Selective sampling algorithms for cost-sensitive multiclass prediction

Alekh Agarwal ALEKHA@MICROSOFT.COM
Microsoft Research, New York NY

Abstract

In this paper, we study the problem of active learning for cost-sensitive multiclass classification. We propose selective sampling algorithms, which process the data in a streaming fashion, querying only a subset of the labels. For these algorithms, we analyze the regret and label complexity when the labels are generated according to a generalized linear model. We establish that the gains of active learning over passive learning can range from none to exponentially large, based on a natural notion of margin. We also present a safety guarantee to guard against model mismatch. Numerical simulations show that our algorithms indeed obtain a low regret with a small number of queries.

1. Introduction

The problem of active learning has received a lot of attention in the context of binary classification, both from a theoretical and an applied perspective. On the theoretical side, a series of works have studied a variety of efficient and inefficient methods with a small query complexity; an incomplete bibliography includes (Cohn et al., 1994; Dasgupta et al., 2007; Beygelzimer et al., 2009; 2010; Hanneke, 2011; Cesa-Bianchi et al., 2009; Dekel et al., 2010). In comparison, there has been relatively little theoretical work on the more general scenario of multiclass classification. Bulk of the work on multiclass active learning has been developed in computer vision, with a focus on scalable algorithms and empirical performance (see e.g. Yan et al., 2003; Jain & Kapoor, 2009; Joshi et al., 2012). Compelling applications also arise in other domains such as text and webpage categorization, computational biology (Luo et al., 2005) and more generally under the umbrella of structured output prediction problems (Roth & Small, 2006). However, little is known about the label complexity and error of these approaches. An interesting aspect of multiclass classification is that the desired criterion is often specified by a general *cost matrix* C. In such scenarios, we would like to further understand how the cost matrix influences our active querying strategy, and how its structure helps or hurts the loss and label complexity of active learning.

In this paper, we study cost-sensitive multiclass classification with a focus on efficient algorithms, as well as guarantees on the error and the label complexity. We build on the selective sampling framework for online active learning, pioneered in the binary setting by Cesa-Bianchi, Gentile and co-authors (Cesa-Bianchi et al., 2009; Orabona & Cesa-Bianchi, 2011; Dekel et al., 2010). In particular, we consider a generalized linear model (GLM) setting for multiclass classification. This is related to, but different from the multilabel setting of Gentile and Orabona (2012) where each label could occur independently, given a data point x. Our first contribution is to establish a connection between conditional probability estimation and cost-sensitive loss minimization. We also show how to obtain consistent conditional probability estimates (for the label to be i, given x). We further construct query rules that utilize these probability estimates in order to select which data points to query the labels for.

Our results bound the regret to the Bayes predictor (under the cost matrix) of our algorithm, as well as the label complexity for our query rules. These guarantees hold for a completely general (possibly adversarial) sequence of data vectors x_t, as long as our GLM assumption holds. We also pose a generalization of the Tsybakov margin condition (Tsybakov, 2004) from binary classification and establish fast rates for active multiclass learning under this condition. Our results show that the gains of active learning over passive are as good as exponential in the most favorable case where a *hard margin* is present between the conditional probabilities of the best and the second best class for each

Proceedings of the 30^{th} International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28. Copyright 2013 by the author(s).

data point. To our knowledge, these are the first such theoretical results for multiclass active learning.

Since our approach is based on online convex optimization, it lends itself to efficient algorithms. We also provide an easy technique to ensure that our algorithm would never do worse than random subsampling even under model mismatch, while performing much better in favorable scenarios. Finally, we complement our theoretical analysis with experimental evaluation in numerical simulations, where our methods do yield label complexity gains, and continue to be robust to model mismatch to a certain degree.

The remainder of this paper is organized as follows. In the next section, we describe our setup and assumptions. Section 3 presents our algorithm along with various query criteria. We describe our main results and their important consequences in Section 4, with simulation results in Section 5. Proofs of our results are deferred to the supplement.

2. Setup and assumptions

We start by describing the generative model we assume for multiclass classification problems along with some assumptions about the model.

