Disagreement-Based Active Learning in Online Settings

Boshuang Huang, Sudeep Salgia, and Qing Zhao, *Fellow, IEEE*

***Abstract*—We study online active learning for classifying streaming instances within the framework of statistical learning theory. At each time, the learner either queries the label of the current instance or predicts the label based on past seen examples. The objective is to minimize the number of queries while constraining the number of prediction errors over a horizon of length T. We develop a disagreement-based online learning algorithm for a general hypothesis space and under the Tsybakov noise and establish its label complexity under a constraint of bounded regret in terms of classification errors. We further establish a matching (up to a poly-logarithmic factor) lower bound, demonstrating the order optimality of the proposed algorithm. We address the tradeoff between label complexity and regret and show that the algorithm can be modified to operate at a different point on the tradeoff curve.**

***Index Terms*—Active learning, label complexity, online learning, regret, statistical learning theory.**

I. Introduction

We CONSIDER online classification of streaming instances within the framework of statistical learning theory. Let $\{X_t\}_{t\geq 1}$ be a sequence of instances drawn independently at random from an unknown underlying distribution $\mathbb{P}_X$ over an instance space $\mathcal{X}$. Each instance X_t has a hidden binary label $Y_t \in \{0,1\}$ that relates probabilistically to the instance according to an unknown conditional distribution $\mathbb{P}_{Y|X}$. The learner is characterized by its hypothesis space $\mathcal{H}$ consisting of all classifiers under consideration. At each time t, the learner decides whether to query the label of the current instance X_t. If yes, Y_t is revealed. Otherwise, the learner predicts the label of X_t using a hypothesis in $\mathcal{H}$ and incurs a classification error if the predicted label does not equal to the true label Y_t. The objective is to minimize the expected number of queries over a horizon of length T while constraining the total number of classification errors. The tension between label complexity and classification error rate needs to be carefully balanced through a sequential strategy governing the query and labeling decisions at each time.

The above problem arises in applications such as spam detection and event detection in real-time surveillance. The key characteristics of these applications are the high-volume streaming of instances and the complex and nuanced definition of labels. While the latter necessitates human intervention to provide annotations for selected instances, such human annotations, time consuming and expensive to obtain, should be sought after sparingly to ensure scalability.

A. Previous Work on Active Learning

The above problem falls under the general framework of active learning. In contrast to passive learning where labeled examples are given *a priori* or drawn at random, active learning asserts control over which labeled examples to learn from by actively querying the labels for carefully selected instances. The hope is that by learning from the most informative examples, the same level of classification accuracy can be achieved with much fewer labels than in passive learning.

1) Offline Active Learning: Active learning has been studied extensively under the Probably Approximately Correct (PAC) model, where the objective is to output an ϵ-optimal classifier with probability $1-\delta$ using as few labels as possible. The PAC model pertains to offline learning since the decision maker does not need to self label any instances during the learning process. An equivalent view is that classification errors that might have incurred during the learning process are inconsequential, and the tension between label complexity and classification errors is absent. If measured purely by label complexity, the decision maker has the luxury of skipping, at no cost, as many instances as needed to wait for the most informative instance to emerge. In the online setting considered in this work, however, self labeling is required in the event of no query, classification errors need to be strictly constrained, and no feedback to the predicted labels is available (thus learning has to rely solely on queried labels). As a result, online active learning faces an essential tradeoff between real-time classification errors and label complexity, which is absent in the offline settings.

A much celebrated offline active learning algorithm was given by Cohn, Atlas, and Ladner [1]. Named after its inventors, the CAL algorithm is applicable to a general hypothesis space $\mathcal{H}$. It, however, relies on the strong assumption of realizability, i.e., the instances are perfectly separable and there exists an error-free classifier in $\mathcal{H}$. In this case, hypotheses inconsistent with a single label can be safely eliminated from further consideration. Based on this key fact, CAL operates by maintaining two sets at each time: the *version space* consisting of all surviving hypotheses (i.e., those that are consistent with all past labels), and the *region*

Manuscript received June 10, 2021; revised December 14, 2021; accepted February 28, 2022. Date of publication March 15, 2022; date of current version May 2, 2022. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ali Tajer. This work was supported by National Science Foundation under Grant CCF-1815559. *(Corresponding author: Sudeep Salgia.)*

The authors are with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14850 USA (e-mail: bh467@cornell.edu; ss3827@cornell.edu; qz16@cornell.edu).

Digital Object Identifier 10.1109/TSP.2022.3159388

of disagreement (RoD), a subset of $\mathcal{X}$ for which there is disagreement among hypotheses in the current version space regarding their labels. CAL queries labels if and only if the instance falls inside the current RoD. Each queried label reduces the version space, which in turn may shrink the RoD, and the algorithm iterates indefinitely. Note that instances outside the RoD are given the same label by all the hypotheses in the current version space. It is thus easy to see that CAL represents a conservative approach: it only disregards instances whose labels can already be perfectly inferred from past labels. Quite surprisingly, by merely avoiding querying labels that carry no additional information, exponential reduction in label complexity can be achieved in a broad class of problems. (See, for example, an excellent survey by Dasgupta [2] and a monograph by Hanneke [3]).

The CAL algorithm was extended to the agnostic setting by Balcan, Beygelzimer, and Langford [4]. In the agnostic setting, instances are not separable, and even the best classifier h^* in $\mathcal{H}$ experiences a non-zero error rate. The main challenge in extending CAL to the agnostic case is the update of the version space: a single inconsistent label can no longer disqualify a hypothesis, and the algorithm needs to balance the desire of quickly shrinking the version space with the irreversible risk of eliminating h^*. Referred to as A^2 (Agnostic Active), the algorithm developed by Balcan, Beygelzimer, and Langford explicitly maintains an ϵ neighborhood of h^* in the version space by examining the empirical errors of each hypothesis. Analysis of the A^2 algorithm can be found in [5]–[8]. Variants of the A^2 algorithm include [9]–[13]. In particular, the DHM algorithm (named after the authors) in [12] simplifies the maintenance of the RoD through a reduction to supervised learning.

The above conservative approach originated from the CAL algorithm is referred to as the disagreement-based approach. The design methodology of this conservative approach focuses on avoiding querying labels that provide no or little additional information. More aggressive approaches that actively seeking out more informative labels to query have been considered in the literature. One such approach is the so-called margin-based. It is specialized for learning homogeneous (i.e. through the origin) linear separators of instances on the unit sphere in $\mathbb{R}^d$ and adopts a specific noise model that assumes linearity in terms of the inner product with the Bayes optimal classifier. In this case, the informativeness of a potential label can be measured by how close the instance is to the current decision boundary. Representative work on the margin-based approach includes [14]–[21]. Another such approach is considered in [22] that combines ideas from both disagreement-based and margin-based approaches.

Besides the stream-based model where instances arrive one at a time, active learning has also been considered under the synthesized instances and the pool-based sampling models [23] and synthesizes instances for various models for applications (see for example, [24]–[26]). These models are less relevant to the online setting considered in this work. In addition to online classification, active learning approaches have also been considered in the context of Bayesian Learning [27], [28] and inference of network topology [29].

2) Online Active Learning: Active learning in the online setting has received much less attention. The work of [30] and [31] extended the margin-based approach to the online setting, focusing, as in the offline case, on homogeneous linear separators for instances on the unit sphere in $\mathbb{R}^d$. A specific noise model was adopted, which assumes that the underlying conditional distribution of the labels is fully determined by the Bayes optimal classifier h^*. In this work, we consider a general instance space and arbitrary classifiers. Tackling the general setting, the proposed algorithm and the analysis are fundamentally different from these two existing studies. Furthermore, we show with simulation examples in Section VI that, even when restricted to the special case of homogeneous linear separators, the algorithm proposed in this work outperforms the margin-based algorithm developed in [30], [31].

The only works we are aware of that extend the disagreement-based approach to the online setting are [32] and [33]. [33] constructs a disagreement graph to relate different hypotheses in the hypothesis space and then uses importance weighted sampling to eliminate them based on observation — an approach that is difficult to extend for the model considered in this work. [32] extends the offline DHM algorithm to a stream-based setting and we discuss in detail the difference between [32] and this work in Section I-B.

B. Main Results

We consider a general instance space $\mathcal{X}$, a general hypothesis space $\mathcal{H}$ of Vapnik-Chervonenkis (VC) dimension d, and the Tsybakov noise model parameterized by $\alpha \in (0, 1]$ [34]. We develop an online active learning algorithm and establish its $O(dT^{\frac{2-2\alpha}{2-\alpha}} \log^2 T)$ label complexity and uniformly bounded regret in prediction errors with respect to the best classifier h^* in $\mathcal{H}$. More specifically, the total expected classification errors in excess to h^* over a horizon of length T is bounded below $1/2$ independent of T, demonstrating that the proposed algorithm offers practically the same level of classification accuracy as h^* with a sublinear label complexity in T. We further establish a matching (up to a poly-logarithmic factor) lower bound, demonstrating the order optimality of the proposed algorithm. We address the tradeoff between label complexity and regret and show that the algorithm can be modified to operate at a different point on the tradeoff curve. Below we contextualize this work with respect to the existing literature by highlighting the differences in three aspects: algorithm design, analysis techniques, and performance comparison.

1) Algorithm Design: Referred to as OLA (OnLine Active), the algorithm developed in this work is rooted in the design principle of the disagreement-based approach. The defining characteristic of the disagreement-based approach is to avoid querying instances that see insufficient disagreement among surviving hypotheses by maintaining, explicitly or inexplicitly, the RoD. Specific algorithm design differs in its temporal structure of when to update the RoD and, more crucially, in the threshold design on what constitutes sufficient disagreement. As detailed below, OLA differs from representative disagreement-based algorithms—the offline A^2 [4] and DHM [12] algorithms and the online ACAL algorithm [32]—in both aspects.

