Learning with a Drifting Target Concept

Steve Hanneke, Varun Kanade, and Liu Yang

[1] Princeton, NJ USA.
steve.hanneke@gmail.com
[2] Département d'informatique, École normale supérieure, Paris, France.
varun.kanade@ens.fr
[3] IBM T.J. Watson Research Center, Yorktown Heights, NY USA.
yangli@us.ibm.com

Abstract. We study the problem of learning in the presence of a drifting target concept. Specifically, we provide bounds on the error rate at a given time, given a learner with access to a history of independent samples labeled according to a target concept that can change on each round. One of our main contributions is a refinement of the best previous results for polynomial-time algorithms for the space of linear separators under a uniform distribution. We also provide general results for an algorithm capable of adapting to a variable rate of drift of the target concept. Some of the results also describe an active learning variant of this setting, and provide bounds on the number of queries for the labels of points in the sequence sufficient to obtain the stated bounds on the error rates.

1 Introduction

Much of the work on statistical learning has focused on learning settings in which the concept to be learned is static over time. However, there are many application areas where this is not the case. For instance, in the problem of face recognition, the concept to be learned actually changes over time as each individual's facial features evolve over time. In this work, we study the problem of learning with a drifting target concept. Specifically, we consider a statistical learning setting, in which data arrive i.i.d. in a stream, and for each data point, the learner is required to predict a label for the data point at that time. We are then interested in obtaining low error rates for these predictions. The target labels are generated from a function known to reside in a given concept space, and at each time t the target function is allowed to change by at most some distance Δ_t: that is, the probability the new target function disagrees with the previous target function on a random sample is at most Δ_t.

This framework has previously been studied in a number of articles. The classic works of [HL91,HL94,BH96,Lon99,BBDK00] and [BL96,BL97] together provide a general analysis of a very-much related setting. Though the objectives in these works are specified slightly differently, the results established there are easily translated into our present framework, and we summarize many of the relevant results from this literature in Section 3.

While the results in these classic works are general, the best guarantees on the error rates are only known for methods having no guarantees of computational efficiency. In a more recent effort, the work of [CMEDV10] studies this problem in the specific context of learning a homogeneous linear separator, when all the Δ_t values are identical. They propose a polynomial-time algorithm (based on the modified Perceptron algorithm of [DKM09]), and prove a bound on the number of mistakes it makes as a function of the number of samples, when the data distribution satisfies a certain condition called "λ-good" (which generalizes a useful property of the uniform distribution on the origin-centered unit sphere). However, their result is again worse than that obtainable by the known computationally-inefficient methods.

Thus, the natural question is whether there exists a polynomial-time algorithm achieving roughly the same guarantees on the error rates known for the inefficient methods. In the present work, we resolve this question in the case of learning homogeneous linear separators under the uniform distribution, by proposing a polynomial-time algorithm that indeed achieves roughly the same bounds on the error rates known for the inefficient methods in the literature. This represents the main technical contribution of this work.

We also study the interesting problem of *adaptivity* of an algorithm to the sequence of Δ_t values, in the setting where Δ_t may itself vary over time. Since the values Δ_t might typically not be accessible in practice, it seems important to have learning methods having no explicit dependence on the sequence Δ_t. We propose such a method below, and prove that it achieves roughly the same bounds on the error rates known for methods in the literature which require direct access to the Δ_t values. Also in the context of variable Δ_t sequences, we discuss conditions on the sequence Δ_t necessary and sufficient for there to exist a learning method guaranteeing a *sublinear* rate of growth of the number of mistakes.

We additionally study an *active learning* extension to this framework, in which, at each time, after making its prediction, the algorithm may decide whether or not to request access to the label assigned to the data point at that time. In addition to guarantees on the error rates (for *all* times, including those for which the label was not observed), we are also interested in bounding the number of labels we expect the algorithm to request, as a function of the number of samples encountered thus far.

2 Definitions and Notation

Formally, in this setting, there is a fixed distribution $\mathcal{P}$ over the instance space $\mathcal{X}$, and there is a sequence of independent $\mathcal{P}$-distributed unlabeled data $X_1, X_2, \ldots$. There is also a concept space $\mathbb{C}$, and a sequence of target functions $\mathbf{h}^* = \{h_1^*, h_2^*, \ldots\}$ in $\mathbb{C}$. Each t has an associated target label $Y_t = h_t^*(X_t)$. In this context, a (passive) learning algorithm is required, on each round t, to produce a classifier $\hat{h}_t$ based on the observations $(X_1, Y_1), \ldots, (X_{t-1}, Y_{t-1})$, and we denote by $\hat{Y}_t = \hat{h}_t(X_t)$ the corresponding prediction by the algorithm for the label of X_t. For any classifier h, we define $\mathrm{er}_t(h) = \mathcal{P}(x : h(x) \neq h_t^*(x))$. We also say the algorithm makes a "mistake" on instance X_t if $\hat{Y}_t \neq Y_t$; thus, $\mathrm{er}_t(\hat{h}_t) = \mathbb{P}(\hat{Y}_t \neq Y_t | (X_1, Y_1), \ldots, (X_{t-1}, Y_{t-1}))$.

For notational convenience, we will suppose the h_t^* sequence is chosen independently from the X_t sequence (i.e., h_t^* is chosen prior to the "draw" of $X_1, X_2, \ldots \sim \mathcal{P}$), and is not random.

In each of our results, we will suppose $\mathbf{h}^*$ is chosen from some set S of sequences in $\mathbb{C}$. In particular, we are interested in describing the sequence $\mathbf{h}^*$ in terms of the magnitudes of *changes* in h_t^* from one time to the next. Specifically, for any sequence $\boldsymbol{\Delta} = \{\Delta_t\}_{t=2}^{\infty}$ in $[0,1]$, we denote by $S_{\boldsymbol{\Delta}}$ the set of all sequences $\mathbf{h}^*$ in $\mathbb{C}$ such that, $\forall t \in \mathbb{N}$, $\mathcal{P}(x : h_t(x) \neq h_{t+1}(x)) \leq \Delta_{t+1}$.

Throughout this article, we denote by d the VC dimension of $\mathbb{C}$ [VC71], and we suppose $\mathbb{C}$ is such that $1 \leq d < \infty$. Also, for any $x \in \mathbb{R}$, define $\mathrm{Log}(x) = \ln(\max\{x, e\})$.

3 Background: $(\boldsymbol{\epsilon}, \boldsymbol{S})$-Tracking Algorithms

As mentioned, the classic literature on learning with a drifting target concept is expressed in terms of a slightly different model. In order to relate those results to our present setting, we first introduce the classic setting. Specifically, we consider a model introduced by [HL94], presented here in a more-general form inspired by [BBDK00]. For a set S of sequences $\{h_t\}_{t=1}^{\infty}$ in $\mathbb{C}$, and a value $\epsilon > 0$, an algorithm $\mathcal{A}$ is said to be (ϵ, S)-*tracking* if $\exists t_\epsilon \in \mathbb{N}$ such that, for any choice of $\mathbf{h}^* \in S$, $\forall T \geq t_\epsilon$, the prediction $\hat{Y}_T$ produced by $\mathcal{A}$ at time T satisfies

$$\mathbb{P}\left(\hat{Y}_T \neq Y_T\right) \leq \epsilon.$$

Note that the value of the probability in the above expression may be influenced by $\{X_t\}_{t=1}^{T}$, $\{h_t^*\}_{t=1}^{T}$, and any internal randomness of the algorithm $\mathcal{A}$.

The focus of the results expressed in this classical model is determining sufficient conditions on the set S for there to exist an (ϵ, S)-tracking algorithm, along with bounds on the sufficient size of t_ϵ. These conditions on S typically take the form of an assumption on the drift rate, expressed in terms of ϵ. Below, we summarize several of the strongest known results for this setting.

3.1 Bounded Drift Rate

The simplest, and perhaps most elegant, results for (ϵ, S)-tracking algorithms is for the set S of sequences with a bounded drift rate. Specifically, for any $\Delta \in [0,1]$, define $S_\Delta = S_{\boldsymbol{\Delta}}$, where $\boldsymbol{\Delta}$ is such that $\Delta_{t+1} = \Delta$ for every $t \in \mathbb{N}$. The study of this problem was initiated in the original work of [HL94]. The best known general results are due to [Lon99]: namely, that for some $\Delta_\epsilon = \Theta(\epsilon^2/d)$, for every $\epsilon \in (0,1]$, there exists an (ϵ, S_Δ)-tracking algorithm for all values of $\Delta \leq \Delta_\epsilon$.[4] This refined an earlier result of [HL94] by a logarithmic

[4] In fact, [Lon99] also allowed the distribution $\mathcal{P}$ to vary gradually over time. For simplicity, we will only discuss the case of fixed $\mathcal{P}$.

factor. [Lon99] further argued that this result can be achieved with $t_\epsilon = \Theta(d/\epsilon)$. The algorithm itself involves a beautiful modification of the one-inclusion graoh prediction strategy of [HLW94]; since its specification is somewhat involved, we refer the interested reader to the original work of [Lon99] for the details.

3.2 Varying Drift Rate: Nonadaptive Algorithm

In addition to the concrete bounds for the case $\mathbf{h}^* \in S_\Delta$, [HL94] additionally present an elegant general result. Specifically, they argue that, for any $\epsilon > 0$, and any $m = \Omega\left(\frac{d}{\epsilon}\mathrm{Log}\frac{1}{\epsilon}\right)$, if $\sum_{i=1}^{m} \mathcal{P}(x : h_i^*(x) \neq h_{m+1}^*(x)) \leq m\epsilon/24$, then for $\hat{h} = \mathrm{argmin}_{h\in\mathbb{C}} \sum_{i=1}^{m} \mathbb{1}[h(X_i) \neq Y_i]$, $\mathbb{P}(\hat{h}(X_{m+1}) \neq h_{m+1}^*(X_{m+1})) \leq \epsilon$.[5] This result immediately inspires an algorithm $\mathcal{A}$ which, at every time t, chooses a value $m_t \leq t-1$, and predicts $\hat{Y}_t = \hat{h}_t(X_t)$, for $\hat{h}_t = \mathrm{argmin}_{h\in\mathbb{C}} \sum_{i=t-m_t}^{t-1} \mathbb{1}[h(X_i) \neq Y_i]$. We are then interested in choosing m_t to minimize the value of ϵ obtainable via the result of [HL94]. However, that result is based on the values $\mathcal{P}(x : h_i^*(x) \neq h_t^*(x))$, which would typically not be accessible to the algorithm. However, suppose instead we have access to a sequence $\boldsymbol{\Delta}$ such that $\mathbf{h}^* \in S_{\boldsymbol{\Delta}}$. In this case, we could approximate $\mathcal{P}(x : h_i^*(x) \neq h_t^*(x))$ by its *upper bound* $\sum_{j=i+1}^{t} \Delta_j$. In this case, we are interested choosing m_t to minimize the smallest value of ϵ such that $\sum_{i=t-m_t}^{t-1} \sum_{j=i+1}^{t} \Delta_j \leq m_t\epsilon/24$ and $m_t = \Omega\left(\frac{d}{\epsilon}\mathrm{Log}\frac{1}{\epsilon}\right)$. One can easily verify that this minimum is obtained at a value

$$m_t = \Theta\left(\underset{m\leq t-1}{\mathrm{argmin}} \frac{1}{m} \sum_{i=t-m}^{t-1} \sum_{j=i+1}^{t} \Delta_j + \frac{d\mathrm{Log}(m/d)}{m}\right),$$

and via the result of [HL94] (applied to the sequence $X_{t-m_t}, \ldots, X_t$) the resulting algorithm has

$$\mathbb{P}\left(\hat{Y}_t \neq Y_t\right) \leq O\left(\min_{1\leq m\leq t-1} \frac{1}{m} \sum_{i=t-m}^{t-1} \sum_{j=i+1}^{t} \Delta_j + \frac{d\mathrm{Log}(m/d)}{m}\right). \quad (1)$$

As a special case, if every t has $\Delta_t = \Delta$ for a fixed value $\Delta \in [0,1]$, this result recovers the bound $\sqrt{d\Delta\mathrm{Log}(1/\Delta)}$, which is only slightly larger than that obtainable from the best bound of [Lon99]. It also applies to far more general and more intersting sequences $\boldsymbol{\Delta}$, including some that allow periodic large jumps (i.e., $\Delta_t = 1$ for some indices t), others where the sequence Δ_t converges to 0, and so on. Note, however, that the algorithm obtaining this bound directly depends on the sequence $\boldsymbol{\Delta}$. One of the contributions of the present work is to remove this requirement, while maintaining essentially the same bound, though in a slightly different form.

[5] They in fact prove a more general result, which also applies to methods approximately minimizing the number of mistakes, but for simplicity we will only discuss this basic version of the result.

