A Robust Zero-Sum Game Framework for Pool-based Active Learning

Dixian Zhu University of Iowa
Zhe Li Apple Inc.
Xiaoyu Wang Intellifusion
Boqing Gong Tencent AI Lab
Tianbao Yang University of Iowa

Abstract

In this paper, we present a novel robust zero-sum game framework for pool-based active learning grounded on advanced statistical learning theory. Pool-based active learning usually consists of two components, namely, learning of a classifier given labeled data and querying of unlabeled data for labeling. Most previous studies on active learning consider these as two separate tasks and propose various heuristics for selecting important unlabeled data for labeling, which may render the selection of unlabeled examples sub-optimal for minimizing the classification error. In contrast, the presented work formulates active learning as a unified optimization framework for learning the classifier, i.e., the querying of labels and the learning of models are unified to minimize a common objective for statistical learning. In addition, the proposed method avoids the issues of many previous algorithms such as inefficiency, sampling bias, and sensitivity to imbalanced data distribution. Besides theoretical analysis, we conduct extensive experiments on benchmark datasets and demonstrate the superior performance of the proposed active learning method over the state-of-the-art methods.

1 Introduction

A classical learning paradigm is assuming that we are given a set of labeled training examples, which is referred to as passive learning. However, in many applications such as natural language processing [1], medical image classification [2], biomedicine and bioinformatics [3], the labeled data are expensive to obtain. Instead, there could be a plenty of unlabeled data available. Active learning addresses the challenge by only querying the labels of a small number of unlabeled data for learning a prediction model. Active learning strategy can be also extended to linear regression [4], outlier detection [5], ensembles [6, 7]. This paper focuses on pool-based active learning (PAL), where a set of unlabeled data is given beforehand.

Although there are extensive studies on PAL, almost all previous methods consist of two alternating steps: (i) training a prediction model based on labeled data; and (ii) selecting some unlabeled data for querying their labels. These two steps are usually alternated for a number of times until either the budget of querying unlabeled data is used up or the performance becomes satisfactory. A common approach for training a prediction model is based on empirical risk minimization over the labeled data. There are various approaches proposed for selecting unlabeled data for querying their labels. These approaches are different in the criterion in terms of what are the optima set of unlabeled data for querying their labels. To this end, many heuristic approaches are proposed, which will be reviewed in next section.

While research about PAL in the traditional route has entered into a bottleneck, in this paper we propose a new framework for PAL, which not only brings a new perspective regarding the two steps in traditional active learning but also improves the performance of start-of-the-art PAL methods. The proposed framework is based on a zero-sum game for training the model and determining the selection probabilities of unlabeled data. In particular, updating the model and selection probabilities can be considered as two players in a zero-sum game, where the player for updating the model aims to minimize a weighted loss over individual data and the player for updating the selection probabilities aims to select the worst weights in a constrained domain to maximize the weighted loss. Different from conventional active learning methods, in the proposed framework the two steps of training prediction models and selecting unlabeled data are unified in a single framework, aiming to minimize a robust

Proceedings of the 22[nd] International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89.

risk for statistical learning.

The proposed robust zero-sum game framework can be considered as a novel extension to the active learning setting of an advanced passive learning approach based on a distributationally robust optimization [8], which is equivalent to using a variance-based regularization and achieves an effect of minimizing both the bias and the variance of the prediction. In addition, our contributions include how to handle the unlabeled data for updating the selection probabilities in order to minimize the robust risk and the convergence analysis of the proposed method. Moreover, we conduct extensive experiments on benchmark datasets to demonstrate the effectiveness of the proposed active learning method.

2 Related Work

Before discussing some related work, we would like to point out that there are a large volume of studies related to PAL and our review cannot be exhausted. we will focus on the core ideas in existing PAL studies and some representative work.

As mentioned before, most PAL methods alternate between two steps, i.e., training a classification model based on labeled (or predicted labeled) data and selecting unlabeled data for querying their labels. The training of a prediction model is usually accomplished by minimizing a certain convex surrogate loss averaged over the labeled (or predicted labeled) data, which is known as empirical risk minimization. Few studies also considered using Bayesian learning to learn a probabilistic model for classification [10, 11].

The core component of all PAL methods is how to select unlabeled data for querying their labels. In general, existing methods can be organized into four categories, namely disagreement-based, margin-based, clustering-based, and optimization-based methods. The idea of disagreement-based methods is to maintain a set of candidate classifiers at each round and select unlabeled examples in the region of disagreement of the candidate classifiers [12] . Based on the queried labels, the algorithm updates the set of candidate classifiers that are sub-optimal and proceeds to the next round. The issue of disagreement-based methods is that they are computational inefficient due to maintaining a set of candidate classifiers and finding examples in the region of disagreement. In margin-based active learning methods [13, 14, 15, 16, 17], a single classifier is maintained in each round and a batch of unlabeled examples that are close to the decision boundary are selected for querying their labels. The issue of margin-based methods is the sampling bias introduced by margin-based query strategy, i.e., the training set quickly diverges from the underlying data distribution [18]. To address this issue, clustering-based approaches and optimization-based approaches are developed.

In clustering-based approaches, the selection of unlabeled data takes the cluster structure of the data into account. Many different algorithms have been proposed in this category with difference lying at how to utilize the cluster structure [18, 19]. Nevertheless, the clustering-based approaches heavily depend on the metric used for clustering. Optimization-based approaches formulate the selection of a subset of unlabeled data as an optimization problem. Many criteria have been proposed for formulating the optimization problem [20, 1, 2]. For example, Hoi et al. [1] formulated the problem by minimizing the ratio between two Fisher information matrices with one computed from all unlabeled data and the other one computed from the selected unlabeled data, which is motivated by that the Fisher information matrix represents the overall uncertainty of a classification model. Wang & Ye [20] motivated the formulation by minimizing the upper bound of true risk. Using standard learning theory of empirical risk minimization, they derived an additional term in the upper bound for active learning, which accounts for the difference between the true data distribution and sampled data distribution and is approximated by a maximum mean discrepancy term computed from all unlabeled data and the selected subset. However, the issue of most optimization-based approaches is that the resulting optimization problem is usually difficult to solve. Previous studies usually seek approximation methods to solve the resulting optimization problems, which could still have polynomial time complexity in the number of unlabeled examples.

In contrast, the proposed active learning framework elegantly avoid these issues mentioned above. First, it is based on a robust optimization formulation, which can be efficiently solved by popular (stochastic) gradient-based methods. As a result, the proposed algorithm has at most linear time complexity and could even enjoy logarithmic time complexity using advanced data structures for selecting unlabeled data. Second, the selection probabilities for all unlabeled data are updated in a systematic way for optimizing the robust risk and their updating rule takes all data including labeled and unlabeled data into account, thus avoiding the issue of sampling bias. In addition, the proposed method is robust to imbalanced data distributions due to minimizing the robust risk. Last but not least, it is grounded on an advanced learning theory instead of classical learning theory of empirical risk minimization, which achieves the best bias and variance tradeoff. We will revisit each of these features in next section.

