Active Learning for Classification with Abstention

Shubhanshu Shekhar
ECE Department, UCSD
shshekha@eng.ucsd.edu

Mohammad Ghavamzadeh
Facebook AI Research
mgh@fb.com

Tara Javidi
ECE Department, UCSD
tjavidi@eng.ucsd.edu

***Abstract*—We consider the problem of binary classification with the caveat that the learner can abstain from declaring a label incurring a cost $\lambda \in [0, 1/2]$ in the process. This is referred to as the problem of binary classification with a fixed-cost of abstention. For this problem, we propose an active learning strategy that constructs a non-uniform partition of the input space and focuses sampling in the regions near the decision boundaries. Our proposed algorithm can work in all the commonly used active learning query models, namely *membership-query*, *pool-based* and *stream-based*. We obtain an upper bound on the excess risk of our proposed algorithm under standard smoothness and margin assumptions and demonstrate its minimax near-optimality by deriving a matching (modulo poly-logarithmic factors) lower bound. The achieved minimax rates are always faster than the corresponding rates in the passive setting, and furthermore the improvement increases with larger values of the smoothness and margin parameters.**

A full version of this paper is accessible at: https://arxiv.org/abs/1906.00303

I. Introduction

Standard binary classification involves using a training set $S_n = \{(X_i, Y_i) \quad : \quad 1 \leq i \leq n\}$ to learn a classifier f which maps elements of an input space $\mathcal{X}$ to the set of binary labels $\mathcal{Y} = \{0, 1\}$. The performance of the learned classifier is measured by its expected probability of error according to some joint distribution P_{XY} on $\mathcal{X} \times \mathcal{Y}$. In this paper, we study a generalization of this problem where in addition to the labels 0 and 1, the learner has an option of *abstaining* from declaring a label (i.e., saying "don't know"). Every time the learner uses this option, it incurs a fixed cost $\lambda \in (0, 1/2]$. We design an active learning strategy for this problem and obtain upper and lower bounds on the excess risk of the resulting classifier in a non-parametric framework.

This problem models several practical scenarios in which it is preferable to withhold a decision at the cost of some additional experimentation, instead of making an incorrect decision and incurring much higher costs. The fixed-cost formulation is particularly suitable for applications in which a precise cost can be assigned to additional experimentation due to using the abstain option. A canonical application of this problem is in automated medical diagnostic systems [1], where classifiers which defer to a human expert on uncertain inputs are more desirable than those that always make a decision. Other key applications include dialog systems and detecting malicious contents on the web.

Active learning is a learning paradigm in which the learner constructs the training set (S_n) by sequentially requesting labels at certain input points selected based on the observed data. This is in contrast to the *passive learning* framework, in which S_n is constructed in an $i.i.d.$ manner. Existing results in the literature, such as [2], [3], have demonstrated the benefits of active (over passive) learning, in terms of improved sample complexity or equivalently, lower excess risk, in standard binary classification. However, in the case of classification with abstention, the design of active learning algorithms and their comparison with their passive counterparts have largely been unexplored. In this paper, we aim to fill this gap in the literature.

Prior Work. The study of the problem of classification with an abstain option was initiated by Chow in [4] and [5]. In [4], the author showed that a threshold-type classifier (see § II) is Bayes optimal for this problem, while in [5] he derived the trade-off curve between the error-rate and abstention-rate. [6] studied this problem in a non-parametric framework (similar to this paper), and derived the convergence rates on the excess risk of plug-in and empirical risk minimization based classifiers.

Another line of work in this area, involves the design of appropriate convex surrogate loss functions and analyzing their consistency. In particular, [7] proposed the Generalized Hinge Loss and proved results on its calibration and excess risk, while [8] obtained the necessary and sufficient conditions for the infinite sample complexity of arbitrary convex surrogate loss functions. Other related works include [9] and [10] that analyzed the binary classification with abstain option with ℓ_1-regularization. More recently, [11] studied abstaining classifiers which are represented by a pair of functions (h, r). The sign of the function h is used for predicting a label, while the sign of r is used to decide whether to abstain or not. They proposed new calibrated convex surrogate loss functions for this problem and obtained generalization and consistency guarantees.

Contributions. Our main contributions are as follows:

1) We begin by proposing an active learning algorithm for the *fixed-cost* setting with knowledge of the smoothness of the regression function, and obtain bounds on its excess risk. The proposed algorithm is general enough to work for the three most commonly used active learning query models: *membership query*, *pool-based*, and *stream-based* (Section III-A).

2) We then demonstrate the minimax near-optimality of our proposed algorithm by deriving matching (modulo poly-logarithmic terms) lower-bound on the excess risk. The lower-bound proof relies on a new comparison inequality for classification with abstention, and a novel construction of a class of *hard* problems (Section III-B).

3) Finally, we use a simple class of learning problems and empirically verify the benefits of active (over passive) learning

as predicted by our theoretical results (Section IV).