3.3 Computational Efficiency

[HL94] also proposed a reduction-based approach, which sometimes yields computationally efficient methods, though the tolerable Δ value is smaller. Specifically, given any (randomized) polynomial-time algorithm $\mathcal{A}$ that produces a classifier $h \in \mathbb{C}$ with $\sum_{t=1}^{m} \mathbb{1}[h(x_t) \neq y_t] = 0$ for any sequence $(x_1, y_1), \ldots, (x_m, y_m)$ for which such a classifier h exists (called the *consistency problem*), they propose a polynomial-time algorithm that is (ϵ, S_{Δ})-tracking for all values of $\Delta \leq \Delta'_{\epsilon}$, where $\Delta'_{\epsilon} = \Theta\left(\frac{\epsilon^2}{d^2 \mathrm{Log}(1/\epsilon)}\right)$. This is slightly worse (by a factor of $d\mathrm{Log}(1/\epsilon)$) than the drift rate tolerable by the (typically inefficient) algorithm mentioned above. However, it does sometimes yield computationally-efficient methods. For instance, there are known polynomial-time algorithms for the consistency problem for the classes of linear separators, conjunctions, and axis-aligned rectangles.

3.4 Lower Bounds

[HL94] additionally prove *lower bounds* for specific concept spaces: namely, linear separators and axis-aligned rectangles. They specifically argue that, for $\mathbb{C}$ a concept space

$$\mathrm{BASIC}_n = \{\cup_{i=1}^{n} [i/n, (i + a_i)/n) : \mathbf{a} \in [0, 1]^n\}$$

on $[0, 1]$, under $\mathcal{P}$ the uniform distribution on $[0, 1]$, for any $\epsilon \in [0, 1/e^2]$ and $\Delta_{\epsilon} \geq e^4 \epsilon^2 / n$, for any algorithm $\mathcal{A}$, and any $T \in \mathbb{N}$, there exists a choice of $\mathbf{h}^* \in S_{\Delta_{\epsilon}}$ such that the prediction $\hat{Y}_T$ produced by $\mathcal{A}$ at time T satisfies $\mathbb{P}\left(\hat{Y}_T \neq Y_T\right) > \epsilon$. Based on this, they conclude that no $(\epsilon, S_{\Delta_{\epsilon}})$-tracking algorithm exists. Furthermore, they observe that the space BASIC_n is embeddable in many commonly-studied concept spaces, including halfspaces and axis-aligned rectangles in $\mathbb{R}^n$, so that for $\mathbb{C}$ equal to either of these spaces, there also is no $(\epsilon, S_{\Delta_{\epsilon}})$-tracking algorithm.

4 Adapting to Arbitrarily Varying Drift Rates

This section presents a general bound on the error rate at each time, expressed as a function of the rates of drift, which are allowed to be *arbitrary*. Most-importantly, in contrast to the methods from the literature discussed above, the method achieving this general result is *adaptive* to the drift rates, so that it requires no information about the drift rates in advance. This is an appealing property, as it essentially allows the algorithm to learn under an *arbitrary* sequence $\mathbf{h}^*$ of target concepts; the difficulty of the task is then simply reflected in the resulting bounds on the error rates: that is, faster-changing sequences of target functions result in larger bounds on the error rates, but do not require a change in the algorithm itself.

4.1 Adapting to a Changing Drift Rate

Recall that the method yielding (1) (based on the work of [HL94]) required access to the sequence $\boldsymbol{\Delta}$ of changes to achieve the stated guarantee on the expected number of mistakes. That method is based on choosing a classifier to predict $\hat{Y}_t$ by minimizing the number of mistakes among the previous m_t samples, where m_t is a value chosen based on the $\boldsymbol{\Delta}$ sequence. Thus, the key to modifying this algorithm to make it adaptive to the $\boldsymbol{\Delta}$ sequence is to determine a suitable choice of m_t without reference to the $\boldsymbol{\Delta}$ sequence. The strategy we adopt here is to use the *data* to determine an appropriate value $\hat{m}_t$ to use. Roughly (ignoring logarithmic factors for now), the insight that enables us to achieve this feat is that, for the m_t used in the above strategy, one can show that $\sum_{i=t-m_t}^{t-1} \mathbb{1}[h_t^*(X_i) \neq Y_i]$ is roughly $\tilde{O}(d)$, and that making the prediction $\hat{Y}_t$ with *any* $h \in \mathbb{C}$ with roughly $\tilde{O}(d)$ mistakes on these samples will suffice to obtain the stated bound on the error rate (up to logarithmic factors). Thus, if we replace m_t with the largest value m for which $\min_{h\in\mathbb{C}} \sum_{i=t-m}^{t-1} \mathbb{1}[h(X_i) \neq Y_i]$ is roughly $\tilde{O}(d)$, then the above observation implies $m \geq m_t$. This then implies that, for $\hat{h} = \mathrm{argmin}_{h\in\mathbb{C}} \sum_{i=t-m}^{t-1} \mathbb{1}[h(X_i) \neq Y_i]$, we have that $\sum_{i=t-m_t}^{t-1} \mathbb{1}[\hat{h}(X_i) \neq Y_i]$ is also roughly $\tilde{O}(d)$, so that the stated bound on the error rate will be achieved (aside from logarithmic factors) by choosing $\hat{h}_t$ as this classifier $\hat{h}$. There are a few technical modifications to this argument needed to get the logarithmic factors to work out properly, and for this reason the actual algorithm and proof below are somewhat more involved. Specifically, consider the following algorithm (the value of the universal constant $K \geq 1$ will be specified below).

0. For $T = 1, 2, \ldots$
1. Let $\hat{m}_T = \max\left\{ m \in \{1, \ldots, T-1\} : \min_{h\in\mathbb{C}} \max_{m' \leq m} \frac{\sum_{t=T-m'}^{T-1} \mathbb{1}[h(X_t)\neq Y_t]}{d\mathrm{Log}(m'/d)+\mathrm{Log}(1/\delta)} < K \right\}$
2. Let $\hat{h}_T = \mathop{\mathrm{argmin}}_{h\in\mathbb{C}} \max_{m' \leq \hat{m}_T} \frac{\sum_{t=T-m'}^{T-1} \mathbb{1}[h(X_t)\neq Y_t]}{d\mathrm{Log}(m'/d)+\mathrm{Log}(1/\delta)}$

Note that the classifiers $\hat{h}_t$ chosen by this algorithm have no dependence on $\boldsymbol{\Delta}$, or indeed anything other than the data $\{(X_i, Y_i) : i < t\}$, and the concept space $\mathbb{C}$.

Theorem 1. *Fix any $\delta \in (0, 1)$, and let $\mathcal{A}$ be the above algorithm. For any sequence $\boldsymbol{\Delta}$ in $[0, 1]$, for any $\mathcal{P}$ and any choice of $\mathbf{h}^* \in S_{\boldsymbol{\Delta}}$, for every $T \in \mathbb{N}\setminus\{1\}$, with probability at least $1 - \delta$,*

$$\mathrm{er}_T\left(\hat{h}_T\right) \leq O\left(\min_{1\leq m\leq T-1} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m} \right).$$

Before presenting the proof of this result, we first state a crucial lemma, which follows immediately from a classic result of [Vap82,Vap98], combined with the fact (from [Vid03], Theorem 4.5) that the VC dimension of the collection of sets $\{\{x : h(x) \neq g(x)\} : h, g \in \mathbb{C}\}$ is at most $10d$.

Lemma 1. *There exists a universal constant $c \in [1, \infty)$ such that, for any class $\mathbb{C}$ of VC dimension d, $\forall m \in \mathbb{N}$, $\forall \delta \in (0,1)$, with probability at least $1-\delta$, every $h, g \in \mathbb{C}$ have*

$$\left| \mathcal{P}(x : h(x) \neq g(x)) - \frac{1}{m}\sum_{t=1}^{m} \mathbb{1}[h(X_t) \neq g(X_t)] \right|$$
$$\leq c\sqrt{\left(\frac{1}{m}\sum_{t=1}^{m} \mathbb{1}[h(X_t) \neq g(X_t)]\right) \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m}}$$
$$+ c\frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m}.$$

We are now ready for the proof of Theorem 1. For the constant K in the algorithm, we will choose $K = 145c^2$, for c as in Lemma 1.

Proof (Proof of Theorem 1). Fix any $T \in \mathbb{N}$ with $T \geq 2$, and define

$$m_T^* = \max\left\{ m \in \{1, \ldots, T-1\} : \forall m' \leq m, \right.$$
$$\left. \sum_{t=T-m'}^{T-1} \mathbb{1}[h_T^*(X_t) \neq Y_t] < K(d\mathrm{Log}(m'/d) + \mathrm{Log}(1/\delta)) \right\}.$$

Note that

$$\sum_{t=T-m_T^*}^{T-1} \mathbb{1}[h_T^*(X_t) \neq Y_t] \leq K(d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta)), \tag{2}$$

and also note that (since $h_T^* \in \mathbb{C}$) $\hat{m}_T \geq m_T^*$, so that (by definition of $\hat{m}_T$ and $\hat{h}_T$)

$$\sum_{t=T-m_T^*}^{T-1} \mathbb{1}[\hat{h}_T(X_t) \neq Y_t] \leq K(d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta))$$

as well. Therefore,

$$\sum_{t=T-m_T^*}^{T-1} \mathbb{1}[h_T^*(X_t) \neq \hat{h}_T(X_t)] \leq \sum_{t=T-m_T^*}^{T-1} \mathbb{1}[h_T^*(X_t) \neq Y_t] + \sum_{t=T-m_T^*}^{T-1} \mathbb{1}[Y_t \neq \hat{h}_T(X_t)]$$
$$\leq 2K(d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta)).$$

Thus, by Lemma 1, for each $m \in \mathbb{N}$, with probability at least $1 - \delta/(6m^2)$, if $m_T^* = m$, then

$$\mathcal{P}(x : \hat{h}_T(x) \neq h_T^*(x)) \leq (2K + c\sqrt{2K} + c)\frac{d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(6(m_T^*)^2/\delta)}{m_T^*}.$$

Furthermore, since $\mathrm{Log}(6(m_T^*)^2) \leq \sqrt{2K} d\mathrm{Log}(m_T^*/d)$, this is at most

$$2(K + c\sqrt{2K}) \frac{d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta)}{m_T^*}.$$

By a union bound (over values $m \in \mathbb{N}$), we have that with probability at least $1 - \sum_{m=1}^{\infty} \delta/(6m^2) \geq 1 - \delta/3$,

$$\mathcal{P}(x : \hat{h}_T(x) \neq h_T^*(x)) \leq 2(K + c\sqrt{2K}) \frac{d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta)}{m_T^*}.$$

Let us denote

$$\tilde{m}_T = \operatorname*{argmin}_{m \in \{1,\ldots,T-1\}} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m}.$$

Note that, for any $m' \in \{1, \ldots, T-1\}$ and $\delta \in (0,1)$, if $\tilde{m}_T \geq m'$, then

$$\begin{aligned} &\min_{m \in \{1,\ldots,T-1\}} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m} \\ &\geq \min_{m \in \{m',\ldots,T-1\}} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j = \frac{1}{m'} \sum_{i=T-m'}^{T-1} \sum_{j=i+1}^{T} \Delta_j, \end{aligned}$$

while if $\tilde{m}_T \leq m'$, then

$$\begin{aligned} &\min_{m \in \{1,\ldots,T-1\}} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m} \\ &\geq \min_{m \in \{1,\ldots,m'\}} \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m} = \frac{d\mathrm{Log}(m'/d) + \mathrm{Log}(1/\delta)}{m'}. \end{aligned}$$

Either way, we have that

$$\begin{aligned} &\min_{m \in \{1,\ldots,T-1\}} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m} \\ &\geq \min \left\{ \frac{d\mathrm{Log}(m'/d) + \mathrm{Log}(1/\delta)}{m'}, \frac{1}{m'} \sum_{i=T-m'}^{T-1} \sum_{j=i+1}^{T} \Delta_j \right\}. \end{aligned} \tag{3}$$

