Batch-Mode Active Learning via Error Bound Minimization

Quanquan Gu
Dept. of Computer Science
University of Illinois
Urbana, IL 61801
qgu3@illinois.edu

Tong Zhang
Department of Statistics
Rutgers University
Piscataway, NJ 08854
tzhang@stat.rutgers.edu

Jiawei Han
Dept. of Computer Science
University of Illinois
Urbana, IL 61801
hanj@cs.uiuc.edu

Abstract

Active learning has been proven to be quite effective in reducing the human labeling efforts by actively selecting the most informative examples to label. In this paper, we present a batch-mode active learning method based on logistic regression. Our key motivation is an out-of-sample bound on the estimation error of class distribution in logistic regression conditioned on any fixed training sample. It is different from a typical PAC-style passive learning error bound, that relies on the i.i.d. assumption of example-label pairs. In addition, it does not contain the class labels of the training sample. Therefore, it can be immediately used to design an active learning algorithm by minimizing this bound iteratively. We also discuss the connections between the proposed method and some existing active learning approaches. Experiments on benchmark UCI datasets and text datasets demonstrate that the proposed method outperforms the state-of-the-art active learning methods significantly.

1 INTRODUCTION

In a typical supervised learning problem, one often requires sufficient labeled data to train an accurate classifier, whereas the labeling process may be expensive and time consuming. This motivates *Active Learning* [11], which has been proven to be effective in reducing the human labeling efforts by actively selecting the most informative examples for labeling. The goal of active learning is to learn a classifier which accurately predicts the labels of new examples, while requesting as few labels as possible.

In the past decades, many active learning methods have been proposed. Depending on the label query strategy, active learning can be roughly categorized into fully sequential active learning [13, 27, 30, 5, 24, 7, 22], batch-mode active learning [20, 17, 21] and one-shot active learning [29, 15, 16, 14]. Fully sequential active learning algorithms select only one example to query its label at one time, and update the classifier. In contrast, batch-mode active learning algorithms select multiple examples at one time. It is more efficient since the classifier is trained fewer times. More importantly, it is able to take into account the information overlap among the multiple examples. Both fully sequential and batch-model active learning are adaptive, as in the query process, the newly labeled data in an earlier iteration can be used to guide the selection of unlabeled data in a latter iteration (e.g., by updating the classifier). In contrast, one-shot active learning is non-adaptive. In this paper, we consider batch-mode active learning, because it is more general than the other two query strategies both in theory and practice. It can be directly adapted to fully sequential and one-short active learning, by simply setting the batch-size to one or to a sufficient large number.

On the other hand, the most widely used criteria for active learning include but not limited to uncertainty sampling [27, 21], query by committee [13], mutual information [24, 16], experimental design [29, 3], and expected error minimization [15, 14]. Besides these practical algorithms mentioned above, there are also several theoretical studies [5, 12, 7, 18, 1], which provide bounds on the label complexity. The method we are going to propose belongs to the family of expected error minimization. The main advantage of the methods in this family is that the criteria are minimizing certain kind of error bounds, which directly relate the label selection procedure with the prediction error. As a result, we are particularly interested in designing such kind of active learning algorithm.

With the above motivation, we present a batch-mode active learning method, which is based on the well-known statistical model of logistic regression [19]. One advantage of logistic regression is that it has an inherent model assumption and thus it is amenable to theoretical analysis. Furthermore, it is in nature a classification model and consequently more suitable for active learning towards classification. We perform a finite sample analysis on the logistic regression

and derive an error bound on the class distribution conditioned on any fixed training sample. This bound is essential because it is different from a typical PAC-style error bound for model-free passive learning [8], that relies on the i.i.d. assumption of the example-class pairs. In contrast, our derived bound allows the training examples to be dependent, which meets the scenario of active learning. Furthermore, the derived error bound does not contain the class labels of the training sample, which allows us to do minimization by choosing training examples without knowing their labels. We propose an active learning criterion to select the examples by minimizing this upper bound iteratively. The resulting method is a combinatorial optimization problem, which is relaxed and solved approximately by projected gradient descent. It is worth noting that the derivation approach we proposed is quite general and is applicable to other generalized linear models beyond logistic regression.

As we mentioned before, although we mainly study batch-mode active learning in this paper, our proposed method supports fully sequential and one-short active learning as well. Furthermore, unlike many active learning methods [5, 12, 7], which rely on sampling the hypothesis space, our method is deterministic and easy to implement. Extensive experiments on UCI datasets and text datasets show that the proposed method significantly outperforms the state-of-the-art active learning methods.

The remainder of this paper is organized as follows. In Section 2, we analyze the logistic regression, and derive a finite sample error bound on its response distribution. In Section 3, we present an active learning criterion based on minimizing the derived error bound, followed by its optimization algorithm. We discuss some related methods in Section 4. The experiments are demonstrated in Section 5. Finally, we draw conclusions and point out the future work in Section 6.

2 FINITE SAMPLE ANALYSIS OF LOGISTIC REGRESSION

In this section, to keep this paper self-contained, we first briefly review logistic regression [19]. Then we derive an estimation error bound for the conditional class distribution based on finite-sample analysis. It is among the main contributions of this paper, and is the theoretical underpinning of the active learning approach proposed in the next section.

2.1 NOTATION

Throughout this paper, we will use lower case letters to denote scalars, lower case bold letters to denote vectors, upper case letters to denote the elements of a matrix or a set, and bold-face upper case letters to denote matrices. $\mathbf{I}$ is an identity matrix with an appropriate size. We use superscript $^\top$ to denote the transpose of a vector or a matrix. The ℓ_2-norm of a vector $\mathbf{x} \in \mathbb{R}^d$ is defined as $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^d x_i^2}$. The spectral norm of a matrix $\mathbf{A}$ is defined as $\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2$. In particular, for a squared matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$, we denote its maximum eigenvalue by $\lambda_{\max}(\mathbf{A})$, and its minimum eigenvalue by $\lambda_{\min}(\mathbf{A})$. We use $[n]$ to denote the index set $\{1, 2, \ldots, n\}$. Given a matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$, $\mathbf{X}_{\mathcal{L}}$ denotes a submatrix of $\mathbf{X}$, which consists of the columns of $\mathbf{X}$ indexed by $\mathcal{L} \subset [n]$. $\mathbf{x}_i$ denotes the i-th column of $\mathbf{X}$. And for a symmetric matrix $\mathbf{D} \in \mathbb{R}^{n\times n}$, $\mathbf{D}_{\mathcal{L}\mathcal{L}}$ denotes a submatrix of $\mathbf{D}$, which contains the rows and columns indexed by $\mathcal{L}$.