3 A Robust Game Framework for PAL

In this section, we present a robust zero-sum game framework for PAL. In subsection 3.1, we present some preliminaries and motivations. In subsection 3.2, we present details of the proposed algorithm, and in subsection 3.3 we present theoretical justification and convergence analysis of the proposed algorithm. Practical implementations and extensions are considered in subsection 3.4.

3.1 Preliminaries and The Basic Idea

Without loss of generality, let us assume that initially we are given a pool of unlabeled examples $\mathcal{U} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{X}\}$ where $\mathcal{X} \subseteq \mathbb{R}^d$, and the labeled pool $\mathcal{L} = \emptyset$. Later on, we discuss how to handle that a set of initial labeled examples are provided. Below, we use capital letters (e.g., Y) to denote a random variable, and small letters (e.g., y_i) to denote an observed variable. Denote by $\mathbb{1}$ a vector of all 1s, by $\mathbf{p} = (p_1, \ldots, p_n)^\top$ a set of probabilities such that $\sum_{i=1}^n p_i = 1$ and $\mathbf{p} \geq 0$. Let $D(\mathbf{p}, \mathbf{q})$ denote a distance metric between two vectors. Two metrics will be discussed in this work, the KL divergence $D(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^n p_i \log \frac{p_i}{q_i}$ and the squared Euclidean distance $D(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^n (p_i - q_i)^2$.

Let $Y_i \in \mathcal{Y}$ denote a (random) label of example $\mathbf{x}_i$. We assume each pair $(\mathbf{x}_i, Y_i)$ follows an unknown distribution $\mathcal{P}_{X,Y} = P(Y|X)\mathcal{P}_X$ over $\mathcal{X} \times \mathcal{Y}$ with the conditional distribution denoted by $P(Y|X)$. We consider the labeling process of $\mathbf{x}_i$ as a sampling $y_i \sim P(Y|\mathbf{x}_i)$. The heart of machine learning methods for learning a prediction model is to minimize the following true risk:

$$\min_{\mathbf{w} \in \Omega} \mathrm{E}_{X,Y}[\ell(\mathbf{w}; X, Y)], \tag{1}$$

where $\ell(\cdot;\cdot)$ is termed the loss function and $\mathbf{w}$ denotes the hypothesis or the model parameter. If we denote by $f(\mathbf{w}; \mathbf{x})$ the prediction score of the model on $\mathbf{x}$, the loss function is usually written as $\ell(\mathbf{w}; x, y) = \ell(f(\mathbf{w}; \mathbf{x}), \mathbf{y})$. For example, if the problem is a binary classification problem $\mathcal{Y} = \{1, -1\}$, commonly used loss functions include hinge loss $\ell(\mathbf{w}; \mathbf{x}, y) = \max(0, 1 - yf(\mathbf{w}; \mathbf{x}))$ and logistic loss $\ell(\mathbf{w}; \mathbf{x}, y) = \log(1 + \exp(-yf(\mathbf{w}; \mathbf{x})))$. The prediction score can be computed as $f(\mathbf{w}; \mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ by a linear model and can be also computed by a deep neural network such that $f(\mathbf{w}; \mathbf{x})$ is the output of the last layer before computing the loss value, where $\mathbf{w}$ denotes the weights in the network.

The empirical risk minimization (ERM) in the passive learning where the labels of training data are provided $\{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$, is motivated by the standard learning theory, i.e., the uniform convergence that bounds the true risk by the empirical risk for any $\mathbf{w} \in \Omega$ [21]:

$$\mathrm{E}_{X,Y}[\ell(\mathbf{w}; X, Y)] \leq \frac{1}{n} \sum_{i=1}^n \ell(\mathbf{w}; \mathbf{x}_i, y_i) + c\frac{\mathcal{C}(\Omega)}{\sqrt{n}}, \tag{2}$$

where c depends on some parameters of the problem and desired confidence socre and $\mathcal{C}(\Omega)$ is some complexity measure of the hypothesis class. Hence, ERM is to solve the following problem:

$$\min_{\mathbf{w} \in \Omega} \frac{1}{n} \sum_{i=1}^n \ell(\mathbf{w}; \mathbf{x}_i, y_i) \tag{3}$$

However, the above formulation has two issues: (i) in active learning paradigm, the labels $\{y_1, \ldots, y_n\}$ are not available beforehand, (ii) the standard learning theory (2) neglects the data-dependence nature of the variance of prediction, which is simply bounded by a constant in deriving (2).

To address these two issues, we propose to solve the following robust optimization problem:

$$\min_{\mathbf{w} \in \Omega} \max_{\mathbf{p} \in \Delta_n} \sum_{i=1}^n p_i \mathrm{E}_{Y|\mathbf{x}_i}[\ell(\mathbf{w}; \mathbf{x}_i, Y)], \tag{4}$$

where $\mathbf{p} = (p_1, \ldots, p_n)^\top$ is a probability vector and $\Delta_n = \{\mathbf{p} \in \mathbb{R}^n : \mathbf{p} \geq 0, \sum_{i=1}^n p_i = 1, D(\mathbf{p}, \mathbb{1}/n) \leq \frac{\rho}{n}\}$ is a constrained domain with $\rho \geq 0$.

The robustness of the framework is due to the maximization over $\mathbf{p} \in \Delta_n$ and can be understood from two viewpoints. First, from the robust optimization perspective, the uniform weights $\mathbb{1}/n$ used in ERM (3) to weigh each individual data is not necessarily optimal for minimizing the true risk (see theoretical justification in next subsection). Hence, using the principle of robust optimization [22], we aim to find $\mathbf{w}$ that can minimize the worst case of weights $\mathbf{p}$ in Δ_n that may include the optimal weights. Second, by controlling the distance between $\mathbf{p}$ and $\mathbb{1}/n$, we can recover two existing learning paradigms, i.e., minimizing an average loss and minimizing the maximal loss. It is easy to see that when $\rho = 0$, the robust optimization problem (4) reduces to an average loss minimization problem, and when $\rho \to \infty$, the robust optimization problem (4) reduces to the maximal loss minimization problem. It has been shown by previous studies that (i) minimizing the average loss is sensitive to imbalanced data distributions (e.g. most data are from the negative class) but is more robust to outliers than minimizing the maximal loss [23]; and (ii) minimizing the maximal loss is sensitive to outliers but is more robust to imbalanced data distributions than minimizing the average loss [24]. Therefore, by controlling the distance between $\mathbf{p}$ and uniform probabilities $\mathbb{1}/n$, the learned model can be robust to both outliers and the imbalanced data distributions.