For any $m \in \{1, \ldots, T-1\}$, applying Bernstein's inequality (see [BLM13], equation 2.10) to the random variables $\mathbb{1}[h_T^*(X_i) \neq Y_i]/d$, $i \in \{T-m, \ldots, T-1\}$, and again to the random variables $-\mathbb{1}[h_T^*(X_i) \neq Y_i]/d$, $i \in \{T-m, \ldots, T-1\}$, together with a union bound, we obtain that, for any $\delta \in (0,1)$, with probability at least $1 - \delta/(3m^2)$,

$$
\begin{aligned}
&\frac{1}{m}\sum_{i=T-m}^{T-1} \mathcal{P}(x : h_T^*(x) \neq h_i^*(x)) \\
&\qquad - \sqrt{\left(\frac{1}{m}\sum_{i=T-m}^{T-1} \mathcal{P}(x : h_T^*(x) \neq h_i^*(x))\right)\frac{2\ln(3m^2/\delta)}{m}} \\
&< \frac{1}{m}\sum_{i=T-m}^{T-1} \mathbb{1}[h_T^*(X_i) \neq Y_i] \\
&< \frac{1}{m}\sum_{i=T-m}^{T-1} \mathcal{P}(x : h_T^*(x) \neq h_i^*(x)) \\
&\qquad + \max\left\{ \begin{array}{l} \sqrt{\left(\frac{1}{m}\sum_{i=T-m}^{T-1} \mathcal{P}(x : h_T^*(x) \neq h_i^*(x))\right)\frac{4\ln(3m^2/\delta)}{m}} \\ \frac{(4/3)\ln(3m^2/\delta)}{m} \end{array} \right. .
\end{aligned} \tag{4}
$$

The left inequality implies that

$$
\frac{1}{m}\sum_{i=T-m}^{T-1} \mathcal{P}(x\!:\!h_T^*(x) \neq h_i^*(x)) \leq \max\left\{\frac{2}{m}\sum_{i=T-m}^{T-1} \mathbb{1}[h_T^*(X_i) \neq Y_i], \frac{8\ln(3m^2/\delta)}{m}\right\}.
$$

Plugging this into the right inequality in (4), we obtain that

$$
\begin{aligned}
\frac{1}{m}\sum_{i=T-m}^{T-1} \mathbb{1}[h_T^*(X_i) \neq Y_i] &< \frac{1}{m}\sum_{i=T-m}^{T-1} \mathcal{P}(x : h_T^*(x) \neq h_i^*(x)) \\
&+ \max\left\{\sqrt{\left(\frac{1}{m}\sum_{i=T-m}^{T-1} \mathbb{1}[h_T^*(X_i) \neq Y_i]\right)\frac{8\ln(3m^2/\delta)}{m}}, \frac{\sqrt{32}\ln(3m^2/\delta)}{m}\right\}.
\end{aligned}
$$

By a union bound, this holds simultaneously for all $m \in \{1, \ldots, T-1\}$ with probability at least $1 - \sum_{m=1}^{T-1} \delta/(3m^2) > 1 - (2/3)\delta$. Note that, on this event, we obtain

$$
\begin{aligned}
\frac{1}{m}\sum_{i=T-m}^{T-1} \mathcal{P}(x : h_T^*(x) \neq h_i^*(x)) &> \frac{1}{m}\sum_{i=T-m}^{T-1} \mathbb{1}[h_T^*(X_i) \neq Y_i] \\
&- \max\left\{\sqrt{\left(\frac{1}{m}\sum_{i=T-m}^{T-1} \mathbb{1}[h_T^*(X_i) \neq Y_i]\right)\frac{8\ln(3m^2/\delta)}{m}}, \frac{\sqrt{32}\ln(3m^2/\delta)}{m}\right\}.
\end{aligned}
$$

In particular, taking $m = m_T^*$, and invoking maximality of m_T^*, if $m_T^* < T - 1$, the right hand side is at least

$$
(K - 6c\sqrt{K})\frac{d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta)}{m_T^*}.
$$

Since $\frac{1}{m}\sum_{i=T-m}^{T-1}\sum_{j=i+1}^{T} \Delta_j \geq \frac{1}{m}\sum_{i=T-m}^{T-1} \mathcal{P}(x : h_T^*(x) \neq h_i^*(x))$, taking $K = 145c^2$, we have that with probability at least $1-\delta$, if $m_T^* < T-1$, then

$$\begin{aligned}
&10(K + c\sqrt{2K}) \min_{m\in\{1,\ldots,T-1\}} \frac{1}{m}\sum_{i=T-m}^{T-1}\sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m} \\
&\geq 10(K + c\sqrt{2K}) \min\left\{ \frac{d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta)}{m_T^*}, \frac{1}{m_T^*}\sum_{i=T-m_T^*}^{T-1}\sum_{j=i+1}^{T} \Delta_j \right\} \\
&\geq 10(K + c\sqrt{2K}) \frac{d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta)}{m_T^*} \\
&\geq \mathcal{P}(x : \hat{h}_T(x) \neq h_T^*(x)).
\end{aligned}$$

Furthermore, if $m_T^* = T-1$, then we trivially have (on the same $1-\delta$ probability event as above)

$$\begin{aligned}
&10(K + c\sqrt{2K}) \min_{m\in\{1,\ldots,T-1\}} \frac{1}{m}\sum_{i=T-m}^{T-1}\sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m} \\
&\geq 10(K + c\sqrt{2K}) \min_{m\in\{1,\ldots,T-1\}} \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta)}{m} \\
&= 10(K + c\sqrt{2K}) \frac{d\mathrm{Log}((T-1)/d) + \mathrm{Log}(1/\delta)}{T-1} \\
&= 10(K + c\sqrt{2K}) \frac{d\mathrm{Log}(m_T^*/d) + \mathrm{Log}(1/\delta)}{m_T^*} \geq \mathcal{P}(x : \hat{h}_T(x) \neq h_T^*(x)).
\end{aligned}$$

4.2 Conditions Guaranteeing a Sublinear Number of Mistakes

One immediate implication of Theorem 1 is that, if the sum of Δ_t values grows sublinearly, then there exists an algorithm achieving an expected number of mistakes growing sublinearly in the number of predictions. Formally, we have the following corollary.

Corollary 1. *If $\sum_{t=1}^{T} \Delta_t = o(T)$, then there exists an algorithm $\mathcal{A}$ such that, for every $\mathcal{P}$ and every choice of $\mathbf{h}^* \in S_{\boldsymbol{\Delta}}$,*

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left[\hat{Y}_t \neq Y_t\right]\right] = o(T).$$

Proof. For every $T \in \mathbb{N}$ with $T \geq 2$, let

$$\tilde{m}_T = \operatorname*{argmin}_{1\leq m\leq T-1} \frac{1}{m}\sum_{i=T-m}^{T-1}\sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta_T)}{m},$$

and define $\delta_T = \frac{1}{\tilde{m}_T}$. Then consider running the algorithm $\mathcal{A}$ from Theorem 1, except that in choosing $\hat{m}_T$ and $\hat{h}_T$ for each T, we use the above value δ_T in place of δ. Then Theorem 1 implies that, for each T, with probability at least $1-\delta_T$,

$$\mathrm{er}_T(\hat{h}_T) \leq O\left(\min_{1\leq m\leq T-1} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta_T)}{m} \right).$$

Since $\mathrm{er}_T(\hat{h}_T) \leq 1$, this implies that

$$\begin{aligned}
&\mathbb{P}\left(\hat{Y}_T \neq Y_T\right) = \mathbb{E}\left[\mathrm{er}_T(\hat{h}_T)\right] \\
&\leq O\left(\min_{1\leq m\leq T-1} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(1/\delta_T)}{m} \right) + \delta_T \\
&= O\left(\min_{1\leq m\leq T-1} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d) + \mathrm{Log}(m)}{m} \right),
\end{aligned}$$

and since $x \mapsto x\mathrm{Log}(m/x)$ is nondecreasing for $x \geq 1$, $\mathrm{Log}(m) \leq d\mathrm{Log}(m/d)$, so that this last expression is

$$O\left(\min_{1\leq m\leq T-1} \frac{1}{m} \sum_{i=T-m}^{T-1} \sum_{j=i+1}^{T} \Delta_j + \frac{d\mathrm{Log}(m/d)}{m} \right).$$

Now note that, for any $t \in \mathbb{N}$ and $m \in \{1,\ldots,t-1\}$,

$$\frac{1}{m} \sum_{s=t-m}^{t-1} \sum_{r=s+1}^{t} \Delta_r \leq \frac{1}{m} \sum_{s=t-m}^{t-1} \sum_{r=t-m+1}^{t} \Delta_r = \sum_{r=t-m+1}^{t} \Delta_s. \tag{5}$$

Let $\beta_t(m) = \max\left\{\sum_{r=t-m+1}^{t} \Delta_r, \frac{d\mathrm{Log}(m/d)}{m}\right\}$, and note that $\sum_{r=t-m+1}^{t} \Delta_r + \frac{d\mathrm{Log}(m/d)}{m} \leq 2\beta_t(m)$. Thus, combining the above with (5), linearity of expectations, and the fact that the probability of a mistake on a given round is at most 1, we obtain

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left[\hat{Y}_t \neq Y_t\right]\right] = O\left(\sum_{t=1}^{T} \min_{m\in\{1,\ldots,t-1\}} \beta_t(m) \wedge 1 \right).$$

Fixing any $M \in \mathbb{N}$, we have that for any $T > M$,

$$\begin{aligned}
&\sum_{t=1}^{T} \min_{m \in \{1,\ldots,t-1\}} \beta_t(m) \wedge 1 \leq M + \sum_{t=M+1}^{T} \beta_t(M) \wedge 1 \\
&\leq M + \sum_{t=M+1}^{T} \mathbb{1}\left[\frac{d\mathrm{Log}(M/d)}{M} \geq \sum_{r=t-M+1}^{t} \Delta_r\right] \frac{d\mathrm{Log}(M/d)}{M} \\
&\qquad + \sum_{t=M+1}^{T} \mathbb{1}\left[\sum_{r=t-M+1}^{t} \Delta_r > \frac{d\mathrm{Log}(M/d)}{M}\right] \\
&\leq M + \frac{d\mathrm{Log}(M/d)}{M} T + \sum_{t=M+1}^{T} \frac{M}{d\mathrm{Log}(M/d)} \sum_{r=t-M+1}^{t} \Delta_r \\
&= \frac{d\mathrm{Log}(M/d)}{M} T + g_M(T),
\end{aligned}$$

where g_M is a function satisfying $g_M(T) = o(T)$ (holding M fixed). Since this is true of *any* $M \in \mathbb{N}$, we have that

$$\begin{aligned}
\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \min_{m \in \{1,\ldots,t-1\}} \beta_t(m) \wedge 1 &\leq \lim_{M\to\infty} \lim_{T\to\infty} \frac{d\mathrm{Log}(M/d)}{M} + \frac{g_M(T)}{T} \\
&= \lim_{M\to\infty} \frac{d\mathrm{Log}(M/d)}{M} = 0,
\end{aligned}$$

so that $\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left[\hat{Y}_t \neq Y_t\right]\right] = o(T)$, as claimed.

For many concept spaces of interest, the condition $\sum_{t=1}^{T} \Delta_t = o(T)$ in Corollary 1 is also a *necessary* condition for *any* algorithm to guarantee a sublinear number of mistakes. For simplicity, we will establish this for the class of *homogeneous linear separators* on $\mathbb{R}^2$, with $\mathcal{P}$ the uniform distribution on the unit circle, in the following theorem. This can easily be extended to many other spaces, including higher-dimensional linear separators or axis-aligned rectangles in $\mathbb{R}^k$, by embedding an analogous setup into those spaces.

Theorem 2. *If* $\mathcal{X} = \{x \in \mathbb{R}^2 : \|x\| = 1\}$, $\mathcal{P}$ *is* Uniform$(\mathcal{X})$, *and* $\mathbb{C} = \{x \mapsto 2\mathbb{1}[w \cdot x \geq 0] - 1 : w \in \mathbb{R}^2, \|w\| = 1\}$ *is the class of homogeneous linear separators, then for any sequence* $\boldsymbol{\Delta}$ *in* $[0,1]$, *there exists an algorithm* $\mathcal{A}$ *such that* $\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left[\hat{Y}_t \neq Y_t\right]\right] = o(T)$ *for* every *choice of* $\mathbf{h}^* \in S_{\boldsymbol{\Delta}}$ if and only if $\sum_{t=1}^{T} \Delta_t = o(T)$.

Proof. The "if" part follows immediately from Corollary 1. For the "only if" part, suppose $\boldsymbol{\Delta}$ is such that $\sum_{t=1}^{T} \Delta_t \neq o(T)$. It suffices to argue that for any algorithm $\mathcal{A}$, there exists a choice of $\mathbf{h}^* \in S_{\boldsymbol{\Delta}}$ for which $\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left[\hat{Y}_t \neq Y_t\right]\right] \neq o(T)$. Toward this end, fix any algorithm $\mathcal{A}$. We proceed by the probabilistic method, constructing a *random* sequence $\mathbf{h}^* \in S_{\boldsymbol{\Delta}}$. Let $B_1, B_2, \ldots$ be independent Bernoulli(1/2) random variables (also independent from the unlabeled data $X_1, X_2, \ldots$). We define the sequence $\mathbf{h}^*$ inductively. For simplicity, we will represent each classifier in *polar* coordinates, writing h_ϕ (for $\phi \in \mathbb{R}$) to denote the classifier that, for $x = (x_1, x_2)$, classifies x as $h_\phi(x) = 2\mathbb{1}[x_1 \cos(\phi) + x_2 \sin(\phi) \geq 0] - 1$; note that $h_\phi = h_{\phi+2\pi}$ for every $\phi \in \mathbb{R}$. As a base case, start by defining a function $h_0^* = h_0$, and letting $\phi_0 = 0$. Now for any $t \in \mathbb{N}$, supposing h_{t-1}^* is already defined to be $h_{\phi_{t-1}}$, we define $\phi_t = \phi_{t-1} + \min\{\Delta_t, 1/2\}\pi B_t$, and $h_t^* = h_{\phi_t}$. Note that $\mathcal{P}(x : h_t^*(x) \neq h_{t-1}^*(x)) = \min\{\Delta_t, 1/2\}$ for every $t \in \mathbb{N}$, so that this inductively defines a (random) choice of $\mathbf{h}^* \in S_{\boldsymbol{\Delta}}$.