2.2 LOGISTIC REGRESSION

Let us consider the binary classification case for simplicity. Given a sample set $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{\pm 1\}$, to have a simpler derivation without considering the bias term θ, one often augments each example with an additional dimension: $\mathbf{x}^\top \leftarrow [\mathbf{x}^\top; 1]$ and $\mathbf{w}^\top \leftarrow [\mathbf{w}^\top; \theta]$. In logistic regression [19], the conditional class probability $\Pr(y|\mathbf{x})$ is given by

$$\Pr(y|\mathbf{x}; \mathbf{w}) = \sigma(y\mathbf{w}^\top \mathbf{x}),$$

where $\sigma(a)$ is the logistic sigmoid function, i.e., $\sigma(a) = 1/(1 + \exp(-a))$. Note that $\sigma(a)$ is a concave function when $a > 0$.

To avoid over-fitting, we place a prior on $\mathbf{w}$ in the form of a zero-mean Gaussian distribution with isotropic covariance, i.e., $\mathcal{N}(\mathbf{0}, 1/\lambda\mathbf{I})$, and seek a $\mathbf{w}$ which maximizes the log-likelihood of the posterior distribution given the training data S, which is equivalent to

$$\widehat{\mathbf{w}} = \arg\min_{\mathbf{w}} \lambda\|\mathbf{w}\|_2^2 - \frac{1}{n}\sum_{i=1}^n \log\sigma(y_i\mathbf{w}^\top\mathbf{x}_i), \qquad (1)$$

where λ is a positive regularization parameter. Eq. (1) is also known as penalized logistic regression, or more precisely, ℓ_2-regularized Logistic regression. It is worth noting that although logistic regression is called "regression", it is in nature a classification model, because its response is binary and it directly estimates the conditional class probability given the data. This is also the reason that we deem that deriving an active learning algorithm based on logistic regression is more natural and effective for classification than inventing one from the real regression models [29, 14].

2.3 ERROR BOUNDS FOR LOGISTIC REGRESSION

In the following, we will analyze ℓ_2-regularized logistic regression reviewed above. First of all, we assume that there exists an unknown true parameter $\mathbf{w}_* \in \mathbb{R}^d$, by which the class label of an example is generated as follows

$$\Pr(y|\mathbf{x}) = \sigma(y\mathbf{w}_*^\top\mathbf{x}). \qquad (2)$$

where $\|\mathbf{w}_*\|_2 \leq R$ for some $R > 0$. This is our model assumption. All the theoretical results we are going to present are built up on this assumption.

Without loss of generality, we assume $\|\mathbf{x}_i\|_2 \leq 1$ for $\forall i$. Then it is easy to verify that

$$\left\| \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right\|_2 \leq 1.$$

The following theorem provides a bound on the estimation error of $\widehat{\mathbf{w}}$, which is central in our theoretical results. The detailed proofs can be found in the supplementary material.

Theorem 1. *For any fixed sample $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where y_i follows the conditional distribution as in Eq. (2), and $\|\mathbf{x}_i\|_2 \leq 1$ for $\forall i$. $\widehat{\mathbf{w}}$ is the estimated weight vector by logistic regression on S, then the estimation error of $\widehat{\mathbf{w}}$ is upper bounded as*

$$\begin{aligned}&\mathbb{E}_{Y|X}\left[\|\widehat{\mathbf{w}} - \mathbf{w}_*\|_2\right] \\ \leq & C_1 \lambda_{\max}\left(\left(\lambda \mathbf{I} + \frac{1}{n}\mathbf{X}\mathbf{D}\mathbf{X}^\top\right)^{-1}\right),\end{aligned}$$

where $\mathbb{E}_{Y|X}$ is the shorthand for $\mathbb{E}_{y_1,\ldots,y_n|\mathbf{x}_1,\ldots,\mathbf{x}_n}$, $C_1 = 1 + 2\lambda R$, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$, $\mathbf{D}$ is a diagonal matrix with diagonal elements defined as follows

$$D_{ii} = \left(1 - \sigma(\mathbf{w}_*^\top \mathbf{x}_i)\right)\sigma(\mathbf{w}_*^\top \mathbf{x}_i). \tag{3}$$

Proof. (Sketch of proof): We use a similar technique adopted in [25]. Define $f(\mathbf{w}) = \lambda\|\mathbf{w}\|_2^2 - 1/n \sum_{i=1}^n \log \sigma(y_i \mathbf{w}^\top \mathbf{x}_i)$. Let $\widehat{\mathbf{w}} = \arg\min_{\mathbf{w}} f(\mathbf{w})$.

Define $g(\Delta)$ as follows

$$\begin{aligned}g(\Delta) =& \mathbb{E}_{Y|X}[\lambda\|\mathbf{w}_* + \Delta\|_2^2 - \frac{1}{n}\sum_{i=1}^{n} \log \sigma(y_i(\mathbf{w}_* + \Delta)^\top \mathbf{x}_i) \\ & -\lambda\|\mathbf{w}_*\|_2^2 + \frac{1}{n}\sum_{i=1}^{n} \log \sigma(y_i \mathbf{w}^\top \mathbf{x}_i)],\end{aligned}$$

where $\Delta = \mathbf{w} - \mathbf{w}_*$. It is easy to verify that $g(0) = 0$. Using the optimality of $\widehat{\mathbf{w}}$, we have $f(\widehat{\mathbf{w}}) \leq f(\mathbf{w}_*)$, yielding

$$\begin{aligned}&\lambda\|\widehat{\mathbf{w}}\|_2^2 - \frac{1}{n}\sum_{i=1}^{n} \log \sigma(y_i \widehat{\mathbf{w}}^\top \mathbf{x}_i) \\ \leq & \lambda\|\mathbf{w}_*\|_2^2 - \frac{1}{n}\sum_{i=1}^{n} \log \sigma(y_i \mathbf{w}_*^\top \mathbf{x}_i).\end{aligned}$$

Therefore, we have $g(\widehat{\Delta}) \leq 0$ with $\widehat{\Delta} = \widehat{\mathbf{w}} - \mathbf{w}_*$. Suppose that we show for some radius $B > 0$, and for $\Delta \in \mathbb{R}^d$ with $\|\Delta\|_2 = B$, we have $g(\Delta) > 0$. We then can claim that $\|\widehat{\Delta}\|_2 \leq B$. We prove it by contradiction: If $\widehat{\Delta}$ lies outside the ball of radius B, then by convexity of $g(\cdot)$, we have

$$g(t\widehat{\Delta} + (1-t)0) \leq t g(\widehat{\Delta}) + (1-t)g(0) \leq 0,$$

for some appropriately chosen $t \in (0, 1)$ such that $t\widehat{\Delta} + (1-t)0$ lies on the boundary of the ball. This is contradict with the fact that $g(t\widehat{\Delta} + (1-t)0) > 0$.