In terms of temporal structure, OLA operates in epochs and updates the RoD at the end of each epoch, where an epoch ends when a fixed number M of labels have been queried. This structure is different from A^2, DHM, and ACAL. In particular, the epochs in A^2 are determined by the time instants when the size of the current RoD shrinks by half due to newly obtained labels. Such an epoch structure, however, requires the knowledge of the marginal distribution $\mathbb{P}_X$ of the instances for evaluating the size of the RoD. The epoch structure of OLA obviates the need for this prior knowledge. DHM, on the other hand, does not operate in epochs and updates (inexplicitly) the RoD at each time. Similarly, ACAL also updates the RoD at each time.[1] Moreover, the updates involve calculating thresholds by solving multiple non-convex optimization problems with randomized nonlinear constraints that can only be checked numerically. In contrast, the epoch-based updates in OLA only involve thresholds that are given in closed-form in terms of empirical errors.

A more crucial improvement in OLA is the design of the threshold that determines the RoD. This is the key algorithm parameter that directly controls the tradeoff between label complexity and classification error rate. By focusing only on empirical errors incurred over significant (X, Y) examples determined by the current RoD, we obtain a tighter concentration inequality and a more aggressive threshold design (See Theorem 1), which leads to significant reduction in label complexity as compared with A^2, DHM, and ACAL, as well as margin-based algorithms (see details on the performance comparison below).

2) Analysis Techniques: Under the offline PAC setting, the label complexity of an algorithm is often analyzed in terms of the suboptimality gap ϵ and the outage probability δ. Under the online setting, however, the label complexity of an algorithm is measured in terms of the horizon length T, which counts both labeled and unlabeled instances. In the analysis of the label complexity of A^2 [4], [5], unlabeled instances are assumed to be cost free, and bounds on the number of unlabeled instances skipped by the algorithm are missing and likely intractable. Without a bound on the unlabeled data usage, the offline label complexity in terms of (ϵ, δ) cannot be translated to its online counterpart.

Yang [32] analyze the label complexity by bounding the excess risk in terms of local Rademacher complexity [35] within each epoch. This technique is restricted to the specific threshold design in ACAL, which is based on expensive non-convex optimization with constraints on randomized Rademacher process.

We adopt new techniques in analyzing the online label complexity of OLA. First we separate the analysis into two stages based on the size of the RoD. For the early stage where the RoD is large, we show that RoD is decreasing exponentially. Then, to upper bound the label complexity, the key idea is to construct a supermartingale $\{S(t)\}_{t\geq 0}$ given by the difference of an exponential function of the total queried labels up to t and a linear function of t. The optimal stopping theorem for supermartingales then leads to an upper bound on the exponential function of the label complexity. A bound on the label complexity thus follows from Jensen's inequality. The remaining label complexity where the RoD is small can be bounded by the product of the size and the remaining time horizon. The separation of the two stages is then optimized to tighten the bound.

The lower bound established in this work is new. We are not aware of any existing lower bound on label complexity in the online setting. Lower bounds for the offline PAC setting (see, e.g., [36], [37]) are inapplicable to the online setting and were established using different techniques.

3) Performance Comparison: We now comment on the performance comparison in terms of both asymptotic orders and finite-time performance.

As stated above, the performance analysis of A^2 is in terms of the PAC parameters (ϵ, δ). The analysis of its online performance is missing. Dasgupta *et al.* provided an upper bound on the unlabeled data usage in DHM [12]. The bound, however, appears to be loose and translates to a linear $O(T)$ label complexity in the online setting. Yang [32] provided an upper bound $O(dT^{\frac{2-2\alpha}{2-\alpha}} \log^3 T)$ on the label complexity of ACAL, which is higher than the $O(dT^{\frac{2-2\alpha}{2-\alpha}} \log^2 T)$ order offered by OLA.

The margin-based algorithm for learning homogeneous linear separators under a uniform distribution of X on the unit sphere is analyzed in [31] under the Tsybakov noise condition. It leads to a regret order of $O(dT^{\frac{2-2\alpha}{3-2\alpha}} \log T)$ and a label complexity of $O(dT^{\frac{2-2\alpha}{2-\alpha}} \log T)$ under the Tsybakov low noise condition. These orders cannot be directly compared with that of OLA due to the restrictions to homogeneous linear separators and the specific form of $\mathbb{P}_{Y|X}$. This margin-based algorithm also operates at a different point on the tradeoff curve between regret and label complexity, offering a slightly lower order in label complexity but a higher order in regret. However, even when restricted to the special case targeted by this margin-based algorithm, the dominating polynomial term is the same, and the finite-time comparison given by simulation examples in Section VI actually show superior performance of OLA in both label complexity and regret.

The finite-time comparison in Section VI also demonstrate significant performance gain offered by OLA over the three representative disagreement-based algorithms: A^2, DHM, and ACAL. In particular, the improvement over the online algorithm ACAL is drastic.

II. Problem Formulation

A. Instances and Hypotheses

Let $\{X_t\}_{t\geq 1}$ be a streaming sequence of instances, each drawn from an instance/sample space $\mathcal{X}$ and characterized by its feature vector. Each subset of $\mathcal{X}$ is a concept. There is a target concept $\mathcal{C} \subset \mathcal{X}$ that the learner aims to learn (e.g., learning the concept "table" from household objects). Relating to the target concept $\mathcal{C}$, each instance X_t has a hidden label Y_t, indicating whether $X_t \in \mathcal{C}$ (i.e., a positive example wherein $Y_t = 1$) or $X_t \notin \mathcal{C}$ (a negative

[1] ACAL has a predetermined epoch structure with geometrically growing epoch length. This epoch structure, however, is not for controlling when to update the RoD, but rather for setting a diminishing sequence of outage probability of eliminating h^*. The algorithm otherwise restarts by forgetting all past experiences at the beginning of each epoch.

example with $Y_t = 0$). The label Y_t relates probabilistically to X_t according to an unknown conditional distribution $\mathbb{P}_{Y|X}$.

The learner is characterized by its hypothesis space $\mathcal{H}$ consisting of all classifiers under consideration. Each hypothesis $h \in \mathcal{H}$ is a measurable function mapping from $\mathcal{X}$ to $\{0, 1\}$. The complexity of the hypothesis space $\mathcal{H}$ is measured by its VC dimension d.

B. Error Rate, Disagreement, and Bayes Optimizer

Recall that $\mathbb{P}_{Y|X}$ denotes the conditional distribution of the true label Y for a given X. Let $\mathbb{P}_X$ denote the unknown marginal distribution of instances X and $\mathbb{P} = \mathbb{P}_X \times \mathbb{P}_{Y|X}$ the joint distribution of an example (X, Y). The error rate of a hypothesis h is given by

$$\epsilon_{\mathbb{P}}(h) = \mathbb{P}[h(X) \neq Y], \tag{1}$$

which is the probability that h misclassifies a random instance. Define the *pseudo-distance* and the *disagreement* between two hypotheses as, respectively,

$$d(h, h') = |\epsilon_{\mathbb{P}}(h) - \epsilon_{\mathbb{P}}(h')|, \rho(h, h') = \mathbb{P}_X[h(X) \neq h'(X)], \tag{2}$$

where the distance is the difference in error rates and the disagreement is the probability mass of the instances over which the two hypotheses disagree. Lastly, $\mathcal{D}(h_1, h_2) = \{x \in \mathcal{X} : h_1(x) \neq h_2(x)\}$ denotes the disagreement region between two hypotheses h_1 and h_2.

Let h^* be the Bayes optimal classifier that minimizes the error rate, i.e., for all $x \in \mathcal{X}$, $h^*(x)$ is the label that minimizes the probability of classification error:

$$h^*(x) = \arg\min_{y=0,1} \mathbb{E}_{\mathbb{P}_{Y|X=x}} \mathbb{1}[Y \neq y], \tag{3}$$

where $\mathbb{1}[\cdot]$ is the indicator function. Let

$$\eta(x) = \mathbb{P}_{Y|X=x}(Y = 1|X = x). \tag{4}$$

It is easy to see that

$$h^*(x) = \begin{cases} 1 & \text{if } \eta(x) \geq \frac{1}{2} \\ 0 & \text{if } \eta(x) < \frac{1}{2} \end{cases}. \tag{5}$$

We assume that $h^* \in \mathcal{H}$.

C. Noise Condition

The function $\eta(x)$ given in (4) is a measure of the feature noise level at x. The noise-free case is when labels are deterministic: $\mathbb{P}_{Y|X=x}$, hence $\eta(x)$, assumes only values of 0 and 1. In this case, the optimal classifier h^* is error-free. This is referred to as the realizable case with perfectly separable data.

In a general agnostic case with arbitrary $\mathbb{P}_{Y|X}$, consistent classifiers may not exist, and even h^* suffers a positive error rate. A particular case, referred to as the Massart bounded noise condition [38], is when $\eta(x)$ is discontinuous at the boundary between positive examples $\mathcal{X}_1^* \triangleq \{x \in \mathcal{X} : h^*(x) = 1\}$ and negative examples $\mathcal{X}_0^* \triangleq \{x \in \mathcal{X} : h^*(x) = 0\}$. Specifically, there exists $\gamma > 0$ such that $|\eta(x) - \frac{1}{2}| \geq \gamma$ for all $x \in \mathcal{X}$.

A more general noise model is the Tsybakov noise condition [34], for which the Massart bounded noise condition is a special case. It allows $\eta(x)$ to pass $\frac{1}{2}$ with a continuous change across the decision boundary and parameterizes the slope around the boundary. Specifically, the Tsybakov noise condition states that there exist $\alpha \in (0, 1]$, $c_0 \geq 0$, such that for all h, we have

$$\rho(h, h^*) \leq c_0 d^{\alpha}(h, h^*). \tag{6}$$

At $\alpha = 1$, the Tsybakov noise reduces to the more benign Massart noise. In terms of the slope around the decision boundary, the above condition can be restated as

$$\mathbb{P}_X\left(\left\{x : \left|\eta(x) - \frac{1}{2}\right| \leq \gamma\right\}\right) \leq c_0' \gamma^{\frac{\alpha}{1-\alpha}} \tag{7}$$

for some constant $c_0' \geq 0$.