II. Preliminaries

Let $\mathcal{X} = [0,1]^D$ denote the input space and $\mathcal{Y} = \{0,1\}$ denote the set of labels to be assigned to points in $\mathcal{X}$. We use d to denote the Euclidean metric on $\mathcal{X}$, i.e., for all $x, x' \in \mathcal{X}$, $d(x,x') := \sqrt{\sum_{i=1}^{D}(x_i - x_i')^2}$. A binary classification problem is completely specified by P_{XY}, i.e., the joint distribution of the input-label random variables. Equivalently, it can also be represented in terms of the marginal over the input space, P_X, and the *regression function* $\eta(x) := P_{Y|X}(Y = 1 \mid X = x)$. A (randomized) abstaining classifier is defined as a mapping $g : \mathcal{X} \mapsto \mathcal{P}(\mathcal{Y}_1)$, where $\mathcal{Y}_1 = \mathcal{Y} \cup \{\Delta\}$, the symbol Δ represents the option of the classifier to abstain from declaring a label, and $\mathcal{P}(\mathcal{Y}_1)$ represents the set of probability distributions on $\mathcal{Y}_1$. Such a classifier g comprises of three functions $g_i : \mathcal{X} \to [0,1]$, for $i \in \mathcal{Y}_1$, satisfying $\sum_{i \in \mathcal{Y}_1} g_i(x) = 1$, for each $x \in \mathcal{X}$. A classifier g is called *deterministic* if the functions g_i take values in $\{0,1\}$. Every deterministic classifier g partitions $\mathcal{X}$ into three disjoint sets (G_0, G_1, G_Δ).

In this paper, we focus on the problem of binary classification with a *fixed cost* of abstention, in which every usage of the abstain option results in a fixed cost $\lambda \in [0, 1/2]$. The corresponding classification loss is defined as $l_\lambda(g,x,y) := \mathbb{1}_{\{g(x) \neq \Delta\}} \mathbb{1}_{\{g(x) \neq y\}} + \lambda \mathbb{1}_{\{g(x) = \Delta\}}$, and the goal is to learn a classifier g which minimizes the expected loss $\mathbb{E}[l_\lambda(g, X, Y)]$. The Bayes optimal classifier for this problem is defined as $g^*_\lambda(x) = 1$, 0, or Δ, depending on whether $1 - \eta(x)$, $\eta(x)$, or λ is the smallest.

Active Learning Models: For this problem, we propose an active classification algorithm for three commonly used active learning models [12, § 2]: (i) *membership query* (MQ), (ii) *pool-based* (PB), and (iii) *stream-based* (SB). MQ is the strongest query model, in which the learner can request labels at any point of the input space. We use a slightly weaker version of MQ in this paper that only requires labels sampled from P_X restricted to certain partitions of $\mathcal{X}$, which we introduce in Definition 1. In the PB model, the learner is provided with a pool of unlabelled samples and must request labels of a subset of the pool. Finally, in the SB model, the learner receives a stream of samples and must decide whether to request a label or discard the sample.

A. Definitions

To construct our active classifier, we will require a hierarchical sequence of partitions of the input space, called the tree of partitions [13], [14].

Definition 1. *A sequence of subsets $\{\mathcal{X}_h\}_{h \geq 0}$ of $\mathcal{X}$ is said to form a* tree of partitions *of $\mathcal{X}$, if they satisfy the following properties:* ***(i)*** *$|\mathcal{X}_h| = 2^h$ and we denote the elements of $\mathcal{X}_h$ by $x_{h,i}$, for $1 \leq i \leq 2^h$,* ***(ii)*** *for every $x_{h,i} \in \mathcal{X}_h$, we denote by $\mathcal{X}_{h,i}$, the* cell *associated with $x_{h,i}$, which is defined as $\mathcal{X}_{h,i} := \{x \in \mathcal{X} \mid d(x, x_{h,i}) \leq d(x, x_{h,j}),\ \forall j \neq i\}$, where ties are broken in an arbitrary but deterministic manner,* ***(iii)*** *we have $\mathcal{X}_{h,i} = \mathcal{X}_{h+1,2i-1} \cup \mathcal{X}_{h+1,2i}$ for all h, i pairs, and* ***(iv)*** *there exist constants $0 < v_2 \leq 1 \leq v_1$ and $\rho \in (0,1)$, such that for all h and i, we have $B(x_{h,i}, v_2\rho^h) \subset \mathcal{X}_{h,i} \subset B(x_{h,i}, v_1\rho^h)$, where $B(x,a) := \{x' \in \mathcal{X} \mid d(x,x') < a\}$ is the open ball in $\mathcal{X}$ centered at x with radius a.*

Remark 1. *For the metric space $(\mathcal{X}, d)$ considered in our paper, i.e., $\mathcal{X} = [0,1]^D$ and d being the Euclidean metric, the cells $\mathcal{X}_{h,i}$ are D-dimensional rectangles. Thus, a suitable choice of parameter values for our algorithms are $\rho = 2^{-1/D}$, $v_1 = 2\sqrt{D}$, and $v_2 = 1/2$.*

Next, we define the dimensionality of the region of the input space at which the regression function $\eta(\cdot)$ is close to some threshold value γ.