For each $t \in \mathbb{N}$, let $Y_t = h_t^*(X_t)$. Now fix any algorithm $\mathcal{A}$, and consider the sequence $\hat{Y}_t$ of predictions the algorithm makes for points X_t, when the target sequence $\mathbf{h}^*$ is chosen as above. Then note that, for any $t \in \mathbb{N}$, since $\hat{Y}_t$ and B_t are independent,

$$\begin{aligned}\mathbb{P}\left(\hat{Y}_t \neq Y_t\right) &\geq \mathbb{E}\left[\mathbb{P}\left(\hat{Y}_t \neq Y_t \middle| \hat{Y}_t, \phi_{t-1}\right)\right] \\ &\geq \mathbb{E}\left[\frac{1}{2}\mathbb{P}\left(h_{\phi_{t-1}+\min\{\Delta_t,1/2\}\pi}(X_t) \neq h_{\phi_{t-1}-\min\{\Delta_t,1/2\}\pi}(X_t) \middle| \phi_{t-1}\right)\right].\end{aligned}$$

Furthermore, since $\min\{\Delta_t, 1/2\}\pi \leq \pi/2$, the regions $\{x : h_{\phi_{t-1}+\min\{\Delta_t,1/2\}\pi}(x) \neq h_{\phi_{t-1}}(x)\}$ and $\{x : h_{\phi_{t-1}-\min\{\Delta_t,1/2\}\pi}(x) \neq h_{\phi_{t-1}}(x)\}$ have zero-probability overlap (indeed, are disjoint if $\Delta_t < 1/2$), the above equals $\min\{\Delta_t, 1/2\}$.

By Fatou's lemma, linearity of expectations, and the law of total expectation, we have that

$$\begin{aligned}\mathbb{E}\left[\limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}[\hat{Y}_t \neq Y_t] \middle| \mathbf{h}^*\right]\right] &\geq \limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{P}\left(\hat{Y}_t \neq Y_t\right) \\ &\geq \limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \min\{\Delta_t, 1/2\}.\end{aligned}$$

Since $\sum_{t=1}^{T} \Delta_t \neq o(T)$, the rightmost expression is strictly greater than zero. Thus, it must be that, with probility strictly greater than 0,

$$\limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}[\hat{Y}_t \neq Y_t] \middle| \mathbf{h}^*\right] > 0.$$

In particular, this implies that there exists a (nonrandom) choice of the sequence $\mathbf{h}^* \in S_{\boldsymbol{\Delta}}$ for which $\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left[\hat{Y}_t \neq Y_t\right]\right] \neq o(T)$. Since this holds for *any* choice of the algorithm $\mathcal{A}$, this completes the proof.

5 Polynomial-Time Algorithms for Linear Separators

In this section, we suppose $\Delta_t = \Delta$ for every $t \in \mathbb{N}$, for a fixed constant $\Delta > 0$, and we consider the special case of learning homogeneous linear separators in

$\mathbb{R}^k$ under a uniform distribution on the origin-centered unit sphere. In this case, the analysis of [HL94] mentioned in Section 3.3 implies that it is possible to achieve a bound on the error rate that is $\tilde{O}(d\sqrt{\Delta})$, using an algorithm that runs in time $\mathrm{poly}(d, 1/\Delta, \log(1/\delta))$ (and independent of t) for each prediction. This also implies that it is possible to achieve expected number of mistakes among T predictions that is $\tilde{O}(d\sqrt{\Delta}) \times T$. [CMEDV10] have since proven that a variant of the Perceptron algorithm is capable of achieving an expected number of mistakes $\tilde{O}((d\Delta)^{1/4}) \times T$.[6]

Below, we improve on this result by showing that there exists an efficient algorithm that achieves a bound on the error rate that is $\tilde{O}(\sqrt{d\Delta})$, as was possible with the inefficient algorithm of [HL94,Lon99] mentioned in Section 3.1. This leads to a bound on the expected number of mistakes that is $\tilde{O}(\sqrt{d\Delta}) \times T$. Furthermore, our approach also allows us to present the method as an *active learning* algorithm, and to bound the expected number of queries, as a function of the number of samples T, by $\tilde{O}(\sqrt{d\Delta}) \times T$. The technique is based on a modification of the algorithm of [HL94], replacing an empirical risk minimization step with (a modification of) the computationally-efficient algorithm of [ABL13].

Formally, define the class of homogeneous linear separators as the set of classifiers $h_w : \mathbb{R}^d \to \{-1, +1\}$, for $w \in \mathbb{R}^d$ with $\|w\| = 1$, such that $h_w(x) = \mathrm{sign}(w \cdot x)$ for every $x \in \mathbb{R}^d$.

5.1 An Improved Guarantee for a Polynomial-Time Algorithm

We have the following result.

Theorem 3. *When $\mathbb{C}$ is the space of homogeneous linear separators (with $d \geq 4$) and $\mathcal{P}$ is the uniform distribution on the surface of the origin-centered unit sphere in $\mathbb{R}^d$, when $\Delta_t = \Delta > 0$ (constant) for all $t \in \mathbb{N}$, for any $\delta \in (0, 1/e)$, there is an algorithm that runs in time* $\mathrm{poly}(d, 1/\Delta, \log(1/\delta))$ *for each time t, such that for every sufficiently large $t \in \mathbb{N}$, with probability at least $1 - \delta$,*

$$\mathrm{er}_t(\hat{h}_t) = O\left(\sqrt{\Delta d \log\left(\frac{1}{\delta}\right)}\right).$$

Also, running this algorithm with $\delta = \sqrt{\Delta d} \wedge 1/e$, the expected number of mistakes among the first T instances is $O\left(\sqrt{\Delta d \log\left(\frac{1}{\Delta d}\right)} T\right)$. Furthermore, the algorithm can be run as an active learning *algorithm, in which case, for this choice of δ, the expected number of labels requested by the algorithm among the first T instances is $O\left(\sqrt{\Delta d} \log^{3/2}\left(\frac{1}{\Delta d}\right) T\right)$.*

[6] This work in fact studies a much broader model of drift, which in fact allows the distribution $\mathcal{P}$ to vary with time as well. However, this $\tilde{O}((d\Delta)^{1/4}) \times T$ result can be obtained from their more-general theorem by calculating the various parameters for this particular setting.

We first state the algorithm used to obtain this result. It is primarily based on a margin-based learning strategy of [ABL13], combined with an initialization step based on a modified Perceptron rule from [DKM09,CMEDV10]. For $\tau > 0$ and $x \in \mathbb{R}$, define $\ell_\tau(x) = \max\left\{0, 1 - \frac{x}{\tau}\right\}$. Consider the following algorithm and subroutine; parameters δ_k, m_k, τ_k, r_k, b_k, α, and κ will all be specified in the context of the proof (see Lemmas 4 and 8); we suppose $M = \sum_{k=0}^{\lceil \log_2(1/\alpha) \rceil} m_k$.

Algorithm: DriftingHalfspaces
0. Let $\tilde{h}_0$ be an arbitrary classifier in $\mathbb{C}$
1. For $i = 1, 2, \ldots$
2. $\quad \tilde{h}_i \leftarrow \text{ABL}(M(i-1), \tilde{h}_{i-1})$

Subroutine: ModPerceptron$(t, \tilde{h})$
0. Let w_t be any element of $\mathbb{R}^d$ with $\|w_t\| = 1$
1. For $m = t+1, t+2, \ldots, t+m_0$
2. $\quad$ Choose $\hat{h}_m = \tilde{h}$ (i.e., predict $\hat{Y}_m = \tilde{h}(X_m)$ as the prediction for Y_m)
3. $\quad$ Request the label Y_m
4. $\quad$ If $h_{w_{m-1}}(X_m) \neq Y_m$
5. $\quad\quad w_m \leftarrow w_{m-1} - 2(w_{m-1} \cdot X_m) X_m$
6. $\quad$ Else $w_m \leftarrow w_{m-1}$
7. Return w_{t+m_0}

Subroutine: ABL$(t, \tilde{h})$
0. Let w_0 be the return value of ModPerceptron$(t, \tilde{h})$
1. For $k = 1, 2, \ldots, \lceil \log_2(1/\alpha) \rceil$
2. $\quad W_k \leftarrow \{\}$
3. $\quad$ For $s = t + \sum_{j=0}^{k-1} m_j + 1, \ldots, t + \sum_{j=0}^{k} m_j$
4. $\quad\quad$ Choose $\hat{h}_s = \tilde{h}$ (i.e., predict $\hat{Y}_s = \tilde{h}(X_s)$ as the prediction for Y_s)
5. $\quad\quad$ If $|w_{k-1} \cdot X_s| \leq b_{k-1}$, Request label Y_s and let $W_k \leftarrow W_k \cup \{(X_s, Y_s)\}$
6. $\quad$ Find $v_k \in \mathbb{R}^d$ with $\|v_k - w_{k-1}\| \leq r_k$, $0 < \|v_k\| \leq 1$, and
$$\sum_{(x,y) \in W_k} \ell_{\tau_k}(y(v_k \cdot x)) \leq \inf_{v : \|v - w_{k-1}\| \leq r_k} \sum_{(x,y) \in W_k} \ell_{\tau_k}(y(v \cdot x)) + \kappa |W_k|$$
7. $\quad$ Let $w_k = \frac{1}{\|v_k\|} v_k$
8. Return $h_{w_{\lceil \log_2(1/\alpha) \rceil - 1}}$

The general idea here is to replace empirical risk minimization in the method of [HL94] discussed above with a computationally efficient method, due to [ABL13]: i.e., the subroutine ABL above. For technical reasons, we apply this method to batches of M samples at a time, and simply use the classifier learned from the previous batch to make the predictions. The method of [ABL13] was originally proposed for the problem of *agnostic* learning, to error rate within a constant factor of the optimal. To use this for our purposes, we set up an analogy between the best achievable error rate in agnostic learning and a value $O(\Delta M)$ in the execution of ABL here. This is reasonable, since we there should exist a classifier with this as its *average* error rate over the M target concepts.

The analysis of [ABL13] required this method to be initialized with a reasonably accurate classifier (constant bound on its error rate). For this, we find (in Lemma 3) that the modified Perceptron algorithm (of [DKM09,CMEDV10]) suffices. The ABL algorithm then iteratively refines a hypothesis w_k by taking a number of samples within a slab of width $b_{k-1} \propto 2^{-k}/\sqrt{d}$ around the previous hypothesis separator w_{k-1}, and optimizing a weighted hinge loss (subject to a constraint that the new hypothesis not be too far from the previous). The analysis (particularly Lemma 8) then reveals that the hypothesis w_k approaches a classifier w^* with error rate $O(\Delta M)$ with respect to all of the target concepts in the give batch.

We note that, even after noting the analogy between the noise rate in agnostic learning and the value $O(\Delta M)$ here, the analysis below does not follow immediately from that of [ABL13]. This is because the sample size M that would be required by the analysis of [ABL13] to achieve error rate within a constant factor of the noise rate would be too large (by a factor of d) for our purposes. In particular, since the value ΔM is increasing in M, converting that original analysis to our present setting would result in a bound on $\mathrm{er}_t(\hat{h}_t)$ larger than that stated in Theorem 3 by roughly a factor of $\sqrt{d}$. The analysis below refines several aspects of the analysis, using stronger concentration arguments for the weighted hinge loss, and generally being more careful in bounding the error rate in terms of the weighted hinge loss performance. We thereby reduce the bound to the result stated in Theorem 3.

Before stating the proof, we have a few additional lemmas that will be needed. The following result for ModPerceptron was proven by [CMEDV10].

Lemma 2. *Suppose* $\Delta < \frac{1}{512}$. *Consider the values* w_m *obtained during the execution of* ModPerceptron$(t, \tilde{h})$. $\forall m \in \{t+1, \ldots, t+m_0\}$, $\mathcal{P}(x : h_{w_m}(x) \neq h^*_m(x)) \leq \mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x))$. *Furthermore, letting* $c_1 = \frac{\pi^2}{d \cdot 400 \cdot 2^{15}}$, *if* $\mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x)) \geq 1/32$, *then with probability at least* $1/64$, $\mathcal{P}(x : h_{w_m}(x) \neq h^*_m(x)) \leq (1-c_1)\mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x))$.