D. Learning Policies and Performance Measure

An online active learning strategy π consists of a sequence of query rules $\{\upsilon_t\}_{t\geq 1}$ and a sequence of prediction rules $\{\lambda_t\}_{t\geq 1}$. Here υ_t and λ_t map from causally available information consisting of past actions, instances, and queried labels to, respectively, the query decision of 0 (no query) or 1 (query) and a predicted label at time t. With a slight abuse of notation, we also let υ_t and λ_t denote the resulting query decision and the predicted label at time t under these respective rules.

The performance of policy $\pi = (\{\upsilon_t\}, \{\lambda_t\})$ over a horizon of length T is measured by the expected number of queries and the expected number of classification errors in excess to that of the Bayes optimal classifier h^*. These two performance measures, referred to as label complexity $\mathbb{E}[Q(T)]$ and regret $\mathbb{E}[R(T)]$, are given as follows.

$$\mathbb{E}[Q(T)] = \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}[\upsilon_t = 1]\right] \tag{8}$$

$$\mathbb{E}[R(T)] = \mathbb{E}\left[\sum_{t\leq T:\upsilon_t=0} \mathbb{1}[\lambda_t \neq Y_t] - \mathbb{1}[h^*(X_t) \neq Y_t]\right], \tag{9}$$

where the expectation is with respect to the stochastic process induced by π. Note that regret measures the expected difference in the *cumulative* classification errors over the entire horizon between a learner employing π and an oracle that uses h^* all through the horizon.

The objective is a learning algorithm that minimizes the label complexity $\mathbb{E}[Q(T)]$ with a constraint on the regret $\mathbb{E}[R(T)]$. The constraint, for example, can be either bounded by a constant independent of T or in a logarithmic order of T.

III. The Online Active Learning Algorithm

A. The Basic Structure

The algorithm operates under an epoch structure. When a fixed number M of labels have been queried in the current epoch, this epoch ends and the next one starts. Note that the epoch length, lower bounded by M, is random due to the real-time active query decisions. The algorithm maintains two sets in each epoch k: the version space $\mathcal{H}_k$ and the RoD $\mathcal{D}(\mathcal{H}_k)$ defined as the region of instances for which there is disagreement among hypotheses in the current version space $\mathcal{H}_k$. More specifically,

$$\mathcal{D}(\mathcal{H}_k) = \{x \in \mathcal{X} : \exists h_1, h_2 \in \mathcal{H}_k, h_1(x) \neq h_2(x)\}. \quad (10)$$

The initial version space is set to the entire hypothesis space $\mathcal{H}$, and the initial RoD is the instance space $\mathcal{X}$. At the end of each epoch, these two sets are updated using the M labels obtained in this epoch, and the algorithm iterates into the next epoch.

At each time instant t of epoch k, the query and prediction decisions are as follows. If $x_t \in \mathcal{D}(\mathcal{H}_k)$, its label is queried. Otherwise, the learner predicts the label of x_t using an arbitrary hypothesis in $\mathcal{H}_k$.

At the end of the epoch, $\mathcal{H}_k$ is updated as follows. Let $\mathcal{Z}_k$ denote the set of the M queried examples in this epoch. For a hypothesis h in $\mathcal{H}_k$, define its empirical error over $\mathcal{Z}_k$ as

$$\epsilon_{\mathcal{Z}_k}(h) = \frac{1}{M} \sum_{(x,y)\in\mathcal{Z}_k} \mathbb{1}[h(x) \neq y]. \quad (11)$$

Let $h_k^* = \arg\min_{h\in\mathcal{H}_k} \epsilon_{\mathcal{Z}_k}(h)$ be the best hypothesis in $\mathcal{H}_k$ in terms of empirical error over $\mathcal{Z}_k$. The version space is then updated by eliminating each hypothesis h whose empirical error over $\mathcal{Z}_k$ exceeds that of h_k^* by a threshold $\Delta_{\mathcal{Z}_k}(h, h_k^*)$ that is specific to h, h_k^*, and $\mathcal{Z}_k$. Specifically,

$$\mathcal{H}_{k+1} = \{h \in \mathcal{H}_k : \epsilon_{\mathcal{Z}_k}(h) - \epsilon_{\mathcal{Z}_k}(h_k^*) < \Delta_{\mathcal{Z}_k}(h, h_k^*)\}. \quad (12)$$

The new RoD $\mathcal{D}(\mathcal{H}_{k+1})$ is then determined by $\mathcal{H}_{k+1}$ as in (10).

B. Threshold Design

We now discuss the key issue of designing the threshold $\Delta_{\mathcal{Z}_k}(h, h_k^*)$ for eliminating suboptimal hypotheses. This elimination threshold controls the tradeoff between two conflicting objectives: quickly shrinking the RoD (thus reducing label complexity) and managing the irreversible risk of eliminating good classifiers (thus increasing future classification errors).

In OLA, we obtain a more aggressive threshold design focusing on empirical errors incurred over significant (X, Y) examples determined by the current RoD.

Specifically, for a pair of hypotheses h_1, h_2, define

$$\epsilon_{\mathbb{P}}(h_1, h_2) = \mathbb{P}(h_1(X) \neq Y \wedge h_2(X) = Y), \quad (13)$$

where $\wedge$ is the logical AND operation. Thus, $\epsilon_{\mathbb{P}}(h_1, h_2)$ is the probability that h_1 misclassifies a random instance but h_2 successfully classified. For a finite set $\mathcal{Z}$ of (x, y) samples, the empirical excess error of h_1 over h_2 on $\mathcal{Z}$ is defined as

$$\epsilon_{\mathcal{Z}}(h_1, h_2) \triangleq \frac{1}{|\mathcal{Z}|} \sum_{(x,y)\in\mathcal{Z}} \mathbb{1}[h_1(x) \neq y \wedge h_2(x) = y]. \quad (14)$$

The elimination threshold $\Delta_{\mathcal{Z}_k}(h, h_k^*)$ is set to:

$$\Delta_{\mathcal{Z}_k}(h, h_k^*) = \beta^2_{\mathcal{H}_k, M} + \beta_{\mathcal{H}_k, M}\left(\sqrt{\epsilon_{\mathcal{Z}_k}(h, h_k^*)} + \sqrt{\epsilon_{\mathcal{Z}_k}(h_k^*, h)}\right), \quad (15)$$

where $\beta_{\mathcal{H}', n} = \sqrt{(4/n)\ln(16T^2\mathcal{S}(\mathcal{H}', 2n)^2)}$ for an arbitrary hypothesis space $\mathcal{H}'$ and positive integer n. Here $\mathcal{S}(\mathcal{H}', n)$ is the n-th shattering coefficient of $\mathcal{H}'$. By Sauer's lemma [39], $\mathcal{S}(\mathcal{H}', n) = O(n^{d'})$ with d' being the VC dimension of $\mathcal{H}'$.

Algorithm 1: The OLA Algorithm.

```
Input: Time horizon $T$, VC dimension $d$, parameter
  $m \in \mathbb{N}^+$.
Initialization: Set $\mathcal{Z}_1 = \emptyset$, Version space $\mathcal{H}_1 = \mathcal{H}$, RoD
  $\mathcal{D}_1 = \mathcal{X}$. Current epoch $k = 1$. $M = \lceil mdT^{\frac{2-2\alpha}{2-\alpha}} \log T \rceil$.
for $t = 1$ to $T$ do
  if $x_t \notin \mathcal{D}_k$ then
    Choose any $h \in \mathcal{H}_k$ and label $x_t$ with $h(x_t)$;
  end if
  if $x_t \in \mathcal{D}_k$ then
    Query label $y_t$ and let $\mathcal{Z}_k = \mathcal{Z}_k \cup \{(x_t, y_t)\}$;
    if $|\mathcal{Z}_k| = M$ then
      Update $\mathcal{H}_{k+1}$ and $\mathcal{D}_{k+1}$ according to (10) and (12)
      with the elimination threshold $\Delta_{\mathcal{Z}_k}$ given in (15);
      Let $k = k + 1$;
    end if
  end if
end for
```

The choice of this specific threshold function will become clear in Section IV-A when the relationship between the empirical error difference of two hypotheses and the ensemble error rate difference under $\mathbb{P}$ is analyzed.

A detailed description of the algorithm is given in Algorithm 1. The algorithm parameter M is set to $\lceil mdT^{\frac{2-2\alpha}{2-\alpha}} \log T \rceil$, where m is a positive integer whose value will be discussed in Section IV-B. We point out that while the horizon length T is used as an input parameter to the algorithm, the standard doubling trick can be applied when T is unknown.

IV. Analysis of Regret and Label Complexity

We first develop the following concentration inequality in Theorem 1 to establish the relationship between the empirical error and ensemble error rate of any pair of hypotheses. The proof employs the normalized uniform convergence VC bound [40]. Details can be found in [41, Appendix A].

Theorem 1: Let $\mathcal{Z}$ be a set of n i.i.d. (X, Y)-samples under distribution $\mathbb{P}$. For all $h_1, h_2 \in \mathcal{H}$, we have, with probability at least $1 - \delta$,

$$\epsilon_{\mathbb{P}}(h_1) - \epsilon_{\mathbb{P}}(h_2) \leq \epsilon_{\mathcal{Z}}(h_1) - \epsilon_{\mathcal{Z}}(h_2) + \gamma_n^2 + \gamma_n\left(\sqrt{\epsilon_{\mathcal{Z}}(h_1, h_2)} + \sqrt{\epsilon_{\mathcal{Z}}(h_2, h_1)}\right), \quad (16)$$

where $\gamma_n = \sqrt{(4/n)\ln(8\mathcal{S}(\mathcal{H}, 2n)^2/\delta)}$.