Definition 2. *For a function $\zeta : [0,\infty) \mapsto [0,\infty)$ and a threshold $\gamma \in (0, 1/2)$, we define the near-γ dimension associated with $(\mathcal{X}, d)$ and the regression function $\eta(\cdot)$ as*

$$D_\gamma(\zeta) := \inf\left\{a \geq 0 \mid \exists C > 0 :\ M\big(\mathcal{X}_\gamma\big(\zeta(r)\big), r\big) \leq C r^{-a},\ \forall r > 0\right\}, \tag{1}$$

where $\mathcal{X}_\gamma\big(\zeta(r)\big) := \big\{x \in \mathcal{X} : |\eta(x) - \gamma| \leq \zeta(r)\big\}$ and $M(S,r)$ is the r packing number of $S \subseteq (\mathcal{X}, d)$.

The above definition is motivated by similar definitions used in the bandit literature, such as the *near-optimality dimension* [13] and the *zooming dimension* [15]. For the case of $\mathcal{X} = [0,1]^D$ considered in this paper, the term $D_\gamma(\zeta)$ must be no greater than D, i.e., $D_\gamma(\zeta) \leq D$. This is because $\mathcal{X}_\gamma\big(\zeta(r)\big) \subset \mathcal{X}$, for all $r > 0$, and there exists a constant $C_D < \infty$, such that $M(\mathcal{X}, r) \leq C_D r^{-D}$, for all $r > 0$.

Remark 2. *We will use an instance of near-γ dimension for stating our results defined as $\tilde{D} = \max_{j=1,2}\{\tilde{D}_j\}$, where $\tilde{D}_j := D_{\gamma_j}(\zeta_1)$ with $\zeta_1(r) = 12(\frac{L_1 v_1}{v_2})^\beta r^\beta$ and $\gamma_j = \frac{1}{2} + (-1)^j(\frac{1}{2} - \lambda)$.*

III. Main Results

A. Active Learning Algorithm

We now propose an active learning algorithm for classification with a fixed-cost λ of abstention, and obtain theoretical bounds on its excess risk under the following two standard assumptions:

(MA) The joint distribution P_{XY} of the input-label pair satisfies the *margin assumption* with parameters $C_0 > 0$ and $\alpha_0 \geq 0$, for $\gamma \in \{1/2 - \lambda, 1/2 + \lambda\}$, which means that for any $0 < t \leq 1$, we have $P_X(|\eta(X) - \gamma| \leq t) \leq C_0 t^{\alpha_0}$.

(HÖ) The regression function η is Hölder continuous with parameters $L > 0$ and $0 < \beta \leq 1$, i.e., for all $x_1, x_2 \in (\mathcal{X}, d)$, we have $|\eta(x_1) - \eta(x_2)| \leq L \times d(x_1, x_2)^\beta$.

The (HÖ) smoothness assumption is standard in nonparametric estimation [16, Chapter 1], while the (MA) condition is a natural generalization of the usual Tsybakov's margin condition (see e.g., [17]) for binary classification, and it has been widely employed in this form in the classification with abstention literature [6], [8], [18]

Outline of Algorithm 1: At any time t, the algorithm maintains a set of active points $\mathcal{X}_t \subset \cup_{h \geq 0} \mathcal{X}_h$, such that the cells associated with the points in $\mathcal{X}_t$ partition the whole $\mathcal{X}$, i.e., $\cup_{x_{h,i} \in \mathcal{X}_t} \mathcal{X}_{h,i} = \mathcal{X}$. The set $\mathcal{X}_t$ is further divided into *classified* active points, $\mathcal{X}_t^{(c)}$, *unclassified* active points, $\mathcal{X}_t^{(u)}$,

Algorithm 1: Active learning with abstention.

Input: n, λ, L, β, v_1, ρ, $h_{\max} = \log n$

1 **Initialize** $t = 1$, $n_e = 0$, $\mathcal{X}_t = \{x_{0,1}\}$, $\mathcal{X}_t^{(u)} = \mathcal{X}_t$, $\mathcal{X}_t^{(c)} = \emptyset$, $\mathcal{X}_t^{(d)} = \emptyset$

2 **while** $n_e \leq n$ **do**

3 **for** $x_{h,i} \in \mathcal{X}_t^{(u)}$ **do**

4 **if** $[l_t(x_{h,i}), u_t(x_{h,i})] \cap \{1/2 - \lambda, 1/2 + \lambda\} = \emptyset$ **then**

5 $\mathcal{X}_t^{(c)} \leftarrow \mathcal{X}_t^{(c)} \cup \{x_{h,i}\}$

6 **end**

7 **end**

8 $x_{h_t,i_t} \in \arg\max_{x_{h,i} \in \mathcal{X}_t^{(u)}} I_t^{(1)}(x_{h,i}) = u_t(x_{h,i}) - l_t(x_{h,i})$