2.1. Generalized linear models for cost-sensitive classification

We assume that we have a total of K classes, and the labels are generated based on a generalized linear model. Specifically, we assume that we have a weight matrix $W^* \in \mathbb{R}^{K\times d}$ with one weight vector per class. We further assume that $W^* \in \mathcal{W} \subseteq \mathbb{R}^{K\times d}$, for some convex set $\mathcal{W}$, with $W^*_K = 0$ wlog to avoid an overcomplete representation. Given a covariate $x \in \mathbb{R}^d$, we associate a label vector $y \in \mathbb{R}^K$ with an entry of 1 for the correct class and zeros elsewhere. Denoting the canonical basis vectors by $\{e_i \in \mathbb{R}^K\}$, we assume that the labels are generated according to the GLM

$$\mathbb{P}(y = e_i \mid W^*, x) = \langle \nabla\Phi(W^*x), e_i\rangle\,, \qquad (1)$$

where $\Phi(\cdot) : \mathbb{R}^K \mapsto \mathbb{R}$ is a convex function. In words, Φ is a function that takes a vector in $\mathbb{R}^K$ and maps it to a probability vector via its gradient. To get some intuition about this definition, consider the special case where $\mathbb{P}(y \mid W^*, x)$ is the canonical exponential family with sufficient statistics y. In this case, the function Φ corresponds to the log-partition function of the exponential family which is always convex (Lauritzen, 1996). As particular special cases, our family includes the multiclass logit model, as well as a linear noise model. We need some additional assumptions regarding the function Φ.

Assumption 1. *The function $\Phi(\cdot)$ is γ_ℓ-strongly convex, that is for all $u, v \in S \subseteq \mathbb{R}^K$, we have*

$$\Phi(u) \geq \Phi(v) + \langle \nabla\Phi(v), (u - v)\rangle + \frac{\gamma_\ell}{2}\|u - v\|_2^2. \quad (2)$$

In applications of the assumption, the set S will be picked so that the assumption is satisfied (with high probability) for all the vectors of form Wx with $W \in \mathcal{W}$ and $x \in \mathbb{R}^d$ (x drawn from underlying population). We also require an analogous upper bound.

Assumption 2. *The function $\Phi(\cdot)$ is γ_u-smooth, that is for all vectors $u, v \in S \subseteq \mathbb{R}^K$, we have*

$$\Phi(u) \leq \Phi(v) + \langle \nabla\Phi(v), (u - v)\rangle + \frac{\gamma_u}{2}\|u - v\|_2^2. \quad (3)$$

We also make one assumption regarding the set of predictors $\mathcal{W}$ and the data x.

Assumption 3. *$\forall x \in \mathcal{X}$, we have $\|x\|_2 \leq R$ and $\forall W \in \mathcal{W}$, we have $\|W^i\|_2 \leq \omega$ for all $i = 1, 2, \ldots, K$.*[1]

In particular, the assumption implies that our predictions $\langle W_i, x\rangle$ are bounded by $R\omega$ for each $i = 1, 2, \ldots, K$. Based on the above model, our methods will be defined in terms of the loss function

$$\ell(Wx, y) = \Phi(Wx) - y^T Wx. \qquad (4)$$

The motivation behind using this definition is that this loss function is calibrated for our noise model, meaning that for each x

$$\arg\min_W \mathbb{E}[\ell(Wx, y) \mid x] = W^*,$$

using our generative model (1). Assumptions 1 and 2 further imply that the loss is smooth and strongly convex as a function of the prediction vector Wx. We now describe a couple of concrete examples of our model to illustrate our assumptions.

2.2. Some motivating examples

Here we focus on examples of the probabilistic model (1) and the corresponding assumptions on the function Φ. We start with a multiclass logistic noise model and then describe a linear model.

