**CSE599i: Online and Adaptive Machine Learning** **Winter 2018**

# Lecture 14: Binary Classification: Disagreement-based Methods

*Lecturer: Kevin Jamieson Scribes: N. Cano, O. Rafieian, E. Barzegary, G. Erion, S. Fatehi, K. Ehsani*

**Disclaimer**: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

## 1 Introduction

Binary classification has been one of the most prevalent problems in machine learning, as it relates to a wide range of empirical problems (*e.g.*, whether or not a user clicks on an ad, likes a photo on social networks, etc). In binary classification, we are given a set of $n$ examples $\{(x_i, y_i)\}_{i=1}^n$, where $x_i \in \mathbb{R}^d$ is the feature vector of the $i^{th}$ example and $y_i \in \{-1, 1\}$ is its label, distributed i.i.d. from some distribution $\mathcal{D}_{XY}$, and the goal of the classifier is to infer the label of unseen examples.

Another important property of a classifier is how fast it learns. That is, for a given hypothesis space, we want to see how fast the error goes down as a function of the number of observations $n$. In this lecture, we focus on the problem of binary classification and examine two approaches to tackle the problem: passive and active learning. In passive learning, the goal of the learner is to infer an accurate predictor from *labeled* training data. The labeled training data are examples of input-output pairs $(x, y)$, where the label $y$ represents the outcome associated with the input $x$. In this setting, the supervised algorithm would obtain the labels for a *random* subset of inputs and learn the classifier. In many situations, however, obtaining labels can be costly and time consuming. For example, in speech recognition, the speech signal is cheap to obtain but labeling would require a lot of time and labor force.

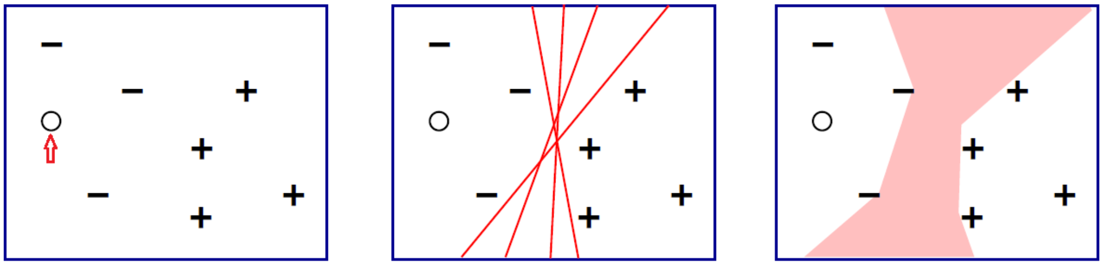
This gives rise to the notion of active learning, which instead, considers situation wherein we are given a set of *unlabeled* data points, (i.e., each training example is an input $x$ without an associated label $y$), and the learner is allowed to request the label $y$ of any particular input $x$. The goal of the active learner is to infer an accurate classifier of labels from inputs without making too many queries. An active learner tries to get the most out of a limited budget by choosing its query points (points for which it asks for the label) in an intelligent and adaptive manner [1].

In this lecture, we start with the problem of passive learning in a realizable setting, where we assume a perfect classifier exists and the data is separable. We examine how fast the error goes down as we increase the sample size. We then consider a particular active learning setting wherein unlabeled data come as a stream, and for each incoming data point, the learner needs to decide whether to query its label or not. We introduce an active learning algorithm proposed by Cohn, Atlas, and Ladner (CAL henceforth) [2] and see how much we can improve the convergence rate as compared with the passive one.

CAL is an example of a broader set of algorithms called disagreement-based methods. The main idea behind these methods is that the active learner maintains a candidate set of hypotheses (often called a version space), and queries the label of a data point only if there is disagreement within this set on how to label the point. As such, each time a new label is seen, the current set of hypothesis that are still "in the running" shrinks. Figure 1, taken from Dasgupta [1], shows an example of a disagreement-based method (CAL) under the assumption that the data is linearly separable. As shown in Figure 1a, seven points were labeled and a new point arrives. Figure 1b shows some of the hypotheses in the current version space, indicating that all these lines are consistent with the labeled data seen so far. Combining all the lines in the version space, we can form the region of disagreement, as shown in Figure 1c. Since the new point does not lie in the disagreement region, the learner is able to infer its label.

Finally, we discuss the agnostic case, where a perfect separator does not exist and the best classifier has some noise. We introduce a disagreement-based algorithm proposed by Dasgupta, Hsu, and Monteleoni

(DHM henceforth) [3] that extends CAL to the agnostic case. We show some theorems on the properties of this algorithm.



(a) Arrival of a new point

(b) Consistent hypotheses

(c) Disagreement region

Figure 1: An Example of A Disagreement-Based Active Learning Algorithm, taken from Dasgupta [1].

## 2 Preliminaries

Before delving into the passive and active learning settings for binary classification, we first introduce a series of notations and definitions that are necessary to characterize the problem. Let $\mathcal{X}$ be the input space, $\mathcal{D}$ the joint distribution over $\mathcal{X} \times \{-1, 1\}$, and $\mathcal{H}$ a class of hypotheses $h : \mathcal{X} \to \{-1, 1\}$.

**Definition 1.** *If a finite dataset is identically and independently distributed (i.i.d.) from some distribution, i.e.,* $\{(x_i, y_i)\}_{i=1}^n \sim D_{X,Y}$*, then the true risk of a hypothesis* $h \in \mathcal{H}$ *is:*

$$R(h) = \Pr(h(X) \neq Y)$$

**Definition 2.** *For a hypothesis class* $\mathcal{H}$ *and dataset* $\{(x_i, y_i)\}_{i=1}^n$ *where* $x_i \in \mathcal{X}$ *and* $y_i \in \{-1, 1\}$*, we call* $\hat{h}$ *the empirical risk minimizer if* $\hat{h} = \arg\min_{h \in \mathcal{H}} \hat{R}_n(h)$ *where:*

$$\hat{R}_n(h) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{h(x_i) \neq y_i\}$$

For i.i.d. draws $\{x_i, y_i\}_{i=1}^n$, then the empirical risk $\hat{R}(h)$ is an unbiased estimator of the true risk $R(h)$ for any hypothesis $h \in \mathcal{H}$. Let $h^* = \arg\min_{h \in \mathcal{H}} R(h)$ be a hypothesis of minimum true risk in $\mathcal{H}$. Then, the goal of the learner is to return a hypothesis $h \in \mathcal{H}$ with true risk $R(h)$ as close as possible to the minimum true risk $R(h^*)$.

## 3 Passive Learning

We start with the problem of passive learning for binary classification. Herein, we have a set of data points and labels $\{(x_i, y_i)\}_{i=1}^n$ drawn i.i.d. from distribution $\mathcal{D}$. Our goal is to find how fast the true risk goes down as a function of $n$. Intuitively, the lower the risk we want, the more data points we need to sample. Further, there must be a positive relationship between the number of hypotheses we have in the hypothesis space and the number of data points we need to identify the hypothesis with the minimum true risk almost surely. The following theorem characterizes these relationships:

**Theorem 1.** *Suppose that we have a finite set of hypotheses* $\mathcal{H}$ *(i.e.,* $|\mathcal{H}| < \infty$*) and* $\hat{h}_n = \arg\min_{h \in \mathcal{H}} \hat{R}_n(h)$*. Also, assume that the data is separable (i.e., the perfect classifier* $h^*$ *with no error exists). For any* $\epsilon, \delta \in$

*$(0,1)$, we have $\Pr\big(R(\hat{h}_n) > \epsilon\big) \leq \delta$ whenever $n \geq \epsilon^{-1}\log(\frac{|\mathcal{H}|}{\delta})$. In other words, for any $\epsilon, \delta \in (0,1)$, with probability $1-\delta$, we have:*

$$R(\hat{h}_n) \leq \frac{\log(\frac{|\mathcal{H}|}{\delta})}{n}$$

Before going over the proof, it is worth noting that the theorem assumes a finite set of hypotheses in the hypothesis class $\mathcal{H}$. Such assumption, besides being practical from an optimization point of view, also simplifies the theory. Given that we can turn an infinite hypothesis class into a finite one just by discretizing the space, it is a reasonable assumption to make that will make the proof simpler. We refer the interested reader to Boucheron et al. [4] in order to better understand how to discretize the space using Vapnik-Chervonenkis (VC) dimension and Rademacher complexity.