This implies the following.

Lemma 3. *Suppose* $\Delta \leq \frac{\pi^2}{400 \cdot 2^{27}(d+\ln(4/\delta))}$. *For* $m_0 = \max\{\lceil 128(1/c_1)\ln(32)\rceil, \lceil 512\ln(\frac{4}{\delta})\rceil\}$, *with probability at least* $1-\delta/4$, ModPerceptron$(t, \tilde{h})$ *returns a vector* w *with* $\mathcal{P}(x : h_w(x) \neq h^*_{t+m_0+1}(x)) \leq 1/16$.

Proof. By Lemma 2 and a union bound, in general we have

$$\mathcal{P}(x : h_{w_m}(x) \neq h^*_{m+1}(x)) \leq \mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x)) + \Delta. \qquad (6)$$

Furthermore, if $\mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x)) \geq 1/32$, then wth probability at least $1/64$,

$$\mathcal{P}(x : h_{w_m}(x) \neq h^*_{m+1}(x)) \leq (1-c_1)\mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x)) + \Delta. \qquad (7)$$

In particular, this implies that the number N of values $m \in \{t+1, \ldots, t+m_0\}$ with either $\mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x)) < 1/32$ or $\mathcal{P}(x : h_{w_m}(x) \neq h^*_{m+1}(x)) \leq$

$(1-c_1)\mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x)) + \Delta$ is lower-bounded by a Binomial$(m, 1/64)$ random variable. Thus, a Chernoff bound implies that with probability at least $1 - \exp\{-m_0/512\} \geq 1 - \delta/4$, we have $N \geq m_0/128$. Suppose this happens.

Since $\Delta m_0 \leq 1/32$, if any $m \in \{t+1, \ldots, t+m_0\}$ has $\mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x)) < 1/32$, then inductively applying (6) implies that $\mathcal{P}(x : h_{w_{t+m_0}}(x) \neq h^*_{t+m_0+1}(x)) \leq 1/32 + \Delta m_0 \leq 1/16$. On the other hand, if all $m \in \{t+1, \ldots, t+m_0\}$ have $\mathcal{P}(x : h_{w_{m-1}}(x) \neq h^*_m(x)) \geq 1/32$, then in particular we have N values of $m \in \{t+1, \ldots, t+m_0\}$ satisfying (7). Combining this fact with (6) inductively, we have that

$$\mathcal{P}(x : h_{w_{t+m_0}}(x) \neq h^*_{t+m_0+1}(x)) \leq (1-c_1)^N \mathcal{P}(x : h_{w_t}(x) \neq h^*_{t+1}(x)) + \Delta m_0$$
$$\leq (1-c_1)^{(1/c_1)\ln(32)} \mathcal{P}(x : h_{w_t}(x) \neq h^*_{t+1}(x)) + \Delta m_0 \leq \frac{1}{32} + \Delta m_0 \leq \frac{1}{16}.$$

Next, we consider the execution of ABL$(t, \tilde{h})$, and let the sets W_k be as in that execution. We will denote by w^* the weight vector with $\|w^*\| = 1$ such that $h^*_{t+m_0+1} = h_{w^*}$. Also denote by $M_1 = M - m_0$.

The proof relies on a few results proven in the work of [ABL13], which we summarize in the following lemmas. Although the results were proven in a slightly different setting in that work (namely, agnostic learning under a fixed joint distribution), one can easily verify that their proofs remain valid in our present context as well.

Lemma 4. *[ABL13] Fix any $k \in \{1, \ldots, \lceil \log_2(1/\alpha) \rceil\}$. For a universal constant $c_7 > 0$, suppose $b_{k-1} = c_7 2^{1-k}/\sqrt{d}$, and let $z_k = \sqrt{r_k^2/(d-1) + b_{k-1}^2}$. For a universal constant $c_1 > 0$, if $\|w^* - w_{k-1}\| \leq r_k$,*

$$\left| \mathbb{E}\left[\sum_{(x,y)\in W_k} \ell_{\tau_k}(|w^* \cdot x|) \middle| w_{k-1}, |W_k| \right] - \mathbb{E}\left[\sum_{(x,y)\in W_k} \ell_{\tau_k}(y(w^* \cdot x)) \middle| w_{k-1}, |W_k| \right] \right|$$
$$\leq c_1 |W_k| \sqrt{2^k \Delta M_1} \frac{z_k}{\tau_k}.$$

Lemma 5. *[BL13] For any $c > 0$, there is a constant $c' > 0$ depending only on c (i.e., not depending on d) such that, for any $u, v \in \mathbb{R}^d$ with $\|u\| = \|v\| = 1$, letting $\sigma = \mathcal{P}(x : h_u(x) \neq h_v(x))$, if $\sigma < 1/2$, then*

$$\mathcal{P}\left(x : h_u(x) \neq h_v(x) \text{ and } |v \cdot x| \geq c' \frac{\sigma}{\sqrt{d}} \right) \leq c\sigma.$$

The following is a well-known lemma concerning concentration around the equator for the uniform distribution (see e.g., [DKM09,BBZ07,ABL13]); for instance, it easily follows from the formulas for the area in a spherical cap derived by [Li11].

Lemma 6. *For any constant $C > 0$, there are constants $c_2, c_3 > 0$ depending only on C (i.e., independent of d) such that, for any $w \in \mathbb{R}^d$ with $\|w\| = 1$, $\forall \gamma \in [0, C/\sqrt{d}]$,*

$$c_2\gamma\sqrt{d} \leq \mathcal{P}\left(x : |w \cdot x| \leq \gamma\right) \leq c_3\gamma\sqrt{d}.$$

Based on this lemma, [ABL13] prove the following.

Lemma 7. *[ABL13] For $X \sim \mathcal{P}$, for any $w \in \mathbb{R}^d$ with $\|w\| = 1$, for any $C > 0$ and $\tau, b \in [0, C/\sqrt{d}]$, for c_2, c_3 as in Lemma 6,*

$$\mathbb{E}\left[\ell_\tau(|w^* \cdot X|)\middle| |w \cdot X| \leq b\right] \leq \frac{c_3\tau}{c_2 b}.$$

The following is a slightly stronger version of a result of [ABL13] (specifically, the size of m_k, and consequently the bound on $|W_k|$, are both improved by a factor of d compared to the original result).

Lemma 8. *Fix any $\delta \in (0, 1/e)$. For universal constants $c_4, c_5, c_6, c_7, c_8, c_9, c_{10} \in (0, \infty)$, for an appropriate choice of $\kappa \in (0, 1)$ (a universal constant), if $\alpha = c_9\sqrt{\Delta d \log\left(\frac{1}{\kappa\delta}\right)}$, for every $k \in \{1, \ldots, \lceil\log_2(1/\alpha)\rceil\}$, if $b_{k-1} = c_7 2^{1-k}/\sqrt{d}$, $\tau_k = c_8 2^{-k}/\sqrt{d}$, $r_k = c_{10}2^{-k}$, $\delta_k = \delta/(\lceil\log_2(4/\alpha)\rceil - k)^2$, and $m_k = \left\lceil c_5 \frac{2^k}{\kappa^2} d \log\left(\frac{1}{\kappa\delta_k}\right)\right\rceil$, and if $\mathcal{P}(x : h_{w_{k-1}}(x) \neq h_{w^*}(x)) \leq 2^{-k-3}$, then with probability at least $1 - (4/3)\delta_k$, $|W_k| \leq c_6 \frac{1}{\kappa^2} d \log\left(\frac{1}{\kappa\delta_k}\right)$ and $\mathcal{P}(x : h_{w_k}(x) \neq h_{w^*}(x)) \leq 2^{-k-4}$.*

Proof. By Lemma 6, and a Chernoff and union bound, for an appropriately large choice of c_5 and any $c_7 > 0$, letting c_2, c_3 be as in Lemma 6 (with $C = c_7 \vee (c_8/2)$), with probability at least $1 - \delta_k/3$,

$$c_2 c_7 2^{-k} m_k \leq |W_k| \leq 4 c_3 c_7 2^{-k} m_k. \tag{8}$$

The claimed upper bound on $|W_k|$ follows from this second inequality.

Next note that, if $\mathcal{P}(x : h_{w_{k-1}}(x) \neq h_{w^*}(x)) \leq 2^{-k-3}$, then

$$\max\{\ell_{\tau_k}(y(w^* \cdot x)) : x \in \mathbb{R}^d, |w_{k-1} \cdot x| \leq b_{k-1}, y \in \{-1, +1\}\} \leq c_{11}\sqrt{d}$$

for some universal constant $c_{11} > 0$. Furthermore, since $\mathcal{P}(x : h_{w_{k-1}}(x) \neq h_{w^*}(x)) \leq 2^{-k-3}$, we know that the angle between w_{k-1} and w^* is at most $2^{-k-3}\pi$, so that

$$\begin{aligned}\|w_{k-1} - w^*\| &= \sqrt{2 - 2w_{k-1} \cdot w^*} \leq \sqrt{2 - 2\cos(2^{-k-3}\pi)} \\ &\leq \sqrt{2 - 2\cos^2(2^{-k-3}\pi)} = \sqrt{2}\sin(2^{-k-3}\pi) \leq 2^{-k-3}\pi\sqrt{2}.\end{aligned}$$

For $c_{10} = \pi\sqrt{2}2^{-3}$, this is r_k. By Hoeffding's inequality (under the conditional distribution given $|W_k|$), the law of total probability, Lemma 4, and linearity of conditional expectations, with probability at least $1 - \delta_k/3$, for $X \sim \mathcal{P}$,

$$\sum_{(x,y)\in W_k} \ell_{\tau_k}(y(w^* \cdot x)) \leq |W_k| \mathbb{E}\left[\ell_{\tau_k}(|w^* \cdot X|) \middle| w_{k-1}, |w_{k-1} \cdot X| \leq b_{k-1}\right]$$
$$+ c_1 |W_k| \sqrt{2^k \Delta M_1} \frac{z_k}{\tau_k} + \sqrt{|W_k|(1/2) c_{11}^2 d \ln(3/\delta_k)}. \quad (9)$$

We bound each term on the right hand side separately. By Lemma 7, the first term is at most $|W_k| \frac{c_3 \tau_k}{c_2 b_{k-1}} = |W_k| \frac{c_3 c_8}{2 c_2 c_7}$. Next,

$$\frac{z_k}{\tau_k} = \frac{\sqrt{c_{10}^2 2^{-2k}/(d-1) + 4c_7^2 2^{-2k}/d}}{c_8 2^{-k}/\sqrt{d}} \leq \frac{\sqrt{2c_{10}^2 + 4c_7^2}}{c_8},$$

while $2^k \leq 2/\alpha$ so that the second term is at most

$$\sqrt{2} c_1 \frac{\sqrt{2c_{10}^2 + 4c_7^2}}{c_8} |W_k| \sqrt{\frac{\Delta m}{\alpha}}.$$

Noting that

$$M_1 = \sum_{k'=1}^{\lceil \log_2(1/\alpha) \rceil} m_{k'} \leq \frac{32 c_5}{\kappa^2} \frac{1}{\alpha} d \log\left(\frac{1}{\kappa\delta}\right), \quad (10)$$

we find that the second term on the right hand side of (9) is at most

$$\sqrt{\frac{c_5}{c_9}} \frac{8c_1}{\kappa} \frac{\sqrt{2c_{10}^2 + 4c_7^2}}{c_8} |W_k| \sqrt{\frac{\Delta d \log\left(\frac{1}{\kappa\delta}\right)}{\alpha^2}} = \frac{8c_1\sqrt{c_5}}{\kappa} \frac{\sqrt{2c_{10}^2 + 4c_7^2}}{c_8 c_9} |W_k|.$$

Finally, since $d \ln(3/\delta_k) \leq 2d \ln(1/\delta_k) \leq \frac{2\kappa^2}{c_5} 2^{-k} m_k$, and (8) implies $2^{-k} m_k \leq \frac{1}{c_2 c_7} |W_k|$, the third term on the right hand side of (9) is at most

$$|W_k| \frac{c_{11} \kappa}{\sqrt{c_2 c_5 c_7}}.$$

Altogether, we have

$$\sum_{(x,y)\in W_k} \ell_{\tau_k}(y(w^* \cdot x)) \leq |W_k| \left(\frac{c_3 c_8}{2 c_2 c_7} + \frac{8c_1\sqrt{c_5}}{\kappa} \frac{\sqrt{2c_{10}^2 + 4c_7^2}}{c_8 c_9} + \frac{c_{11}\kappa}{\sqrt{c_2 c_5 c_7}} \right).$$