By some calculations, we have

$$g(\Delta) \geq C_{\min} B^2 - B - 2\lambda R B + \lambda B^2,$$

where $C_{\min} = \lambda_{\min}(1/n \sum_{i=1}^n \sigma(\mathbf{w}_*^\top \mathbf{x}_i)(1 - \sigma(\mathbf{w}_*^\top \mathbf{x}_i))\mathbf{x}_i \mathbf{x}_i^\top)$.

It is easy to show that $B = (1 + 2\lambda R)/(C_{\min} + \lambda)$ makes $g(\Delta) > 0$. Based on previous argument, since $g(\widehat{\Delta}) \leq 0$, we have

$$\begin{aligned}\|\widehat{\Delta}\|_2 \leq & \frac{1 + 2\lambda R}{C_{\min} + \lambda} \\ = & \frac{1 + 2\lambda R}{\lambda_{\min}\left(\lambda \mathbf{I} + \frac{1}{n}\sum_{i=1}^{n} \sigma(\mathbf{w}_*^\top \mathbf{x}_i)(1 - \sigma(\mathbf{w}_*^\top \mathbf{x}_i))\mathbf{x}_i \mathbf{x}_i^\top\right)} \\ = & (1 + 2\lambda R)\lambda_{\max}\left(\left(\lambda \mathbf{I} + \frac{1}{n}\sum_{i=1}^{n} D_{ii}\mathbf{x}_i \mathbf{x}_i^\top\right)^{-1}\right).\end{aligned}$$

□

Remark 1: The above bound is derived by analyzing the second-order Taylor expansion of Eq. (1). If we simply use the strongly convex property of Eq. (1), we cannot get the desired bound, because the information of the second-order derivative will not be fully utilized. Consequently, the above bound is sharper than the bound derived by strong convexity.

In logistic regression, the classification of a new example is solely based on its estimated conditional class probability. Therefore, we aim to bound the estimation error of the conditional class probability rather than $(\widehat{\mathbf{w}}^\top \mathbf{v} - \mathbf{w}_*^\top \mathbf{v})^2$ as in linear regression. Based on Theorem 1, we can prove the following theorem, which achieves our goal.

Theorem 2. *For any fixed sample $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where y_i follows the conditional distribution as in Eq. (2), and $\|\mathbf{x}_i\|_2 \leq 1$ for $\forall i$. $\widehat{\mathbf{w}}$ is the estimated weight vector by logistic regression on S. Then the estimated conditional class probability on a validation set $\{\mathbf{v}_j\}_{j=1}^m$ is upper bounded as*

$$\begin{aligned}&\mathbb{E}_{Y|X}\left[\sum_{j=1}^{m}\left(\Pr(y|\mathbf{v}_j; \hat{\mathbf{w}}) - \Pr(y|\mathbf{v}_j; \mathbf{w}_*)\right)^2\right] \\ \leq & C_2 tr\left(\left(\lambda \mathbf{I} + \frac{1}{n}\mathbf{X}\mathbf{D}\mathbf{X}^\top\right)^{-1} \mathbf{V}\mathbf{\Sigma}\mathbf{V}^\top\right),\end{aligned}$$

where $C_2 = (1+\lambda R)^2(\lambda+1)^2/\lambda^2$, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ and $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_m]$, $\mathbf{\Sigma}$ is a diagonal matrix with diagonal elements defined as follows

$$\Sigma_{jj} = \left(1 - \sigma(\tilde{\mathbf{w}}^\top \mathbf{v}_i)\right)\sigma(\tilde{\mathbf{w}}^\top \mathbf{v}_i), \tag{4}$$

with $\tilde{\mathbf{w}} = \mathbf{w}_* + \alpha(\widehat{\mathbf{w}} - \mathbf{w}_*)$ for some $\alpha \in [0, 1]$.

Proof. (Sketch of proof): Consider the second-order Taylor expansion of $\sigma(\widehat{\mathbf{w}}^\top \mathbf{v}_j)$, we have the following inequality,

$$\sigma(\widehat{\mathbf{w}}^\top \mathbf{v}_j) = \sigma(\mathbf{w}_*^\top \mathbf{v}_j) + \sigma(\tilde{\mathbf{w}}^\top \mathbf{v}_j)\left(1 - \sigma(\tilde{\mathbf{w}}^\top \mathbf{v}_j)\right)\mathbf{v}_j \cdot \widehat{\Delta},$$

where $\tilde{\mathbf{w}} = \mathbf{w}_* + \alpha(\widehat{\mathbf{w}} - \mathbf{w}_*) = \mathbf{w}_* + \alpha\widehat{\Delta}$ for some $\alpha \in [0, 1]$.

Then we have

$$\begin{aligned}
&\mathbb{E}_{Y|X}\left[\sum_{j=1}^{m}\left(\Pr(y|\mathbf{v}_j, \widehat{\mathbf{w}}) - \Pr(y|\mathbf{v}_j, \mathbf{w}^*)\right)^2\right] \\
=&\sum_{j=1}^{m}\left(\sigma(\widehat{\mathbf{w}}^\top \mathbf{v}_j) - \sigma(\mathbf{w}_*^\top \mathbf{v}_j)\right)^2 \\
=&\sum_{j=1}^{m}\left(\sigma(\tilde{\mathbf{w}}^\top \mathbf{v}_j)\left(1 - \sigma(\tilde{\mathbf{w}}^\top \mathbf{v}_j)\right)\mathbf{v}_j \cdot \widehat{\Delta}\right)^2 \\
=&\sum_{j=1}^{m}\left(\Sigma_{jj}\mathbf{v}_j \cdot \widehat{\Delta}\right)^2,
\end{aligned}$$

which can be further bounded by Theorem 1. □

Remark 2: All the above theoretical results hold under the conditional expectation with respect to the conditional class distribution $\Pr(Y|X)$, given any fixed design matrix $\mathbf{X}$. They do not require either $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ or $\{\mathbf{x}_i\}_{i=1}^n$ to be i.i.d., which is the common assumption in passive learning. In addition, the derived bounds do not depend on the class labels of the training sample explicitly.

It can be observed from Theorem 2 that, the expected estimation error of the conditional class probability $P(Y|X)$ on a validation set is upper bounded by a term which can be approximately computed based on the training set together with the validation set without their labels. Therefore, they can be used to guide the design of active learning algorithms, because the examples in the pool are not only dependent (starting from the second round of label query) in active learning, but also unlabeled. It also explains why we need to derive such a kind of bounds to design active learning algorithms rather than using existing PAC-style bounds for model-free learning [8]. In a nutshell, we can minimize this bound by choosing a subsample of the training set. We will discuss this in details in the next section.