Since all samples in $\mathcal{D}_k$ are queried at epoch k in the proposed OLA algorithm, we can see that $\mathcal{Z}_k$ is an i.i.d. sample of size M from distribution $\mathbb{P}|\mathcal{D}_k$, which is defined as

$$\mathbb{P}|\mathcal{D}_k(x) = \begin{cases} \mathbb{P}(x)/\phi(\mathcal{D}_k) & \text{if } x \in \mathcal{D}_k \\ 0 & \text{otherwise} \end{cases}, \quad (17)$$

where $\phi(\mathcal{D}) = \mathbb{P}(X \in \mathcal{D})$ for $\mathcal{D} \subseteq \mathcal{X}$.

Therefore, we can apply Theorem 1 to each epoch k with $\mathcal{Z}_k$ and $\mathbb{P}|\mathcal{D}_k$, which gives us the following corollary.

Corollary 1: Let $\beta_n = \sqrt{(4/n)\ln(16T^2\mathcal{S}(\mathcal{H}, 2n)^2)}$. With probability at least $1 - \frac{1}{2T}$, for all $k \geq 1$ and for all $h \in \mathcal{H}_k$, we have

$$\epsilon_{\mathbb{P}|\mathcal{D}_k}(h_k^*) - \epsilon_{\mathbb{P}|\mathcal{D}_k}(h) \le \epsilon_{\mathcal{Z}_k}(h_k^*) - \epsilon_{\mathcal{Z}_k}(h) + \beta_M^2 + \beta_M \left(\sqrt{\epsilon_{\mathcal{Z}_k}(h_k^*, h)} + \sqrt{\epsilon_{\mathcal{Z}_k}(h, h_k^*)} \right). \quad (18)$$

A. Regret

Next, using Theorem 1 we show that the expected regret of the proposed OLA algorithm is bounded by $1/2$.

Theorem 2: The expected regret $\mathbb{E}[R(T)]$ of the OLA algorithm is bounded as follows:

$$\mathbb{E}[R(T)] \le \frac{1}{2}.$$

Proof: Here we provide the sketch of the proof. The detailed proof can be found in [41, Appendix B]. First we show that if the inequalities in Corollary 1 hold simultaneously for all $k \ge 1$, we have $h^* \in \mathcal{H}_k$ for all $k \ge 1$, which implies $R(T) = 0$. Therefore by Corollary 1, we have $\mathbb{P}(R(T) > 0) \le \frac{1}{2T}$. Note that $R(T) \le T$, we have $\mathbb{E}[R(T)] \le \frac{1}{2T} \cdot T = \frac{1}{2}$ as desired. □

B. Label Complexity

For the purpose of label complexity analysis, we define the following online disagreement coefficient, which is slightly different from the disagreement coefficient defined for offline active learning in [5]. Recall the psuedo-metric ρ defined in (2). The online disagreement coefficient $\theta = \theta(\mathbb{P}, \mathcal{H})$ is defined as

$$\theta = \sup \left\{ \frac{\phi[\mathcal{D}(B(h^*, r))]}{r} : r > 0 \right\}, \quad (19)$$

where $B(h, r) = \{h \in \mathcal{H} : \rho(h, h') < r\}$ is a "hypothesis ball" centered at h with radius r.

The quantity θ bounds the rate at which the disagreement mass of the ball $B(h^*, r)$ grows with the radius r. It is bounded by $\sqrt{d}$ when $\mathcal{H}$ is d-dimensional homogeneous separators [5].

Next we upper bound the label complexity for the proposed online active learning algorithm.

Theorem 3: Let $\mathbb{E}[Q(T)]$ be the expected label complexity of OLA. If $m > 324(\theta c_0)^{\frac{2}{\alpha}}$, then there exists $C_1 > 0$ such that

$$\mathbb{E}[Q(T)] \le C_1 m d T^{\frac{2-2\alpha}{2-\alpha}} (\log T + 1)^2, \quad (20)$$

where $\theta = \theta(\mathbb{P}_X, \mathcal{H})$ is the disagreement coefficient.

Note that m is a constant determined by the algorithm, the label complexity $\mathbb{E}[Q(T)]$ has an order of $O(dT^{\frac{2-2\alpha}{2-\alpha}} \log^2 T)$. For the Massart noise condition at $\alpha = 1$, the label complexity is $O(d \log^2 T)$.

Proof: We have discussed in Section I-B the key ideas and techniques used in the proof. The detailed proof can be found in [41, Appendix C]. □

C. Order Optimality

We now establish the order optimality of the label complexity (upto poly-logarthmic factors) of OLA under a bounded regret constraint. This is obtained by establishing a lower bound on the label complexity feasible under any policy with a bounded regret.

Theorem 4: Consider the Tsybakov noise satisfying the following condition with a parameter $\alpha \in (0, 1)$: there exist constants c_1 and c_2 independent of $x \in \mathcal{X}$ such that $\frac{c_1}{2} r_0(x)^{\frac{1}{\alpha}-1} \le |\eta(x) - \frac{1}{2}| \le \frac{c_2}{2} r_0(x)^{\frac{1}{\alpha}-1}$ holds for all $x \in \mathcal{X}$ where $r_0(x) = \inf_{\{h: x \in \mathcal{D}(h, h^*)\}} \rho(h, h^*)$. The label complexity of all policies with bounded regret is of order $\Omega(T^{\frac{2-2\alpha}{2-\alpha}})$.

Note that a lower bound on the noise (i.e., an upper bound specified through the constant c_2 on the slope of $\eta(x)$ passing $1/2$) is further imposed in order to establish a tight lower bound on label complexity for a specific noise level. We point out that while we focused on the constraint of a bounded regret, the analysis can be easily modified to obtain lower bounds under regret constraints of different orders. Specifically, we can show a lower bound of $\Omega(\min\{T^{2(1-\alpha)(1-\epsilon)}, T^{\frac{2-2\alpha}{2-\alpha}}\})$ under a regret constraint of order $O(T^{\epsilon})$ for some $\epsilon > 0$. It is also straightforward to modify the lower bound analysis to accommodate different problem models (e.g., those studied in [31], [32]).

Proof: The key in establishing the lower bound is to identify a limiting subproblem inherent to the online classification problem that determines the label complexity. We show that an inherent binary hypothesis testing problem presents such a limit. For this specific subproblem, we show that the probability of the event where label complexity is capped at $\Omega(T^{\frac{2-2\alpha}{2-\alpha}})$ is small if the regret on the subproblem has to be bounded. The detailed proof is given in [41, Appendix D]. □

For the case of Massart Noise, we can establish a lower bound of $\Omega(\log T)$ under the constraint of a sublinear regret budget. The basic proof technique follows similar ideas as that for the Tsybakov noise but with a simplified analysis (See [41]). We summarize the lower bound for the case of Massart Noise in the following theorem.

Theorem 5: Consider the Massart Noise model where $\eta(x)$ is bounded away from $1/2$ by a parameter $\gamma > 0$, i.e., for all $x \in \mathcal{X}$, $|\eta(x) - 1/2| \ge \gamma$. The label complexity of all policies that achieve a sublinear regret under the above Massart Noise model is of the order $\Omega(\log T)$.

For constraints of sublinear but unbounded regret, the above lower bound is tight since it matches with the upper bound on the label complexity of Random Walk OLA or RW-OLA for short (see Section V). Under the constraint of bounded regret, however, we conjecture a $\Omega(\log^2 T)$ lower bound on label complexity for a general hypothesis space with an infinite number of hypotheses. [2]

[2] The intuition behind this conjecture is as follows. Let $N(\epsilon)$ denote the ϵ-covering number of $\mathcal{H}$ and $\mathcal{C}$ be an associated ϵ-cover that has $N(\epsilon)$ hypotheses. Specifically, an ϵ-cover $\mathcal{C}$ of $\mathcal{H}$ is a subset of hypothesis $\{h_1, \ldots, h_N\}$ such that for any $h \in \mathcal{H}$ there exists an $i \in \{1, 2, \ldots, N\}$ such that $\rho(h, h_i) \le \epsilon$ and the ϵ-covering number of $\mathcal{H}$ is the size of the smallest ϵ-cover of $\mathcal{H}$. Following the same line of arguments in the proof of Theorem 5, we can show that for a hypothesis $h \in \mathcal{C}$, the policy needs to query $\Omega(\log T)$ instances in $\mathcal{D}(h, h^*)$ to ensure a bounded regret and this needs to hold simultaneously for all $h \in \mathcal{C}$ by the end of the time horizon. Using the uniformity of the cover $\mathcal{C}$, we can show that the expected number of queries to hit $\Omega(\log T)$ queries in $\mathcal{D}(h, h^*)$ for all $h \in \mathcal{C}$ is $\Omega(\log T \log N(\epsilon))$. Choosing $\epsilon \sim T^{-1/2}$ results in a bound of $\Omega(\log^2 T)$ on label complexity.

V. Extensions and Discussions

A. *Tradeoff Between Label Complexity and Regret*

In this section, we show that OLA can be modified to operate on a different point on the tradeoff curve of regret vs. label complexity.

In OLA, the threshold for elimination is constructed conservatively to achieve a bounded regret. More specifically, the outage probability of eliminating h^* from the version space (i.e., the parameter δ in Theorem 1) in each epoch is capped at a small value $1/T^2$ that diminishes with T. We now consider a variant of OLA in which the elimination probability δ is set to a constant in order to quickly shrink RoD for a lower label complexity. To mitigate the high probability of h^* being eliminated, which may result in a linear regret order, we build in a verification stage at the beginning of each epoch for the algorithm to self recognize and recover from the event of h^* being eliminated. The key idea is to devise a biased random walk on the version spaces that allows the algorithm to trace back to a previous version space in the event of h^* being eliminated. The bias of the random walk, however, ensures that with high probability the trajectories of the random walk concentrate on subsets of $\mathcal{H}$ containing h^* with disagreement regions diminishing at a desired rate. As a result, after taking $O(\log T)$ steps, the random walk arrives at a version space with a disagreement region in the order of $O(1/T)$ in terms of its probability mass. In summary, compared with a deterministic transition between consecutive version spaces as in OLA, the random walk approach allows us to move more swiftly on average across version spaces with the option of tracing back and hence correcting previous erroneous moves. It is crucial to the objective of operating at a different point on the label complexity-regret curve. We refer to this variant of OLA as RW-OLA.