Taking $c_9 = 1/\kappa^3$ and $c_8 = \kappa$, this is at most

$$\kappa |W_k| \left(\frac{c_3}{2 c_2 c_7} + 8 c_1 \sqrt{c_5} \sqrt{2c_{10}^2 + 4c_7^2} + \frac{c_{11}}{\sqrt{c_2 c_5 c_7}} \right).$$

Next, note that because $h_{w_k}(x) \neq y \Rightarrow \ell_{\tau_k}(y(v_k \cdot x)) \geq 1$, and because (as proven above) $\|w^* - w_{k-1}\| \leq r_k$,

$$|W_k| \mathrm{er}_{W_k}(h_{w_k}) \leq \sum_{(x,y)\in W_k} \ell_{\tau_k}(y(v_k \cdot x)) \leq \sum_{(x,y)\in W_k} \ell_{\tau_k}(y(w^* \cdot x)) + \kappa |W_k|.$$

Combined with the above, we have

$$|W_k|\mathrm{er}_{W_k}(h_{w_k}) \leq \kappa|W_k|\left(1 + \frac{c_3}{2c_2c_7} + 8c_1\sqrt{c_5}\sqrt{2c_{10}^2 + 4c_7^2} + \frac{c_{11}}{\sqrt{c_2c_5c_7}}\right).$$

Let $c_{12} = 1 + \frac{c_3}{2c_2c_7} + 8c_1\sqrt{c_5}\sqrt{2c_{10}^2 + 4c_7^2} + \frac{c_{11}}{\sqrt{c_2c_5c_7}}$. Furthermore,

$$\begin{aligned}|W_k|\mathrm{er}_{W_k}(h_{w_k}) &= \sum_{(x,y)\in W_k} \mathbb{1}[h_{w_k}(x) \neq y] \\ &\geq \sum_{(x,y)\in W_k} \mathbb{1}[h_{w_k}(x) \neq h_{w^*}(x)] - \sum_{(x,y)\in W_k} \mathbb{1}[h_{w^*}(x) \neq y].\end{aligned}$$

For an appropriately large value of c_5, by a Chernoff bound, with probability at least $1 - \delta_k/3$,

$$\sum_{s=t+\sum_{j=0}^{k-1} m_j+1}^{t+\sum_{j=0}^{k} m_j} \mathbb{1}[h_{w^*}(X_s) \neq Y_s] \leq 2e\Delta M_1 m_k + \log_2(3/\delta_k).$$

In particular, this implies

$$\sum_{(x,y)\in W_k} \mathbb{1}[h_{w^*}(x) \neq y] \leq 2e\Delta M_1 m_k + \log_2(3/\delta_k),$$

so that

$$\sum_{(x,y)\in W_k} \mathbb{1}[h_{w_k}(x) \neq h_{w^*}(x)] \leq |W_k|\mathrm{er}_{W_k}(h_{w_k}) + 2e\Delta M_1 m_k + \log_2(3/\delta_k).$$

Noting that (10) and (8) imply

$$\begin{aligned}\Delta M_1 m_k &\leq \Delta\frac{32c_5}{\kappa^2}\frac{d\log\left(\frac{1}{\kappa\delta}\right)}{c_9\sqrt{\Delta d\log\left(\frac{1}{\kappa\delta}\right)}}\frac{2^k}{c_2c_7}|W_k| \leq \frac{32c_5}{c_2c_7c_9\kappa^2}\sqrt{\Delta d\log\left(\frac{1}{\kappa\delta}\right)}2^k|W_k| \\ &= \frac{32c_5}{c_2c_7c_9^2\kappa^2}\alpha 2^k|W_k| = \frac{32c_5\kappa^4}{c_2c_7}\alpha 2^k|W_k| \leq \frac{32c_5\kappa^4}{c_2c_7}|W_k|,\end{aligned}$$

and (8) implies $\log_2(3/\delta_k) \leq \frac{2\kappa^2}{c_2c_5c_7}|W_k|$, altogether we have

$$\begin{aligned}\sum_{(x,y)\in W_k} \mathbb{1}[h_{w_k}(x) \neq h_{w^*}(x)] &\leq |W_k|\mathrm{er}_{W_k}(h_{w_k}) + \frac{64ec_5\kappa^4}{c_2c_7}|W_k| + \frac{2\kappa^2}{c_2c_5c_7}|W_k| \\ &\leq \kappa|W_k|\left(c_{12} + \frac{64ec_5\kappa^3}{c_2c_7} + \frac{2\kappa}{c_2c_5c_7}\right).\end{aligned}$$

Letting $c_{13} = c_{12} + \frac{64ec_5}{c_2c_7} + \frac{2}{c_2c_5c_7}$, and noting $\kappa \leq 1$, we have $\sum_{(x,y)\in W_k} \mathbb{1}[h_{w_k}(x) \neq h_{w^*}(x)] \leq c_{13}\kappa|W_k|$.

Lemma 1 (applied under the conditional distribution given $|W_k|$) and the law of total probability imply that with probability at least $1-\delta_k/3$,

$$|W_k|\mathcal{P}\left(x : h_{w_k}(x) \neq h_{w^*}(x) \Big| |w_{k-1}\cdot x| \leq b_{k-1}\right) \leq \sum_{(x,y)\in W_k} \mathbb{1}[h_{w_k}(x) \neq h_{w^*}(x)] + c_{14}\sqrt{|W_k|(d\log(|W_k|/d) + \log(1/\delta_k))},$$

for a universal constant $c_{14} > 0$. Combined with the above, and the fact that (8) implies $\log(1/\delta_k) \leq \frac{\kappa^2}{c_2c_5c_7}|W_k|$ and

$$\begin{aligned} d\log(|W_k|/d) &\leq d\log\left(\frac{8c_3c_5c_7\log\left(\frac{1}{\kappa\delta_k}\right)}{\kappa^2}\right) \\ &\leq d\log\left(\frac{8c_3c_5c_7}{\kappa^3\delta_k}\right) \leq 3\log(8\max\{c_3,1\}c_5)c_5 d\log\left(\frac{1}{\kappa\delta_k}\right) \\ &\leq 3\log(8\max\{c_3,1\})\kappa^2 2^{-k}m_k \leq \frac{3\log(8\max\{c_3,1\})}{c_2c_7}\kappa^2|W_k|, \end{aligned}$$

we have

$$\begin{aligned} &|W_k|\mathcal{P}\left(x : h_{w_k}(x) \neq h_{w^*}(x) \Big| |w_{k-1}\cdot x| \leq b_{k-1}\right) \\ &\leq c_{13}\kappa|W_k| + c_{14}\sqrt{|W_k|\left(\frac{3\log(8\max\{c_3,1\})}{c_2c_7}\kappa^2|W_k| + \frac{\kappa^2}{c_2c_5c_7}|W_k|\right)} \\ &= \kappa|W_k|\left(c_{13} + c_{14}\sqrt{\frac{3\log(8\max\{c_3,1\})}{c_2c_7} + \frac{1}{c_2c_5c_7}}\right). \end{aligned}$$

Thus, letting $c_{15} = \left(c_{13} + c_{14}\sqrt{\frac{3\log(8\max\{c_3,1\})}{c_2c_7} + \frac{1}{c_2c_5c_7}}\right)$, we have

$$\mathcal{P}\left(x : h_{w_k}(x) \neq h_{w^*}(x) \Big| |w_{k-1}\cdot x| \leq b_{k-1}\right) \leq c_{15}\kappa. \tag{11}$$

Next, note that $\|v_k - w_{k-1}\|^2 = \|v_k\|^2 + 1 - 2\|v_k\|\cos(\pi\mathcal{P}(x : h_{w_k}(x) \neq h_{w_{k-1}}(x)))$. Thus, one implication of the fact that $\|v_k - w_{k-1}\| \leq r_k$ is that $\frac{\|v_k\|}{2} + \frac{1-r_k^2}{2\|v_k\|} \leq \cos(\pi\mathcal{P}(x : h_{w_k}(x) \neq h_{w_{k-1}}(x)))$; since the left hand side is positive, we have $\mathcal{P}(x : h_{w_k}(x) \neq h_{w_{k-1}}(x)) < 1/2$. Additionally, by differentiating, one can easily verify that for $\phi \in [0,\pi]$, $x \mapsto \sqrt{x^2+1-2x\cos(\phi)}$ is minimized at $x = \cos(\phi)$, in which case $\sqrt{x^2+1-2x\cos(\phi)} = \sin(\phi)$. Thus, $\|v_k - w_{k-1}\| \geq \sin(\pi\mathcal{P}(x : h_{w_k}(x) \neq h_{w_{k-1}}(x)))$. Since $\|v_k - w_{k-1}\| \leq r_k$, we have $\sin(\pi\mathcal{P}(x : h_{w_k}(x) \neq h_{w_{k-1}}(x))) \leq r_k$. Since $\sin(\pi x) \geq x$ for all $x \in [0,1/2]$, combining this with the fact (proven above) that $\mathcal{P}(x : h_{w_k}(x) \neq h_{w_{k-1}}(x)) < 1/2$ implies $\mathcal{P}(x : h_{w_k}(x) \neq h_{w_{k-1}}(x)) \leq r_k$.

In particular, we have that both $\mathcal{P}(x : h_{w_k}(x) \neq h_{w_{k-1}}(x)) \leq r_k$ and $\mathcal{P}(x : h_{w^*}(x) \neq h_{w_{k-1}}(x)) \leq 2^{-k-3} \leq r_k$. Now Lemma 5 implies that, for any universal constant $c > 0$, there exists a corresponding universal constant $c' > 0$ such that

$$\mathcal{P}\left(x : h_{w_k}(x) \neq h_{w_{k-1}}(x) \text{ and } |w_{k-1} \cdot x| \geq c' \frac{r_k}{\sqrt{d}}\right) \leq c r_k$$

and

$$\mathcal{P}\left(x : h_{w^*}(x) \neq h_{w_{k-1}}(x) \text{ and } |w_{k-1} \cdot x| \geq c' \frac{r_k}{\sqrt{d}}\right) \leq c r_k,$$

so that (by a union bound)

$$\begin{aligned}
&\mathcal{P}\left(x : h_{w_k}(x) \neq h_{w^*}(x) \text{ and } |w_{k-1} \cdot x| \geq c' \frac{r_k}{\sqrt{d}}\right) \\
&\leq \mathcal{P}\left(x : h_{w_k}(x) \neq h_{w_{k-1}}(x) \text{ and } |w_{k-1} \cdot x| \geq c' \frac{r_k}{\sqrt{d}}\right) \\
&+ \mathcal{P}\left(x : h_{w^*}(x) \neq h_{w_{k-1}}(x) \text{ and } |w_{k-1} \cdot x| \geq c' \frac{r_k}{\sqrt{d}}\right) \leq 2 c r_k.
\end{aligned}$$

In particular, letting $c_7 = c' c_{10}/2$, we have $c' \frac{r_k}{\sqrt{d}} = b_{k-1}$. Combining this with (11), Lemma 6, and a union bound, we have that

$$\begin{aligned}
&\mathcal{P}\left(x : h_{w_k}(x) \neq h_{w^*}(x)\right) \\
&\leq \mathcal{P}\left(x : h_{w_k}(x) \neq h_{w^*}(x) \text{ and } |w_{k-1} \cdot x| \geq b_{k-1}\right) \\
&\quad + \mathcal{P}\left(x : h_{w_k}(x) \neq h_{w^*}(x) \text{ and } |w_{k-1} \cdot x| \leq b_{k-1}\right) \\
&\leq 2 c r_k + \mathcal{P}\left(x : h_{w_k}(x) \neq h_{w^*}(x) \middle| |w_{k-1} \cdot x| \leq b_{k-1}\right) \mathcal{P}\left(x : |w_{k-1} \cdot x| \leq b_{k-1}\right) \\
&\leq 2 c r_k + c_{15} \kappa c_3 b_{k-1} \sqrt{d} = \left(2^5 c c_{10} + c_{15} \kappa c_3 c_7 2^5\right) 2^{-k-4}.
\end{aligned}$$

Taking $c = \frac{1}{2^6 c_{10}}$ and $\kappa = \frac{1}{2^6 c_3 c_7 c_{15}}$, we have $\mathcal{P}(x : h_{w_k}(x) \neq h_{w^*}(x)) \leq 2^{-k-4}$, as required.

By a union bound, this occurs with probability at least $1 - (4/3)\delta_k$.

Proof (Proof of Theorem 3). We begin with the bound on the error rate. If $\Delta > \frac{\pi^2}{400 \cdot 2^{27}(d + \ln(4/\delta))}$, the result trivially holds, since then $1 \leq \frac{400 \cdot 2^{27}}{\pi^2} \sqrt{\Delta(d + \ln(4/\delta))}$. Otherwise, suppose $\Delta \leq \frac{\pi^2}{400 \cdot 2^{27}(d + \ln(4/\delta))}$.