3 ACTIVE LEARNING BASED ON ERROR BOUND MINIMIZATION

Before presenting the new active learning method, let us recall the basic setting of batch-mode active learning as follows. Given a training data matrix, i.e., $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, and an initial labeled set $\mathcal{L}$, together with a set of unlabeled examples, i.e., $\mathcal{U}$. Batch-mode active learning operates in T iterations. In each iteration, the learner will choose b examples (denoted by $\mathcal{B}$) from the unlabeled set $\mathcal{U}$ to label, and add these labeled examples into the existing labeled set $\mathcal{L}$ (also remove $\mathcal{B}$ from the unlabeled set $\mathcal{U}$). The goal of batch-mode active learning is to find bT examples in total, which are the most informative examples, namely selected subsample set, to query their labels.

3.1 THE CRITERION

The proposed active learning method is motivated by Theorem 2. In Theorem 2, we can see that the estimation error of the conditional class probability is upper bounded by $\mathrm{tr}(\mathbf{\Sigma}^{\frac{1}{2}}\mathbf{V}^\top(\lambda\mathbf{I} + 1/n\mathbf{X}\mathbf{D}\mathbf{X}^\top)^{-1}\mathbf{V}^\top\mathbf{\Sigma}^{\frac{1}{2}})$, where $\mathbf{D}$ and $\mathbf{\Sigma}$ are depending on $\mathbf{w}_*$ and $\tilde{\mathbf{w}}$. Since $\mathbf{w}_*$ and $\tilde{\mathbf{w}}$ are unknown, we cannot calculate $\mathbf{D}$ and $\mathbf{\Sigma}$ exactly. Instead, we use the current $\widehat{\mathbf{w}}$ to approximate $\mathbf{w}_*$ and $\tilde{\mathbf{w}}$. Based on the approximate $\mathbf{D}$ and $\mathbf{\Sigma}$, we can choose b examples from $\mathcal{U}$ which minimizes the upper bound. Then we will use these newly labeled b examples together with existing labeled examples to update the classifier. After that, we may get better approximations to $\mathbf{D}$ and $\mathbf{\Sigma}$. This process is repeated until the label budget is used out.

More specifically, in the t-th iteration, we have labeled set $\mathcal{L}$ and unlabeled set $\mathcal{U}$. We also have the classifier $\widehat{\mathbf{w}}_t$, based on which we can get approximations of $\mathbf{D}$ and $\mathbf{\Sigma}$. Then we are going to choose the next b examples by minimizing the following criterion,

$$\arg\min_{\mathcal{B}\subset\mathcal{U}} \mathrm{tr}\left(\mathbf{\Sigma}^{\frac{1}{2}}\mathbf{V}^\top\left(\lambda\mathbf{I} + \mathbf{X}_\mathcal{B}\mathbf{D}_{\mathcal{B}\mathcal{B}}\mathbf{X}_\mathcal{B}^\top\right)^{-1}\mathbf{V}\mathbf{\Sigma}^{\frac{1}{2}}\right),$$

where we absorb $1/n$ into λ. By introducing $\tilde{\mathbf{v}}_j = \sqrt{\Sigma_{jj}}\mathbf{v}_j$ and $\tilde{\mathbf{x}}_i = \sqrt{D_{ii}}\mathbf{x}_i$, the above optimization problem can be simplified as

$$\arg\min_{\mathcal{B}\subset\mathcal{U}} \mathrm{tr}\left(\tilde{\mathbf{V}}^\top\left(\lambda\mathbf{I} + \tilde{\mathbf{X}}_\mathcal{B}\tilde{\mathbf{X}}_\mathcal{B}^\top\right)^{-1}\tilde{\mathbf{V}}\right). \tag{5}$$

It is a combinatorial optimization problem. Similar problems have been encountered in previous work [29]. One way to solve it is applying the sequential minimization algorithm derived in [29] b times, to get a batch $\mathcal{B}$. However, this sacrifices the advantage of batch-mode active learning, because it neglects the information overlap among examples. Another way is formulating it as a semi-definite programming [9], which is computationally very expensive. Here, we do some relaxation and use the projected gradient descent to solve it, following the idea adopted in [14].

3.2 OPTIMIZATION

We introduce a selection matrix $\mathbf{S} \in \mathbb{R}^{|\mathcal{U}|\times b}$, which is defined as

$$S_{ij} = \begin{cases} 1, & \text{if the } i\text{-th example in } \mathcal{U} \text{ is selected} \\ & \text{as the } j\text{-point in } \mathcal{B} \\ 0, & \text{otherwise.} \end{cases}$$

It is easy to check that each column of $\mathbf{S}$ has one and only one 1, and each row has at most one 1. We denote the constraint set for $\mathbf{S}$ by $\mathcal{S}_1 = \{\mathbf{S}|\mathbf{S} \in \{0,1\}^{|\mathcal{U}|\times b}, \mathbf{S}^\top\mathbf{S} = \mathbf{I}\}$.

With $\mathbf{S}$, we have $\tilde{\mathbf{X}}_{\mathcal{B}} = \tilde{\mathbf{X}}\mathbf{S}$. Then Eq. (5) can be simplified as

$$\arg\min_{\mathbf{S}\in\mathcal{S}_1} \text{tr}\left(\tilde{\mathbf{V}}^\top(\lambda\mathbf{I} + \tilde{\mathbf{X}}\mathbf{S}\mathbf{S}^\top\tilde{\mathbf{X}}^\top)^{-1}\tilde{\mathbf{V}}\right).$$

The above optimization problem is almost continuous, except the constraint set $\mathcal{S}_1$. In order to apply continuous optimization algorithms, we relax it into the following continuous domain, i.e., $\mathcal{S}_2 = \{\mathbf{S}|\mathbf{S} \geq 0, \mathbf{S}^\top\mathbf{S} = \mathbf{I}\}$.