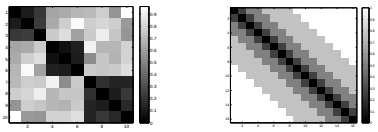
Example 1 (Multiclass logistic regression)**.** *The multiclass logistic model corresponds to choosing the function $\Phi(Wx) = \log(\sum_{i=1}^K \exp(x^T W^i))$. This gives rise to the conditional probability model*

$$\mathbb{P}(Y = i \mid W, x) = \frac{\exp(x^T W^i)}{\sum_{j=1}^K \exp(x^T W^j)},$$

which is the well-known multinomial logit model. It is easily checked that the loss function (4) for this setting is the multiclass logistic loss $\log(1+\sum_{i\neq y} \exp(x^T W^i - x^T W^y))$. For this setting, we assume that Assumption 3 is satisfied with $\omega = R = 1$. With these bounds, it can be checked that the function Φ satisfies Assumptions 1 and 2 with constants $1/(eK^2)$ and 1, resp. [2]

[1] $W^i \in \mathbb{R}^d$ is the i_{th} row of W.

[2] Strong convexity can be improved by rescaling the loss to instead use $\exp(x^T W^i/\sigma)$ for some $\sigma > 0$.



(a) (b)

Figure 1. Examples of structured cost matrices (see text)

Example 2 (Multiclass linear regression)**.** *Unlike the multiclass logistic case, there is no standard definition for a multiclass linear model. We consider*

$$\mathbb{P}(Y = i|W, x) = x^T W^i - (\sum_{j=1}^{K} x^T W^j - 1)/K.$$

The induced probabilities are non-negative assuming $x^T W^i - x^T W^j \leq 1/K$ *for all* $i \neq j$*, and they always add up to 1. This is also the natural generalization of the linear model for binary classification (Cesa-Bianchi et al., 2009). The induced function* Φ *for this case is*

$$\sum_{i=1}^{K} (x^T W^i)^2/2 - \big(\sum_{j=1}^{K} x^T W^j - 1\big)^2/(2K).$$

It is easily checked that Assumptions 1 and 2 are satisfied with constants $1 - 1/K$ *and* 1 *respectively.*

2.3. Cost-sensitive multiclass classification

In the problem of cost-sensitive multiclass classification, we are given a cost matrix $C \in \mathbb{R}^{K \times K}$ with non-negative entries and zeros on the diagonal. These assumptions are without loss of generality. Here $C(i, j)$ is the cost of predicting j when the true label is i. The simplest example of a cost matrix is the one corresponding to the 0/1 loss for multiclass classification:

$$C(i, j) = \begin{cases} 0 & \text{if } i = j \\ 1 & \text{otherwise} \end{cases} . \tag{5}$$

However, the more general setting allows us to penalize mistakes involving different class pairs differently. For instance, one could imagine a block-structured matrix with zeros on the diagonal blocks (Fig. 1(a)). This corresponds to groups of similar classes, with no penalty for mistakes within a group and a high penalty for mistakes across groups. Another example is a tree structured cost matrix, where the classes are organized into a tree hierarchy (e.g. in hierarchical classification) and the cost of a mistake is the tree-distance between the two classes (Fig. 1(b)).

Given such a cost-matrix, the quality of a prediction $\hat{y}$ for a point x is measured by the expected cost:

$$\mathbb{E}[C(Y, \hat{y}) \mid x] = \sum_{i=1}^{K} C(i, \hat{y})(\nabla\Phi(W^* x))_i.$$

In the sequel, we will measure the performance of our algorithms in the regret to the best weight matrix W^*, as measured by this expected cost-sensitive loss.

3. Selective sampling for multiclass classification

In this section we present our algorithms for the cost-sensitive multiclass classification setting. We first present an algorithm for an arbitrary choice of a query function. We then give concrete examples of query functions that we consider in our work.

3.1. Algorithm

Our algorithms build on a growing body of work on selective sampling algorithms for online active learning by Cesa-Bianchi, Gentile and co-authors (Cesa-Bianchi et al., 2009; Orabona & Cesa-Bianchi, 2011; Dekel et al., 2010). In order to describe the algorithm, we need some additional notation.