Fix any $i \in \mathbb{N}$. Lemma 3 implies that, with probability at least $1 - \delta/4$, the w_0 returned in Step 0 of $\mathrm{ABL}(M(i-1), \tilde{h}_{i-1})$ satisfies $\mathcal{P}(x : h_{w_0}(x) \neq h^*_{M(i-1)+m_0+1}(x)) \leq 1/16$. Taking this as a base case, Lemma 8 then inductively implies that, with probability at least

$$1 - \frac{\delta}{4} - \sum_{k=1}^{\lceil \log_2(1/\alpha) \rceil} (4/3) \frac{\delta}{2(\lceil \log_2(4/\alpha) \rceil - k)^2} \geq 1 - \frac{\delta}{2}\left(1 + (4/3)\sum_{\ell=2}^{\infty} \frac{1}{\ell^2}\right) \geq 1 - \delta,$$

every $k \in \{0, 1, \ldots, \lceil \log_2(1/\alpha) \rceil\}$ has

$$\mathcal{P}(x : h_{w_k}(x) \neq h^*_{M(i-1)+m_0+1}(x)) \leq 2^{-k-4}, \tag{12}$$

and furthermore the number of labels requested during $\mathrm{ABL}(M(i-1), \tilde{h}_{i-1})$ total to at most (for appropriate universal constants $\hat{c}_1, \hat{c}_2$)

$$\begin{aligned} m_0 + \sum_{k=1}^{\lceil \log_2(1/\alpha) \rceil} |W_k| &\leq \hat{c}_1 \left(d + \ln\left(\frac{1}{\delta}\right) + \sum_{k=1}^{\lceil \log_2(1/\alpha) \rceil} d \log\left(\frac{(\lceil \log_2(4/\alpha) \rceil - k)^2}{\delta} \right) \right) \\ &\leq \hat{c}_2 d \log\left(\frac{1}{\Delta d}\right) \log\left(\frac{1}{\delta}\right). \end{aligned}$$

In particular, by a union bound, (12) implies that for every $k \in \{1, \ldots, \lceil \log_2(1/\alpha) \rceil\}$, every

$$m \in \left\{ M(i-1) + \sum_{j=0}^{k-1} m_j + 1, \ldots, M(i-1) + \sum_{j=0}^{k} m_j \right\}$$

has

$$\begin{aligned} &\mathcal{P}(x : h_{w_{k-1}}(x) \neq h^*_m(x)) \\ &\leq \mathcal{P}(x : h_{w_{k-1}}(x) \neq h^*_{M(i-1)+m_0+1}(x)) + \mathcal{P}(x : h^*_{M(i-1)+m_0+1}(x) \neq h^*_m(x)) \\ &\leq 2^{-k-3} + \Delta M. \end{aligned}$$

Thus, noting that

$$\begin{aligned} M &= \sum_{k=0}^{\lceil \log_2(1/\alpha) \rceil} m_k = \Theta\left(d + \log\left(\frac{1}{\delta}\right) + \sum_{k=1}^{\lceil \log_2(1/\alpha) \rceil} 2^k d \log\left(\frac{\lceil \log_2(1/\alpha) \rceil - k}{\delta} \right) \right) \\ &= \Theta\left(\frac{1}{\alpha} d \log\left(\frac{1}{\delta}\right) \right) = \Theta\left(\sqrt{\frac{d}{\Delta} \log\left(\frac{1}{\delta}\right)} \right), \end{aligned}$$

with probability at least $1 - \delta$,

$$\mathcal{P}(x : h_{w_{\lceil \log_2(1/\alpha) \rceil - 1}}(x) \neq h^*_{Mi}(x)) \leq O\left(\alpha + \Delta M\right) = O\left(\sqrt{\Delta d \log\left(\frac{1}{\delta}\right)} \right).$$

In particular, this implies that, with probability at least $1 - \delta$, every $t \in \{Mi + 1, \ldots, M(i+1) - 1\}$ has

$$\begin{aligned} \mathrm{er}_t(\hat{h}_t) &\leq \mathcal{P}(x : h_{w_{\lceil \log_2(1/\alpha) \rceil - 1}}(x) \neq h^*_{Mi}(x)) + \mathcal{P}(x : h^*_{Mi}(x) \neq h^*_t(x)) \\ &\leq O\left(\sqrt{\Delta d \log\left(\frac{1}{\delta}\right)} \right) + \Delta M = O\left(\sqrt{\Delta d \log\left(\frac{1}{\delta}\right)} \right), \end{aligned}$$

which completes the proof of the bound on the error rate.

Setting $\delta = \sqrt{\Delta d}$, and noting that $\mathbb{1}[\hat{Y}_t \neq Y_t] \leq 1$, we have that for any $t > M$,

$$\mathbb{P}\left(\hat{Y}_t \neq Y_t\right) = \mathbb{E}\left[\mathrm{er}_t(\hat{h}_t)\right] \leq O\left(\sqrt{\Delta d \log\left(\frac{1}{\delta}\right)}\right) + \delta = O\left(\sqrt{\Delta d \log\left(\frac{1}{\Delta d}\right)}\right).$$

Thus, by linearity of the expectation,

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left[\hat{Y}_t \neq Y_t\right]\right] \leq M + O\left(\sqrt{\Delta d \log\left(\frac{1}{\Delta d}\right)} T\right) = O\left(\sqrt{\Delta d \log\left(\frac{1}{\Delta d}\right)} T\right).$$

Furthermore, as mentioned, with probability at least $1 - \delta$, the number of labels requested during the execution of ABL$(M(i-1), \tilde{h}_{i-1})$ is at most

$$O\left(d \log\left(\frac{1}{\Delta d}\right) \log\left(\frac{1}{\delta}\right)\right).$$

Thus, since the number of labels requested during the execution of ABL$(M(i-1), \tilde{h}_{i-1})$ cannot exceed M, letting $\delta = \sqrt{\Delta d}$, the expected number of requested labels during this execution is at most

$$\begin{aligned} O\left(d \log^2\left(\frac{1}{\Delta d}\right)\right) + \sqrt{\Delta d} M &\leq O\left(d \log^2\left(\frac{1}{\Delta d}\right)\right) + O\left(d \sqrt{\log\left(\frac{1}{\Delta d}\right)}\right) \\ &= O\left(d \log^2\left(\frac{1}{\Delta d}\right)\right). \end{aligned}$$

Thus, by linearity of the expectation, the expected number of labels requested among the first T samples is at most

$$O\left(d \log^2\left(\frac{1}{\Delta d}\right) \left\lceil \frac{T}{M} \right\rceil\right) = O\left(\sqrt{\Delta d} \log^{3/2}\left(\frac{1}{\Delta d}\right) T\right),$$

which completes the proof.

Remark: The original work of [CMEDV10] additionally allowed for some number K of "jumps": times t at which $\Delta_t = 1$. Note that, in the above algorithm, since the influence of each sample is localized to the predictors trained within that "batch" of M instances, the effect of allowing such jumps would only change the bound on the number of mistakes to $\tilde{O}\left(\sqrt{d\Delta} T + \sqrt{\frac{d}{\Delta}} K\right)$. This compares favorably to the result of [CMEDV10], which is roughly $O\left((d\Delta)^{1/4} T + \frac{d^{1/4}}{\Delta^{3/4}} K\right)$. However, the result of [CMEDV10] was proven for a slightly more general setting, allowing distributions $\mathcal{P}$ that are not quite uniform (though they do require a relation between the angle between any two separators and the probability mass they disagree on, similar to that holding for the uniform distribution, which seems to require that the distributions approximately retain some properties of the uniform distribution). It is not clear whether Theorem 3 can be generalized to this larger family of distributions.

6 General Results for Active Learning

As mentioned, the above results on linear separators also provide results for the number of queries in *active learning*. One can also state quite general results on the expected number of queries and mistakes achievable by an active learning algorithm. This section provides such results, for an algorithm based on the the well-known strategy of *disagreement-based* active learning. Throughout this section, we suppose $\mathbf{h}^* \in S_\Delta$, for a given $\Delta \in (0,1]$: that is, $\mathcal{P}(x : h^*_{t+1}(x) \neq h^*_t(x)) \leq \Delta$ for all $t \in \mathbb{N}$.

First, we introduce a few definitions. For any set $\mathcal{H} \subseteq \mathbb{C}$, define the *region of disagreement*

$$\mathrm{DIS}(\mathcal{H}) = \{x \in \mathcal{X} : \exists h, g \in \mathcal{H} \text{ s.t. } h(x) \neq g(x)\}.$$

The analysis in this section is centered around the following algorithm. The Active subroutine is from the work of [Han12] (slightly modified here), and is a variant of the A^2 (Agnostic Acive) algorithm of [BBL06]; the appropriate values of M and $\hat{T}_k(\cdot)$ will be discussed below.

Algorithm: DriftingActive
0. For $i = 1, 2, \ldots$
1. Active($M(i-1)$)

Subroutine: Active(t)
0. Let $\hat{h}_0$ be an arbitrary element of $\mathbb{C}$, and let $V_0 \leftarrow \mathbb{C}$
1. Predict $\hat{Y}_{t+1} = \hat{h}_0(X_{t+1})$ as the prediction for the value of Y_{t+1}
2. For $k = 0, 1, \ldots, \log_2(M/2)$
3. $Q_k \leftarrow \{\}$
4. For $s = 2^k + 1, \ldots, 2^{k+1}$
5. Predict $\hat{Y}_s = \hat{h}_k(X_s)$ as the prediction for the value of Y_s
6. If $X_s \in \mathrm{DIS}(V_k)$
7. Request the label Y_s and let $Q_k \leftarrow Q_k \cup \{(X_s, Y_s)\}$
8. Let $\hat{h}_{k+1} = \mathrm{argmin}_{h \in V_k} \sum_{(x,y)\in Q_k} \mathbb{1}[h(x) \neq y]$
9. Let $V_{k+1} \leftarrow \{h \in V_k : \sum_{(x,y)\in Q_k} \mathbb{1}[h(x) \neq y] - \mathbb{1}[\hat{h}_{k+1}(x) \neq y] \leq \hat{T}_k\}$

To express an abstract bound on the number of labels requested by this algorithm, we will make use of a quantity known as the *disagreement coefficient* [Han07], defined as follows. For any $r \geq 0$ and any classifier h, define $\mathrm{B}(h, r) = \{g \in \mathbb{C} : \mathcal{P}(x : g(x) \neq h(x)) \leq r\}$. Then for $r_0 \geq 0$ and any classifier h, define the disagreement coefficient of h with respect to $\mathbb{C}$ under $\mathcal{P}$:

$$\theta_h(r_0) = \sup_{r > r_0} \frac{\mathcal{P}(\mathrm{DIS}(\mathrm{B}(h, r)))}{r}.$$

Usually, the disagreement coefficient would be used with h equal the target concept; however, since the target concept is not fixed in our setting, we will make use of the worst-case value of the disagreement coefficient: $\theta_{\mathbb{C}}(r_0) = \sup_{h\in\mathbb{C}} \theta_h(r_0)$. This quantity has been bounded for a variety of spaces $\mathbb{C}$ and distributions $\mathcal{P}$ (see e.g., [Han07,EYW12,BL13]). We have the following result[7].

Theorem 4. *For an appropriate universal constant $c_1 \in [1,\infty)$, if $\mathbf{h}^* \in S_\Delta$ for some $\Delta \in (0,1]$, then taking $M = \left\lceil c_1\sqrt{\frac{d}{\Delta}}\right\rceil_2$, and $\hat{T}_k = \log_2(1/\sqrt{d\Delta}) + 2^{2k+2}e\Delta$, and defining $\epsilon_\Delta = \sqrt{d\Delta}\mathrm{Log}(1/(d\Delta))$, the above* DriftingActive *algorithm makes an expected number of mistakes among the first T instances that is*

$$O\left(\epsilon_\Delta \mathrm{Log}(d/\Delta)T\right) = \tilde{O}\left(\sqrt{d\Delta}\right)T$$

and requests an expected number of labels among the first T instances that is

$$O\left(\theta_{\mathbb{C}}(\epsilon_\Delta)\epsilon_\Delta \mathrm{Log}(d/\Delta)T\right) = \tilde{O}\left(\theta_{\mathbb{C}}(\sqrt{d\Delta})\sqrt{d\Delta}\right)T.$$

The proof of Theorem 4 relies on an analysis of the behavior of the Active subroutine, characterized in the following lemma.