Since the projection onto $\{\mathbf{S} : \mathbf{S}^\top\mathbf{S} = \mathbf{I}\}$ is computationally expensive, we would like to design an algorithm in which the constraint $\mathbf{S}^\top\mathbf{S} = \mathbf{I}$ is automatically satisfied after each gradient descent. To cope with $\mathbf{S}^\top\mathbf{S} = \mathbf{I}$, we introduce a Lagrange multiplier $\mathbf{\Lambda} \in \mathbb{R}^{b\times b}$, and write down the Lagrangian function as

$$\begin{aligned} L(\mathbf{S}) =& \text{tr}\left(\tilde{\mathbf{V}}^\top(\lambda\mathbf{I} + \tilde{\mathbf{X}}\mathbf{S}\mathbf{S}^\top\tilde{\mathbf{X}}^\top)^{-1}\tilde{\mathbf{V}}\right) \\ &+\text{tr}\left(\mathbf{\Lambda}(\mathbf{S}^\top\mathbf{S} - \mathbf{I})\right). \end{aligned}$$

The derivative of $L(\mathbf{S})$ with respect to $\mathbf{S}$ is

$$\frac{\partial L}{\partial \mathbf{S}} = -2\tilde{\mathbf{X}}^\top\mathbf{B}\tilde{\mathbf{X}}\mathbf{S} + 2\mathbf{S}\mathbf{\Lambda}, \tag{6}$$

where $\mathbf{B} = \mathbf{A}^{-1}(\tilde{\mathbf{V}}\tilde{\mathbf{V}}^\top)\mathbf{A}^{-1}$ and $\mathbf{A} = \lambda\mathbf{I} + \tilde{\mathbf{X}}\mathbf{S}\mathbf{S}^\top\tilde{\mathbf{X}}^\top$. Using the fact that $\mathbf{S}^\top\mathbf{S} = \mathbf{I}$ yields $\mathbf{\Lambda} = \mathbf{S}^\top\tilde{\mathbf{X}}^\top\mathbf{B}\tilde{\mathbf{X}}\mathbf{S}$. Substituting the Lagrange multiplier $\mathbf{\Lambda}$ back into Eq. (6), we obtain the derivative depending solely on $\mathbf{S}$. Then following [14], we can use projected gradient descent to find a local optimal solution for Eq. (6), where the projection is only onto $\{\mathbf{S} : \mathbf{S} \geq \mathbf{0}\}$. After the local optimal $\mathbf{S}^*$ is obtained, we can discretize it to obtain the desired solution. The analysis of the gap between the local optima and the global optima is challenging and perhaps an open problem. It may be helpful to realize that $\mathcal{S}_2$ is a matching polytope [23] for such kind of analysis.

In summary, we present the whole algorithm for active learning based on error bound minimization in Algorithm 1. Since our algorithm is designed from logistic regression, we call it **Logistic Bound**. In the special case that $b = 1$, i.e., fully sequential active learning, we do not need to use projected gradient descent in each iteration. In that case, we can find the best single example by sorting.

We emphasize that α in Theorem 2 is some parameter within $[0, 1]$. This parameter comes from the mean value theorem in the derivation. It is not a parameter of our algorithm, because we use $\widehat{\mathbf{w}}$ to approximate $\tilde{\mathbf{w}}$ in Algorithm 1. So we do not need to tune α at all.

Algorithm 1 Batch-Mode Active Learning Based on Error Bound Minimization (Logistic Bound)

Input: $\mathbf{X}$, $\mathbf{V}$, number of iterations T, batch size b, regularization parameter λ, initial labeled set $\mathcal{L}$ and unlabeled set $\mathcal{U}$;
for $t = 1 \to T$ **do**
  Compute $\widehat{\mathbf{w}}_t$ based on $\mathcal{L}$;
  Compute $\mathbf{D}$ and $\mathbf{\Sigma}$ based on Eqs. (3) and (4);
  Compute $\mathcal{B} \subset \mathcal{U}$ based on Eq. (5);
  Update $\mathcal{L} = \mathcal{L} \cup \mathcal{B}$ and $\mathcal{U} = \mathcal{U} \setminus \mathcal{B}$;
end for

3.3 TIME COMPLEXITY

In this subsection, we analyze the time complexity of the proposed active learning algorithm. The computation of Eq. (6) involves $\mathbf{A}^{-1}$, which is the inverse of a $d \times d$ matrix. However, we do not need to compute it directly. Since $\mathbf{A}^{-1} = (\lambda\mathbf{I} + \mathbf{X}\mathbf{S}\mathbf{S}^\top\mathbf{X}^\top)^{-1}$, by applying the Woodbury matrix identity, we have $\mathbf{A}^{-1} = 1/\lambda\mathbf{I} - 1/\lambda\mathbf{X}\mathbf{S}(\lambda\mathbf{I} + \mathbf{S}^\top\mathbf{X}^\top\mathbf{X}\mathbf{S})^{-1}\mathbf{S}^\top\mathbf{X}^\top$. Thus we only need to calculate the inverse of $(\lambda\mathbf{I} + \mathbf{S}^\top\mathbf{X}^\top\mathbf{X}\mathbf{S})$, whose size is $b \times b$, where b is the batch size. So the time complexity of computing the gradient in Eq. (6) can be reduced to $O(ndb + db^2 + b^3)$, which is dominated by $O(ndb)$ because b is often set to 5 to 50. The total complexity of the projected gradient descent is $O(ndbt)$, where t is the iteration number. The time complexity is clearly linear to the sample size n, and the dimension of the input space d.

4 RELATED WORK

In this section, we show the connections of the proposed approach with some existing active learning methods.

One thread of related work is experimental design [2]. For instance, Yu et al. [29] proposed transductive experimental design (TED), whose intent is to select the examples to learn a least squares regression function which has minimum prediction variance on the validation data. As we mentioned before, it is a *non-adaptive* active learning method, because the label information of the selected examples cannot be utilized to select subsequent unlabeled examples. Intuitively, taking into account the labels of queried examples is beneficial for subsequent label query. Our method is able to utilize the labeled examples obtained up to now to choose the next batch of examples through $\mathbf{D}$ and $\mathbf{\Sigma}$. Recall that in our method, each example in the pool is weighted by $D_{ii} = \sigma(\widehat{\mathbf{w}}^\top\mathbf{x}_i)(1 - \sigma(\widehat{\mathbf{w}}^\top\mathbf{x}_i))$. Apparently, the more uncertain an example is, the bigger its weight will be (because D_{ii} is maximized when $\sigma(\widehat{\mathbf{w}}^\top\mathbf{x}_i) = 1/2$). In the special case, if $\sigma(\widehat{\mathbf{w}}^\top\mathbf{x}_i) = 1/2$ for every example in the pool, then all the examples are equally weighted, and our method will degrade to TED. By applying the derivation technique to ridge regression, we

can obtain a very similar result to [29]. However, using the derivation technique from experimental design, we cannot get the results in this paper. So the derivation technique used in this paper is more general.

The second line of related work is active learning based on logistic regression [30, 26]. Based on the asymptotic analysis, Zhang and Oles [30] derived the inverse of the Fisher information matrix of the maximum likelihood estimation (MLE) of logistic regression, which measures the variance of model. Thus they proposed an active learning criterion by minimizing the variance. Our criterion derived in Theorem 2 is from finite sample analysis, which is non-asymptotic and different from the criterion derived in [30]. Since finite sample error bounds characterize the behavior of a classifier provided with a finite training sample, they provide more accurate guidance on algorithm design than asymptotic analysis. Following this seminal work, several incremental studies were presented, which solve the same criterion using different sophisticated optimization algorithms. For example, Hoi et al. [20] proposed to reformulate it as a submodular function maximization problem. On the other hand, Guo and Schuurmans [17] proposed a batch-mode active learning algorithm based on logistic regression, by maximizing the likelihood on the labeled training sample and minimizing the entropy on the selected unlabeled training sample. The main innovation lies in the optimization part rather than the theoretical results.