Algorithm 1 A Robust Zero-Sum Framework for PAL

1: **Input**: $\mathbf{w}_1 \in \Omega$, initial stepsize: η_0, α_0
2: Initialize: $\mathbf{p}_1 = (1/n, \ldots, 1/n) \in \mathbb{R}^n$, $\mathcal{U}^1 = \mathcal{U}$ and $\mathcal{L}^1 = \mathcal{L}$
3: **for** $t = 1, \ldots, T$ **do**
4: Sample $i_t \in [n] \sim \mathbf{p}_t$
5: **if** $\mathbf{x}_{i_t} \in \mathcal{U}^t$ **then**
6: Query for the label y_{i_t} of $\mathbf{x}_{i_t}$, and update $\mathcal{L}^{t+1} = \mathcal{L}^t \cup \{(\mathbf{x}_{i_t}, y_{i_t}\}$, $\mathcal{U}^{t+1} = \mathcal{U}^t \setminus \{\mathbf{x}_{i_t}\}$
7: **end if**
8: Update $\mathbf{w}_{t+1} = \Pi_\Omega[\mathbf{w}_t - \eta_t \nabla \ell(\mathbf{w}_t; \mathbf{x}_{i_t}, y_{i_t})]$
9: Compute $\hat{\mathbf{v}}_t = (\hat{v}_1^t, \ldots, \hat{v}_n^t) \in \mathbb{R}^n$ by (5)
10: Compute $\mathbf{q}_t = \mathbf{p}_t \circ \exp(\alpha_t \hat{\mathbf{v}}_t)$
11: Update $\mathbf{p}_{t+1} = \arg\min_{\mathbf{p} \in \Delta_n} D(\mathbf{p}, \mathbf{q}_t)$
12: **end for**
13: **Output**: $\widehat{\mathbf{w}}_T = \sum_{t=1}^T \mathbf{w}_t / T$

Before ending this subsection, we mention that if an initial set of labeled data $\{\mathbf{x}_i, y_i, i = 1, \ldots n_l\}$ is provided, we can replace the i-th component $\mathrm{E}_{Y|\mathbf{x}_i}[\ell(\mathbf{w}; \mathbf{x}_i, Y)]$ by $\ell(\mathbf{w}; \mathbf{x}_i, y_i)$.

3.2 The Algorithmic Framework

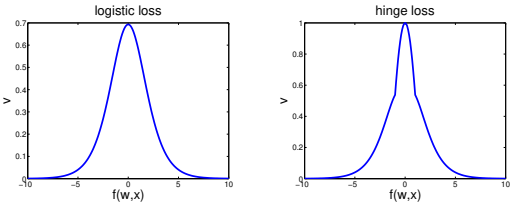
Next, we will present an algorithmic framework for minimizing the robust objective in (4), which is a unified algorithm for learning the model parameters $\mathbf{w}$ and selecting the unlabeled data for querying its label. Here, we focus on the binary classification problem where $\mathcal{Y} = \{-1, 1\}$ and the extension to multi-class classification problem is presented in subsection 3.4. The basic steps of the proposed algorithm are shown in Algorithm 1. Please note that, Algorithm 1 does not impose a budget on the number of queries. In subsection 3.4, we will discuss how to handle the case with a budget on the number of queries for unlabeled data.

At each iteration, the algorithm selects one data point from the set of all examples $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ according to the the sampling probabilities given by the current probability vector $\mathbf{p}_t$. Let $i_t \in \{1, \ldots, n\} \sim \mathbf{p}_t$ denote the index of the sampled example at the t-th iteration. If $\mathbf{x}_{i_t} \in \mathcal{U}^t$, we query for the label $y_{i_t} \sim P(Y|\mathbf{x}_{i_t})$ (the labeling step) and then add $\mathbf{x}_{i_t}$ into $\mathcal{L}^t$ and delete it from $\mathcal{U}^t$. If $\mathbf{x}_{i_t} \in \mathcal{L}^t$, the labeling step is skipped. Then the algorithm proceeds to update $\mathbf{w}$ and $\mathbf{p}$.

In Step 8, $\eta_t \geq 0$ is a step size (see the convergence analysis in the next subsection), and $\Pi_\Omega[\cdot]$ is a projection operator defined as

$$\Pi_\Omega[\mathbf{u}] = \arg\min_{\mathbf{w} \in \Omega} \frac{1}{2} \|\mathbf{w} - \mathbf{u}\|_2^2,$$

and the update for $\mathbf{w}$ can be explained as a stochastic gradient update for the objective in (4). In particular,



(a) logistic loss (b) hinge loss

Figure 1: the value of $\hat{v}_i$ vs $f(\mathbf{w}; \mathbf{x}_i)$

it is easy to show that

$$\begin{aligned}\mathrm{E}[\nabla \ell(\mathbf{w}_t; \mathbf{x}_{i_t}, y_{i_t})] &= \mathrm{E}_{i_t} \mathrm{E}_{y_{i_t}|\mathbf{x}_{i_t}}[\nabla \ell(\mathbf{w}_t; \mathbf{x}_{i_t}, y_{i_t})] \\ &= \sum_{i=1}^n p_i \mathrm{E}_{Y|\mathbf{x}_i}[\nabla \ell(\mathbf{w}_t; \mathbf{x}_i, Y)]\end{aligned}$$

The update of $\mathbf{p}_t$ in Step 10 & Step 11 can be understood as a mirror descent update using a Bregman divergence $D(\mathbf{p}, \mathbf{q})$ [25], where $\alpha_t \geq 0$ is a step size (see convergence analysis in the supplement). The detailed computation of $\mathbf{p}_{t+1}$ depends on the choice of Bregman divergence and is postponed to subsection 3.4. The motivation of Step 9 is to compute a gradient of the robust objective in terms of $\mathbf{p}_t$ given $\mathbf{w}_t$. In particular, the gradient of p_i^t is given as

$$\begin{aligned}v_i^t = \mathrm{E}_{Y|\mathbf{x}_i}[\ell(\mathbf{w}_t; \mathbf{x}_i, Y)] &= \ell(\mathbf{w}_t; \mathbf{x}_i, 1) \Pr(Y = 1|\mathbf{x}_i) \\ &+ \ell(\mathbf{w}_t; \mathbf{x}_i, -1) \Pr(Y = -1|\mathbf{x}_i)\end{aligned}$$

