k-1}, \tilde{\Gamma}_k)$ satisfies $(O(\phi_k^{\frac{1}{\kappa}-1}), \kappa, \Omega(\frac{\epsilon_k}{\phi_k}))$-approximate Tsybakov noise condition. Third, we call Lemmas 8 and 9 respectively to compute the number of label queries in epoch k. Finally we sum the number of label queries in each epoch k up. Our main results are summarized in the following two theorems.

Theorem 8. *Suppose Algorithm 6 is run with excess error guarantee* ϵ *and failure probability* δ*.* $err(h^*) = \nu$*. Then with probability* $1 - \delta$*, the output* $\hat{h}$ *satisfies* $err(\hat{h}) - err(h^*) \leq \epsilon$*. The number of label queries is at most*

$$\tilde{O}(\sup_{r\geq\epsilon} \frac{\Phi(B(h^*, 2\nu + r), r/256)}{2\nu + r} \cdot d \ln \frac{1}{\epsilon} \cdot (\ln \frac{1}{\epsilon} + \frac{\nu^2}{\epsilon^2}))$$

Theorem 9. *Suppose Algorithm 6 is run with excess error guarantee ϵ and failure probability δ. $(\mathcal{H}, D)$ satisfies (C, κ)-Tsybakov noise condition. Then with probability $1-\delta$, the output $\hat{h}$ satisfies $err(\hat{h}) - err(h^*) \leq \epsilon$. The number of labels queries is at most*

$$\tilde{O}(\sup_{r \geq \epsilon} \frac{\Phi(B(h^*, Cr^{\frac{1}{\kappa}}), r/256)}{r^{\frac{1}{\kappa}}} \cdot d \ln^2 \frac{1}{\epsilon} \cdot \epsilon^{\frac{2}{\kappa}-2})$$

We remark that since $\Phi(\mathrm{B}(h^*, r), \eta) \leq \Phi(\mathrm{B}(h^*, r), 0) \leq \mathbb{P}(\mathrm{DIS}(\mathrm{B}(h^*, r)))$, Theorem 8 implies that the label complexity is always no worse than $\tilde{O}(\theta(2\nu^* + \epsilon) \cdot d \ln \frac{1}{\epsilon} \cdot (\frac{\nu^2}{\epsilon^2} + \ln \frac{1}{\epsilon}))$ when $\mathrm{err}(h^*) = \nu^*$. Similarly, Theorem 9 implies that the label complexity is always not worse than $\tilde{O}(C\theta(Cr^{\frac{1}{\kappa}}) \cdot d \ln \frac{1}{\epsilon} \cdot \epsilon^{\frac{2}{\kappa}-2})$. Therefore, up to logarithmic factors, our bound is always no worse than the bounds known for disagreement-based algorithms.

Specifically, if $\mathcal{H}$ is the set of homogeneous linear classifiers, and $D_{\mathcal{X}}$ is isotropic log-concave in $\mathbb{R}^d$, our label complexity bounds essentially recovers the results of [BL13], which is generally a $O(\sqrt{d})$ multiplicative factor improvement compared to known upper bounds for disagreement-based algorithms since $\theta(r) = \tilde{O}(\sqrt{d} \ln \frac{1}{\epsilon})$.

Corollary 2. *Suppose Algorithm 6 is run with excess error guarantee ϵ and failure probability δ. $D_{\mathcal{X}}$ is isotropic log-concave on $\mathbb{R}^d$. $\mathcal{H}$ is the class of homogeneous linear classifiers in $\mathbb{R}^d$. Then the following results holds.*
(1) Suppose $\inf_{h \in \mathcal{H}} err(h^*) = \nu^*$, then with probability $1-\delta$, the number of labels requested is at most

$$\tilde{O}(d \ln^2 \frac{1}{\epsilon} \cdot (\ln \frac{1}{\epsilon} + \frac{\nu^2}{\epsilon^2}))$$

(2) Suppose $(\mathcal{H}, D)$ satisfies (C, κ)-Tsybakov noise condition, then with probability $1-\delta$, the number of labels requested is at most

$$\tilde{O}(d \ln^3 \frac{1}{\epsilon} \cdot \epsilon^{\frac{2}{\kappa}-2})$$

5 Conclusion and Open Questions

By utilizing a novel notion of uncertainty based on confidence-rated predictor with guaranteed error, we have proposed an aggressive agnostic active learning algorithm. This opens up several interesting research directions, such as:

- Find other settings of $(\mathcal{H}, D)$ that confidence-based active learning algorithms can offer asymptotically lower label requirements compared to known disagreement-based active learning algorithms.
- In epoch k of Algorithm 6, we crucially rely on that our confidence-rated predictor P satisfying $\mathbb{P}(P(x) \neq h(x), P(x) \neq \perp) \leq O(\eta_k)$ for all $h \in V_{k-1}$. Then triangle inequality is applied to ensure for all h in V_{k-1}, $\mathbb{P}(h(x) \neq y, P(x) \neq \perp) - \mathbb{P}(h^*(x) \neq y, P(x) \neq \perp) \leq O(\eta_k)$. Can we design agnostic algorithms that can directly guarantee the latter?
- Design computationally efficient alternatives of the algorithm. Note that Algorithm 6 crucially relies on keeping a candidate set V_k in epoch k to ensure consistency, which is computationally intractable. Can implicit candidate set representation, e.g. the ones in [DHM07, Han09, BHLZ10] help in reducing computational costs?