9 **if** $\left(e_t\big(n_{h_t,i_t}(t)\big) < L(v_1 \rho^{h_t})^{\beta}\right)$ *and* $(h_t < h_{\max})$ **then**

10 $\mathcal{X}_t^{(u)} \leftarrow \mathcal{X}_t^{(u)} \setminus \{x_{h_t,i_t}\} \cup \{x_{h_t+1,2i_t-1}, x_{h_t+1,2i_t}\}$

11 $u_t(x_{h_t+1,i'}) \leftarrow u_t(x_{h_t,i_t})$, $l_t(x_{h_t+1,i'}) \leftarrow l_t(x_{h_t,i_t})$, for $i' \in \{2i_t - 1, 2i_t\}$

12 **else**

13 call REQUEST_LABEL

14 **end**

15 $t \leftarrow t + 1$

16 **end**

Output: $\hat{g}$ defined by Eq. (2)

and *discarded* points, $\mathcal{X}_t^{(d)}$. The classified points are those at which the value of η has been estimated sufficiently well so that we do not need to evaluate them further. The unclassified points require further evaluation and perhaps refinement before making a decision. The discarded points are those for which we do not have sufficiently many unlabelled samples in their cells (in the *stream-based* and *pool-based* settings). For every active point, the algorithm computes high probability upper and lower bounds on the maximum and minimum η values in the cell associated with the point. The difference of these upper and lower bounds can be considered as a surrogate for the uncertainty in the η value in a cell. In every round, the algorithm selects a candidate point from the unclassified set that has the largest value of this uncertainty. Having chosen the candidate point, the algorithm either refines the cell or asks for a label at that point.

At a high level, Algorithm 1 involves repeating the following two steps: **1)** Maintaining a partition of the input space, and for each set in the partition, constructing upper and lower confidence bounds for the maximum and minimum (respectively) η values in the cell, and **2)** Based on these confidence bounds, either refine the partition or request a label. Finally, when the sampling budget is exhausted, **3)** aggregate the information gathered by the sampling strategy to define an abstaining classifier. We now describe these three steps in more details.

a) Confidence Interval Construction: At $t \geq 1$, for any cell $\mathcal{X}_{h,i}$ associated with $x_{h,i} \in \mathcal{X}_t$ and $n_{h,i}(t)$ denoting the number of queries in the cell $\mathcal{X}_{h,i}$ before time t, we compute an upper-bound on the maximum η value in the cell as $u_t(x_{h,i}) := \min\{u_{t-1}(x_{h,i}), \bar{u}_t(x_{h,i})\}$, where $\bar{u}_t(x_{h,i}) = \hat{\eta}_t(x_{h,i}) + e_t(n_{h,i}(t)) + V_h$. Here we have $\hat{\eta}_t(x_{h,i}) = \frac{1}{n_{h,i}(t)} \sum_{s=1}^{t} \mathbb{1}_{\{x_{h_t,i_t} \in \mathcal{X}_{h,i}\}} y_t$, $e_t(n_{h,i}(t)) = \sqrt{\frac{2\log(2\pi^2 t^3 n/3)}{n_{h,i}(t)}}$ (see [19, Lemma 3]), and $V_h = L\left(v_1 \rho^h\right)^{\beta}$ is an upper-bound on the maximum variation of the η value in a cell at level h of the tree of partitions $(\mathcal{X}_h)_{h \geq 0}$. We can define the lower-bound on the minimum η value in the cell in a similar manner, $l_t(x_{h,i}) := \max\{l_{t-1}(x_{h,i}), \bar{l}_t(x_{h,i})\}$, where $\bar{l}_t(x_{h,i}) := \hat{\eta}_t(x_{h,i}) - e_t(n_{h,i}(t)) - V_h$. We set $l_0(x_{h,i}) = -\infty$ and $u_0(x_{h,i}) = +\infty$ for all $x_{h,i}$.

b) Refine or Request Label: In order to select a candidate point, Algorithm 1 selects an *unclassified* point with maximum amount of uncertainty in its value. The uncertainty is measured by the index $I_t^{(1)}(x_{h,i}) = u_t(x_{h,i}) - l_t(x_{h,i})$ (Line 8). Having selected a candidate point x_{h_t,i_t} at time t, the algorithm either *refines* the cell (Lines 9-11) or requests a label depending on the relative magnitudes of $e_t(n_{h_t,i_t}(t))$ and V_{h_t} (Line 13). The label request depends on the query model and consists

Algorithm 2: REQUEST_LABEL

Input: Mode, x_{h_t,i_t}, n_e, $\mathcal{X}_t^{(d)}$, $\mathcal{X}_t^{(u)}$

1 Flag $\leftarrow$ False;

2 **if** *Mode*=='*Membership*' **then**

3 $x_t \sim P_X(\cdot \mid \mathcal{X}_{h_t,i_t})$, $y_t \sim$ Bernoulli$(\eta(x_t))$, Increment $\leftarrow$ True ;