The last but not least related work is the family of expected error minimization-based approaches. Recently, Gu et al. [14] proposed an active learning method based on minimizing the out-of-sample error bound for Laplacian regularized least squares (LapRLS) [6], a semi-supervised version of least squares regression. It sheds light on designing active learning algorithms via deriving certain kind of error bounds, which do not rely on the i.i.d assumption of the training sample nor the class labels. The method is non-adaptive[1], raising a question that whether we can design an adaptive active learning algorithm along this line. Our result in this paper is in the affirmative. However, the linear regression model as well as the derivation technique used there are not capable to achieve this goal. So we study logistic regression with a new analyzing technique instead. Note also that in the supervised case, the methods proposed in [14] and [29] are identical in terms of the criterion.

[1]It is nonadaptive, in the sense that when the model ($\widehat{\mathbf{w}}$) is updated using the newly labeled examples, the algorithm is not able to use the information from the updated model ($\widehat{\mathbf{w}}$) to choose next batch of examples to label. In fact, the active learning algorithm in [14] does not use any information from $\widehat{\mathbf{w}}$ in the process of active learning, because its criterion does not depend on $\widehat{\mathbf{w}}$.

5 EXPERIMENTS

In this section, we evaluate the proposed method on both UCI datesets [3] (wdbc, wpbc, sonar, heart, australian, diabetes, splice) and two text datasets (Text1 and Text2) generated from the famous 20-newsgroups data set[2]. For the text datasets, the original number of features (words) is $8,014$. We apply principal component analysis (PCA) to reduce the input dimensionality by projecting the data onto its leading principal components, where the number of principal component is determined such that it accounts for 95% of its total variance. For each example, we normalize it into a vector with unit ℓ_2-norm.

5.1 EXPERIMENTAL SETUP AND BASELINES

In order to randomize the experiments, in each run of experiments, we use 50% data as the training examples. The remaining 50% data is used as test set. We use the training set for active learning and evaluate the prediction performance on the fixed test set. This random split was repeated 20 times, thus we can do statistical significance test. We study a difficult case of active learning, where we start with one randomly selected example per class. All the algorithms start with the same initial labeled set, unlabeled set and test set.

To demonstrate the effectiveness of our proposed method, we compare it with existing state-of-the art algorithms, including one fully sequential active learning approach, one non-adaptive active learning algorithm, and three batch-mode active learning methods. We summarize these methods as follows:

Random Sampling (**Random**): It is the simplest baseline, which uniformly selects examples from the candidate set as training data.

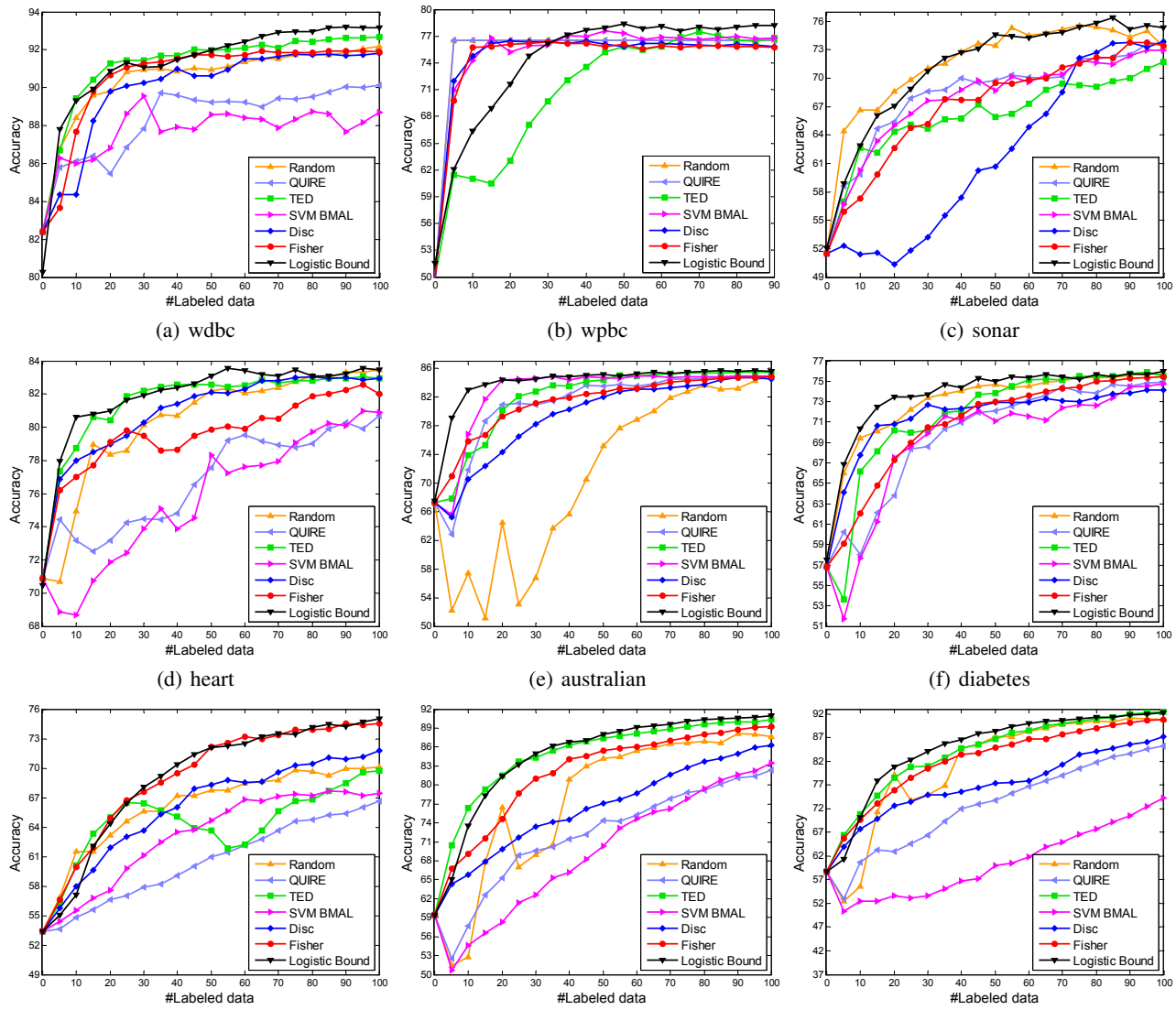
Query the informative and representative examples (**QUIRE**) [22]: it is a fully sequential active learning algorithm.