For labeled examples $\mathbf{x}_i \in \mathcal{L}^{t+1}$, we can calculate an unbiased estimate by $\hat{v}_i^t = \ell(\mathbf{w}_t; \mathbf{x}_i, y_i)$, and for unlabeled examples $\mathbf{x}_i \in \mathcal{U}^{t+1}$, we estimate $\mathrm{E}_{y|\mathbf{x}_i}[\ell(\mathbf{w}_t^\top \mathbf{x}_i, y)]$ by using an estimation of $P(Y|\mathbf{x}_i)$ by $\widehat{P}_t(Y|\mathbf{x}_i)$, i.e.,

$$\hat{v}_i^t = \begin{cases} \ell(\mathbf{w}_t; \mathbf{x}_i, y_i) & i \in \mathcal{L}^{t+1} \\ \ell(\mathbf{w}_t; \mathbf{x}_i, 1)\widehat{P}_t(Y = 1|\mathbf{x}_i) + & i \in \mathcal{U}^{t+1} \\ \ell(\mathbf{w}_t; \mathbf{x}_i, -1)\widehat{P}_t(Y = -1|\mathbf{x}_i) & \end{cases} \quad (5)$$

Estimation of $P(Y|\mathbf{x}_i)$ for linear and deep models We estimate the true condition distribution by a sigmoid function using the prediction score of the current model on each example $\mathbf{x}_i$, i.e.,

$$\widehat{P}_t(Y|\mathbf{x}_i) = \frac{1}{1 + \exp(-Y f(\mathbf{w}_t; \mathbf{x}_i))}, \quad (6)$$

where $f(\mathbf{w}_t; \mathbf{x}_i)$ is a real-valued prediction score of the model $\mathbf{w}_t$ on the data $\mathbf{x}_i$ such that the larger the value of $f(\mathbf{w}; \mathbf{x})$, the higher probability for the data belonging to the positive class.

More Understanding of the Selection of Unlabeled Data Before ending this subsection, we present more understanding of the selection of unlabeled data, which sheds more insights on the updating of $\mathbf{p}_{t+1}$. We will use the KL divergence $D(\mathbf{p}, \mathbf{q}) = \sum_i p_i \log \frac{p_i}{q_i}$ as an example. However, the discussion

can be extended to the Euclidean distance. The lemma below shows that p_i^{t+1} increases as $\hat{v}_i^t$ increases.

Lemma 1. *If KL divergence is used in Algorithm 1, then there exists $\gamma > 0$ such that $p_i^{t+1} \propto (q_i^t)^\gamma, \forall i \in \{1, \ldots, n\}$, where $\mathbf{q}_t = (q_1^t, \ldots, q_n^t) = \mathbf{p}_t \circ \exp(\alpha_t \hat{\mathbf{v}}_t)$.*

The proof of the above lemma can be found in [26] (please refer to their Appendix C.2). For two commonly used loss functions for binary classification, namely hinge loss $\ell(\mathbf{w}; \mathbf{x}, y) = \max(0, 1 - yf(\mathbf{w}; \mathbf{x}))$ and $\ell(\mathbf{w}; \mathbf{x}, y) = \log(1 + \exp(-yf(\mathbf{w}; \mathbf{x})))$, we can also show that v_i^t is a monotonically decreasing function of $|f(\mathbf{w}; \mathbf{x})|$ (please see Figure 1). Therefore, Lemma 1 implies that the smaller the margin $|f(\mathbf{w}; \mathbf{x})|$, the higher probability value of p_i^{t+1}, i.e., the higher chance for the i-th example to be selected in next round. In other word, among unlabeled examples those with smaller margins will have higher probabilities to be sampled for querying their labels. This effect is the same as margin-based PAL methods [13, 14, 15, 16, 17]. However, the proposed algorithm does not suffer from the issue of sampling bias as it is proposed for minimizing the robust objective in (4) with sampling performed on all data points $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$.

3.3 Theoretical Analysis

In this subsection, we provide a theoretical justification of the distributionally robust optimization (4) and a convergence analysis of the proposed algorithm. For simplicity of presentation, we define $\bar{\ell}(\mathbf{w}; X) = \mathrm{E}_{Y|X}[\ell(\mathbf{w}; |X, Y)]$ and $\bar{\ell}(\mathbf{w}) = (\bar{\ell}(\mathbf{w}; \mathbf{x}_1), \ldots, \bar{\ell}(\mathbf{w}; \mathbf{x}_n))^\top$.

The distributionally robust optimization (4) is motivated by a refined upper bound of the true risk (e.g., by Bennett's inequality) [27],

$$\mathrm{E}_X[\bar{\ell}(\mathbf{w}; X)] \leq \frac{\sum_{i=1}^n \bar{\ell}(\mathbf{w}; \mathbf{x}_i)}{n} + c_1 \sqrt{\frac{\mathrm{Var}_X(\bar{\ell}(\mathbf{w}; X))}{n}} + \frac{c_2}{n}, \quad (7)$$

where $\mathbf{w} \in \Omega$, c_1, c_2 depend on some parameters of the problem and desired confidence score, and Var_X denotes the variance over random variable X. The above upper bound includes both the bias (the first term) and the variance (the second term) of the prediction, which are two key components in the testing error. Therefore, a good model should optimize the above upper bound that balances between bias and variance. However, the above upper bound is a non-convex function of $\mathbf{w}$ and depends on unknown distribution involved in $\mathrm{Var}_X(\bar{\ell}(\mathbf{w}; X))$, which make it difficulty to optimize. To this end, a distributationaly robust optimization problem was proposed in [8]. Below, we extend the results in [8] for passive learning to our active learning setting to justify the proposed algorithm.

Theorem 1. *Assume that $\bar{\ell}(\mathbf{w}; X) \in [0, M]$ for $\mathbf{w} \in \Omega$, and $D(p, q)$ is the squared Euclidean distance. If $n \geq \max(5, \frac{M^2}{\sigma^2} \max(8\sigma, 44))$, with probability $1 - \exp(n\sigma^2/11M^2)$, we have*

$$\max_{\mathbf{p} \in \Delta_n} \sum_{i=1}^n p_i \bar{\ell}(\mathbf{w}; \mathbf{x}_i) = \frac{1}{n} \sum_{i=1}^n \bar{\ell}(\mathbf{w}; \mathbf{x}_i) + \sqrt{\frac{\rho}{n} Var_n(\bar{\ell}(\mathbf{w}; X))},$$

where $\sigma = Var_X(\bar{\ell}(\mathbf{w}; X))$, and $Var_n(\bar{\ell}(\mathbf{w}; X))$ denotes the empirical variance of $\bar{\ell}(\mathbf{w}; X)$ computed based on the given data. If additionally $n \geq \rho/2 \geq 9 \log 12$ and $\widehat{\mathbf{w}}$ denotes the optimal solution to (4), then

$$\mathrm{E}_X[\bar{\ell}(\widehat{\mathbf{w}}; X)] \leq \frac{6M\rho}{n} + \min_{\mathbf{w} \in \Omega} \left\{ \mathrm{E}_X[\bar{\ell}(\mathbf{w}; X)] + 2\sqrt{\frac{\rho}{n} Var_X(\bar{\ell}(\mathbf{w}; X))} \right\}$$