Before delving into the details of the verification stage, we define the parent and child relationship between version spaces. For each epoch k, if $\mathcal{H}_k$ is obtained by eliminating some of the hypotheses in the version space $\mathcal{H}_{r(k)}$ of a previous epoch $r(k)$, we say that $\mathcal{H}_{r(k)}$ is the parent of $\mathcal{H}_k$ and $\mathcal{H}_k$ is a child of $\mathcal{H}_{r(k)}$. Note that $\mathcal{H}_k$ is a subset of $\mathcal{H}_{r(k)}$.

1) Verification Stage: In the verification stage of epoch k, the query and prediction decision are based on its parent version space $\mathcal{H}_{r(k)}$ and its corresponding RoD. When a fixed number M of labels have been queried, we start the verification process as follows.

Let $\mathcal{Z}'_k$ denote the set of the M queried examples in the verification stage. We examine two values in terms of the empirical error over $\mathcal{Z}'_k$: (1) $\min_{h\in\mathcal{H}_k}\epsilon_{\mathcal{Z}'_k}(h)$: the minimum empirical error inside $\mathcal{H}_k$; (2) $\min_{h\notin\mathcal{H}_k}\epsilon_{\mathcal{Z}'_k}(h)$: the minimum empirical error outside $\mathcal{H}_k$. Intuitively, if $h^*\in\mathcal{H}_k$, the difference

$$\min_{h\notin\mathcal{H}_k}\epsilon_{\mathcal{Z}'_k}(h)-\min_{h\in\mathcal{H}_k}\epsilon_{\mathcal{Z}'_k}(h) \tag{21}$$

will be large, and vice versa. We hence determine the outcome of the verification stage based on whether this gap between the empirical errors outside and inside the current version space $\mathcal{H}_k$ exceeds a properly designed threshold. If the verification passes, indicating that $h^*\in\mathcal{H}_k$ with a sufficiently high probability, the epoch proceeds in the same way as in OLA, and the current version space $\mathcal{H}_k$ is further pruned to form a new version space $\mathcal{H}_{k+1}$. If, on the other hand, the verification fails, the current epoch ends, and the algorithm traces back to the parent of $\mathcal{H}_k$ by setting $\mathcal{H}_{k+1}=\mathcal{H}_{r(k)}$. The evolution of the version spaces across epochs follows a biased random walk as detailed below.

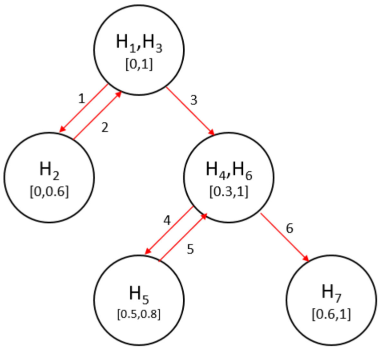

Fig. 1. A typical random walk on a version space tree.

2) Random Walk on a Version-Space Tree: Based on the outcome of the verification stage, the version space $\mathcal{H}_{k+1}$ of epoch $k+1$ is either a child or a parent of $\mathcal{H}_k$. In particular, following the evolution of the version spaces, we can construct a growing tree to record the parent-children relationship between version spaces. In this tree structure, each node represents a version space, and the version space sequence $\{\mathcal{H}_k\}_{k\geq 1}$ forms a random walk on the tree. Illustrated in Fig. 1 is a sample path of the random walk on a tree for a hypothesis space consisting of threshold classifiers $\mathcal{H}=\{h_z|0\leq z\leq 1\}$ on $\mathcal{X}=[0,1]$ where $h_z=[z,1]$. We can see that on this tree, the verification failed in epochs 2 and 5 and but passed in epochs 1, 3, 4, and 6.

3) Threshold design: In addition to the elimination threshold for pruning the version space as in OLA, RW-OLA also requires a verification threshold. As explained below, these two thresholds are coupled and need to be designed jointly to ensure the desired performance of the algorithm.

The verification stage performs a binary detection problem: whether h^* is inside the current version space $\mathcal{H}_k$. On one hand, to ensure that the random walk on the version spaces is biased toward the direction of correct pruning, the verification threshold needs to be chosen to ensure a sufficiently accurate detection outcome. On the other hand, the hardness of this detection problem is determined by how close the two cases of $h^*\in\mathcal{H}_k$ and $h^*\notin\mathcal{H}_k$ are. More specifically, the hardness of this binary detection problem is determined by the error rate difference between the best hypothesis inside $\mathcal{H}_k$ and the best hypothesis outside $\mathcal{H}_k$:

$$\min_{h\notin\mathcal{H}_k}\epsilon_{\mathbb{P}_{r(k)}}(h)-\min_{h\in\mathcal{H}_k}\epsilon_{\mathbb{P}_{r(k)}}(h), \tag{22}$$

which, in turn, is determined by the elimination threshold in epoch $r(k)$ when $\mathcal{H}_k$ is obtained. More specifically, while a smaller elimination threshold leads to more aggressive pruning of the version space, it results in a smaller performance gap between hypotheses outside $\mathcal{H}_k$ and those inside $\mathcal{H}_k$ since near-optimal hypotheses may be eliminated from the version space. Consequently, the verification stage faces a harder detection problem. In summary, the two thresholds need to be designed jointly to balance the label complexity associated with verification and with normal learning.

Let p denote the desired bias of the random walk. This implies that (i) when $h^* \in \mathcal{H}_k$, the verification passes with a probability no smaller than p; (ii) when $h^* \notin \mathcal{H}_k$, the verification fails with a probability no smaller than p. Let

$$\Delta(M, \delta) = 2\sqrt{2\frac{\log \mathcal{S}(\mathcal{H}, 2M) + \log \frac{2}{\delta}}{M}}. \tag{23}$$

We set the elimination threshold to $6\Delta(M, 1 - \sqrt{p})$ so that the error rate difference in (22), which determines the hardness of the verification problem, is at least $4\Delta(M, 1 - \sqrt{p})$ with high probability. The verification threshold is set to $2\Delta(M, 1 - \sqrt{p})$ to guarantee the desired bias of p. A detailed derivation of the thresholds is given in [41, Appendix E].

A detailed description of the algorithm is given in Algorithm 2. The algorithm parameter M is set to $\lceil md \rceil$, where m is a positive integer whose value will be discussed in the analysis below.

Algorithm 2: The Random Walk OLA (RW-OLA) Algorithm.

```
Input: VC dimension $d$, parameter $m \in \mathbb{N}^+$.
Initialization: Set Version space $\mathcal{H}_1 = \mathcal{H}$, RoD
 $\mathcal{D}_1 = \mathcal{X}$. Current epoch $k = 1$. $M = md$. Parents
 $r(1) = 1$
while $t \leq T$ do
  Verification:
  Let $Z'_k = \emptyset$
  while $|Z'_k| < M$ do
    Let $t = t + 1$
    if $x_t \notin \mathcal{D}_{r(k)}$ then
      Choose any $h \in \mathcal{H}_{r(k)}$ and label $x_t$ with $h(x_t)$;
    else
      Query label $y_t$ and let $\mathcal{Z}'_k = \mathcal{Z}'_k \cup \{(x_t, y_t)\}$;
    end if
  end while
  if
 $\min_{h \notin \mathcal{H}_k} \epsilon_{\mathcal{Z}'_k}(h) - \min_{h \in \mathcal{H}_k} \epsilon_{\mathcal{Z}'_k}(h) < 2\Delta(M, 1 - p)$
 then
    Let $\mathcal{H}_{k+1} = \mathcal{H}_{r(k)}$, $\mathcal{D}_{k+1} = \mathcal{D}_{r(k)}$,
 $r(k + 1) = r(r(k))$, $k \leftarrow k + 1$;
    continue;
  end if
  Elimination:
  Let $Z_k = \emptyset$
  while $|Z_k| < M$ do
    Let $t = t + 1$
    if $x_t \notin \mathcal{D}_k$ then
      Choose any $h \in \mathcal{H}_k$ and label $x_t$ with $h(x_t)$;
    else
      Query label $y_t$ and let $\mathcal{Z}_k = \mathcal{Z}_k \cup \{(x_t, y_t)\}$;
    end if
  end while
  Update $\mathcal{H}_{k+1}$ as following:
      $\mathcal{H}_{k+1} = \{h \in \mathcal{H}_k : \epsilon_{\mathcal{Z}_k}(h) - \epsilon_{\mathcal{Z}_k}(h_k^*) <$
                  $6\Delta(M, 1 - \sqrt{p})\}$          (24)
  Update $\mathcal{D}_{k+1}$ according to (10).
  Let $r(k + 1) = k$, $k = k + 1$;
end while
```

4) Analysis of Regret and Label Complexity: *Theorem 6:* Let $\mathbb{E}[Q(T)]$ be the expected label complexity of the RW-OLA algorithm. If $m > 1024(\theta c_0)^2$, under Massart noise condition, there exists $C_2 > 0$ such that

$$\mathbb{E}[Q(T)] \leq C_2 md \log T, \tag{25}$$

where $\theta = \theta(\mathbb{P}_X, \mathcal{H})$ is the disagreement coefficient.