4 **else if** *Mode*=='*Pool*' **then**

5 **if** $Z_t \cap \mathcal{X}_{h_t,i_t} \neq \emptyset$ **then**

6 choose $\tilde{x}_{h_t,i_t} \in Z_t \cap \mathcal{X}_{h_t,i_t}$ arbitrarily ;

7 $y_t \sim$ Bernoulli$(\eta(\tilde{x}_{h_t,i_t}))$, $Z_t \leftarrow Z_t \setminus \{\tilde{x}_{h_t,i_t}\}$, Increment $\leftarrow$ True;

8 **else**

9 $\mathcal{X}_t^{(d)} \leftarrow \mathcal{X}_t^{(d)} \cup \{x_{h_t,i_t}\}$, $\mathcal{X}_t^{(u)} \leftarrow \mathcal{X}_t^{(u)} \setminus \{x_{h_t,i_t}\}$;

10 **end**

11 **else**

12 counter $\leftarrow 1$, discard $\leftarrow$ True, Flag $\leftarrow$ True ;

13 **while** $\left(\textit{counter} \leq N_n\right)$ *AND Flag* **do**

14 Observe next element of the stream $x \sim P_X$;

15 **if** $x \in \mathcal{X}_{h_t,i_t}$ **then**

16 $y_t \sim$ Bernoulli$(\eta(x))$, discard $\leftarrow$ False, Increment $\leftarrow$ True, ;

17 Break

18 **end**

19 counter $\leftarrow$ counter $+1$;

20 **end**

21 **if** *discard* **then**

22 $\mathcal{X}_t^{(d)} = \mathcal{X}_t^{(d)} \cup \{x_{h_t,i_t}\}$, $\mathcal{X}_t^{(u)} = \mathcal{X}_t^{(u)} \setminus \{x_{h_t,i_t}\}$;

23 **end**

24 **if** *Increment* **then**

25 $n_e \leftarrow n_e + 1$;

26 **end**

27 **end**

of the following steps: (i) In the *membership query model* (MQ), the point x_t for which we request the label is drawn from the distribution P_X restricted to the cell $\mathcal{X}_{h_t,i_t}$. (ii) In the *pool-based model* (PB), we request the label if there is an unlabelled sample remaining in the cell $\mathcal{X}_{h_t,i_t}$, otherwise, we remove x_{h_t,i_t} from $\mathcal{X}_t^{(u)}$ and add it to $\mathcal{X}_t^{(d)}$. (iii) In the *stream-based model* (SB), we discard the samples until a point in $\mathcal{X}_{h_t,i_t}$ arrives. If $N_n = 2n^2\log(n)$ samples have been discarded, we remove x_{h_t,i_t} from $\mathcal{X}_t^{(u)}$ and add it to $\mathcal{X}_t^{(d)}$. The pseudo-code of the above three steps is provided in the subroutine REQUEST_LABEL in Algorithm 2.

c) Classifier Definition.: Let t_n denote the time at which the n'th query is made and Algorithm 1 halts. We define the final estimate of the regression function as $\hat{\eta}(x) = \hat{\eta}_{t_n}\big(\pi_{t_n}(x)\big)$, where $\pi_{t_n}(x) \coloneqq \big\{x_{h,i} \in \mathcal{X}_{t_n} \mid x \in \mathcal{X}_{h,i}\big\}$, and the discarded region of the input space as $\tilde{\mathcal{X}}_n^{(d)} \coloneqq \cup_{x_{h,i}\in\mathcal{X}_{t_n}^{(d)}} \mathcal{X}_{h,i}$. Finally, the classifier returned by the algorithm is defined as

$$\hat{g}(x) = \begin{cases} 1 & \text{if } \; u_{t_n}\big(\pi_{t_n}(x)\big) > 1-\lambda \;\text{ or }\; x \in \tilde{\mathcal{X}}_n^{(d)}, \\ 0 & \text{if } \; l_{t_n}\big(\pi_{t_n}(x)\big) < \lambda \;\text{ and }\; x \notin \tilde{\mathcal{X}}_n^{(d)}, \\ \Delta & \text{otherwise.} \end{cases} \tag{2}$$

Analysis: Before stating an upper-bound on the excess risk of Algorithm 1, we show (Lemma 1) that it will suffice to prove this bound for the MQ model. More specifically, we show that under mild assumptions, the P_X measure of $\tilde{\mathcal{X}}_n^{(d)}$ in PB and SB models is no larger than $1/n$ with probability at least $(1-1/n)$. This implies that in these two models, with high probability, the misclassification risk of $\hat{g}$ can be upper-bounded by $1/n + P_{XY}\big(\hat{g}(X) \neq Y,\ \hat{g}(X) \neq \Delta,\ X \notin \tilde{\mathcal{X}}_n^{(d)}\big)$, where the analysis of the second term is identical for all three active learning models.