*Proof.* First, we can show that $\hat{R}_n(\hat{h}_n) = 0$, since we have:

$$\hat{R}_n(\hat{h}_n) = \min_{h\in\mathcal{H}} \hat{R}_n(h) \leq \hat{R}_n(h^*) = 0$$

Now we can write:

$$\Pr\big(R(\hat{h}_n) > \epsilon\big) = \Pr\Big(\bigcup_{h\in\mathcal{H}} \{R(h) > \epsilon \wedge \ \hat{R}_n(h) = 0\}\Big) \leq \sum_{h\in\mathcal{H}} \Pr\big(\{R(h) > \epsilon \wedge \ \hat{R}_n(h) = 0\}\big), \tag{1}$$

where we performed a union bound to obtain the inequality. We can now find a bound for each element— $\Pr\big(\{R(h) > \epsilon, \ \hat{R}_n(h) = 0\}\big)$. This is the probability that a hypothesis with true risk greater than $\epsilon$ shows zero empirical risk in $n$ points drawn i.i.d. from $\mathcal{D}$. Since the true risk for this hypothesis is greater than $\epsilon$, the probability that this hypothesis correctly identifies a random point is lower than $1-\epsilon$. Thus, we can write:

$$\sum_{h\in\mathcal{H}} \Pr\big(\{R(h) > \epsilon \wedge \ \hat{R}_n(h) = 0\}\big) \leq \sum_{h\in\mathcal{H}} (1-\epsilon)^n \tag{2}$$

Finally, using the approximation $1 - x \leq e^{-x}$ for $x \geq 0$, we can write:

$$\sum_{h\in\mathcal{H}} (1-\epsilon)^n \leq |\mathcal{H}|e^{-\epsilon n} \tag{3}$$

Combining (1), (2), and (3), we have the following:

$$\Pr\big(R(\hat{h}_n) > \epsilon\big) \leq |\mathcal{H}|e^{-\epsilon n} = \delta \tag{4}$$

Solving for $n$, we find $n \geq \epsilon^{-1}\log(\frac{|\mathcal{H}|}{\delta})$, and this completes the proof. □

**Example 1.** *As a concrete example, let us assume $x$ being uniform on $[0,1]$, and that the hypothesis class is defined as $\mathcal{H} = \{sign(x - \frac{i-1}{m-1}) : i = 1, ..., m\}$; that is, there are $m$ classifiers uniformly spaced in our hypothesis class (i.e., $|\mathcal{H}| = m$). Figure 2 shows one of those classifiers (at $\frac{3}{m}$ threshold).*

*If $y = h^*(x)$ for some $h^* \in \mathcal{H}$ (i.e., the perfect classifier exists in the hypothesis class), then we can apply Theorem 1 and say that after $n$ observations with probability at least $1-\delta$:*

$$R(\hat{h}_n) \leq \frac{\log(m/\delta)}{n}$$

*Using the definition of expectation, the expected risk is given by:*

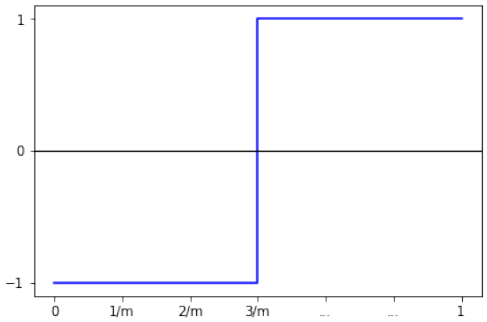


Figure 2: An Example of a Hypothesis

$$\mathbb{E}[R(\hat{h}_n)] = \int_{t=0}^{1} \Pr(R(\hat{h}_n) > t)dt$$

*In view of Theorem 1, we know that the probability term in RHS is bounded by* $me^{-tn}$. *Indeed, we know that probability cannot exceed 1. Thus, we can upper bound the probability term as follows:*

$$\begin{aligned}\mathbb{E}[R(\hat{h}_n)] &\leq \int_{0}^{1} \min\{1, me^{-tn}\}dt \\ &\leq \int_{0}^{\frac{\log(m)}{n}} 1dt + \int_{\frac{\log(m)}{n}}^{1} e^{-tn}dt \\ &\leq \frac{\log(m)}{n} + \int_{\frac{\log(m)}{n}}^{1} e^{-tn}dt \\ &\leq \frac{2\log(m)}{n}\end{aligned}$$

*which shows that the expected risk goes down with* $1/n$ *rate. We expect to improve it using some active (interactive) learning strategy, as we shall see next.*

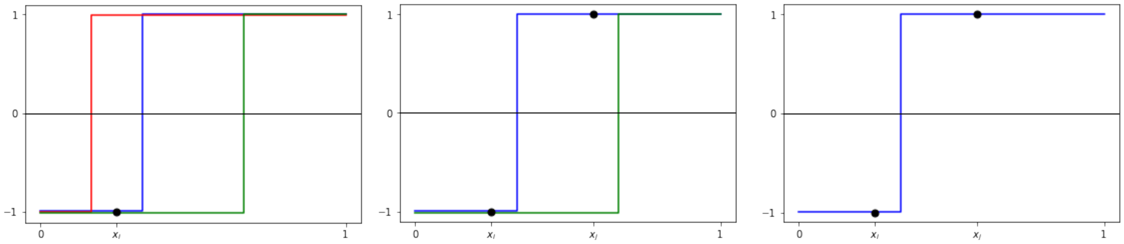
# 4 Active Learning

We now turn our attention into active learning. As mentioned before, the motivation behind active learning comes from the cases in which querying labels is costly. Thus, the main goal of an active learning algorithm is to identify the true hypothesis with fewer queries compared to passive learning. In particular, we focus on a streaming setting wherein unlabeled data come as a stream and the active learner has to decide whether to query for particular labels or not. An example of such streaming environments is spam detection where contents come as a stream and the algorithm classifies whether the content is spam or not, upon receiving any content. However, if the algorithm is uncertain about the label, it must ask for the label by showing the content to the user and getting her feedback.

In the active learning framework, we essentially want to see whether we can achieve the same risk as passive learning with lower number of labels requested. Before introducing any further notation, let us revisit Example 1 with the simple one-dimensional data points to give some intuition on why active learning might help. If the data is separable, its hidden labels are a sequence of $-$'s followed by a sequence of $+$'s. As such, when we get the label $-1$ for a data point, we can infer that all the points left to this point to be $-1$.

Therefore, we do not query for these points and save many potential queries that we would have made in passive learning.

Figure 3 illustrates this idea of successive elimination of the hypotheses in the hypothesis class. Suppose that we are left with three hypotheses in our hypothesis space and the learner receives the data point $x_i$ and asks for its label, which turns out to be $-1$ (Figure 3a). The learner can then rule out inconsistent hypotheses (in this case, the red classifier), and only keep the consistent ones (blue and green). Later in time, the learner asks for the label of point $x_j$ which is $+1$ (Figure 3b). The learner is now able to rule out even more hypotheses (in this case, the green classifier), which means that the space of possible classifiers keeps on shrinking. Over time, this shrinkage allows the learner to end up with a single best classifier (the blue classifier as shown in Figure 3c).



(a) Elimination of red hypothesis (b) Elimination of green hypothesis (c) Identifying the best hypothesis

Figure 3: Successive Elimination of Hypotheses in Active Learning

The above example assumes that the data is separable and that there exists a perfect classifier that makes no mistakes. This is known as the *realizable* setting. Besides this setting, there is also the *agnostic* case, in which we do not make the separability assumption (*i.e.*, the best separator could make mistakes). We show how active learning works in those settings as well.