Proof: Here we provide a sketch of the proof. The detailed proof can be found in the [41, Appendix E]. We first show that the bias of the random walk is indeed bounded above p with the chosen thresholds. We then show that when the verification passes, the RoD in the next epoch will shrink with a fixed rate c. Based on these two statements, we can show that the expected RoD is decreasing exponentially with rate $c_1 = 1 - p + c^2 p < 1$. The same submartingale technique used in analyzing OLA is then used to bound the label complexity of RW-OLA. □

Under Massart noise, the epoch length for OLA is $md \log T$, which leads to $O(\log^2 T)$ label complexity. For RW-OLA, the epoch length is only md, which makes the label complexity only $O(\log T)$. In other words, to make sure RoD decreases exponentially, OLA requires epoch length to be $O(d \log T)$ but RW-OLA only requires it to be $O(1)$.

Theorem 7: Let $\mathbb{E}[R(T)]$ be the expected regret of the RW-OLA algorithm. If $m > 1024(\theta' c_0)^2$ and $\theta' > 0$, under Massart noise condition, there exists $C_3 > 0$ such that

$$\mathbb{E}[R(T)] \leq C_3 md \log T, \tag{26}$$

where

$$\theta' = \inf\left\{\frac{\phi[\mathcal{D}(B(h^*, r))]}{r} : r > 0\right\}. \tag{27}$$

is the modified disagreement coefficient.

Proof: Since an epoch k with $h^* \in \mathcal{H}_k$ incurs no regret, we only need to consider the case where $h^* \notin \mathcal{H}_k$. In this case, based on the RW-OLA algorithm, regret incurs if and only if the instance falls into a subset outside of RoD. Based on the bias of the random walk, we can show that the expected ratio of that subset to the current RoD is bounded by a constant. Since queries occur whenever the instances fall inside the RoD, we can show that the expected regret is upper bounded by this constant multiplying the expected label complexity, which is $O(d \log T)$. See [41, Appendix F] for the detailed proof.

B. Implementation for Homogeneous Linear Classification

There are several steps in OLA and RW-OLA that can be computationally expensive, which is inherent to the disagreement-based approach. Specifically, maintaining the version space

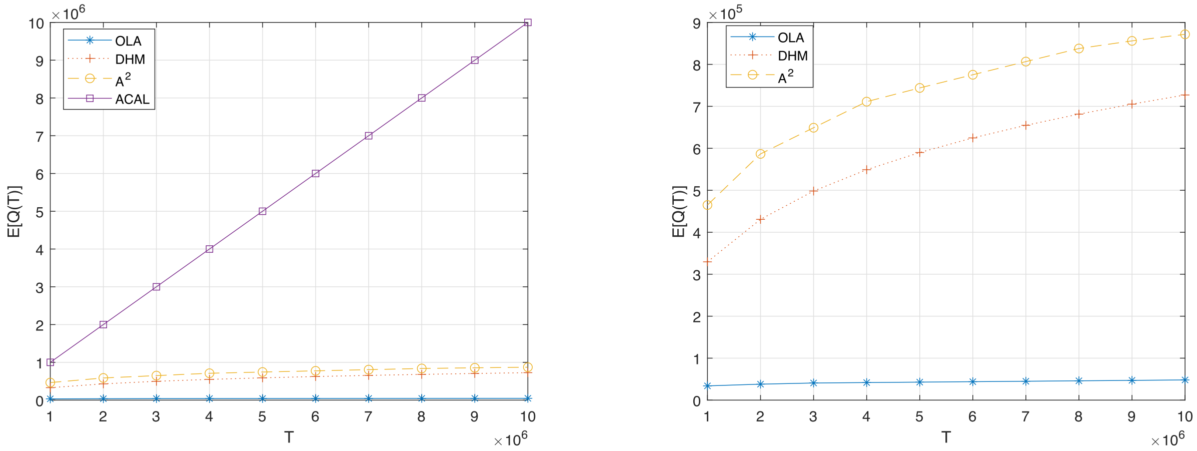

Fig. 2. Comparison with A^2, DHM, and ACAL ($d = 1$, Tsybakov noise with $\alpha = 1$ and $c_0 = 5$, $h^* = h_{0.5}$).

and RoD, and computing the best empirical hypothesis h_k^* can be costly. We discuss here approximate implementations with manageable computational complexity for homogeneous linear classification, drawing inspiration from techniques of using surrogate loss [13] and the Query-by-Committee approaches [14], [42].

In homogeneous linear classification, $\mathcal{X}$ is the surface of the d-dimension unit Euclidean sphere. Each hypothesis, as a linear separator that passes the origin, is given by a unit vector $\mathbf{u} \in \mathbb{R}^d$ such that the corresponding concept is $\{\mathbf{x} \in \mathcal{X} : \mathbf{ux} \geq 0\}$.

To estimate the best empirical hypothesis h_k^*, we use hinge loss function $l(z) = \max\{1 - z, 0\}$ to replace the 0-1 loss function in (11). Then, the best empirical hypothesis $\hat{h}_k^*$ under hinge loss is given by

$$\hat{h}_k^* = \min_{h \in \mathcal{H}_k} \sum_{(x,y) \in \mathcal{Z}_k} \max\{1 - (2y - 1)\mathbf{u}x, 0\}. \tag{28}$$

Then standard linear classification algorithms such as SVM can be employed to compute $\hat{h}_k^*$.

The version space is approximated with N constituent hypotheses sampled uniformly at random. Specifically, at $t = 1$, we sample N hypotheses uniformly at random from the entire hypothesis space $\{\mathbf{u} \in \mathbb{R}^d, ||\mathbf{u}|| = 1\}$ and form $\hat{\mathcal{H}}_1$. At each epoch k, for each hypothesis $h \in \hat{\mathcal{H}}_k$, we check whether it should be eliminated based on (12) and label them as $+1$ or -1 accordingly. Then, we run a linear classification algorithm to separate the hypotheses labeled $+1$ from those labeled -1 in the above process. This yields a linear classifier of the form $\omega\mathbf{u} + b \geq 0$ that represents an approximation of the new version space. To obtain an approximation of the new version space $\hat{\mathcal{H}}_{k+1}$, we again sample N hypotheses uniformly at random from $\{\mathbf{u} : \omega\mathbf{u} + b \geq 0\}$ to form the next version space $\hat{\mathcal{H}}_{k+1}$.

Since the version space is estimated by a finite number of hypotheses, instead of maintaining the RoD explicitly, we check whether $x_t \notin \mathcal{D}_k$ by checking whether all $h \in \hat{\mathcal{H}}_k$ agree on x_t. A detailed description of the algorithm is given in Algorithm 3.

For RW-OLA, both the verification stage and elimination stage can be implemented similarly. In particular, the elimination stage of RW-OLA is exactly the same as OLA except the threshold will be different. Therefore, it can be done by replacing the threshold in step 2 to maintain the version space and RoD. For the verification stage, which involves finding the best empirical hypothesis inside and outside of $\mathcal{H}_k$, can be done using the hinge loss replacement in (28) with a standard linear classification algorithm as well.

Algorithm 3: OLA for Homogeneous Linear Classification.

```
Initialization: Set $\mathcal{Z}_0 = \emptyset$, Random sample $\hat{\mathcal{H}}_0$
 uniformly from $\mathcal{H}$.
for $t = 1$ to $T$ do
  if All $h \in \hat{\mathcal{H}}_k$ agree on $x_t$ then
    Choose any $h \in \hat{\mathcal{H}}_k$ and label $x_t$ with $h(x_t)$;
  else
    Query label $y_t$ and let $\mathcal{Z}_k = \mathcal{Z}_k \cup \{(x_t, y_t)\}$;
    if $|\mathcal{Z}_k| = M$ then
      1. Find $h_k^*$ using $\mathcal{Z}_k$ and (28);
      2. For all $h \in \hat{\mathcal{H}}_k$, check whether $h \in \mathcal{H}_{k+1}$ based
 on (12) and label them accordingly;
      3. Find linear classifier $\omega\mathbf{u} + b \geq 0$ for $\hat{\mathcal{H}}_k$ and its
 label;
      4. Random sample $\hat{\mathcal{H}}_{k+1}$ from $\{\mathbf{u} : \omega\mathbf{u} + b \geq 0\}$;
    end if
  end if
end for
```

VI. Simulation Examples

We first compare the label complexity of OLA and RW-OLA with existing disagreement-based active learning algorithms. We first consider a one-dimensional instance space $\mathcal{X} = [0, 1]$ and threshold classifiers with $\mathcal{H} = \{h_z | 0 \leq z \leq 1\}$ where $h_z = [z, 1]$. Note that the VC dimension $d = 1$. We set $\mathbb{P}_X$ to be the uniform distribution. Fig. 2 and 3 show the comparison under different Tsybakov noise conditions.

In Fig. 4, we consider the same instance space $\mathcal{X} = [0, 1]$ and uniformly distributed instances, but a hypothesis space $\mathcal{H} = \{h_{z_1,z_2} | 0 \leq z_1, z_2 \leq 1\}$ consisting of all intervals $h_{z_1,z_2} =$

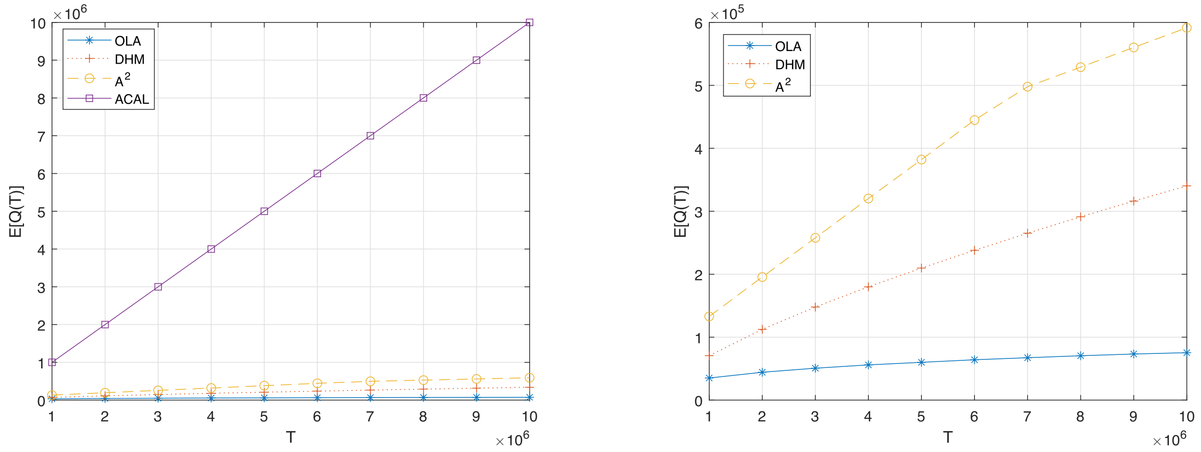

Fig. 3. Comparison with A^2, DHM, and ACAL ($d = 1$, Tsybakov noise with $\alpha = 0.5$ and $c_0 = 1$, $h^* = h_{0.5}$).