Given a weight matrix W and a data point x, it will be convenient to define the score of a class i as

$$S^x_W(i) = \sum_{j=1}^{K} (\max_{a,b} C(a, b) - C(j, i))(\nabla\Phi(Wx))_j. \tag{6}$$

In the simpler setting with the 0/1 multiclass loss, we see that $S^x_W(i) = (\nabla\Phi(Wx))_i$. We start with an easy lemma.

Lemma 1. *Given a cost matrix* C*, suppose the class conditional probabilities follow the probabilistic model* (1) *based on a weight matrix* W^**. Then the Bayes optimal classifier predicts as* $\arg\max_i S^x_{W^*}(i)$*.*

This intuition will be important in going from scores to predictions in our algorithm. Before describing the algorithm, we mention a couple of more important notations. We use the indicator variables $Z_t \in \{0, 1\}$ to indicate whether the label was queried at time t or not. Given $\gamma > 0$, we define the matrix

$$M_t = \sum_{s=1}^{t-1} Z_s x_s x_s^T + \frac{\gamma}{\gamma_\ell} I. \tag{7}$$

At time t, we denote the history of past x's and the queried labels as H_t. Formally,

$$H_t = \{x_s \ : \ 1 \leq s < t \text{ and } y_s \ : \ Z_s = 1\} \tag{8}$$

In Algorithm 1, we describe a generic algorithmic template that takes a query function $Q \ : \ \mathcal{X} \times \{\mathcal{X} \times \{1, 2, \ldots, K\}\}^{t-1} \mapsto \{0, 1\}$ and queries y_t if $Q(x_t, H_t) = 1$. We give examples of the query function after presenting the general algorithm.

The update rule (9) does not use the cost matrix because our algorithm is based on consistent conditional probability estimation under the generative model (1). The update rule (9) estimates a weight matrix W_t which is close to W^*, which is then mapped to a prediction as in Lemma 1.

Algorithm 1 CS-Selectron algorithm for selective sampling in cost-sensitive multiclass classification

Require: Query function Q, regularization parameter $\gamma > 0$ and cost matrix C.
Initialize $W_1 = 0$, $M_1 = \gamma I/\gamma_\ell$.
for all time steps $t = 1, 2, \ldots, T$ **do**
Observe instance $x_t \in \mathcal{X}$, $H_{t+1} = H_t \cup \{x_t\}$.
Predict $\hat{y}_t$ as $\arg\max_{i=1,2,\ldots,K} S^{x_t}_{W_t}(i)$.
if $Q(x_t, H_t) = 1$ **then**
Query label y_t
Update $Z_t = 1$, $H_{t+1} = H_{t+1} \cup \{y_t\}$ and $M_{t+1} = M_t + x_t x_t^T$.
Update W_t according to the rule

$$W_{t+1} = \arg\min_{W \in \mathcal{W}} \left\{ \sum_{s=1}^{t} Z_s \ell(W x_s, y_s) + \gamma \|W\|_F^2 \right\}. \tag{9}$$

end if
end for

Before we move on to discuss the query functions, we will make some remarks about the computational properties of Algorithm 1. The algorithm might seem computationally challenging since it requires us to solve a loss minimization problem over all the queried samples at each step. This is not an issue, however, since warm-start at the previous solution is a fairly good guess in most cases. Indeed, the most expensive step of our algorithm is not the update rule (since it only occurs when we query), but the computation of the quadratic form $x_t M_t^{-1} x_t$ at each step t, which will be used in all our query criteria. While this computation seems unavoidable to us at this time, it seems possible to use approximate SVD computations using ideas from randomized linear algebra (Halko et al., 2011; Clarkson & Woodruff, 2009) which exploit the low-rank structures common to natural datasets.