## 4.1 Realizable Case

In this setting we assume there exists a hypothesis $h^* \in \mathcal{H}$ that correctly labels every example, that is, $Pr(h^*(X) = Y) = 1$. In other words, we assume that $h^*$ has no error, i.e., $R(h^*) = 0$.

### 4.1.1 Preliminaries

Here we include some definitions that we will use throughout this section.

**Definition 3.** *After observing n labels we call the set of all hypotheses still in the running for being the empirical risk minimizer over all observed data the version space and denote it $V_n \subset \mathcal{H}$. If $h^* = \arg\min_{h\in\mathcal{H}} R(h)$ and $R(h^*) = 0$, then*

$$V_n = \{h \in \mathcal{H} : h(x) = y \;\; \forall (x, y) \in Z\}$$

*the subset of hypotheses in $\mathcal{H}$ consistent with the examples in $Z$.*

**Definition 4.** *For some hypothesis class $\mathcal{H}$ and set $\mathcal{X}$ where for $h \in \mathcal{H}, h : \mathcal{X} \to \{-1, 1\}$, the region of disagreement is*

$$DIS(\mathcal{H}) = \{x \in \mathcal{X} : \exists\, h, h' \in \mathcal{H} \; s.t. \; h(x) \neq h'(x)\}$$

*the set of unlabeled examples $x$ for which there are hypotheses in $\mathcal{H}$ that disagree on how to label $x$.*

### 4.1.2 Algorithm: CAL

The CAL algorithm [2] (see Algorithm 1) proceeds by examining the unlabeled data points one at a time, and decides after each point $x_t$ whether to query its label $y_t$ or not. CAL decides to query $y_t$ if there is disagreement about how to label $x_t$ among the version space $V_t$. In other words, CAL chooses to query the label $y_t$ if and only if $x_t$ is in the region of disagreement $DIS(V_{t-1})$. Figure 4 shows an example of the disagreement region between the two classifiers.

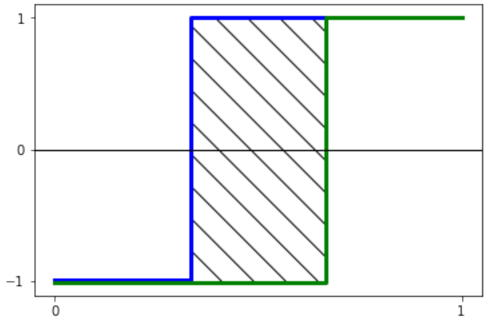


Figure 4: Region of Disagreement between the blue and green classifiers

**Algorithm 1** CAL

```
Initialize: $Z_0 = \emptyset$, $V_0 = \mathcal{H}$
for $t = 1, 2...n$ do
    Obtain unlabeled data point $x_t$
    if $\exists h, h' \in \mathcal{V}_{t-1}$ such that $h(x_t) \neq h'(x_t)$ then
        Query $y_t$, and set $Z_t = Z_{t-1} \cup (x_t, y_t)$
    else
        Set $\hat{y}_t = h(x_t)$ for any $h \in V_{t-1}$, and set $Z_t = Z_{t-1} \cup (x_t, \hat{y}_t)$
    end if
    Set $V_t = \{h \in \mathcal{H} : h(x_i) = y_i \;\; \forall (x_i, y_i) \in Z_t\}$
end for
return any $h \in V_n$
```

### 4.1.3 Disagreement Coefficient

CAL (and other disagreement-based methods in general) use a concept called the disagreement coefficient for analyzing the label complexity. The label complexity is the number of data points for which we query for their labels. It is an important concept since, as we mentioned in the introduction, asking for a data point's label can be costly.

**Definition 5.** *For a random variable $X \in \mathcal{X}$, the disagreement (pseudo) metric $\rho$ on $\mathcal{H}$ is defined by:*

$$\rho(h, h') = Pr(h(X) \neq h'(X))$$

Let $B(h, r) = \{h' \in \mathcal{H} : \rho(h, h') \leq r\}$ denote the closed ball centered at $h \in \mathcal{H}$ with radius $r$. The ball basically denotes all the hypotheses that are close to the hypothesis $h$ in the hypothesis space. Using this ball, we can now define the disagreement coefficient.

**Definition 6.** *The disagreement coefficient of $h \in \mathcal{H}$ with respect to a hypothesis class $\mathcal{H}$ and distribution $D_{X,Y}$ is defined as*

$$\theta_h = \sup_r \frac{Pr_x(DIS(B(h,r)))}{r}$$

To give intuition about the disagreement coefficient, let us go through the definition step by step for the optimal hypothesis. According to definition, $DIS(B(h^*,r))$ denotes the disagreement region in the ball with radius $r$ around the optimal hypothesis $h^*$. Consequently, $Pr_x(DIS(B(h^*,r)))$ denotes the probability that we get a data point in the disagreement region of the optimal hypothesis to label. The disagreement coefficient essentially allow us to see how fast the probability that the next data point would land in the disagreement region around the optimal hypothesis changes.

The size of the ball around the optimal hypothesis ($B(h^*,r)$) is important. A smaller ball means that we are closer to finding the optimal hypothesis, and that there are less hypotheses that agree in the currently assigned labels. It is of foremost importance to understand at what rate the size of this ball decreases. The disagreement coefficient is a measure that captures it; the higher the disagreement coefficient, the harder the problem of finding the optimal hypothesis is.

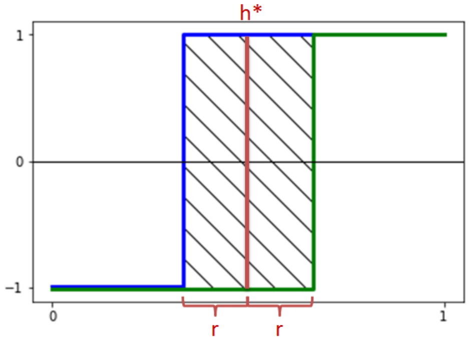


Figure 5: The region of disagreement for $B(h^*,r)$

Let us calculate the disagreement coefficient for our previous example. As you can see in Figure 5, the shaded area shows the disagreement region for $B(h^*,r)$. Since the distribution of data points is uniform in $[0,1]$, the value of $Pr(DIS(B(h^*,r)))$ is equal to $2r$, the size of the shaded area. Therefore, the disagreement coefficient is equal to

$$\theta_{h^*} = \sup_r \frac{Pr(DIS(B(h^*,r)))}{r} = \frac{2r}{r} = 2.$$

With the exception of very nice situations (uniform distribution, symmetric geometry, etc.) the disagreement coefficient is often impossible to calculate. Below are the disagreement coefficients for some classes of problems:

- Thresholds on $\mathbb{R}: \theta = 2$.
- Homogeneous hyperplanes in $\mathbb{R}^d$ with data uniformly distributed on a sphere: $\theta \leq \sqrt{d}$
- General hyperplanes in $\mathbb{R}^d$ with the data density bounded below: $\theta = O(d)$
- Intervals $[a,b]$ on $\mathbb{R}$: $\theta = \infty$

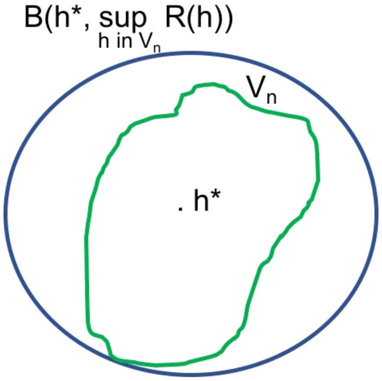


Figure 6: $V_n$ and the ball $B$ around $h^*$ with the worst hypothesis

### 4.1.4 Label Complexity

We now present the label complexity analysis of CAL.