Lemma 1. *Assume that in the pool-based model, the pool size $M_n > \max\{2n^3, 16n^2\log(n)\}$, and in the stream-based model, $N_n = 2n^2\log(n)$. Then, we have $\mathbb{P}\big(P_X(\tilde{\mathcal{X}}_n^{(d)}) > 1/n\big) \leq 1/n$.*

Thus, given Lemma 1, we can proceed with the analysis under the MQ model, with the knowledge that the same result holds for the other two models with an additional $1/n$ term. We now obtain an upper-bound on the excess risk of the classifier constructed by Algorithm 1.

Theorem 1. *Suppose that the assumptions (MA) and (HÖ) hold, and let $\tilde{D}$ be the dimension term defined in Remark 2. For $a > \tilde{D}$ and the corresponding C_a, assume n is large enough to ensure $(\frac{n}{\log n}) \geq (\frac{64C_a}{L^2 v_1^{2\beta} v_2^a})(\frac{8Lv_1^\beta}{\rho^\beta})^{(2\beta+a)/\beta}$. Then, for the classifier $\hat{g}$ defined by (2), with probability at least $1-2/n$, we have $R_\lambda(\hat{g}) - R_\lambda(g_\lambda^*) = \tilde{\mathcal{O}}\big(n^{-\beta(\alpha_0+1)/(2\beta+a)}\big)$, where the hidden constant depends on the parameters L, β, v_1, v_2, ρ, C_0, and a.*

Remark 3. *The term $\tilde{D}$ depends on both the smoothness parameter β and the margin parameter α_0, and is always upper bounded by the ambient dimension D. If additionally, P_X satisfies the* strong-density assumption*, i.e., it admits a density p_X w.r.t. the Lebesgue measure such that $p_X \geq c_0 > 0$ for all $x \in \mathcal{X}$, we have $\tilde{D} \leq \max\{0, D-\alpha_0\beta\}$ (see [19, App. H.1] for proof).*

Remark 4 (Comparison with Passive Algorithms)**.** *Theorem 1 implies that the worst case excess risk under **(MA)** and **(HÖ)** conditions achieved by Algorithm 1 is $\tilde{\mathcal{O}}\big(n^{-\beta(\alpha_0+1)/(2\beta+D)}\big)$. In Theorem 2, we will show that this rate cannot be improved by deriving matching (modulo poly-log factors) lower bound, thus establishing the minimax near-optimality of Algorithm 1. In the passive setting, under the same assumptions, the plug-in approach of [6] using the estimator of [17] achieves an excess risk bound of the order $\tilde{\mathcal{O}}\big(n^{-\beta(1+\alpha_0)/(D+2\beta+\alpha_0\beta)}\big)$. This rate can be shown to be minimax near-optimal by combining Lemma 2 with the lower-bound construction of [17]. Thus, due to the additional $\alpha_0\beta$ term, the minimax rate in the passive setting is always slower than that in the active setting, and furthermore, the gap in performance increases for smoother regression function (large β) and larger margin parameter (α_0).*

B. Lower Bound

We now derive minimax lower-bounds on the expected excess risk of the fixed-cost setting and the membership query model. The proof follows the general outline for obtaining lower-bounds described in works, such as [17], [20], reducing the estimation problem to an appropriate multiple hypothesis testing problem, and then applying Theorem 2.5 of [16]. The novel elements of our result are the construction of an appropriate class of *hard* regression functions and the comparison inequality presented in Lemma 2. The details of the construction as well the proofs are in [19, Appendix G].

Lemma 2. *In the fixed-cost of abstention setting with the cost $\lambda < 1/2$, let g represent any abstaining classifier and g_λ^* represent the Bayes optimal one. Then, we have $R_\lambda(g) - R_\lambda(g_\lambda^*) \geq cP_X\big((G_\lambda^* \backslash G_\lambda) \cup (G_\lambda \backslash G_\lambda^*)\big)^{(1+\alpha_0)/\alpha_0}$, where $c > 0$ is a constant and α_0 is the parameter of the assumption (MA).*

Lemma 2 aids our lower-bound proof in several ways: **1)** it motivates our construction of *hard* problem instances in which it is difficult to distinguish between the 'abstain' and 'not-abstain' options, **2)** it suggests a natural definition of pseudo-metric (see Thm. [16, Theorem 2.5]), and **3)** it allows us to convert the lower-bound on the hypothesis testing problem to that on the excess risk. We now state the main result of this section.

Theorem 2. *Let $\mathcal{A}$ be any active learning algorithm in the fixed-cost $\lambda < 1/2$ abstention setting and $\hat{g}_n$ be the abstaining classifier learned by $\mathcal{A}$ with n label queries. Let $\mathcal{P}(L, \beta, \alpha_0)$ represent the class of joint distributions P_{XY} satisfying the margin assumption (MA) with exponent $\alpha_0 > 0$, whose regression function is (L,β) Hölder continuous with $L \geq 3$ and $0 < \beta \leq 1$. Then, we have $\inf_{\mathcal{A}} \sup_{P_{XY}\in\mathcal{P}(L,\beta,\alpha_0)} \big(\mathbb{E}\left[R_\lambda(\hat{g}_n) - R_\lambda(g_\lambda^*)\right]\big) = \Omega\big(n^{-\beta(1+\alpha_0)/(2\beta+D)}\big)$.*