Remark: The first equality in the above theorem shows that the robust risk $\max_{\mathbf{p} \in \Delta_n} \sum_{i=1}^n p_i \bar{\ell}(\mathbf{w}; \mathbf{x}_i)$ is a good approximation of the upper bound of the true risk in (7) when the number of examples n is large enough, which justifies the distributationaly robust optimization problem (4) for active learning. The second inequality implies that if the variance of the optimal solution $\mathbf{w}_*$ for minimizing the true risk (1) is small (e.g., $\mathrm{Var}_X(\bar{\ell}(\mathbf{w}_*; X)) \leq O(1/n)$), then the excess risk (or called the statistical error) of the optimal solution to (4) is in the order of $O(1/n)$, i.e.,

$$\mathrm{E}_X[\bar{\ell}(\widehat{\mathbf{w}}; X)] - \mathrm{E}_X[\bar{\ell}(\mathbf{w}_*; X)] \leq O(1/n), \quad (8)$$

which is smaller than the standard statistical error of $O(1/\sqrt{n})$ of ERM [21].

Next, we present the convergence analysis of Algorithm 1 for minimizing the robust risk. The purpose of our convergence analysis is to demonstrate that in some restrictive settings, Algorithm 1 can converge to the optimal solution to (4) up to a statistical error, lending further justification to the updates in Algorithm 1. In particular, we consider linear model $f(\mathbf{w}; \mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ and the loss function $\ell(\mathbf{w}; \mathbf{x}, y) = \ell(y\mathbf{w}^\top \mathbf{x})$ to be a convex function of $\mathbf{w}$. Let $\mathcal{R}(\mathbf{w}) = \max_{\mathbf{p} \in \Delta_n} \sum_{i=1}^n p_i \bar{\ell}(\mathbf{w}; \mathbf{x}_i)$ denote the robust risk, $\widehat{\mathbf{w}}$ denote the optimal solution to minimizing the robust risk (4) and $\mathbf{w}_*$ denote the optimal solution to minimizing the true risk (1). We make the following assumptions for our convergence analysis.

Assumption 2. *We assume that*

- *there exists $c > 0$ and $\beta \in (0, 1]$ such that $\|\widehat{\mathbf{w}} - \mathbf{w}_*\|_2 \leq c \left(\mathrm{E}_X[\bar{\ell}(\widehat{\mathbf{w}}; X)] - \mathrm{E}_X[\bar{\ell}(\mathbf{w}_*; X)]\right) \leq O(1/n^\beta)$;*
- *there exists $\hat{c} > 0$ such that $\|\mathbf{w} - \widehat{\mathbf{w}}\|_2 \leq \hat{c}(\mathcal{R}(\mathbf{w}) - \mathcal{R}(\widehat{\mathbf{w}}))$ for any $\mathbf{w} \in \Omega$;*
- *For any $\mathbf{w} \in \Omega$, there exist r, M, G such*

that $\|\mathbf{w}\|_2 \leq r$, $\ell(\mathbf{w}; \mathbf{x}_i, Y) \in [0, M]$, $\|\nabla \ell(\mathbf{w}; \mathbf{x}_i, y_i)\|_2 \leq G$, *and all examples are normalized such that* $\|\mathbf{x}_i\|_2 \leq R$, *and* $MR\hat{c} \leq 1/2$;

- *the conditional distribution is given by* $P(Y|\mathbf{x}) = \frac{1}{1+\exp(-Y\mathbf{w}_*^\top \mathbf{x})}$.

Remark: The first and second assumptions assume that the problem (1) and (4) satisfy certain regularity condition (namely weak sharp minimum condition [28]). It can be satisfied e.g., when $\ell(\mathbf{w}; \mathbf{x}, y)$ is a hinge loss and $\mathbf{w}^\top \mathbf{x} \leq 1$ [29, 30]. $O(1/n^\beta)$ denotes the statistical error of the robust risk minimizer $\widehat{\mathbf{w}}$. The third assumption is a standard assumption for convergence analysis. The last assumption is based on that $\mathbf{w}_*$ is the Bayes optimal solution (i.e., the loss function is Fisher consistent), which is true for many surrogate loss functions such as hinge loss and logistic loss [31]. Under the above assumptions, we prove the following convergence results.

Theorem 3. *Under Assumption 2, with* $\alpha_t = \alpha \propto \sqrt{\frac{1}{T}}, \eta_t = \eta \propto \sqrt{\frac{1}{T}}$ *we have:*

$$\mathrm{E}[\mathcal{R}(\mathbf{w}_\tau) - \mathcal{R}(\widehat{\mathbf{w}})] \leq O\left(\frac{1}{\sqrt{T}} + \frac{1}{n^\beta}\right)$$

where $\tau \in 1, \ldots, T$ *is uniformly sampled from* $\{1, \ldots, T\}$.

Remark: The above theorem shows that a random iterate $\mathbf{w}_\tau$ of Algorithm 1 converges to the optimal solution $\widehat{\mathbf{w}}$ (in expectation) up to a statistical error $O(1/n^\beta)$ when $T = O(n^{2\beta})$. Although the convergence is provided for a randomly selected solution $\mathbf{w}_\tau$, in practice the averaged solution $\widehat{\mathbf{w}}_T$ is more robust.

3.4 Implementation and Extension

In this subsection, we provide some implementation details and extensions.

Efficient Update of $\mathbf{p}_{t+1}$. [26] discussed the updates of $\mathbf{p}_{t+1}$ for different Bregaman divergences that defines the constrained domain Δ_n. In this work, to avoid additional overhead handling the constraint $D(\mathbf{p}, \mathbb{1}/n) \leq \rho/n$, we consider minimizing a regularized version of the robust risk, i.e.,

$$\max_{\mathbf{p} \geq 0, \sum_i p_i = 1} \mathbf{p}^\top \bar{\ell}(\mathbf{w}; \mathbf{x}_i) - \lambda D(\mathbf{p}, \mathbb{1}/n) \qquad (9)$$

To tackle this objective, the Step 11 of Algorithm 1 can be replaced by using a proximal mapping of the regularizer:

$$\mathbf{p}_{t+1} = \arg \min_{\mathbf{p} \geq 0, \sum_i p_i = 1} D(\mathbf{p}, \mathbf{q}_t) + \alpha_t \lambda D(\mathbf{p}, \mathbf{1}/n)$$

which can be computed in a closed form for both KL divergence and squared Euclidean divergence. We refer the readers to [26] for more details due to limit of space. Therefore the update of $\mathbf{p}_t$ can be performed in a time complexity of $O(n)$. The same convergence result as in Theorem 3 can be established.