**Theorem 2.** *Let $h^* = \arg\min_{h \in \mathcal{H}} R(h)$ and $R(h^*) = 0$, and suppose $n$ samples have been taken producing a version space $V_n \subseteq \mathcal{H}$ s.t. $V_n = \{h \in \mathcal{H} : \hat{R}(h) = 0\}$. If we request $\lambda$ additional labels only when the samples lie in the disagreement region $DIS(V_n)$, where $\lambda = 2\theta_{h^*} \log(|\mathcal{H}|/\delta)$, then, with probability greater than $1 - \delta/n$:*

$$\sup_{h \in V_{n+\lambda}} R(h) \leq \sup_{h \in V_n} \frac{1}{2} R(h) \tag{5}$$

Theorem 2 is saying that if we have taken $n$ samples and we take $\lambda$ more, then the risk of the worst possible hypothesis in the new version space $V_{n+\lambda}$ is smaller than the risk of the worst previous hypothesis in the version space $V_n$, divided by two; that is, we are cutting the error by half every time we take $\lambda$ samples.

*Proof.* The disagreement coefficient allows for a bound that relates the region of disagreement to the true risk of any $h \in V_n$. First, observe that:

$$\frac{Pr(DIS(V_n))}{\sup_{h \in V_n} R(h)} \leq \frac{Pr(DIS(B(h^*, \sup_{h \in V_n} R(h))))}{\sup_{h \in V_n} R(h)} \tag{6}$$

$$\leq \theta_{h^*} \tag{7}$$

where (6) falls from the fact that in the RHS we replace $V_n$ with a bigger set; that is, the ball around $h^*$ that contains the worst hypothesis $\sup_{h \in V_n} R(h)$, as shown in Figure 6. We can then upper bound the RHS of (6) by $\theta_{h^*}$, just by using the definition of the disagreement coefficient.
By the definition of risk we have:

$$\sup_{h \in V_{n+\lambda}} R(h) = \sup_{h \in V_{n+\lambda}} Pr(h(X) \neq Y) \tag{8}$$

Using the following probability rules:

$$P(A) = P(A, B) + P(A, \neg B) = P(A|B)P(B) + P(A|\neg B)P(\neg B)$$

If we condition, we have that (8) is equal to:

$$\sup_{h \in V_{n+\lambda}} Pr(h(X) \neq Y | X \in DIS(V_n)) Pr(X \in DIS(V_n)) + Pr(h(X) \neq Y | X \notin DIS(V_n)) Pr(X \notin DIS(V_n)) \tag{9}$$

Note that $h \in V_{n+\lambda}$, and as we only remove hypotheses from the version space, then $V_{n+\lambda} \subset V_n$. Given that $h^* \in V_n$ at all times and that all hypotheses in the version space agree on the labels of points outside the disagreement region $DIS(V_n)$, the second term in the summation (9) has to be zero.
We are left with the first term of (9), where we only look at $\lambda$ points that land in the disagreement region of $V_n$. This is like a brand new problem, where we can do passive learning only with points that land in $DIS(V_n)$. If we apply Theorem 1 and condition on the new version space $V_{n+\lambda}$, then the risk of any classifier in the new version space if we see only $\lambda$ samples satisfies:

$$Pr(h(X) \neq Y | X \in DIS(V_n)) \leq \frac{\log(|V_{n+\lambda}|/\delta)}{\lambda} \leq \frac{\log(|\mathcal{H}|/\delta)}{\lambda} \tag{10}$$

Plugging in the definition of $\lambda$ in (10), $\log(|\mathcal{H}|/\delta)$ cancels out and we are left with $1/2\theta_{h^*}$. Going back to (9), then we have:

$$\sup_{h \in V_{n+\lambda}} R(h) \leq \frac{1}{2\theta_{h^*}} Pr(X \in DIS(V_n)) \tag{11}$$

By multiplying both sides of (7) by $\sup_{h \in V_n} R(h)$, we can replace $Pr(X \in DIS(V_n))$ in (11):

$$\begin{aligned} \sup_{h \in V_{n+\lambda}} R(h) \leq & \frac{1}{2\theta_{h^*}} \theta_{h^*} \sup_{h \in V_n} R(h) \\ \leq & \sup_{h \in V_n} \frac{1}{2} R(h) \end{aligned}$$

which concludes the proof. □

Finally, we need to do the previous procedure $\log_2(1/\epsilon)$ epochs in order to achieve $\epsilon$-error, meaning that the bound holds simultaneously for all epochs. By taking a union bound over $\lambda, 2\lambda, ..., \lceil n/\lambda \rceil$, we have that after $n \geq \lambda \lceil \log_2(1/\epsilon) \rceil$ labels, the true risk of any classifier satisfies:

$$R(\hat{h}_n) \leq \exp(-n/(\theta_{h^*} \log(|\mathcal{H}| \log_2(1/\epsilon)/\delta)))$$

which is equivalent to say that $R(\hat{h}_n) < \epsilon$ when:

$$n \geq \log_2(1/\epsilon) \theta_{h^*} \log\left(\frac{|\mathcal{H}| \log_2(1/\epsilon)}{\delta}\right)$$

We see that the disagreement coefficient governs how fast the risk goes down, and is going exponentially fast as opposed to the passive learning rate, which was $O(1/n)$.

## 4.2 Agnostic Case

The CAL [2] algorithm introduced in the previous section has two major shortcomings: 1) it explicitly maintains the *region of uncertainty* which would be computationally cumbersome in most cases, and 2) it assumes separability but data is not usually separable in practice. To address these issues, Dasgupta, Hsu, and Monteleoni [3] developed an agnostic active learning algorithm, DHM, that does not assume separability and extends CAL [2] to this setting.

### 4.2.1 Preliminaries

Before formally stating DHM, we introduce a series of notations and definitions that are necessary in characterizing the problem. These notations are fairly similar to those in the previous section on realizable setting. However, here we use the original notation in DHM [3] for two reasons. First, while CAL and DHM are similar in many aspects, they are built on fundamentally different assumption (realizable vs. agnostic). Therefore, transforming the notation in favor of consistency might cause confusion for the readers. Second, we want readers to easily refer to the original article in case of any confusion as proofs we present here can be complicated.

Similar to previous condition, let $\mathcal{X}$ be the input space, $\mathcal{D}$ the joint distribution over $\mathcal{X} \times \{-1, 1\}$, and $\mathcal{H}$ a class of hypotheses $h : \mathcal{X} \to \{-1, 1\}$ with VC dimension $vcdim(\mathcal{H}) = d < \infty$. The setting is similar to CAL [2] – unlabeled points are drawn from $\mathcal{D}_{\mathcal{X}}$ (marginal distribution of $\mathcal{D}$ over $\mathcal{X}$), and we can optionally request the label $y$ sampled from the joint distribution $\mathcal{D}$.

We first define the error of a hypothesis $h$ under $\mathcal{D}$ as follows:

$$err_{\mathcal{D}}(h) = \Pr_{(x,y)\sim\mathcal{D}}[h(x) \neq y]$$

We can then define the empirical error on a finite sample $Z \subset \mathcal{X} \times \{-1, 1\}$ as follows:

$$err(h, Z) = \frac{1}{|Z|} \sum_{(x,y)\in Z} \mathbf{1}\{h(x) \neq y\} \tag{12}$$

It is important to notice that since we are in the agnostic case, the best classifier $h^*$ has some error $\nu$.

### 4.2.2 Algorithm: DHM

We now present DHM (see Algorithm 2). The main idea of this algorithm is to divide the data points into two groups $\hat{S}$ and $T$ to extend the notion of uncertainty to the agnostic setting. The former contains the points for which we do not request the label, while the latter contains the data points for which we explicitly request labels.