This result shows the minimax near-optimality of Algorithm 1, as its excess risk upper-bound matches the lower-bound up to poly-logarithmic factors in the worst case when $\tilde{D} = D$.

b \ n	100	500	1000	2000	3000
0.2	**0.0421**	**0.4812**	**0.0475**	**0.0469**	**0.0472**
0.5	0.0137	**0.0170**	**0.0188**	**0.0214**	**0.0227**
0.8	0.0010	0.0036	0.0073	0.0081	**0.0081**
1.0	-0.0260	0.0000	-0.0007	0.0077	0.0077

(a) With increasing b, the difference in performance between active and passive algorithms reduces.

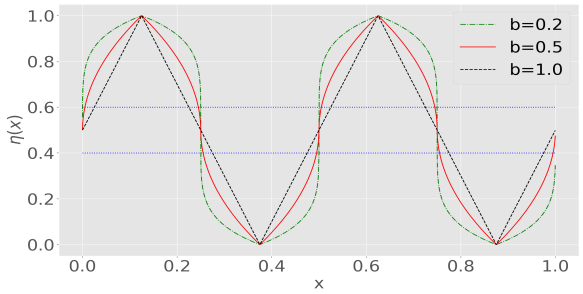
a \ n	100	500	1000	1500	2000
1.0	0.0134	0.0060	0.0084	**0.0103**	**0.0131**
0.8	0.0145	**0.0173**	**0.0171**	**0.0207**	**0.0216**

(b) Decreasing the amplitude a results in decreasing α_0, with β fixed. Thus, the performance gap between active and passive algorithms increases as a is lowered.

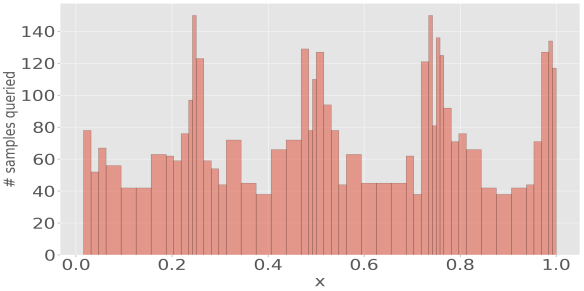
TABLE I: The tables show the difference in empirical risks of the passive and active algorithms, i.e., $(r_p - r_a)$. Bold indicates statistically significant.

Due to space constraints, we defer the proof of this result to Appendix **??** of [19].

IV. Numerical Illustration



(a) Plots of the regression function η used in the experiments for $a = 1.0$ and $b \in \{0.2, 0.5, 1.0\}$. The dotted blue lines represent the thresholds $\lambda = 0.4$ and $1 - \lambda = 0.6$.



(b) Histogram of the points sampled by Algorithm 1 for $n = 4000$, $a = 1.0$, and $b = 0.5$.

Fig. 1

We now verify the advantages of active (over passive) learning shown by our theoretical results on a class of toy problems. In these problems, we fix $\mathcal{X} = [0, 1]$, P_X as the uniform distribution, and the cost of abstention at $\lambda = 0.4$, and set $\eta(x) = 0.5\Big(1 + a\big(\sum_{k=0}^{3}(-1)^k(4x-k)^b\big)\Big)$, for $a, b \in [0, 1]$. The regression functions η are Hölder continuous with $L = 4^b$ and $\beta = b$. Moreover, with the above choice of η and P_X, the (MA) assumption holds with $\alpha_0 = 1/b$.

To provide a benchmark for comparison with Algorithm 1, we consider a simple passive classifier which implements a classification rule based on a piecewise constant estimator of η using a uniform partition of $\mathcal{X}$ with a bandwidth $bw = 0.1$. We ran the following two experiments with these algorithms: **1)** Change b for a fixed a, and **2)** Change a for a fixed b.

Expected Performance. When a is fixed and b is varied, the parameters β and α_0 both change such that $\alpha_0\beta = 1$. In this case, given the exponents of n in the excess risk of active (Theorem 1) and passive (Remark 4), we expect the gap between the performance of active and passive algorithms decreases with increasing b. When a is varied with b fixed, β is unchanged and α_0 decreases with a. Thus, again based on the result of Theorem 1, we expect the gap increases with a.

Observed Performance. For every combination of parameters a, b and n, we ran 50 repetitions of the active and passive algorithms and computed the empirical risk with 10000 test samples. We denote by r_a and r_p the average (over 50 runs) empirical risk of the active and passive algorithms. We tabulate the $r_p - r_a$ values for the two experiments in Tables 1 and 2, respectively. To test the statistical significance of the results, we use z-score test with 95% confidence [21, § 4.2].

We see in Table 1 that the performance difference between active and passive algorithms decreases with increasing b. In the case of $b = 1.0$, the first statistically significant difference was observed for $n > 3000$. Similarly in Table 2, we observe that the performance gap decreases with decreasing a (or decreasing α_0) as predicted by theory.