3.2. Query functions

There have been different query functions that have been considered in previous works on selective sampling in the binary classification setting and we describe their multiclass variants below. In order to define the criteria, we need define some additional notation. We define the following quantities of interest:

$$y_t^* = \arg\max_{i=1,\ldots,K} S^{x_t}_{W^*}(i), \quad y_t^{'} = \arg\max_{i \neq y_t^*} S^{x_t}_{W^*}(i)$$

$$\hat{y}_t = \arg\max_{i=1,\ldots,K} S^{x_t}_{W_t}(i), \quad y_t^{''} = \arg\max_{i \neq \hat{y}_t} S^{x_t}_{W_t}(i). \tag{10}$$

In words, y_t^* and $y_t^{'}$ are the optimal and second-best classes as per the true weight matrix W^*. $\hat{y}_t$ and $y_t^{''}$ are our best estimates of these classes based on our weight matrix W_t. We now define our query rules. We will use $\mathbb{1}\{A\}$ to denote the indicator of an event A.

- **BBQ selection rule:** This rule was introduced in the work of Cesa-Bianchi et al. (2009):

$$Q(x_t, H_t) = \mathbb{1}\left\{ \|x_t\|^2_{M_t^{-1}} \geq t^{-\kappa} \right\}, \tag{11}$$

for some $\kappa \in (0, 1)$. This rule turns out to be applicable in the multiclass setting as is. The intuition behind this rule is that if the current data point x_t is captured well by the linear span of the previously queried data points, then we can make a confident prediction regarding the label y_t. The exponent κ is the parameter that governs the trade-off between the number of queries the algorithm makes and the regret it incurs.

- **BBQ$_\epsilon$ selection rule:** This rule is a slight modification of the BBQ query criterion, and uses a query function

$$Q(x_t, H_t) = \mathbb{1}\left\{ \eta_\epsilon \|x_t\|^2_{M_t^{-1}} \geq \epsilon^2 \right\}, \tag{12}$$

where $\eta_\epsilon > 0$ is a function dependent on C and Φ which controls the distance between W_t and W^*, to be specified later. ϵ is a parameter of the algorithm. The intuition behind this rule is that at the rounds where we don't query, we will be guaranteed that the difference between predictions of W_t and W^* on x_t is at most ϵ whp.

- **DGS selection rule:** This query criterion is a modification of a rule that was proposed in the work of Dekel et al. (2010) in the context of binary classification, and takes not only the previous covariates, but also the observed labels y_s into account. The query function for this criterion in the multiclass setting is

$$\mathbb{1}\left\{ S^{x_t}_{W_t}(\hat{y}_t) - S^{x_t}_{W_t}(y_t^{''}) \leq \eta_{DGS} \|x_t\|_{M_t^{-1}} \right\}. \tag{13}$$

The intuition behind this rule is that on the rounds where we do not query the label y_t, we are guaranteed (whp) that either $\hat{y}_t = y_t^*$, or the regret is small.

4. Main results and their consequences

In this section we state the main results regarding the performance and the query complexity of Algorithm 1, and obtain some illustrative corollaries. We conclude the section with a safety guarantee for scenarios where our modeling assumption (1) is not valid.

4.1. Regret and label complexity

At a high level, we will demonstrate that the average regret of our algorithm vanishes with the number of queries N_T at a rate which adapts to the hardness of the problem. In the worst case, the rate is

$\widetilde{O}(1/\sqrt{N_T})$, which is also achieved by random subsampling. In the best case, our average regret vanishes exponentially fast in N_T, while random subsampling can only achieve an error of $\widetilde{O}(1/N_T)$ in this case (Daniely et al., 2011). An extension of Tsybakov's margin condition (Tsybakov, 2004) allows for a smooth interpolation between the two extremes, yielding rates that are $\widetilde{O}(N_T^{-(1+\alpha)/2})$ as α ranges from 0 (noisy) to ∞ (hard-margin).