Extension to multi-class classification problem. For multi-class classification problem with $\mathcal{Y} = \{1, \ldots, K\}$, the cross-entropy cost function $\ell(\mathbf{w}; \mathbf{x}, y) = -\sum_{k=1}^K \mathbb{I}(y = k) \log \frac{\exp(f_k(\mathbf{w}, \mathbf{x}_i))}{\sum_{l=1}^K \exp(f_l(\mathbf{w}; \mathbf{x}_i))}$ can be used, where $\mathbb{I}(\cdot)$ is an indication function. The estimation of $\hat{v}_i^t$ for unlabeled data can be computed by $\sum_{k=1}^K \ell(\mathbf{w}; \mathbf{x}_i, k) \hat{P}_t(Y = k|\mathbf{x}_i)$, where $\hat{P}_t(Y = k|\mathbf{x}_i)$ can be computed by softmax function using current model parameters.

An implementation with $\log(n)$ time complexity per iteration. Using advanced data structures and a stochastic update of $\mathbf{p}_{t+1}$ (i.e., using a stochastic gradient to update $\mathbf{p}$), each iteration can cost up to $O(\log(n))$ time complexity. We omit the details due to limit the space (please refer to [26]). However, the number of iterations could be increased due to the variance in the stochastic gradient of $\mathbf{p}_{t+1}$. In our experiments, we use the full version as outlined in Algorithm 1.

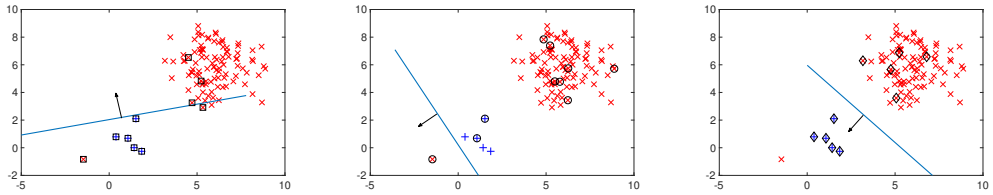
A Budget on the Number of Queries. If there is a budget on the number of queries, we can change Algorithm 1 in the following way. After the budget of querying the labels of unlabeled data is used up at some iteration t, we continue the training by (i) only sampling data from labeled pool with a probability vector $\hat{\mathbf{p}}_t \in \mathbb{R}^{n_l}$ (n_l is the size of labeled pool) that is normalization of components in $\mathbf{p}_t$ corresponding to the labeled data, and (ii) by updating the probability vector $\hat{\mathbf{p}}_t \in \mathbb{R}^{n_l}$ and the model parameter $\mathbf{w}_t$ using the labeled data at each iteration, until the process converges.

4 Experiments

In this section, we present some experimental results to justify the proposed method for PAL and also compare with existing PAL methods on learning both linear models and deep neural networks.

4.1 Synthetic Data

We first conduct experiments on a synthetic data to verify the robustness of the proposed PAL framework to imbalanced data and outliers as articulated in Section 3.1. To this end, we generate an imbalanced 2-dimension data set as shown in Figure 2, where there are 5 positive examples denoted by + and 100 negative examples denoted by ×. The label budget is 10 and iteration number is 10000. Examples of the two classes follow a Gaussian distribution. In order to create an outlier, we flip the label of the point at the lower left corner to the negative class. Then we run



(a) maximal RM, $\lambda = 0$ (b) average RM, $\lambda = \infty$ (c) robust RM $\lambda = 1$

Figure 2: '+' represent positive samples and '×' represent negative samples. There is an outlier '×' on the lower left corner. The blue lines are the learned decision boundary and the arrows point to the half-space that is classified as positive. The squares, circles and diamonds represent the data points selected by active learning algorithm for labeling. Maximal loss can be easily misled by noise or outlier; while average loss may neglect small amount of misclassified data. 'RM': risk minimization.

the proposed PAL with different values of the regularization parameter λ as in equation (9). In particular, we use three values, $\lambda = 0$ corresponding to maximal risk minimization, $\lambda = \infty$ corresponding to average risk minimization, and $\lambda = 1$ corresponding to a robust risk minimization. The step size $\alpha_t = 1/\sqrt{t}$ and $\eta_t = 0.1/\sqrt{t}$. The result is shown in Figure 2. From the result, we can see that (i) minimizing the maximal risk is sensitive to the outlier and the learned classifier is misled by the outlier; (ii) minimizing the average loss is robust to outlier but sensitive to the imbalanced data distribution, and the learned classifier predicts most examples as negative; (iii) minimizing a robust risk is robust to both the outlier and the imbalanced data distribution, yielding the best prediction result. This experiment clearly justifies the robustness of the proposed robust risk minimization for PAL.

4.2 Active Learning of Linear Models

In this subsection, we compare the proposed method with several existing PAL methods for active learning of linear models:

- RS: querying by random sampling. At each round it selects a batch of samples uniformly at random from the unlabeled data pool for labeling.
- MADI: margin-based querying by incorporating the diversity of selected examples [15]. This is a batch-mode margin-based PAL method improved by incorporating data diversity.
- RMADI: similar to the MADI method except that the classifier is learned by minimizing the robust risk [8] instead of by ERM. We compare with this method in order to verify that the proposed unified framework is better than this ad-hoc approach.
- MRFI: querying by minimizing the ration of Fisher Information matrices to preserve data distribution [1]. This is an optimization based batch-mode PAL method.
- BMDR: querying by balancing informative and representative samples [20]. This is also an optimization based batch-mode PAL method.