For $A, B \subset \mathcal{X} \times \{-1, 1\}$, let $\mathbf{learn}_{\mathcal{H}}(A, B)$ be a black-box supervised learner that takes as input a data set and returns any classifier from $\mathcal{H}$ consistent with $A$ and with the minimum error on $B$. Consistent with $A$ means that the classifier perfectly classifies the examples in $A$. Instead, minimum error on $B$ indicates that the classifier can make some mistakes on the set $B$. Upon receiving $x_t$, the DHM algorithm considers two hypotheses, $h_{\hat{y}} = \mathbf{learn}_{\mathcal{H}}(S_{t-1} \cup \{(x_t, \hat{y})\}, T_{t-1})$ for $\hat{y} \in \{-1, 1\}$, where $\hat{y}$ refers to predicted labels, and compares their empirical errors on $\hat{S}_{t-1} \cup T_{t-1}$ ($\hat{S}_{t-1}$ and $T_{t-1}$ refers to the state of the $\hat{S}$ and $T$ sets at time $t-1$ respectively). If the difference is large enough (denoted by $\Delta_{t-1}$), we infer the label for $x_t$ and add it to $\hat{S}_{t-1}$. Otherwise, the algorithm requests the label $y_t$, and adds $(x_t, y_t)$ to $T_{t-1}$. In the following subsection we will show how the error difference, $\Delta_t$, gets updated.

### 4.2.3 Error differences

As mentioned before, upon arriving any new point, we build two classifiers $h_{-\hat{y}}$ and $h_{\hat{y}}$ that are consistent with the new point being classified as $-1$ or $+1$ respectively. DHM algorithm then infers the label if the error difference between these two hypotheses exceeds $\Delta_t$. This difference actually reflects the extent to which we are certain about the true label of the new point: if the errors are close enough, we need to request the label. Intuitively, $\Delta_t$ should reflect how closely empirical errors on a sample approximate true errors on the distribution $\mathcal{D}$. In this section, we derive this threshold for the error difference.

It is worth noting that we do not necessarily have the true labels for the points in $\hat{S}_t$, as we might have made mistake in inferring their label. However, the setting of $\Delta_t$ can only depends on observable quantities (including those inferred and not necessarily the true labels of the points in $\hat{S}_t$). As such, to ensure that it does not cause statistical problems, we need to highlight the distinction between errors on $\hat{S}_t \cup T_t$ (observable) and $S_t \cup T_t$ (unobservable) as presented in the following definition.

**Algorithm 2** DHM

```
Initialize: $\hat{S}_0 = \emptyset$, $T_0 = \emptyset$
for $t = 1, 2...n$ do
    Obtain unlabeled data point $x_t$
    For each $\hat{y} \in \{-1, 1\}$, let $h_{\hat{y}} = \mathbf{learn}_{\mathcal{H}}(\hat{S}_{t-1} \cup \{(x_t, \hat{y})\}, T_{t-1})$
    if $err(h_{-\hat{y}}, \hat{S}_{t-1} \cup T_{t-1}) - err(h_{\hat{y}}, \hat{S}_{t-1} \cup T_{t-1}) > \Delta_{t-1}$ for some $\hat{y} \in \{-1, 1\}$
    (or if no such $h_{-\hat{y}}$ is found) then
        Set $\hat{S}_t = \hat{S}_{t-1} \cup (x_t, \hat{y})$ and $T_t = T_{t-1}$
    else
        Request $y_t$, and set $\hat{S}_t = \hat{S}_{t-1}$, $T_t = T_{t-1} \cup (x_t, y_t)$
    end if
end for
return $h_f = \mathbf{learn}_{\mathcal{H}}(\hat{S}_n, T_n)$
```

**Definition 7.** *Let $S_t$ be the set of labeled examples identical to those in $\hat{S}_t$, except with the true hidden labels swapped in $S_t$, whereas the labels in $\hat{S}_t$ are inferred. We distinguish the error on true from the inferred set as follows:*

$$err_t(h) = err(h, S_t \cup T_t)$$
$$\widehat{err}_t(h) = err(h, \hat{S}_t \cup T_t)$$

With the definitions above, we now have three different notions of error. To avoid confusion, we briefly describe each of them here:

- $err_{\mathcal{D}}(h)$ is defined as the true error defined over the joint distribution $\mathcal{D}$ (this is similar to $R(h)$ presented in realizable case)
- $err_t(h)$ is the empirical error over $t$ examples with their true labels ($S_t \cup T_t$)
- $\widehat{err}_t(h)$ is the empirical error on the sample with some inferred labels ($\hat{S}_t \cup T_t$)

We would like to examine whether the $\widehat{err}_t(h)$ converges to the true error $err_{\mathcal{D}}(h)$, with specific setting of $\Delta_t$. To do so, we need to take a two-step approach to prove the consistency. In the first step, we examine the convergence of $err_t(h)$ and $err_{\mathcal{D}}(h)$. In the second step, we show the consistency of $\widehat{err}_t(h)$ and $err_{\mathcal{D}}(h)$.

For the analysis of this algorithm, we use the following normalized uniform convergence bound. Before stating the lemma, we define the $n$-th shattering coefficient $\mathcal{S}(\mathcal{H}, n)$ as the maximum number of ways in which $\mathcal{H}$ can label a set of $n$ points; by Sauers lemma, this is at most $O(n^d)$ [5].

**Lemma 1.** (Vapnik and Chervonenkis [6]) *Let $\mathcal{F}$ be a family of measurable functions $f : \mathcal{Z} \to \{0, 1\}$ over a space $\mathcal{Z}$. Denote by $\mathbb{E}_Z f$ the empirical average of $f$ over a subset $Z \subset \mathcal{Z}$. Let $\alpha_t = \sqrt{(4/t)\ln(8\mathcal{S}(\mathcal{F}, 2t)/\delta)}$. If $Z$ is an i.i.d. sample of size $t$ from a fixed distribution over $\mathcal{Z}$, then, with probability at least $1 - \delta$, for all $f \in \mathcal{F}$:*

$$-\min\left(\alpha_t\sqrt{\mathbb{E}_Z f}, \alpha_t^2 + \alpha_t\sqrt{\mathbb{E} f}\right) \le \mathbb{E} f - \mathbb{E}_Z f \le \min\left(\alpha_t^2 + \alpha_t\sqrt{\mathbb{E}_Z f}, \alpha_t\sqrt{\mathbb{E} f}\right)$$

In light of Lemma 1, we can view $\mathbb{E} f - \mathbb{E}_Z f$ as the difference between $err_{\mathcal{D}}(h)$ and $err_t(h)$. Since the sample in $S_t \cup T_t$ is i.i.d., one idea is to apply Lemma 1 to the $err_t(h)$ function and obtain convergence bounds. However, there are some issues with this approach. First, we cannot reliably compute $err_t(h)$, since we do not request the true labels for points in $\hat{S}_t$. Instead, we can compute error differences for pairs of hypotheses $(h, h')$ that agree on $\hat{S}_t$, because we have:

$$err_t(h) - err_t(h') = \widehat{err}_t(h) - \widehat{err}_t(h') \tag{13}$$

The second issue is that these empirical error differences are means of $\{-1, 0, 1\}$-valued random variables. In order to apply Lemma 1, we need to rewrite them in terms of $\{0, 1\}$-valued random variables, as specified for the range of the function $f$. The following definition helps us apply Lemma 1:

**Definition 8.** *For a pair of $(h, h') \in \mathcal{H} \times \mathcal{H}$, define:*

$$\begin{aligned} g^{+}_{h,h'}(x, y) &= \mathbf{1}[h(x) \neq y \wedge \mathbf{1}(h'(x) = y] \\ g^{-}_{h,h'}(x, y) &= \mathbf{1}[h(x) = y \wedge \mathbf{1}(h'(x) \neq y] \end{aligned}$$

Now, we can write the following lemma:

**Lemma 2.** *Let $\alpha_t = \sqrt{(4/t)\ln(8\mathcal{S}(\mathcal{H}, 2t)^2/\delta)}$. With probability $1 - \delta$ over an i.i.d. sample $Z$ of size $t$ from $\mathcal{D}$, we have for all $(h, h') \in \mathcal{H} \times \mathcal{H}$,*

$$err(h, Z) - err(h', Z) \leq err_{\mathcal{D}}(h) - err_{\mathcal{D}}(h') + \alpha_t^2 + \alpha_t\left(\sqrt{\mathbb{E}_Z[g^{+}_{h,h'}]} + \sqrt{\mathbb{E}_Z[g^{-}_{h,h'}]}\right)$$