Transductive experiment design (**TED**) [29]: it is a non-adaptive active learning method. Note that it selects all the examples to label at one shot.

SVM batch-mode active learning (**SVM BMAL**) [21]: in our empirical study, we found that it consistently outperforms SVM active learning [27], so we only demonstrate its results while omit the results of SVM active learning. We use linear kernel for SVM. In fact, SVM active learning can be seen as a special case of SVM BMAL, where the batch size is equal to 1.

Discriminative batch-mode active learning (**Disc**) [17]: it is a batch-mode active learning algorithm based on logistic regression.

[2]http://people.csail.mit.edu/jrennie/20Newsgroups/



(a) wdbc (b) wpbc (c) sonar

(d) heart (e) australian (f) diabetes

(g) splice (h) Text1 (i) Text2

Figure 1: Comparison of active learning methods on both UCI and text datasets with batch side $b = 5$, and $T = 20$ iterations. The x-axis is number of labeled examples, and the y-axis is the classification accuracy (%).

Table 1: Win/tie/loss counts for the proposed method versus the other methods during the whole active learning process, based on paired t-test at 95% significance level. The first column is the dataset name (#examples/#features).

DATASETS	VS RANDOM	VS QUIRE	VS TED	VS SVM_BMAL	VS DISC	VS FISHER
WDBC(569/30)	14/6/0	18/2/0	8/12/0	18/2/0	17/3/0	10/10/0
WPBC(198/33)	9/6/5	9/6/5	13/7/0	4/12/4	10/6/4	11/6/3
SONAR(208/60)	2/16/2	13/7/0	18/2/0	14/6/0	20/0/0	19/1/0
HEART(270/13)	9/11/0	19/1/0	6/13/1	20/0/0	5/15/0	14/6/0
AUSTRALIAN(690/14)	14/6/0	20/0/0	10/10/0	10/10/0	20/0/0	20/0/0
DIABETES(768/8)	1/19/0	20/0/0	11/9/0	20/0/0	17/3/0	15/5/0
SPLICE(1000/60)	15/4/1	18/2/0	14/5/1	18/2/0	16/4/0	0/19/1
TEXT1(1980/991)	20/0/0	20/0/0	13/5/2	20/0/0	19/1/0	19/1/0
TEXT2(1990/768)	18/2/0	20/0/0	9/10/1	20/0/0	18/2/0	18/1/1

Fisher information of logistic regression (**Fisher**) [20]: It is also a batch-mode active learning based on logistic regression. However, it is derived from non-asymptotic analysis of logistic regression.

For our method, the validation set $\mathbf{V}$ is set to the same as the pool of unlabeled examples. Recall that our method does not require the labels of the validation set either.

For each dataset, we let the active learning methods incrementally choose $b = 5$ examples to label, and perform $T = 20$ iterations in total (except for wpbc, where we only perform $T = 19$ iterations due to limited examples). We did not compare with [16], because we were not able to acquire a working implementation of this algorithm. According to the experimental results (Table 1) reported in [16], its performance is statistically similar to Disc [17]. We did not use semi-supervised classifiers. Hence the approach proposed in [14] reduces to TED. Most of the implementations are provided by the authors of the corresponding papers.

One issue with most of the active learning methods we investigated is that they are invented based on different classifiers. For example, TED is designed for ridge regression. Disc and Fisher are developed based on logistic regression. We use different classifiers for different active learning approaches, because we found that using the classifier based on which the active learning method is derived can lead to better results than using other classifiers. Furthermore, for each active learning method, its parameter and the parameter of its corresponding classifier are tuned by 5-fold cross validation on the labeled set through searching the grid $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$.

5.2 RESULTS AND DISCUSSIONS

The experimental results are shown in Figure 1. In all subfigures, the x-axis represents the number of labeled examples, while the y-axis is the averaged classification accuracy on the test data over 20 runs.

We compare all the active learning methods during the entire query process. Recall that in Figure 1, there are 20 query points (except for wpbc, which has only 19), with 20 results on each of them. We therefore run a 2-sided paired t-test at each query point, at 95% significance level. The results of t-test can be categorized into three cases: (i) our method outperforms a specific algorithm significantly, denoted by "win"; (ii) our method is significantly worse than a specific algorithm, denoted by "lose"; (iii) otherwise, denoted by "tie". We summarize the t-test results in terms of the count of "win", "tie" and "lose" in Table 1.

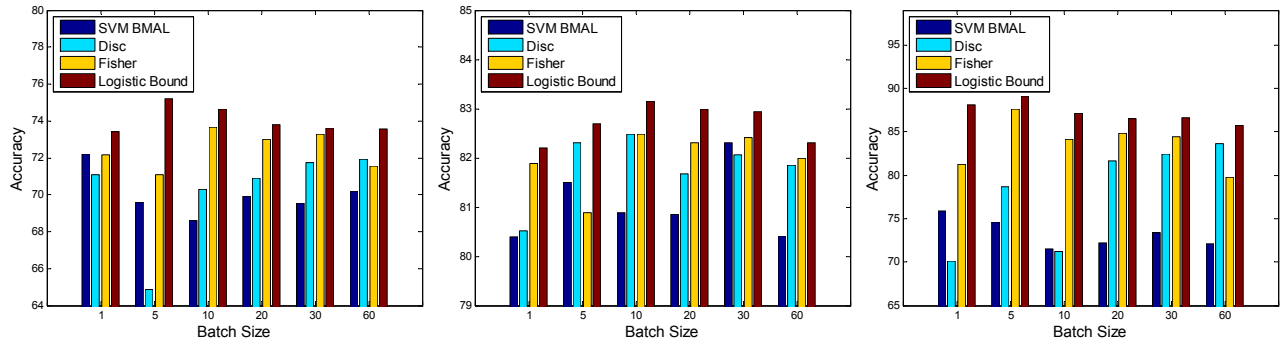
We observe that the proposed method outperforms the other methods significantly at most cases. SVM_BMAL and QUIRE are often the worst. The reason is probably that their criteria are not related to prediction performance. The performance of Disc is satisfactory. Yet it performs well on some datasets while not very well on other datasets. The performance of TED and Fisher are comparable. Although TED aims to minimize the variance of prediction, [14] showed that it is actually consistent with minimizing the out-of-sample error of ridge regression. This explains its good performance. However, since TED is a nonadaptive active learning method, it cannot fully utilize the label information during the query process. This limits its performance on many datasets. Fisher minimizes the uncertainty of the model, which does not necessarily lead to small generalization error. However, it happens that the uncertain reduction criterion for logistic regression derived in [30] is a little similar to our criterion. This may interpret its general good performance. The superior performance of our method is attributed to its theoretical foundation, which guarantees that the classifier can achieve small prediction error on the unseen data. Lastly, we found that the performance of random sampling is not bad. As an unbiased label selection procedure, random sampling is at least a consistent algorithm to choose the training sample, as is widely done in passive learning. This is consistent with the result reported in [17].