The key reason for the benefit of active learning scheme over passive, is that the active algorithm focuses sampling in the *difficult* regions of the input space near the decision boundaries, as shown in the histogram in Figure 1b. This effect becomes more pronounced when P_X puts small mass in these boundary regions, such as when b is small or a is large.

V. Conclusion

In this paper, we proposed and analyzed an active learning algorithm for the problem of binary classification with *fixed-cost* of abstention under three most commonly used active learning query models: *membership-query*, *pool-based*, and *stream-based*. We obtained upper-bound on the excess risk of our algorithm and demonstrated their minimax (near)-optimality by deriving lower-bound. We then presented some numerical results which verify the theoretical predictions.

In the full version of this manuscript [19], we also consider the following extensions: **(a)** an adaptive version of Algorithm 1 which does not require the knowledge of the smoothness parameters (L, β), and **(b)** We also consider another abstention setting, the *bounded-rate* setting, in which the learner is allowed to abstain for up to a fixed fraction $\delta \in (0, 1)$ of inputs without incurring any cost.

REFERENCES

[1] P. Rubegni, G. Cevenini, M. Burroni, R. Perotti, G. Dell'Eva, P. Sbano, C. Miracco, P. Luzi, P. Tosi, P. Barbini *et al.*, "Automated diagnosis of pigmented skin lesions," *International Journal of Cancer*, vol. 101, no. 6, pp. 576–580, 2002.

[2] S. Dasgupta, "Coarse sample complexity bounds for active learning," in *Advances in neural information processing systems*, 2006, pp. 235–242.

[3] R. M. Castro and R. D. Nowak, "Minimax bounds for active learning," *IEEE Transactions on Information Theory*, vol. 54, no. 5, pp. 2339–2353, 2008.

[4] C.-K. Chow, "An optimum character recognition system using decision functions," *IRE Transactions on Electronic Computers*, no. 4, pp. 247–254, 1957.

[5] C. Chow, "On optimum recognition error and reject tradeoff," *IEEE Transactions on information theory*, vol. 16, no. 1, pp. 41–46, 1970.

[6] R. Herbei and M. Wegkamp, "Classification with reject option," *Canadian Journal of Statistics*, vol. 34, no. 4, pp. 709–721, 2006.

[7] P. Bartlett, M. Jordan, and J. McAuliffe, "Convexity, classification, and risk bounds," *Journal of the American Statistical Association*, vol. 101, no. 473, pp. 138–156, 2006.

[8] M. Yuan, M.and Wegkamp, "Classification methods with reject option based on convex risk minimization," *Journal of Machine Learning Research*, vol. 11, pp. 111–130, 2010.

[9] M. Wegkamp, "Lasso type classifiers with a reject option," *Electronic Journal of Statistics*, vol. 1, pp. 155–168, 2007.

[10] M. Wegkamp and M. Yuan, "Support vector machines with a reject option," *Bernoulli*, vol. 17, no. 4, pp. 1368–1385, 2011.

[11] C. Cortes, G. DeSalvo, and M. Mohri, "Learning with rejection," in *International Conference on Algorithmic Learning Theory*, 2016, pp. 67–82.

[12] B. Settles, "Active learning literature survey," University of Wisconsin-Madison Department of Computer Sciences, Tech. Rep., 2009.

[13] S. Bubeck, R. Munos, G. Stoltz, and C. Szepesvári, "X-armed bandits," *Journal of Machine Learning Research*, vol. 12, no. May, pp. 1655–1695, 2011.

[14] R. Munos *et al.*, "From bandits to Monte-Carlo Tree Search: The optimistic principle applied to optimization and planning," *Foundations and Trends® in Machine Learning*, vol. 7, no. 1, pp. 1–129, 2014.

[15] R. Kleinberg, A. Slivkins, and E. Upfal, "Bandits and experts in metric spaces," *arXiv preprint arXiv:1312.1277*, 2013.

[16] A. B. Tsybakov, *Introduction to Nonparametric Estimation*, 1st ed. Springer Publishing Company, Incorporated, 2009.

[17] J.-Y. Audibert and A. Tsybakov, "Fast learning rates for plug-in classifiers," *The Annals of statistics*, vol. 35, no. 2, pp. 608–633, 2007.

[18] P. Bartlett and M. Wegkamp, "Classification with a reject option using a hinge loss," *Journal of Machine Learning Research*, vol. 9, pp. 1823–1840, 2008.

[19] S. Shekhar, M. Ghavamzadeh, and T. Javidi, "Active Learning for Binary Classification with Abstention," *arXiv preprint arXiv:1906.00303*, 2019.

[20] S. Minsker, "Plug-in approach to active learning," *Journal of Machine Learning Research*, vol. 13, no. Jan, pp. 67–90, 2012.

[21] S. Raschka, "Model evaluation, model selection, and algorithm selection in machine learning," *arXiv preprint arXiv:1811.12808*, 2018.