In order to define regret, we recall our earlier definition (8) of H_t and further define $\mathcal{F}_t = \sigma\{H_t \cup x_t\}$. In words, $\mathcal{F}_t$ is the sigma field generated by $x_1, \ldots, x_t$ along with all the labels we have seen before round t. Our results are stated in terms of the regret:

$$R_T = \sum_{t=1}^{T} \left(\mathbb{E}[C(Y_t, \hat{y}_t) \mid \mathcal{F}_t] - \mathbb{E}[C(Y_t, y_t^*) \mid \mathcal{F}_t]\right) \quad (14)$$

Observe that the regret is incurred on each round, regardless of whether we query or not. Our results will involve the following quantity which counts the number of *hard* to classify points, modulated at a level ϵ

$$T_\epsilon = \{1 \le t \le T \ : \ S^{x_t}_{W^*}(y_t^*) - S^{x_t}_{W^*}(y_t^{'}) \le \epsilon\}. \quad (15)$$

For any class i, we define the average cost as $\bar{C}_i = \sum_j C(j,i)/K$ and the column-variation in the costs as

$$\sigma^2(C) = \max_{i=1,2,\ldots,K} \sum_{j=1}^{K} (C(j,i) - \bar{C}_i)^2. \quad (16)$$

This definition captures the variation of the cost matrix, making it invariant to adding a constant to each column of the cost matrix. We also use the shorthand

$$\psi(C, \Phi) = \sigma^2(C)\gamma_u^2/\gamma_\ell^2, \quad (17)$$

which will capture our dependence on the cost matrix and the link function Φ. With this notation, we can now state our main results. We start with a result for the BBQ_ϵ query criterion. We do not give any results for the BBQ criterion, but similar guarantees can be obtained by combining our techniques with the previous works of Cesa-Bianchi et al. (2009; 2011).

For ease of presentation of our results, let us define

$$\theta_t = \frac{8\sqrt{dK}}{\gamma_\ell}\sqrt{\log\left(1+\frac{2R^2\gamma_\ell}{\gamma}\right)\log\frac{dKt}{\delta}} + \sqrt{\frac{2\gamma\omega^2}{\gamma_\ell}}. \quad (18)$$

In the first theorem, we use BBQ_ϵ rule with $\eta_\epsilon = 4\sigma^2(C)\gamma_u^2\theta_t^2$.

Theorem 1 (BBQ_ϵ rule)**.** *Suppose we receive labels generated according to the model* (1) *and Assumptions 1-3 are satisfied. Suppose we run Algorithm 1 with the BBQ_ϵ query criterion using some $\epsilon > 0$ and $\gamma = \gamma_\ell$. Then, for $T \ge 3$ and $0 < \delta < 1/e$, with probability $1 - 2\delta$ the regret is at most*

$$R_T = \widetilde{O}\left(\epsilon T_\epsilon + \psi(C,\Phi)\,\frac{d}{\epsilon}\log\frac{1}{\delta}\right).$$

The number of queries made is at most

$$N_T = \widetilde{O}\left(\psi(C,\Phi)\,\frac{d^2K}{\epsilon^2}\right)$$

A qualitatively similar result also holds for the DGS criterion. In this case we use $\eta_{DGS} = 2\sigma(C)\gamma_u\,\theta_t$.

Theorem 2 (DGS rule)**.** *Under conditions of Theorem 1, suppose we run Algorithm 1 with the cost-sensitive DGS criterion. Then, for $T \ge 3$ and $0 < \delta < 1/e$, with probability $1 - 2\delta$ the regret is at most*

$$R_T = \widetilde{O}\left(\inf_{\epsilon>0}\left\{\epsilon T_\epsilon + \psi(C,\Phi)\,\frac{d}{\epsilon}\log\frac{1}{\delta}\right\}\right),$$

For any $\epsilon > 0$, with probability at least $1 - \delta$, the number of queries made is at most

$$N_T = \widetilde{O}\left(T_\epsilon + \psi(C,\Phi)\,\frac{d^2K}{\epsilon^2}\right)$$

Observe that ϵ is a parameter of the algorithm in Theorem 1, but a free parameter in Theorem 2. A few remarks about these results are in order.

(a) We reiterate that the above results hold for an arbitrary sequence x_t, much like earlier results on selective sampling (Orabona & Cesa-Bianchi, 2011). In order to interpret the results, we observe that setting $T_\epsilon = T$ and optimizing over ϵ yields a regret of $\widetilde{O}(1/\sqrt{T})$ and $\widetilde{O}(N_T)$—recovering the passive learning results. However, for nicer problems with $T_\epsilon = o(T)$ for ϵ small enough, we expect strict improvements in label complexity.