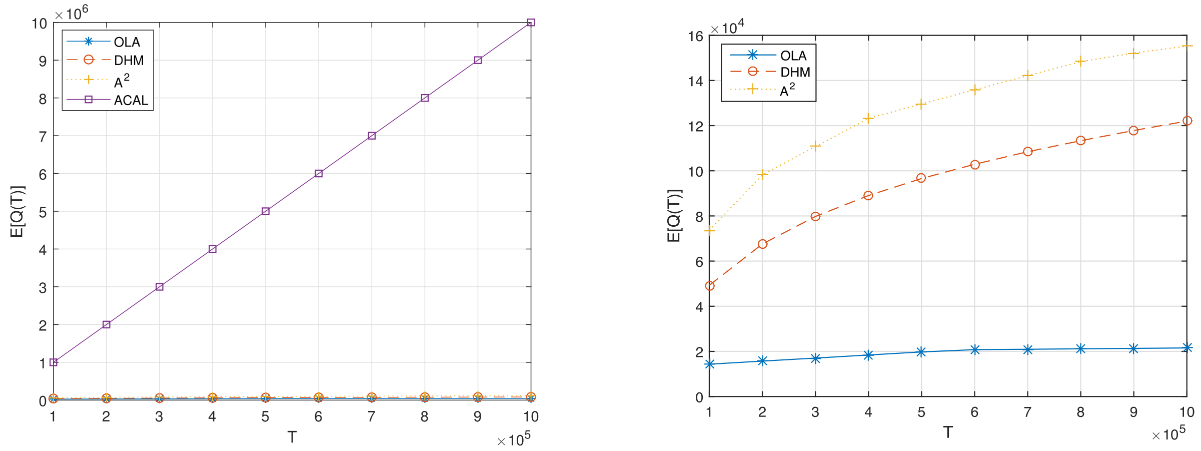

Fig. 4. Comparison with A^2, DHM, and ACAL for ($d = 2$, Tsybakov $\alpha = 1$ and $c_0 = 1$ $h^* = h_{0,25,0.75}$).

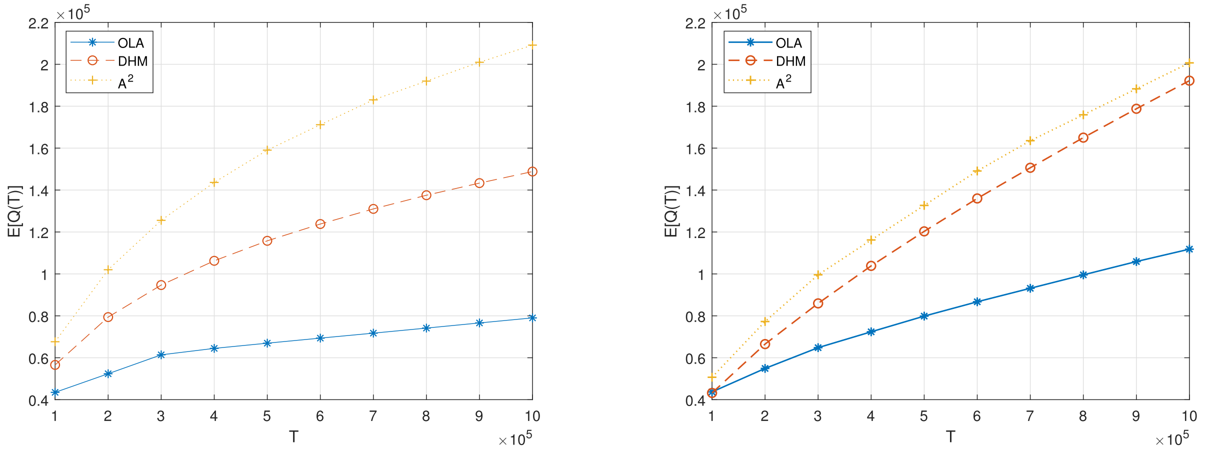

Fig. 5. Comparison with A^2 and DHM for ($d = 3$, $N = 50000$, Tsybakov noise with $\alpha = 1$ and $c_0 = 1$, $\alpha = 0.5$ and $c_0 = 5$, $h^* = (1, 0, 0)$).

$[z_1, z_2]$. Note that in this case, the VC dimension $d = 2$. For both cases, since the hypothesis space can be effectively represented by N^d points (with $d = 1, 2$) after taking N samples, we implemented an exact version of all the algorithms. For ACAL, the optimization problem to obtain the threshold was solved numerically.

Since the label complexity for ACAL is much larger than the others, we plot the others in the right figure. The significant reduction in label complexity offered by OLA and RW-OLA is evident from Figs. 2–4. The simulated classification errors are near zero for all the algorithms.

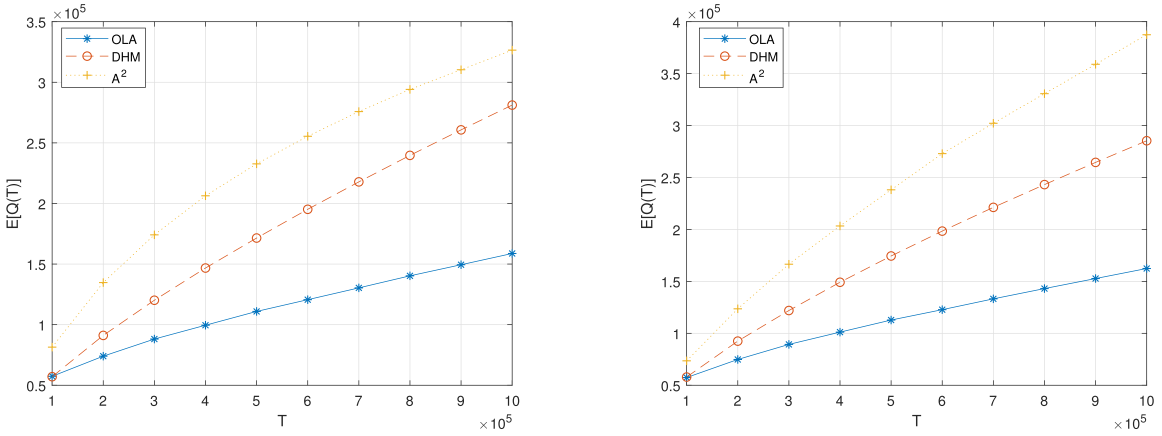

Fig. 6. Comparison with A^2 and DHM for ($d = 4$, $N = 50000$, Tsybakov noise with $\alpha = 1$ and $c_0 = 1$, $\alpha = 0.5$ and $c_0 = 5$ $h^* = (1, 0, 0, 0)$).

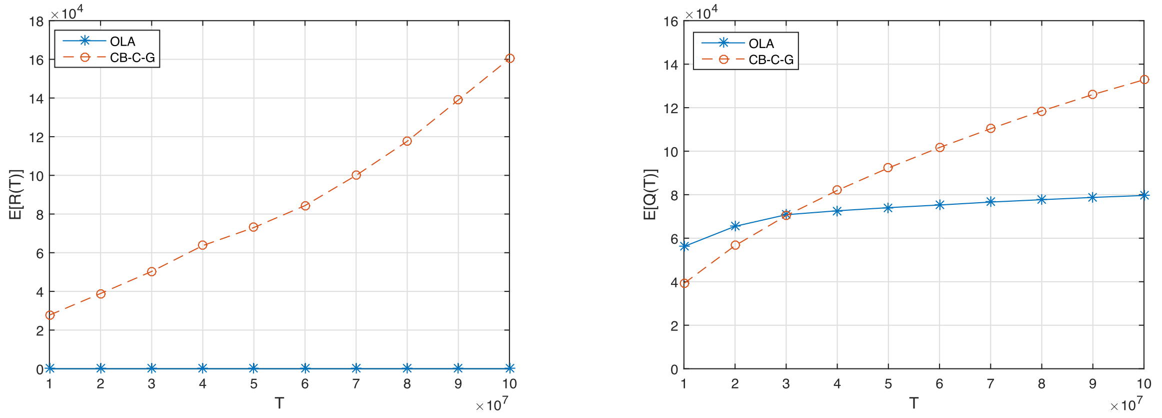

Fig. 7. Comparison with CB-C-G [30] under Tsybakov noise with $\alpha = 0.5$ and $c_0 = 1$.

Next we consider d-dimension homogeneous linear classification setting. Fig. 5 and 6 show the comparison for $d = 3, 4$. DHM and A^2 are implemented with similar methods as discussed in Section V-B. The simulated classification errors are near zero for all three algorithms.