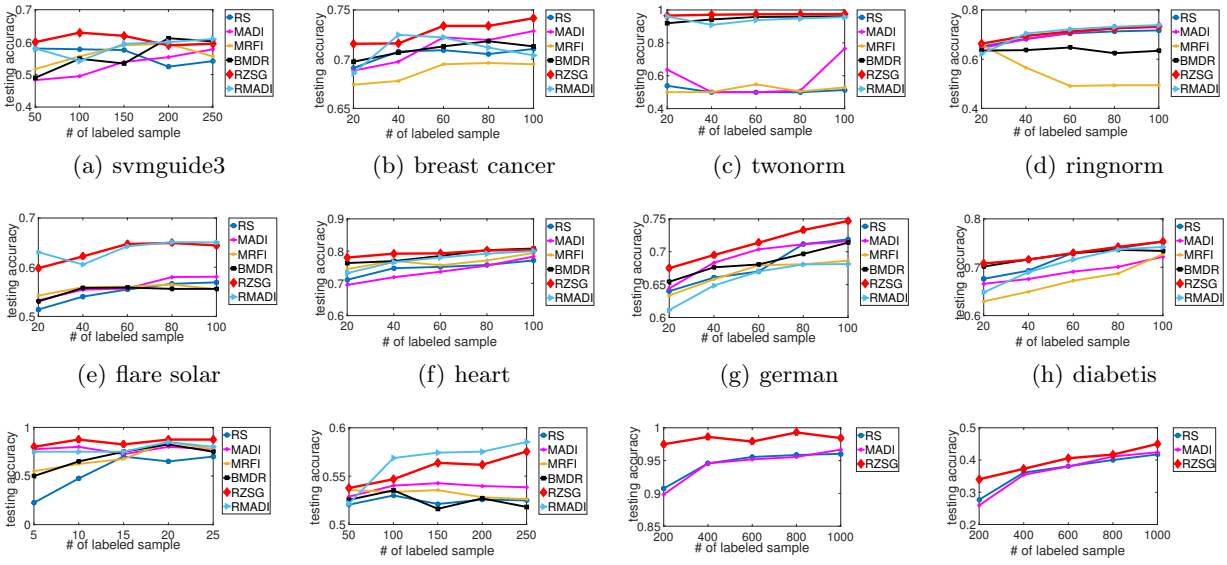
The proposed PAL method is referred to as **RZSG** - a robust zero-sum game framework for PAL.

We conduct experiments on 10 binary classification datasets from UCI Machine Learning Repository [33] and LIBSVM Data website [34], namely, madelon, svmguide3, breast cancer, twonorm, ringnorm, flare solar, heart, german, diabetis, and duke breast cancer. [1] The statistics of these datasets are summarized in the Supplement.

We note that all methods have regularization parameters for learning the classifier on the labeled data points. The values of the regularization parameters of each method are tuned by 5-fold cross validation. In particular, we split the training into 5 folds and do cross validation by using the corresponding passive learning algorithm for each PAL algorithm. The best value of the regularization parameter is selected based on the validation performance and used for running each PAL algorithm on the original training data. Finally, the learned classifier of each PAL algorithm is evaluated on the testing data for comparison. For batch-model PAL methods (RS, MADI, MRFI, BMDR, RMADI), we use various batch sizes for different data sets. In particular, for datasets with a training size < 100, the batch size is 5; with a training size in $(100, 1000]$, the batch size is 20; and with a training size in $(1000, 5000]$, the batch size is 50. For all PAL algorithms, we independently repeat the experiments 10 times and report the averaged testing accuracy.

For all baseline methods except for RMADI, we use the LIBSVM [34] library to learn a SVM classifier. The hinge loss as employed by SVM is also used in our method. There are also some other parameters for

[1] Some of UCI benchmark datasets have been preprocessed and can be downloaded at http://theoval.cmp.uea.ac.uk/matlab/.



(a) svmguide3 (b) breast cancer (c) twonorm (d) ringnorm

(e) flare solar (f) heart (g) german (h) diabetis

(i) duke breast cancer (j) madelon (k) MNIST (l) CIFAR10

Figure 3: Active learning of linear models (a)∼(j), and of deep neural networks (k,l).

each PAL method. They are set to their suggested values in the original papers. For example, MADI and RMADI have a trade-off parameter between quality and diversity, which is set to 0.5; BMDR has a tunable parameter β, which is set to 1000. Our method has two other parameters α_t and η_t for the step sizes, which are set based on the theoretical analysis (cf. the Supplement).

In each figure 3 (a∼j), the x-axis represents different budget values and the y-axis represents the testing accuracy. From the results, we can see the superiority of the proposed PAL method. In general, not only robust regularizer can improve active learning performance (RMADI is better than MADI), but also our unified active learning strategy can further boost performance.

4.3 Active Learning of Deep Neural Networks

Next, we present some experimental results for active learning of deep neural networks. To this end, we use two benchmark datasets, namely MNIST [35] and CIFAR-10 [36] (dataset statistics are in the supplement). The two baseline methods MRFI and BMDR are proposed for binary classification and they are not easily extended to multi-class classification. The baseline RS can be implemented without any change. We extend the baseline MADI to the multi-class classification by using the entropy of estimated class probabilities as the quality measure. The selection of top examples for MADI is based on a sampling method with sampling probabilities proportional to the prediction score, which we find to be more effective than the strategy in the original paper, i.e., simply selecting the examples based on the margin score. We use convolutional neural networks (CNN) as the prediction model with cross-entropy as the loss. The CNN for MNIST consists of two convolutional layers with 5×5 filters and ReLU activation function, two max pooling layers with 2×2 filters and stride of 2, and two fully connected layers with 1024 and 10 neurons; and that for CIFAR-10 is a simplified AlexNet [37]. For our method RZSG, the step size parameter is set to $\alpha_t = 10$, $\eta_t = 0.001$ for MNIST. For CIFAR10, they are set to $\alpha_t = 0.01$ and $\eta_t = 0.001$. Adam [38] is employed as the optimizer for CNN. The results are shown in Figure 3 (k, l), which again demonstrate the the proposed method is much better than the baselines.

5 Conclusions

We have proposed a novel robust zero-sum game framework for pool-based active learning. It is the first work that uses a unified framework for selecting unlabeled examples and for updating the models to minimize a robust risk. We have analyzed the proposed method from different perspectives and demonstrated that it is robust to imbalanced data distribution and outliers, avoids sampling bias, and is efficient. We also conduct extensive experiments to justify and verify the effectiveness of the proposed method, which clearly demonstrate its superior performance comparing with the state-of-the-art pool based active learning methods.

Acknowledgements

We thank the anonymous reviewers for their helpful comments. D. Zhu and T. Yang are partially supported by National Science Foundation (IIS-1545995).

References

[1] Hoi, Steven CH and Jin, Rong and Lyu, Michael R (2006). Large-scale text categorization by batch mode active learning *Proceedings of the 15th international conference on World Wide Web*, 633–642. Edinburgh, Scotland, Uk: ACM.

[2] Hoi Steven CH and Rong jin and Jianke Zhu and Michael R. Lyu (2006). Batch mode active learning and its application to medical image classification *Proceedings of the 23rd international conference on Machine learning*, 417–424. Pittsburgh, PA, USA: ACM.

[3] Warmuth, Manfred K and Rätsch, Gunnar and Mathieson, Michael and Liao, Jun and Lemmen, Christian (2002). Active learning in the drug discovery process *Advances in Neural information processing systems*, 1449–1456. Vancouver, British Columbia, Canada: MIT Press.

[4] Sugiyama, Masashi (2006). Active learning in approximately linear regression based on conditional expectation of generalization error *Journal of Machine Learning Research*, **7**(Jan):141–166.