(b) We expect a similar result to hold for an update rule where we just do an Online Newton Step (Hazan et al., 2007) instead of our current rule (9), by combining the techniques of Gentile & Orabona (2012) with our results.

(c) An important assumption in the implementation of Algorithm 1 is that we know the correct link function in order to pick the right loss function. This is currently a limitation of our theory, but the algorithm is stable to small perturbations. That is, if $\mathbb{E}[Y \mid x]$ is close to $\nabla\Phi$ for some function Φ in a pointwise sense, then it suffices to use the loss function defined by Φ.

In order to discuss concrete examples of the benefits of active learning, we now focus on the setting of *i.i.d.* x's. In binary classification problems, one assumption that helps to capture the benefits of active learning is the Tsybakov noise condition (Tsybakov, 2004) which governs the fraction of data that lies close to the classification boundary. We now describe a multiclass version of this assumption, and then provide improved regret guarantees under this assumption.

Assumption 4 (Multiclass Tsybakov noise condition)**.** *We say that a distribution $\mathbb{P}$ over $\mathbb{R}^d$ satisfies*

the multiclass Tsybakov noise condition with parameters (ϵ_0, α, c) *for some* $\epsilon_0 > 0$ *and* $\alpha \geq 0$ *if for all* $0 \leq \epsilon \leq \epsilon_0$,

$$\mathbb{P}\left(S^X_{W^*}(y^*(X)) - S^X_{W^*}(y^{'}(X)) \leq \epsilon\right) \leq c\,\epsilon^\alpha.$$

In words, the fraction of points where the scores of the best and the second best classes are within ϵ is at most $c\,\epsilon^\alpha$. In the special case of 0/1 loss, this yields the following more intuitive condition. For all $0 \leq \epsilon \leq \epsilon_0$, $\mathbb{P}\left((\nabla\Phi(W^*X))_{y^*(X)} - (\nabla\Phi(W^*X))_{y'(X)} \leq \epsilon\right) \leq c\,\epsilon^\alpha$. That is, we control the fraction of points x where the probabilities of the best and the second-best class are closer than ϵ at a level $c\epsilon^\alpha$. In particular, $\alpha = 0$ is a tautology for $c = 1$, while $\alpha \to \infty$ imposes a hard margin of size ϵ_0. This is analogous to controlling the difference $|\mathbb{P}(y = 1 \mid x) - 0.5|$ in the binary case, and provides the natural extension of the Tsybakov noise condition from the binary classification case. An immediate consequence of the assumption is that we obtain $T_\epsilon = \widetilde{O}(T\epsilon^\alpha)$ for all $\epsilon \leq \epsilon_0$, both in expectation and with high probability (Dekel et al., 2010). Under the assumption, we can obtain the following simplified corollaries of our earlier results.

Corollary 1. *Under conditions of Theorem 2, assume further that the covariate sequence is drawn* i.i.d. *according to a distribution that satisfies Assumption 4. Then with probability at least* $1-2\delta$*, the average regret of Algorithm 1 with the DGS query criterion is at most*

$$\frac{R_T}{T} = \widetilde{O}\left(\left(\psi(C,\Phi)\,\frac{d}{T}\right)^{\frac{1+\alpha}{2+\alpha}}\right).$$

With probability at least $1-\delta$*, the number of queries is at most* $N_T = \widetilde{O}\left(T^{\frac{2}{2+\alpha}}\left(\psi(C,\Phi)\,d^2K\right)^{\frac{\alpha}{2+\alpha}}\right)$.

A similar result also holds for the BBQ_ϵ query rule. From the result, we can see that as $\alpha \to \infty$, N_T approaches $\mathcal{O}(\log T)$ and the average regret approaches $\mathcal{O}(1/T)$, which is the best possible scaling in T even if we query all T labels (Daniely et al., 2011). In order to further understand the gains of active learning in such low noise problems, it is instructive to study the average regret as a function of the number of queries made. Doing so, we obtain the following corollary.