Next we compare OLA with the online margin-based algorithm CB-C-G proposed in [30], [31]. It is specialized in learning homogeneous separators under specific noise model: there exists a fixed and unknown vector $\mathbf{u} \in \mathbb{R}^d$ with Euclidean norm $||\mathbf{u}|| = 1$ such that $\eta(\mathbf{x}) = (1 + \mathbf{u}^\top \mathbf{x})/2$. Then, the Bayes optimal classifier $h^*(\mathbf{x}) = \mathbb{1}[\mathbf{u}^\top \mathbf{x} \geq 0]$. Shown in Fig. 7 are the label complexity and classification error comparisons under this specific noise model with $d = 2, \mathbf{u} = (1, 0)$, and uniform $\mathbb{P}_X$. It shows that even when comparing under this special setting, OLA offers considerable reduction in label complexity and drastic improvement in classification accuracy. These simulation results suggest that the more conservative disagreement-based approach might be more suitable in the online setting than the aggressive margin-based approach, especially in terms of bounding the online classification errors. The reason is that the margin-based approaches aggressively seek out the most informative samples to query, which may have an advantage in the offline setting when unqueried samples can be skipped without labeling. In the online setting, however, such approaches result in more frequent self-labeling, hence higher rate of online classification errors.

VII. Conclusion

Online active learning has received considerably less attention than its offline counterpart. Real-time stream-based applications, however, necessitate a better understanding of this problem. The proposed algorithms and the established lower bounds in this work represent only initial attempts at addressing this problem. Much remains open. In particular, the characterization of the regret vs. label complexity tradeoff is incomplete, and online learning algorithms that can operate at any given point on the tradeoff curve require further investigation. Furthermore, an immediate direction is to consider the multi-class equivalent of this binary classification problem. A promising approach to this direction is via the use of output codes [43], [44].

Several other future directions include extending OLA for spaces that are difficult to characterize by VC dimension but can be more conveniently characterized by covering/bracketing numbers. Another interesting direction would to be consider the scenario of potentially correlated samples for which one would need to redesign the threshold and update schemes to account for the correlation among samples. Thus, this work offers a stepping stone towards a rich set of future directions.

References

[1] D. Cohn, L. Atlas, and R. Ladner, "Improving generalization with active learning," *Mach. Learn.*, vol. 15, no. 2, pp. 201–221, 1994.

[2] S. Dasgupta, "Two faces of active learning," *Theor. Comput. Sci.*, vol. 412, no. 19, pp. 1767–1781, 2011.

[3] S. Hanneke *et al.*, "Theory of disagreement-based active learning," *Foundations Trends Mach. Learn.*, vol. 7, no. 2-3, pp. 131–309, 2014.

[4] M.-F. Balcan, A. Beygelzimer, and J. Langford, "Agnostic active learning," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 65–72.

[5] S. Hanneke, "A bound on the label complexity of agnostic active learning," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 353–360.

[6] S. Hanneke, "Adaptive rates of convergence in active learning," in *Proc. 22nd Conf. Learn. Theory (COLT)*, 2009.

[7] V. Koltchinskii, "Rademacher complexities and bounding the excess risk in active learning," *J. Mach. Learn. Res.*, vol. 11, no. 9, pp. 2457–2485, 2010.

[8] S. Hanneke *et al.*, "Rates of convergence in active learning," *Ann. Statist.*, vol. 39, no. 1, pp. 333–361, 2011.

[9] A. Beygelzimer, S. Dasgupta, and J. Langford, "Importance weighted active learning," in *Proc. 26th Int. Conf. Mach. Learn.*, 2009, pp. 49–56.

[10] A. Beygelzimer, D. J. Hsu, J. Langford, and T. Zhang, "Agnostic active learning without constraints," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 199–207.

[11] A. Beygelzimer, D. Hsu, N. Karampatziakis, J. Langford, and T. Zhang, "Efficient active learning," in *Proc. Workshop On-line Trading Exploration Exploitation*, 2011.

[12] S. Dasgupta, D. J. Hsu, and C. Monteleoni, "A general agnostic active learning algorithm," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 353–360.

[13] S. Hanneke and L. Yang, "Surrogate losses in passive and active learning," *Electron. J. Statist.*, vol. 13, no. 2, pp. 4646–4708, Jul. 2019.

[14] S. Dasgupta, A. T. Kalai, and C. Monteleoni, "Analysis of perceptron-based active learning," in *Proc. Int. Conf. Comput. Learn. Theory*, 2005, pp. 249–263.

[15] M.-F. Balcan, A. Broder, and T. Zhang, "Margin based active learning," in *Proc. Int. Conf. Comput. Learn. Theory*, 2007, pp. 35–50.

[16] M.-F. Balcan and P. Long, "Active and passive learning of linear separators under log-concave distributions," in *Proc. Conf. Learn. Theory*, 2013, pp. 288–316.

[17] P. Awasthi, M. F. Balcan, and P. M. Long, "The power of localization for efficiently learning linear separators with noise," in *Proc. 46th Annu. ACM Symp. Theory Comput.*, 2014, pp. 449–458.

[18] P. Awasthi, M.-F. Balcan, N. Haghtalab, and R. Urner, "Efficient learning of linear separators under bounded noise," in *Conf. Learn. Theory*, 2015, pp. 167–190.

[19] P. Awasthi, M. F. Balcan, N. Haghtalab, and H. Zhang, "Learning and 1-bit compressed sensing under asymmetric noise," *J. Mach. Learn. Res.*, vol. 49, no. 6, pp. 152–192, 2016.

[20] C. Zhang, "Efficient active learning of sparse halfspaces," in *Proc. Mach. Learn. Res.*, 2018, vol. 75, pp. 1–26.

[21] C. Cortes, G. DeSalvo, C. Gentile, M. Mohri, and N. Zhang, "Region-based active learning," in *Proc. 22nd Int. Conf. Artif. Intell. Statist.*, 2019, pp. 2801–2809.

[22] C. Zhang and K. Chaudhuri, "Beyond disagreement-based agnostic active learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 1, 2014, vol. 1, pp. 442–450. [Online]. Available: https://arxiv.org/abs/1407.2657v2

[23] B. Settles, "Active learning," *Synth. Lectures Artif. Intell. Mach. Learn.*, vol. 6, no. 1, pp. 1–114, 2012.

[24] J. Haupt, R. M. Castro, and R. Nowak, "Distilled sensing: Adaptive sampling for sparse detection and estimation," *IEEE Trans. Inf. Theory*, vol. 57, no. 9, pp. 6222–6235, Sep. 2011.

[25] T. Tsiligkaridis, B. M. Sadler, and A. O. Hero, "Collaborative 20 questions for target localization," *IEEE Trans. Inf. Theory*, vol. 60, no. 4, pp. 2233–2252, Apr. 2014.

[26] J. Lipor, B. P. Wong, D. Scavia, B. Kerkez, and L. Balzano, "Distance-penalized active learning using quantile search," *IEEE Trans. Signal Process.*, vol. 65, no. 20, pp. 5453–5465, Oct. 2017.

[27] J. Paisley, X. Liao, and L. Carin, "Active learning and basis selection for kernel-based linear models: A Bayesian perspective," *IEEE Trans. Signal Process.*, vol. 58, no. 5, pp. 2686–2700, May 2010.

[28] A. Tsakmalis, S. Chatzinotas, and B. Ottersten, "Constrained Bayesian active learning of a linear classifier," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, 2018, pp. 6663–6667.

[29] P. Sattari, M. Kurant, A. Anandkumar, A. Markopoulou, and M. G. Rabbat, "Active learning of multiple source multiple destination topologies," *IEEE Trans. Signal Process.*, vol. 62, no. 8, pp. 1926–1937, Apr. 2014.

[30] N. Cesa-Bianchi, A. Conconi, and C. Gentile, "Learning probabilistic linear-threshold classifiers via selective sampling," in *Learning Theory and Kernel Machines*. Berlin, Germany: Springer, 2003, pp. 373–387.

[31] G. Cavallanti, N. Cesa-Bianchi, and C. Gentile, "Linear classification and selective sampling under low noise conditions," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 249–256.

[32] L. Yang, "Active learning with a drifting distribution," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2079–2087.

[33] C. Cortes, G. DeSalvo, C. Gentile, M. Mohri, and N. Zhang, "Active learning with disagreement graphs," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 2468–2483. [Online]. Available: https://proceedings.mlr.press/v97/cortes19b.html

[34] A. B. Tsybakov *et al.*, "Optimal aggregation of classifiers in statistical learning," *Ann. Statist.*, vol. 32, no. 1, pp. 135–166, 2004.

[35] V. Koltchinskii *et al.*, "Local Rademacher complexities and oracle inequalities in risk minimization," *Ann. Statist.*, vol. 34, no. 6, pp. 2593–2656, 2006.

[36] R. M. Castro and R. D. Nowak, "Minimax bounds for active learning," *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 2339–2353, May 2008.

[37] S. Hanneke and L. Yang, "Minimax analysis of active learning," *J. Mach. Learn. Res.*, vol. 16, no. 1, pp. 3487–3602, 2015.

[38] P. Massart *et al.*, "Risk bounds for statistical learning," *Ann. Statist.*, vol. 34, no. 5, pp. 2326–2366, 2006.

[39] O. Bousquet, S. Boucheron, and G. Lugosi, "Introduction to statistical learning theory," in *Advanced Lectures on Machine Learning*. Berlin, Germany: Springer, 2004, pp. 169–207.

[40] V. N. Vapnik and A. Y. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities," in *Measures of Complexity*. Berlin, Germany: Springer, 2015, pp. 11–30.

[41] B. Huang, S. Salgia, and Q. Zhao, "Disagreement-based active learning in online settings," 2020, *arXiv:1904.09056*.

[42] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Mach. Learn.*, vol. 28, no. 2-3, pp. 133–168, 1997.

[43] M. F. Balcan, T. Dick, and Y. Mansour, "Label efficient learning by exploiting multi-class output codes," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2015, pp. 1735–1741. [Online]. Available: http://arxiv.org/abs/1511.03225

[44] S. Gu, Y. Cai, J. Shan, and C. Hou, "Active learning with error-correcting output codes," *Neurocomputing*, vol. 364, pp. 182–191, Oct. 2019.