Lemma 9. *Fix any $t \in \mathbb{N}$, and consider the values obtained in the execution of* Active(t)*. Under the conditions of Theorem 4, there is a universal constant $c_2 \in [1,\infty)$ such that, for any $k \in \{0,1,\ldots,\log_2(M/2)\}$, with probability at least $1 - 2\sqrt{d\Delta}$, if $h^*_{t+1} \in V_k$, then $h^*_{t+1} \in V_{k+1}$ and $\sup_{h\in V_{k+1}} \mathcal{P}(x : h(x) \neq h^*_{t+1}(x)) \leq c_2 2^{-k} d\mathrm{Log}(c_1/\sqrt{d\Delta})$.*

Proof. By a Chernoff bound, with probability at least $1 - \sqrt{d\Delta}$,

$$\sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[h^*_{t+1}(X_s) \neq Y_s] \leq \log_2(1/\sqrt{d\Delta}) + 2^{2k+2}e\Delta = \hat{T}_k.$$

Therefore, if $h^*_{t+1} \in V_k$, then since every $g \in V_k$ agrees with h^*_{t+1} on those points $X_s \notin \mathrm{DIS}(V_k)$, in the update in Step 9 defining V_{k+1}, we have

$$\begin{aligned}
&\sum_{(x,y)\in Q_k} \mathbb{1}[h^*_{t+1}(x) \neq y] - \mathbb{1}[\hat{h}_{k+1}(x) \neq y] \\
&= \sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[h^*_{t+1}(X_s) \neq Y_s] - \min_{g\in V_k} \sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[g(X_s) \neq Y_s] \\
&\leq \sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[h^*_{t+1}(X_s) \neq Y_s] \leq \hat{T}_k,
\end{aligned}$$

so that $h^*_{t+1} \in V_{k+1}$ as well.

[7] Here, we define $\lceil x\rceil_2 = 2^{\lceil \log_2(x)\rceil}$, for $x \geq 1$.

Furthermore, if $h^*_{t+1} \in V_k$, then by the definition of V_{k+1}, we know every $h \in V_{k+1}$ has

$$\sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[h(X_s) \neq Y_s] \leq \hat{T}_k + \sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[h^*_{t+1}(X_s) \neq Y_s],$$

so that a triangle inequality implies

$$\begin{aligned}\sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[h(X_s) \neq h^*_{t+1}(X_s)] &\leq \sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[h(X_s) \neq Y_s] + \mathbb{1}[h^*_{t+1}(X_s) \neq Y_s] \\ &\leq \hat{T}_k + 2\sum_{s=2^k+1}^{2^{k+1}} \mathbb{1}[h^*_{t+1}(X_s) \neq Y_s] \leq 3\hat{T}_k.\end{aligned}$$

Lemma 1 then implies that, on an additional event of probability at least $1 - \sqrt{d\Delta}$, every $h \in V_{k+1}$ has

$$\begin{aligned}&\mathcal{P}(x : h(x) \neq h^*_{t+1}(x)) \\ &\leq 2^{-k}3\hat{T}_k + c2^{-k}\sqrt{3\hat{T}_k(d\mathrm{Log}(2^k/d) + \mathrm{Log}(1/\sqrt{d\Delta}))} \\ &\quad + c2^{-k}(d\mathrm{Log}(2^k/d) + \mathrm{Log}(1/\sqrt{d\Delta})) \\ &\leq 2^{-k}3\log_2(1/\sqrt{d\Delta}) + 2^k 12e\Delta + c2^{-k}\sqrt{6\log_2(1/\sqrt{d\Delta})d\mathrm{Log}(c_1/\sqrt{d\Delta})} \\ &\quad + c2^{-k}\sqrt{2^{2k}24e\Delta d\mathrm{Log}(c_1/\sqrt{d\Delta})} + 2c2^{-k}d\mathrm{Log}(c_1/\sqrt{d\Delta}) \\ &\leq 2^{-k}3\log_2(1/\sqrt{d\Delta}) + 12ec_1\sqrt{d\Delta} + 3c2^{-k}\sqrt{d}\mathrm{Log}(c_1/\sqrt{d\Delta}) \\ &\quad + \sqrt{24e}c\sqrt{d\Delta\mathrm{Log}(c_1/\sqrt{d\Delta})} + 2c2^{-k}d\mathrm{Log}(c_1/\sqrt{d\Delta}),\end{aligned}$$

where c is as in Lemma 1. Since $\sqrt{d\Delta} \leq 2c_1 d/M \leq c_1 d 2^{-k}$, this is at most

$$\left(5 + 12ec_1^2 + 3c + \sqrt{24e}cc_1 + 2c\right)2^{-k}d\mathrm{Log}(c_1/\sqrt{d\Delta}).$$

Letting $c_2 = 5 + 12ec_1^2 + 3c + \sqrt{24e}cc_1 + 2c$, we have the result by a union bound.

We are now ready for the proof of Theorem 4.

Proof (Proof of Theorem 4). Fix any $i \in \mathbb{N}$, and consider running Active($M(i-1)$). Since $h^*_{M(i-1)+1} \in \mathbb{C}$, by Lemma 9, a union bound, and induction, with probability at least $1 - 2\sqrt{d\Delta}\log_2(M/2) \geq 1 - 2\sqrt{d\Delta}\log_2(c_1\sqrt{d/\Delta})$, every $k \in \{0, 1, \ldots, \log_2(M/2)\}$ has

$$\sup_{h \in V_k} \mathcal{P}(x : h(x) \neq h^*_{M(i-1)+1}(x)) \leq c_2 2^{1-k} d\mathrm{Log}(c_1/\sqrt{d\Delta}). \tag{13}$$

Thus, since $\hat{h}_k \in V_k$ for each k, the expected number of mistakes among the predictions $\hat{Y}_{M(i-1)+1}, \ldots, \hat{Y}_{Mi}$ is

$$
\begin{aligned}
&1 + \sum_{k=0}^{\log_2(M/2)} \sum_{s=2^k+1}^{2^{k+1}} \mathbb{P}(\hat{h}_k(X_{M(i-1)+s}) \neq Y_{M(i-1)+s}) \\
&\leq 1 + \sum_{k=0}^{\log_2(M/2)} \sum_{s=2^k+1}^{2^{k+1}} \mathbb{P}(h^*_{M(i-1)+1}(X_{M(i-1)+s}) \neq Y_{M(i-1)+s}) \\
&\quad + \sum_{k=0}^{\log_2(M/2)} \sum_{s=2^k+1}^{2^{k+1}} \mathbb{P}(\hat{h}_k(X_{M(i-1)+s}) \neq h^*_{M(i-1)+1}(X_{M(i-1)+s})) \\
&\leq 1 + \Delta M^2 + \sum_{k=0}^{\log_2(M/2)} 2^k \left(c_2 2^{1-k} d\mathrm{Log}(c_1/\sqrt{d\Delta}) + 2\sqrt{d\Delta}\log_2(M/2) \right) \\
&\leq 1 + 4c_1^2 d + 2c_2 d\mathrm{Log}(c_1/\sqrt{d\Delta}) \log_2(2c_1\sqrt{d/\Delta}) + 4c_1 d \log_2(c_1\sqrt{d/\Delta}) \\
&= O\left(d\mathrm{Log}(d/\Delta)\mathrm{Log}(1/(d\Delta))\right).
\end{aligned}
$$

Furthermore, (13) implies the algorithm only requests the label $Y_{M(i-1)+s}$ for $s \in \{2^k+1, \ldots, 2^{k+1}\}$ if $X_{M(i-1)+s} \in \mathrm{DIS}(\mathrm{B}(h^*_{M(i-1)+1}, c_2 2^{1-k} d\mathrm{Log}(c_1/\sqrt{d\Delta})))$, so that the expected number of labels requested among $Y_{M(i-1)+1}, \ldots, Y_{Mi}$ is at most

$$
\begin{aligned}
&1 + \sum_{k=0}^{\log_2(M/2)} 2^k \Big(\mathbb{E}[\mathcal{P}(\mathrm{DIS}(\mathrm{B}(h^*_{M(i-1)+1}, c_2 2^{1-k} d\mathrm{Log}(c_1/\sqrt{d\Delta}))))] \\
&\qquad\qquad + 2\sqrt{d\Delta}\log_2(c_1\sqrt{d/\Delta}) \Big) \\
&\leq 1 + \theta_{\mathbb{C}}\left(4c_2 d\mathrm{Log}(c_1/\sqrt{d\Delta})/M\right) 2c_2 d\mathrm{Log}(c_2/\sqrt{d\Delta}) \log_2(2c_1\sqrt{d/\Delta}) \\
&\qquad\qquad + 4c_1 d \log_2(c_1\sqrt{d/\Delta}) \\
&= O\left(\theta_{\mathbb{C}}\left(\sqrt{d\Delta}\mathrm{Log}(1/(d\Delta))\right) d\mathrm{Log}(d/\Delta)\mathrm{Log}(1/(d\Delta))\right).
\end{aligned}
$$

Thus, the expected number of mistakes among indices $1, \ldots, T$ is at most

$$
O\left(d\mathrm{Log}(d/\Delta)\mathrm{Log}(1/(d\Delta)) \left\lceil \frac{T}{M} \right\rceil\right) = O\left(\sqrt{d\Delta}\mathrm{Log}(d/\Delta)\mathrm{Log}(1/(d\Delta))T\right),
$$

and the expected number of labels requested among indices $1, \ldots, T$ is at most

$$
\begin{aligned}
&O\left(\theta_{\mathbb{C}}\left(\sqrt{d\Delta}\mathrm{Log}(1/(d\Delta))\right) d\mathrm{Log}(d/\Delta)\mathrm{Log}(1/(d\Delta)) \left\lceil \frac{T}{M} \right\rceil\right) \\
&\qquad = O\left(\theta_{\mathbb{C}}\left(\sqrt{d\Delta}\mathrm{Log}(1/(d\Delta))\right) \sqrt{d\Delta}\mathrm{Log}(d/\Delta)\mathrm{Log}(1/(d\Delta))T\right).
\end{aligned}
$$

References

[ABL13] P. Awasthi, M.-F. Balcan, and P. M. Long. The power of localization for efficiently learning linear separators with noise. *arXiv:1307.8371v2*, 2013.

[BBDK00] P. L. Bartlett, S. Ben-David, and S. R. Kulkarni. Learning changing concepts by exploiting the structure of change. *Machine Learning*, 41:153–174, 2000.

[BBL06] M. F. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. In *Proc. of the 23rd International Conference on Machine Learning*, 2006.

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

[BH96] P. L. Bartlett and D. P. Helmbold. Learning changing problems, 1996. (unpublished).

[BL96] R. D. Barve and P. M. Long. On the complexity of learning from drifting distributions. In *Proceedings of the ninth annual conference on Computational learning theory*, COLT '96, pages 122–130, 1996.

[BL97] R. D. Barve and P. M. Long. On the complexity of learning from drifting distributions. *Inf. Comput.*, 138(2):170–193, 1997.

[BL13] M.-F. Balcan and P. M. Long. Active and passive learning of linear separators under log-concave distributions. In *Proceedings of the* 26^{th} *Conference on Learning Theory*, 2013.

[BLM13] S. Boucheron, G. Lugosi, and P. Massart. *Concentration Inequalities*. Oxford University Press, 2013.

[CMEDV10] K. Crammer, Y. Mansour, E. Even-Dar, and J. Wortman Vaughan. Regret minimization with concept drift. In *COLT*, pages 168–180, 2010.

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

[EYW12] R. El-Yaniv and Y. Wiener. Active learning via perfect selective classification. *Journal of Machine Learning Research*, 13:255–279, 2012.

[Han07] S. Hanneke. A bound on the label complexity of agnostic active learning. In *Proceedings of the* 24^{th} *International Conference on Machine Learning*, 2007.

[Han12] S. Hanneke. Activized learning: Transforming passive to active with improved label complexity. *Journal of Machine Learning Research*, 13(5):1469–1587, 2012.

[HL91] D. P. Helmbold and P. M. Long. Tracking drifting concepts using random examples. In *COLT*, pages 13–23, 1991.

[HL94] D. P. Helmbold and P. M. Long. Tracking drifting concepts by minimizing disagreements. *Machine Learning*, 14(1):27–45, 1994.

[HLW94] D. Haussler, N. Littlestone, and M. Warmuth. Predicting $\{0,1\}$-functions on randomly drawn points. *Information and Computation*, 115:248–292, 1994.

[Li11] S. Li. Concise formulas for the area and volume of a hyperspherical cap. *Asian Journal of Mathematics and Statistics*, 4(1):66–70, 2011.

[Lon99] P. M. Long. The complexity of learning according to two models of a drifting environment. *Machine Learning*, 37(3):337–354, 1999.

[Vap82] V. Vapnik. *Estimation of Dependencies Based on Empirical Data*. Springer-Verlag, New York, 1982.

[Vap98] V. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, Inc., New York, 1998.

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

[Vid03] M. Vidyasagar. *Learning and Generalization with Applications to Neural Networks*. Springer-Verlag, 2^{nd} edition, 2003.