*Proof.* Let $\mathcal{G} = \{g^{+}_{h,h'} : (h, h') \in \mathcal{H} \times \mathcal{H}\} = \{g^{-}_{h,h'} : (h, h') \in \mathcal{H} \times \mathcal{H}\}$. Using the fact that $\mathcal{S}(\mathcal{G}, 2t) \leq \mathcal{S}(\mathcal{H}, 2t)^2$ and the LHS inequality of Lemma 1, we can write:

$$\mathbb{E}_Z[g^{+}_{h,h'}] \leq \mathbb{E}[g^{+}_{h,h'}] + \min\left(\alpha_t\sqrt{\mathbb{E}_Z[g^{+}_{h,h'}]}, \alpha_t^2 + \alpha_t\sqrt{\mathbb{E}[g^{+}_{h,h'}]}\right) \leq \mathbb{E}[g^{+}_{h,h'}] + \alpha_t\sqrt{\mathbb{E}_Z[g^{+}_{h,h'}]} \tag{14}$$

We can also use the RHS inequality of Lemma 1 and the function $g^{-}_{h,h'}$, and obtain the following inequality:

$$-\mathbb{E}_Z[g^{-}_{h,h'}] \leq -\mathbb{E}[g^{-}_{h,h'}] + \min\left(\alpha_t^2 + \alpha_t\sqrt{\mathbb{E}_Z[g^{-}_{h,h'}]}, \alpha_t\sqrt{\mathbb{E}[g^{-}_{h,h'}]}\right) \leq -\mathbb{E}[g^{-}_{h,h'}] + \alpha_t^2 + \alpha_t\sqrt{\mathbb{E}_Z[g^{-}_{h,h'}]} \tag{15}$$

We add (14) and (15) and derive the following inequality:

$$\mathbb{E}_Z[g^{+}_{h,h'}] - \mathbb{E}_Z[g^{-}_{h,h'}] \leq \mathbb{E}[g^{+}_{h,h'}] - \mathbb{E}[g^{-}_{h,h'}] + \alpha_t^2 + \alpha_t\left(\sqrt{\mathbb{E}_Z[g^{+}_{h,h'}]} + \sqrt{\mathbb{E}_Z[g^{-}_{h,h'}]}\right) \tag{16}$$

By definition (Eq. 12 and Def. 8), we have $\mathbb{E}_Z[g^{+}_{h,h'}] - \mathbb{E}_Z[g^{-}_{h,h'}] = err(h, Z) - err(h', Z)$. Hence, we can write:

$$err(h, Z) - err(h', Z) \leq err_{\mathcal{D}}(h) - err_{\mathcal{D}}(h') + \alpha_t^2 + \alpha_t\left(\sqrt{\mathbb{E}_Z[g^{+}_{h,h'}]} + \sqrt{\mathbb{E}_Z[g^{-}_{h,h'}]}\right)$$

□

For $Z = S_t \cup T_t$, we can use (13) and obtain the inequality for difference in errors that can be empirically determined (i.e., $\widehat{err}_t(h) - \widehat{err}_t(h')$). However, to obtain bounds for this difference in empirical errors, we need to fix the terms in the square root on the RHS of Lemma 2. To do so, we can write the following inequalities:

$$\mathbb{E}_{S_t \cup T_t}[g^{+}_{h,h'}] = \frac{1}{t}\sum_{(x,y)\in T_t} \mathbf{1}[h(x) \neq y \wedge h'(x) = y] \leq \frac{1}{t}\sum_{(x,y)\in T_t} \mathbf{1}[h(x) \neq y] = \widehat{err}_t(h) \tag{17}$$

$$\mathbb{E}_{S_t \cup T_t}[g^{-}_{h,h'}] = \frac{1}{t}\sum_{(x,y)\in T_t} \mathbf{1}[h(x) = y \wedge h'(x) \neq y] \leq \frac{1}{t}\sum_{(x,y)\in T_t} \mathbf{1}[h'(x) \neq y] = \widehat{err}_t(h') \tag{18}$$

We take the sum over $T_t$, because $h$ and $h'$ agree on the labels for the examples in $S_t$. The last equality also holds because both $h$ and $h'$ are consistent with $\hat{S}_t$ by construction.

Now, we can state a corollary of Lemma 2 as follows:

**Corollary 1.** *Let $\beta_t = \sqrt{(4/t)\ln(8(t^2+t)\mathcal{S}(\mathcal{H},2t)^2/\delta)}$. Then, with probability $1-\delta$, for all $t \geq 1$ and all $(h,h') \in \mathcal{H} \times \mathcal{H}$ consistent with $\hat{S}_t$, we have:*

$$\widehat{err}_t(h) - \widehat{err}_t(h') \leq err_{\mathcal{D}}(h,Z) - err_{\mathcal{D}}(h',Z) + \beta_t^2 + \beta_t\Big(\sqrt{\widehat{err}_t(h)} + \sqrt{\widehat{err}_t(h')}\Big)$$

*Proof.* For each $t \geq 1$, we apply Lemma 2 using $Z = S_t \cup T_t$ and $\delta = \delta/(t^2+t)$. Therefore, we will have bounds with different probabilities, *i.e.*, $1 - \delta/(t^2+t)$ for each $t$. We then apply a union bound over all $t \geq 1$. Thus, with probability at least $1-\delta$, the bounds in Lemma 2 hold simultaneously for all $t \geq 1$ and all $(h,h') \in \mathcal{H} \times \mathcal{H}$, with $Z = S_t \cup T_t$.[1] So we can write:

$$err(h,Z) - err(h',Z) \leq err_{\mathcal{D}}(h) - err_{\mathcal{D}}(h') + \beta_t^2 + \beta_t\Big(\sqrt{\mathbb{E}_Z[g^+_{h,h'}]} + \sqrt{\mathbb{E}_Z[g^-_{h,h'}]}\Big) \tag{19}$$

Using Eq. 13 for the LHS and Eq. 17 and 18 for the RHS, we can show:

$$\widehat{err}_t(h) - \widehat{err}_t(h') \leq err_{\mathcal{D}}(h,Z) - err_{\mathcal{D}}(h',Z) + \beta_t^2 + \beta_t\Big(\sqrt{\widehat{err}_t(h)} + \sqrt{\widehat{err}_t(h')}\Big)$$

□

Corollary 1 shows a very interesting fact: although $\hat{S}_t \cup T_t$ is not an i.i.d. sample from $\mathcal{D}$, we can obtain the normalized uniform convergence bounds for empirical error differences in $\hat{S}_t \cup T_t$. In light of this finding, we can define $\Delta_t$ as follows:

$$\Delta_t := \beta_t^2 + \beta_t\Big(\sqrt{\widehat{err}_t(h_{+1})} + \sqrt{\widehat{err}_t(h_{-1})}\Big), \tag{20}$$

where $\beta_t = \sqrt{(4/t)\ln(8(t^2+t)\mathcal{S}(\mathcal{H},2t)^2/\delta)} = \tilde{O}(\sqrt{d\log n/n})$, since $\mathcal{S}(\mathcal{H},2t) \leq (2t)^d$.

### 4.2.4 Correctness and Fall-back

Now we prove that the set of hypotheses that are consistent with $\hat{S}$ always includes the optimal hypothesis, $h^*$.