5.3 STUDY ON THE BATCH SIZE

In previous experiments, we fixed the batch size to 5, which could be biased in comparison. So we will compare our method with those batch-mode active learning algorithms under different settings of batch size here. We vary the batch size using the grid $\{1, 5, 10, 20, 30, 60\}$ and show the results with 60 labeled examples in Figure 2. We only show the results on three datasets (Sonar, Heart and Text1). Similar results can be observed on the other datasets.

It can be seen that under different batch sizes, our method outperforms the other batch-mode active learning algorithms in most cases. This strengthens the superiority of our method over the others. In addition, we also observe some interesting results. For example, for some batch-mode active learning algorithms such as SVM BMAL and Disc, their performance of using a batch size of more than one example sometimes seems not as good as choosing a single example at each round. This implies that they may not be able to address the information overlap among examples very well. In contrast, our method is able to exploit the interdependence among examples, because our method usually achieves better results with batch-size larger than one.

In addition, we found that our method obtains the best result when the batch size is either not too small ($b = 1$) nor too large ($b = 60$). This is quite reasonable, because when $b = 1$, it is a fully sequential strategy, and we cannot utilize the dependence among examples. On the contrary, if the batch size is too large (such as one-short active learning in



(a) sonar (b) heart (c) Text1

Figure 2: Comparison of batch-mode active learning methods on three datasets with different batch size ranging from $b = 1$ to $b = 60$. The x-axis represents the batch size, and the y-axis is the classification accuracy (%) with 60 labeled examples.

the extreme case), the information contained in the newly labeled examples cannot be immediately exploited through updating the classifier, which may limit its performance. This somehow implies the superiority of batch-mode active learning against both fully sequential and one-short active learning. It also suggests us to choose a medium size of batch in practice. More rigorous analysis is required in the future work.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we present a novel active learning method based on out-of-sample error bound minimization. We use logistic regression as a running example to derive the algorithm. We would like to emphasize that the derivation technique developed in this paper applies to other generalized linear models, or even more sophisticated graphical models. In our future work, we will study these alternatives. We also plan to conduct comparisons with some other batch-mode active learning methods proposed recently [4] [10] [28]. On the other hand, we aim to develop an algorithm solving Eq. (5) with provable guarantee.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments. This work was supported in part by the U.S. Army Research Laboratory under Cooperative Agreement No. W911NF-09-2-0053 (NS-CTA), the U.S. Army Research Office under Cooperative Agreement No. W911NF-13-1-0193, U.S. National Science Foundation grants CNS-0931975, IIS-1017362, IIS-1320617, IIS-1354329, DTRA, NASA NRA-NNH10ZDA001N, and MIAS, a DHS-IDS Center for Multimodal Information Access and Synthesis at UIUC. The first author was supported by IBM Ph.D. Fellowship (2013-2014).

References

[1] A. Agarwal. Selective sampling algorithms for cost-sensitive multiclass prediction. In *ICML (3)*, pages 1220–1228, 2013.

[2] A. D. Anthony Atkinson and R. Tobias. *Optimum Experimental Designs*. Oxford Statistical Science Series. Oxford University Press, May 2007.

[3] A. Asuncion and D. Newman. UCI machine learning repository, 2007.

[4] J. Azimi, A. Fern, and X. Fern. Batch bayesian optimization via simulation matching. In *NIPS*, pages 109–117, 2010.

[5] M.-F. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. In *ICML*, pages 65–72, 2006.

[6] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[7] A. Beygelzimer, D. Hsu, J. Langford, and T. Zhang. Agnostic active learning without constraints. In *NIPS*, pages 199–207, 2010.

[8] O. Bousquet, S. Boucheron, and G. Lugosi. Introduction to statistical learning theory. In *Advanced Lectures on Machine Learning*, pages 169–207, 2003.

[9] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, Cambridge, 2004.

[10] Y. Chen and A. Krause. Near-optimal batch mode active learning and adaptive submodular optimization. In *International Conference on Machine Learning (ICML)*, 2013.

[11] D. A. Cohn, L. E. Atlas, and R. E. Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.

[12] S. Dasgupta, D. Hsu, and C. Monteleoni. A general agnostic active learning algorithm. In *NIPS*, 2007.

[13] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby. Selective sampling using the query by committee algorithm. *Machine Learning*, 28(2-3):133–168, 1997.

[14] Q. Gu, T. Zhang, C. H. Q. Ding, and J. Han. Selective labeling via error bound minimization. In *NIPS*, pages 332–340, 2012.

[15] A. Guillory and J. A. Bilmes. Label selection on graphs. In *NIPS*, pages 691–699, 2009.

[16] Y. Guo. Active instance sampling via matrix partition. In *NIPS*, pages 802–810, 2010.

[17] Y. Guo and D. Schuurmans. Discriminative batch mode active learning. In *NIPS*, 2007.

[18] S. Hanneke. Rates of convergence in active learning. *The Annals of Statistics*, 39(1):333–361, 2011.

[19] T. Hastie, R. Tibshirani, and J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction: with 200 full-color illustrations*. New York: Springer-Verlag, 2001.

[20] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu. Batch mode active learning and its application to medical image classification. In *ICML*, pages 417–424, 2006.

[21] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu. Semi-supervised svm batch mode active learning for image retrieval. In *CVPR*, 2008.

[22] S.-J. Huang, R. Jin, and Z.-H. Zhou. Active learning by querying informative and representative examples. In *NIPS*, pages 892–900, 2010.

[23] B. Korte and J. Vygen. *Combinatorial Optimization: Theory and Algorithms*. Springer Publishing Company, Incorporated, 4th edition, 2007.

[24] A. Krause, A. P. Singh, and C. Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9:235–284, 2008.

[25] P. Ravikumar, M. J. Wainwright, and J. D. Lafferty. High-dimensional ising model selection using ℓ1-regularized logistic regression. *The Annals of Statistics*, 38(3):1287C1319, 2010.

[26] A. I. Schein and L. H. Ungar. Active learning for logistic regression: an evaluation. *Machine Learning*, 68(3):235–265, 2007.

[27] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. In *ICML*, pages 999–1006, 2000.

[28] Z. Wang and J. Ye. Querying discriminative and representative samples for batch mode active learning. In *KDD*, pages 158–166, 2013.

[29] K. Yu, J. Bi, and V. Tresp. Active learning via transductive experimental design. In *ICML*, pages 1081–1088, 2006.

[30] T. Zhang and F. J. Oles. A probability analysis on the value of unlabeled data for classification problems. In *ICML*, 2000.