Acknowledgement

The author would like to thank his research exam committee: Vineet Bafna, Sanjoy Dasgupta and Lawrence Saul for their time and thoughtful comments. The author also thanks his advisor, Kamalika Chaudhuri for the guidance and collaboration on this work, along with help through the preparation of the exam. Thanks are also due to the author's undergraduate advisor Liwei Wang for introducing this topic and instructive discussions.

References

[ABL14] P. Awasthi, M-F. Balcan, and P. M. Long. The power of localization for efficiently learning linear separators with noise. In *STOC*, 2014.

[Ang87] Dana Angluin. Queries and concept learning. *Machine Learning*, 2(4):319–342, 1987.

[Ang04] Dana Angluin. Queries revisited. *Theor. Comput. Sci.*, 313(2):175–194, 2004.

[BBL09] M.-F. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. *J. Comput. Syst. Sci.*, 75(1):78–89, 2009.

[BBZ07] M.-F. Balcan, A. Z. Broder, and T. Zhang. Margin based active learning. In *COLT*, 2007.

[BHLZ10] A. Beygelzimer, D. Hsu, J. Langford, and T. Zhang. Agnostic active learning without constraints. In *NIPS*, 2010.

[BL13] M.-F. Balcan and P. M. Long. Active and passive learning of linear separators under log-concave distributions. In *COLT*, 2013.

[BZ74] M. V. Burnashev and K. Sh. Zigangirov. An interval estimation problem for controlled observations. *Problems Inform. Transmission*, 10:223–231, 1974.

[CAL94] D. A. Cohn, L. E. Atlas, and R. E. Ladner. Improving generalization with active learning. *Machine Learning*, 15(2), 1994.

[CCG11] Giovanni Cavallanti, Nicolò Cesa-Bianchi, and Claudio Gentile. Learning noisy linear classifiers via adaptive and selective sampling. *Machine Learning*, 83(1):71–102, 2011.

[CGZ04] Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. Worst-case analysis of selective sampling for linear-threshold algorithms. In *Advances in Neural Information Processing Systems 17 [Neural Information Processing Systems, NIPS 2004, December 13-18, 2004, Vancouver, British Columbia, Canada]*, 2004.

[CN07] Rui Castro and Robert D. Nowak. Minimax bounds for active learning. In *COLT*, pages 5–19, 2007.

[Das05] S. Dasgupta. Coarse sample complexity bounds for active learning. In *NIPS*, 2005.

[DGS12] Ofer Dekel, Claudio Gentile, and Karthik Sridharan. Selective sampling and active learning from single and multiple teachers. *Journal of Machine Learning Research*, 13:2655–2697, 2012.

[DH08] S. Dasgupta and D. Hsu. Hierarchical sampling for active learning. In *ICML*, 2008.

[DHM07] S. Dasgupta, D. Hsu, and C. Monteleoni. A general agnostic active learning algorithm. In *NIPS*, 2007.

[EYW10] R. El-Yaniv and Y. Wiener. On the foundations of noise-free selective classification. *JMLR*, 2010.

[EYW11] R. El-Yaniv and Y. Wiener. Agnostic selective classification. In *NIPS*, 2011.

[EYW12] R. El-Yaniv and Y. Wiener. Active learning via perfect selective classification. *JMLR*, 2012.

[FSST97] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby. Selective sampling using the query by committee algorithm. *Machine Learning*, 28(2-3):133–168, 1997.

[Han07] S. Hanneke. A bound on the label complexity of agnostic active learning. In *ICML*, 2007.

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

[Han12] Steve Hanneke. Activized learning: Transforming passive to active with improved label complexity. *Journal of Machine Learning Research*, 13:1469–1587, 2012.

[Han13] S. Hanneke. A statistical theory of active learning. Manuscript, 2013.

[Kää06] M. Kääriäinen. Active learning in the non-realizable case. In *ALT*, 2006.

[KMT93] Sanjeev R. Kulkarni, Sanjoy K. Mitter, and John N. Tsitsiklis. Active learning using arbitrary binary valued queries. *Machine Learning*, 11:23–35, 1993.

[Kol10] V. Koltchinskii. Rademacher complexities and bounding the excess risk in active learning. *JMLR*, 2010.

[MT99] E Mammen and A. B. Tsybakov. Smooth discrimination analysis. *Annals of Statistics*, 27:1808–1829, 1999.

[Now11] R. D. Nowak. The geometry of generalized binary search. *IEEE Transactions on Information Theory*, 57(12):7893–7906, 2011.

[SC08] Burr Settles and Mark Craven. An analysis of active learning strategies for sequence labeling tasks. In *EMNLP*, pages 1070–1079, 2008.

[TK01] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. pages 45–66, 2001.

[Tsy04] A. B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *Annals of Statistics*, 32:135–166, 2004.

[VC71] V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16(2):264–280, 1971.

[WS14] Yining Wang and Aarti Singh. Noise-adaptive margin-based active learning for multi-dimensional data. *CoRR*, abs/1406.5383, 2014.