**Lemma 3.** *With probability of at least $1-\delta$, the hypothesis $h^* = \arg\inf_{h\in\mathcal{H}} err_{\mathcal{D}}(h)$ is consistent with $\hat{S}_t$ for all $t \geq 0$ in Algorithm 2.*

*Proof.* Apply the bounds in Corollary 1 and proceed by induction on $t$. The base case of induction is trivial since $\hat{S}_0 = \varnothing$. Now assume that $h^*$ is in the set of consistent hypotheses with $\hat{S}_t$. Suppose upon arriving $x_{t+1}$, we discover $\widehat{err}_t(h_{+1}) - \widehat{err}_t(h_{-1}) > \Delta_t$. Using contradiction, we will show that $h^*(x_{t+1}) = -1$. Suppose that $h^*(x_{t+1}) = +1$. Since we assumed that $\widehat{err}_t(h_{+1}) - \widehat{err}_t(h_{-1}) > \Delta_t$, we can write:

$$\widehat{err}_t(h_{+1}) - \widehat{err}_t(h_{-1}) > \beta_t^2 + \beta_t\Big(\sqrt{\widehat{err}_t(h_{+1})} + \sqrt{\widehat{err}_t(h_{-1})}\Big) \implies \sqrt{\widehat{err}_t(h_{+1})} > \beta_t \tag{21}$$

In addition, since our algorithm chooses $h_{+1}$, we know that $\widehat{err}_t(h^*) \geq \widehat{err}_t(h_{+1})$. Together with Eq. 21, we have:

$$\begin{aligned}
\widehat{err}_t(h^*) - \widehat{err}_t(h_{-1}) &= \big(\widehat{err}_t(h^*) - \widehat{err}_t(h_{+1})\big) + \big(\widehat{err}_t(h_{+1}) - \widehat{err}_t(h_{-1})\big)\\
&> \sqrt{\widehat{err}_t(h_{+1})}\big(\sqrt{\widehat{err}_t(h^*)} - \sqrt{\widehat{err}_t(h_{+1})}\big) + \beta_t^2 + \beta_t\big(\sqrt{\widehat{err}_t(h_{+1})} - \sqrt{\widehat{err}_t(h_{-1})}\big)\\
&> \beta_t\big(\sqrt{\widehat{err}_t(h^*)} - \sqrt{\widehat{err}_t(h_{+1})}\big) + \beta_t^2 + \beta_t\big(\sqrt{\widehat{err}_t(h_{+1})} - \sqrt{\widehat{err}_t(h_{-1})}\big)\\
&= \beta_t^2 + \beta_t\big(\sqrt{\widehat{err}_t(h^*)} - \sqrt{\widehat{err}_t(h_{-1})}\big).
\end{aligned}$$

Now since we have $\widehat{err}_t(h^*) - \widehat{err}_t(h_{-1}) > \beta_t^2 + \beta_t\big(\sqrt{\widehat{err}_t(h^*)} - \sqrt{\widehat{err}_t(h_{-1})}\big)$, Corollary 1 implies that $err_{\mathcal{D}}(h^*) > err_{\mathcal{D}}(h_{-1})$, which is a contradiction since $h^*$ is the optimal hypothesis. The proof is similar for the case where $\widehat{err}_t(h_{-1}) - \widehat{err}_t(h_{+1}) > \Delta_t$. □

[1] The reason the union bound gives us the probability $1-\delta$ is because of the term $t^2+t$ in $\beta_t$. For each $t \geq 1$, we have the bounds in Lemma 2 with probability $1-\delta/(t^2+t)$. Thus, the probability for union bound will be $1 - \sum_{t=1}^{\infty}\delta/(t^2+t) = 1-\delta$.

Recall that the best hypothesis $h^*$ in the hypothesis space has some error $\nu$. Here we prove that for $t = \widetilde{O}((d/\epsilon)(1 + \nu/\epsilon))$, Algorithm 2 returns a hypothesis with error at most $\epsilon + \nu$. To be more specific, we prove the following theorem.

**Theorem 3.** *Let $\nu = \inf_{h \in \mathcal{H}} err_{\mathcal{D}}(H)$ and $d = vcdim(\mathcal{H})$. There exist a constant $c > 0$ such that the following holds. If Algorithm 2 is given a stream of $n$ unlabeled examples, then with the probability of least $1 - \delta$, the algorithm returns a hypothesis with error at most $\nu + c(\alpha^2 + \sqrt{\nu}\alpha)$ where $\alpha$ is equal to $\sqrt{(1/n)(d \log n + log(1/\delta))}$.*

*Proof.* In Lemma 3 we proved that $h^*$ is consistent with $\hat{S}_n$ with probability at least $1 - \delta$. We know that $\widehat{err}_n(h_f) \leq \widehat{err}_n(h^*)$, where $h_f$ is the hypothesis returned from Algorithm 2. Also, using a similar approach as Corollary 1, we can have $\widehat{err}_n(h^*) - \widehat{err}_n(h_f) \leq err_{\mathcal{D}}(h^*) - err_{\mathcal{D}}(h_f) + \beta_n^2 + \beta_n\Big(\sqrt{\widehat{err}_{\mathcal{D}}(h^*)} + \sqrt{\widehat{err}_{\mathcal{D}}(h_f)}\Big)$. Therefore,[2]

$$\begin{aligned}
err_{\mathcal{D}}(h_f) &\leq err_{\mathcal{D}}(h^*) + \widehat{err}_n(h_f) - \widehat{err}_n(h^*) + \beta_n^2 + \beta_n\big(\sqrt{\widehat{err}_{\mathcal{D}}(h^*)} + \sqrt{\widehat{err}_{\mathcal{D}}(h_f)}\big) \\
&\leq err_{\mathcal{D}}(h^*) + \beta_n^2 + \beta_n\big(\sqrt{\widehat{err}_{\mathcal{D}}(h^*)} + \sqrt{\widehat{err}_{\mathcal{D}}(h_f)}\big) \\
&\leq \nu + \beta_n^2 + \beta_n\sqrt{\nu} + \beta_n\sqrt{\widehat{err}_{\mathcal{D}}(h_f)} \\
&\leq \nu + 3\beta_n^2 + 2\beta_n\sqrt{\nu}
\end{aligned}$$

Thus, the error term is at most $\nu + c(\beta_n^2 + \sqrt{\nu}\beta_n)$. Also, we discussed in Corollary 1 that $\beta_n = \widetilde{O}(\sqrt{\frac{d \log n}{n}})$ which concludes the proof. □

In view of Theorem 3, Algorithm 2 returns a hypothesis with error at most $\nu + \epsilon$ when $n = \tilde{O}\big((d/\epsilon)(1 + \nu/\epsilon)\big)$; this is (asymptotically) the usual sample complexity of supervised learning. Since the algorithm requests at most $n$ labels, its label complexity is always at most $n = \tilde{O}\big((d/\epsilon)(1 + \nu/\epsilon)\big)$.

### 4.2.5 Label Complexity

We can also propose a higher bound on the expected number of labels requested by DHM in terms of the disagreement coefficient. Before proving the bound, we need to find some bounds for the probability of requesting a label. The main idea for proving this is that the probability of requesting a label is intimately related to the disagreement coefficient and the empirical errors of the hypothesis in the previous step.

**Lemma 4.** *There exist constants $c_1, c_2 > 0$ such that, with probability at least $1 - 2\delta$ for all $t \geq 1$, the following holds. Let $\hat{y} = h^*(x_{t+1})$, where $h^* = \arg\inf_{h \in \mathcal{H}} err_{\mathcal{D}}(h)$, then the probability that Algorithm 2 requests the label $y_{t+1}$ is*

$$Pr_{x_{t+1} \sim \mathcal{D}_{\mathcal{X}}}[Request\ y_{t+1}] < Pr_{x_{t+1} \sim \mathcal{D}_{\mathcal{X}}}[err_{\mathcal{D}}(h_{-\hat{y}}) \leq c_1\nu + c_2\beta_t^2] \tag{22}$$

*where $\nu$ and $\beta$ are as defined in Theorem 3 and Corollary 1, respectively.*

*Proof.* Define $\gamma_t = \sqrt{(4/t)\ln\left(8(t^2 + t)\mathcal{S}(\mathcal{H}, 2t)/\delta\right)} \leq \beta_t$. With probability at least $1 - 2\delta$:

1. By Lemma 1, for all $t \geq 1$ and $h \in \mathcal{H}$:

$$-\gamma_t^2 - \gamma_t\sqrt{err_{\mathcal{D}}(h)} \leq err_{\mathcal{D}}(h) - err_t(h) \leq \gamma_t\sqrt{err_{\mathcal{D}}(h)}$$

2. By Lemma 3, $h^*$ is consistent with $\hat{\mathcal{S}}_t$ for all $t \geq 0$.

[2] In order to derive the inequality, in (14) and (15), we select the terms with $\mathbb{E}[g^-_{h,h'}]$ and $\mathbb{E}[g^+_{h,h'}]$ in the min function instead of the terms with $\mathbb{E}_{\mathcal{Z}}[g^-_{h,h'}]$ and $\mathbb{E}_{\mathcal{Z}}[g^+_{h,h'}]$. Also in (17) and (18) we take expectation instead of empirical expectation.