[5] Abe, Naoki and Zadrozny, Bianca and Langford, John (2006). Outlier detection by active learning *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 504–509. Philadelphia, PA, USA: ACM.

[6] Abe, Naoki and Mamitsuka, Hiroshi (1998). Query learning strategies using boosting and bagging *Machine learning: proceedings of the fifteenth international conference (ICML'98)*, **7**. Morgan Kaufmann Pub.

[7] Melville, Prem and Mooney, Raymond J (2004). Diverse ensembles for active learning *Proceedings of the twenty-first international conference on Machine learning*, p74. Banff, Alberta, Canada: ACM.

[8] Hongseok Namkoong and John C. Duchi (2017). Variance-based Regularization with Convex Objectives *Advances in Neural Information Processing Systems 30 (NIPS)*, 2975–2984. Long Beach, CA, USA: ACM.

[9] Beygelzimer, Alina and Dasgupta, Sanjoy and Langford, John (2009). Importance Weighted Active Learning *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, 49–56. Montreal, Quebec, Canada: ACM.

[10] Nguyen Viet Cuong and Wee Sun Lee and Nan Ye (2014). Near-optimal Adaptive Pool-based Active Learning with General Loss *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence (UAI)*, 122–131. Quebec City, Quebec, Canada: AUAI Press.

[11] Nguyen, Viet Cuong and Lee, Wee Sun and Ye, Nan and Chai, Kian Ming Adam and Chieu, Hai Leong (2013). Active Learning for Probabilistic Hypotheses Using the Maximum Gibbs Error Criterion. *Advances in Neural Information Processing Systems 26 (NIPS)*, 1457-1465. Lake Tahoe, NV, USA: MIT Press.

[12] Balcan, Maria-Florina and Beygelzimer, Alina and Langford, John (2006). Agnostic Active Learning. *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, 65–72. Pittsburgh, PA, USA: ACM.

[13] Pranjal Awasthi and Maria-Florina Balcan and Philip M. Long (2017). The Power of Localization for Efficiently Learning Linear Separators with Noise. *J. ACM*, **63**(6):50:1–50:27.

[14] Tong, Simon and Koller, Daphne (2001). Support vector machine active learning with applications to text classification. *Journal of machine learning research*, **2**(Nov):45–66.

[15] Brinker Klaus (2003). Incorporating diversity in active learning with support vector machines. *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 59–66. Washington, DC, USA: AAAI Press.

[16] Zhang, Chicheng(2018). Efficient active learning of sparse halfspaces. *arXiv preprint arXiv*:1805.02350.

[17] Awasthi, Pranjal and Balcan, Maria-Florina and Haghtalab, Nika and Zhang, Hongyang (2016). Learning and 1-bit compressed sensing under asymmetric noise. *Conference on Learning Theory (COLT)*, 152–192. Columbia University, NY, USA: JMLR Workshop and Conference Proceedings.

[18] Dasgupta, Sanjoy and Hsu, Daniel (2008). Hierarchical Sampling for Active Learning. *Proceedings of the 25th International Conference on Machine Learning (ICML)*, 208–215. Helsinki, Finland: ACM.

[19] Ruth Urner and Sharon Wulff and Shai Ben-David (2013). PLAL: Cluster-based active learning. *The 26th Annual Conference on Learning Theory (COLT)*, 376–397. Princeton University, NJ, USA: JMLR Workshop and Conference Proceedings.

[20] Wang Zheng and Jieping Ye (2015). Querying discriminative and representative samples for batch mode active learning. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, **9**(3):17.

[21] Vapnik, Vladimir N. (1998). *Statistical Learning Theory.* Wiley-Interscience.

[22] Ben-Tal, A. and El Ghaoui, L. and Nemirovski, A.S. (2009). *Robust Optimization.* Princeton Series in Applied Mathematics: Princeton University Press.

[23] Yanbo Fan and Siwei Lyu and Yiming Ying and Bao-Gang Hu (2017). Learning with Average Top-k Loss. *Advances in Neural Information Processing Systems 30 (NIPS)*, 497–505. Long Beach, CA, USA: MIT Press.

[24] Shalev-Shwartz, Shai, and Yonatan Wexler. (2016). Minimizing the Maximal Loss: How and Why. *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, 793–801. New York City, NY, USA: ACM.

[25] Nemirovski, Arkadi and Juditsky, Anatoli and Lan, Guanghui and Shapiro, Alexander (2009). Robust Stochastic Approximation Approach to Stochastic Programming. *SIAM Journal on Optimization*, **19**(36):1574–1609.

[26] Hongseok Namkoong and John C. Duchi (2016). Stochastic Gradient Methods for Distributionally Robust Optimization with f-divergences. *Advances in Neural Information Processing Systems 29 (NIPS)*, 2208–2216. Barcelona, Spain: MIT Press.

[27] Maurer, Andreas and Pontil, Massimiliano (2009). Empirical Bernstein Bounds and Sample-Variance Penalization. *Proceedings of the 22nd Annual Conference on Learning Theory (COLT).* Princeton University, NJ, USA: JMLR Workshop and Conference Proceedings.

[28] Burke, James V., and Sien Deng (2005). "Weak sharp minima revisited, part II: application to linear regularity and error bounds. *Math. Program.*, **104**(2-3):235–261. Princeton Series in Applied Mathematics: Princeton University Press.

[29] Yi Xu and Qihang Lin and Tianbao Yang. (2017). Stochastic Convex Optimization: Faster Local Growth Implies Faster Global Convergence. *Proceedings of the 34rd International Conference on Machine Learning (ICML)*, 3821-3830. Sydney, Australia: ACM.

[30] Liu, Mingrui, Xiaoxuan Zhang, Lijun Zhang, Rong Jin, and Tianbao Yang. (2018). Fast Rates of ERM and Stochastic Approximation: Adaptive to Error Bound Conditions. *arXiv preprint arXiv:*, 1805.04577.

[31] Bartlett, Peter L. and Jordan, Michael I. and McAuliffe, Jon D (2006). Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, **101**(473):138–156.

[32] John C. Duchi and Shai Shalev-Shwartz and Yoram Singer and Tushar Chandra. (2008). Efficient projections onto the l_1-ball for learning in high dimensions. *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML)*, 272–279. Helsinki, Finland: ACM.

[33] Asuncion, Arthur and Newman, David (2007). UCI machine learning repository.

[34] Chang, Chih-Chung and Lin, Chih-Jen (2011). LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, **2**(3):27.

[35] Deng, Li (2012). The MNIST database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, **29**(6):141–142.

[36] Krizhevsky, Alex and Nair, Vinod and Hinton, Geoffrey (2014). The CIFAR-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html.*

[37] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 1097-1105. Lake Tahoe, NV: MIT Press.

[38] Kingma, Diederik and Ba, Jimmy (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv*:1412.6980.