Now suppose that $h^*(x_{t+1}) = -1$ and Algorithm 2 requests the label $y_{t+1}$. So we need to prove that the RHS of (22) equals one for some $c_1, c_2 > 0$. Since the label is requested:

$$\widehat{err}_t(h_{+1}) - \widehat{err}_t(h_{-1}) \leq \beta_t^2 + \beta_t(\sqrt{\widehat{err}_t(h_{+1})} + \sqrt{\widehat{err}_t(h_{-1})}) \tag{23}$$

A lower bound for the equation above is:

$$err_t(h_{+1}) - err_t(h^*) = \widehat{err}_t(h_{+1}) - \widehat{err}_t(h^*) \leq \widehat{err}_t(h_{+1}) - \widehat{err}_t(h_{-1}) \tag{24}$$

This lower bound and the fact that $\widehat{err}_t(h_{+1}) \leq err_t(h_{+1})$ and $\widehat{err}_t(h_{-1}) \leq err_t(h^*)$ implies that,

$$err_t(h_{+1}) \leq err_t(h^*) + \beta_t^2 + \beta_t(\sqrt{err_t(h_{+1})}) + \beta_t\sqrt{err_t(h^*)}$$

Due to the fact that $A \leq B + C\sqrt{A} \to A \leq B + C^2 + C\sqrt{B}$, it can be implied that:

$$err_t(h_{+1}) \leq err_t(h^*) + 2\beta_t^2 + \beta_t\sqrt{err_t(h^*)} + \beta_t\sqrt{err_t(h^*) + \beta_t^2 + \beta_t\sqrt{err_t(h^*)}}$$

Uniform convergence of errors results in $err_{\mathcal{D}}(h_{+1}) + \gamma_t\sqrt{err_{\mathcal{D}}(h_{+1})} \leq err_t(h_{+1})$ and it also implies that $err_t(h^*) \leq \nu + \gamma_t^2 + \gamma_t\sqrt{\nu}$, which sums up to the conclusion that $err_{\mathcal{D}}(h_{+1}) \leq 3\nu + (12 + 2\sqrt{3})\beta_t^2$, as needed. □

Using this upper bound on the probability of requesting a label, Lemma 5 presents another upper bound in terms of the disagreement coefficient. Herein, we use a slight variation of the disagreement coefficient introduced in the realizable setting, defined as follows:

**Definition 9.** *The disagreement coefficient $\theta = \theta(\mathcal{D}, \mathcal{H}, \epsilon) > 0$ is:*

$$\theta = \sup \frac{Pr_{x\sim\mathcal{D}_\mathcal{X}}[\exists h \in B(h^*, r) \ s.t. \ h(x) \neq h^*(x)]}{r} : r \geq v + \epsilon$$

where $h^* = \arg\inf_{h\in\mathcal{H}} err_{\mathcal{D}}(h)$ and $v = err_{\mathcal{D}}(h^*)$.

**Lemma 5.** *In the same setting as Lemma 4, there exists a constant $c > 0$ such that*

$$Pr_{x_{t+1}\sim\mathcal{D}_\mathcal{X}}[Request \ y_{t+1}] \leq c\theta(\nu + \beta_t^2),$$

*where $\theta = \theta(\mathcal{D}, \mathcal{H}, 3\beta_n^2 + 2\beta_m\sqrt{\nu})$ is the disagreement coefficient defined in Definition 9.*

*Proof.* Suppose $h^*(x_{t+1}) = -1$. By triangle inequality, $err_{\mathcal{D}}(h_{+1}) \geq \rho(h_{+1}, h^*) - \nu$, where $\rho$ is the disagreement (pseudo) metric. By Lemma 4, there exists $c_1$ and $c_2$:

$$Pr_{x_{t+1}\sim\mathcal{D}_\mathcal{X}}[\text{Request } y_{t+1}] < Pr_{x_{t+1}\sim\mathcal{D}_\mathcal{X}}[\rho(h_{+1}, h^*) \leq (c_1 + 1)\nu + c_2\beta_t^2] \tag{25}$$

We can choose the constants $c_1, c_2 > 0$ such that $(c_1 + 1)\nu + c_2\beta_t^2 \geq \nu + 3\beta_n^2 + 2\beta_m\sqrt{\nu}$. By Definition 5, $Pr_{x_{t+1}\sim\mathcal{D}_\mathcal{X}}[\rho(h_{+1}, h^*) \leq (c_1 + 1)\nu + c_2\beta_t^2] \leq \theta.((c_1 + 1)\nu + c_2\beta_t^2)$ □

Now, using this lemma, calculating the complexity bound for agnostic active learning and obtaining the upper bounds on the number of labels requested by the algorithm is straightforward.

**Theorem 4.** *Let $n$ be the number of unlabeled data given to Algorithm 2, $d = vcdim(\mathcal{H})$ and $\nu, \beta, \theta$ as defined in Theorem 3, Corollary 1 and Lemma 5, respectively. There exists constants $c_1 > 0$ such that for any $c_2 \geq 1$, with probability higher than $1 - 2\delta$:*

1. *If $\nu \leq (c2 - 1)\beta_n^2$, Algorithm 2 returns a hypothesis with error as bounded in Theorem 3 and $L \leq 1 + c_1c_2\theta.(d\log^2 n + log\frac{1}{\delta}\log n)$*

*2. Otherwise, the same holds except that* $L \leq 1 + c_1\theta(\nu n + d\log^2 n + \log\frac{1}{\delta}\log n)$

*where* $L$ *is the expected number of labels requested. Furthermore, with the probability at least* $1-\delta'$*, the algorithm requests no more than* $L + \sqrt{3L\log(1/\delta')}$ *labels.*

*Proof.* Using a Chernoff bound for the Poisson trials and applying Lemma 5 the proof is concluded. □

In conclusion, with the substitution $\epsilon = 3\beta_n^2 + 2\beta_m\sqrt{\nu}$, these theorems prove that for any hypothesis class and data distribution for which $\theta \leq \frac{1}{\epsilon+\nu}$, Algorithm 2 using only $\widetilde{O}(\theta d\log^2(1/\epsilon))$ labels, can achieve the error $\epsilon \approx \nu$, and using only $\widetilde{O}(\theta d\log^2(1/\epsilon) + \theta d(\nu/\epsilon)^2)$ labels, it can achieve the error $\epsilon \ll \nu$.

# References

[1] Sanjoy Dasgupta. Two faces of active learning. *Theor. Comput. Sci.*, 412(19):1767–1781, April 2011.

[2] David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Mach. Learn.*, 15(2):201–221, May 1994.

[3] Sanjoy Dasgupta, Daniel J Hsu, and Claire Monteleoni. A general agnostic active learning algorithm. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 353–360. Curran Associates, Inc., 2008.

[4] Bousquet Olivier Lugosi Gábor Boucheron, Stéphane. Theory of classification : a survey of some recent advances. *ESAIM: Probability and Statistics*, 9:323–375, 2005.

[5] Ofer Dekel and Andrew Guillory. Lecture notes in cse522 winter 2011, learning theory, February 2011.

[6] Vladimir N Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of complexity*, pages 11–30. Springer, 2015.