Corollary 2. *Under conditions of Corollary 1, we have the following with probability at most* $1-2\delta$*.*

(a) For the BBQ_ϵ *rule with* $\epsilon^* = \left(\frac{d\psi(C,\Phi)}{T}\right)^{1/(\alpha+2)}$*, assuming* $\epsilon^* \leq \epsilon_0$*, the average regret satisfies*

$$\frac{R_T}{T} = \widetilde{O}\left(\left(\psi(C,\Phi)\,\frac{d^2K}{N_T}\right)^{\frac{1+\alpha}{2}}\right).$$

(b) For the DGS rule, the average regret satisfies

$$\frac{R_T}{T} = \widetilde{O}\left(d^{\frac{(1+\alpha)^2}{(2+\alpha)}}K^{\frac{\alpha(1+\alpha)}{2(2+\alpha)}}\left(\frac{\psi(C,\Phi)}{N_T}\right)^{\frac{1+\alpha}{2}}\right),$$

assuming $\epsilon^* \leq \epsilon_0$*.*

In terms of the scaling of the average regret with respect to the number of queries, both the methods achieve a guarantee of $N_T^{-\frac{(1+\alpha)}{2}}$, which is known to be optimal under Assumption 4 in the binary classification setting (Castro & Nowak, 2008). In particular, as $\alpha \to \infty$, the average regret decays exponentially in N_T, meaning we query only $\mathcal{O}(\log T)$ labels. This behavior is similar to the selective sampling algorithms for binary classification (Dekel et al., 2010; Orabona & Cesa-Bianchi, 2011). A crucial difference between the two parts of the corollary is that while BBQ_ϵ needs knowledge of the noise level in setting the parameter ϵ, the DGS query rule adapts to it.

4.2. Conclusions for specific cost matrices

In order to better understand our results, we now specialize to the case of specific cost matrices, providing concrete values of $\sigma^2(C)$.

0/1 multiclass loss: In this special case, the cost matrix takes the form (5). It is easy to check that the parameter $\sigma^2(C) = 1 - 1/K$ in this case.

This immediately yields bounds on regret and query complexity for our algorithms in the multiclass 0/1 loss scenario, as corollaries of our Theorems. In order to better understand the scalings with respect to the dimension d and the number of classes K, we observe from Corollary 1 that our regret bound takes the form

$$\frac{R_T}{T} = \widetilde{O}\left(\epsilon^{1+\alpha} + \frac{d\gamma_u^2}{\gamma_\ell^2\epsilon T}\right) = \widetilde{O}\left(\left(\frac{d\gamma_u^2}{\gamma_\ell^2 T}\right)^{\frac{1+\alpha}{2+\alpha}}\right),$$

where the second equality optimizes for the best ϵ. It might seem at the first glance that our rates are completely independent of K. However, that is not the case in general. The condition number of the Hessian introduced through the ratio γ_u/γ_ℓ can often depend on K (such as in Example 1). Understanding optimal scalings with respect to d and K remains an interesting question for future research.

Block structured cost matrix: We consider a simple version of the block-structured cost matrix example illustrated in Fig. 1(a). Suppose that our cost matrix consists of r blocks, each of size K/r. The cost matrix is zero on the diagonal blocks corresponding to the groups, and identically 1 on the off-diagonal blocks. In this case, it is easily checked by a direct calculation that $\sigma^2(C) = K/r(1-1/r)$.We see that we do not incur any substantial costs if we have a large number of small, homogeneous groups. In contrast, a small number of large, homogeneous groups can force an additional factor of $\mathcal{O}(K)$ in our results. This not just an artifact of our analysis, but seems like an actual prob-

lem case for Algorithm 1. For large groups, we still estimate the probabilities for individual classes (to exploit the GLM assumption), but predict based on sum of class probabilities over a large group which has an error potentially larger by a factor of K/r.[3]

Tree structured cost matrix: This is the setting illustrated in Fig. 1(b). We assume that our K classes are arranged at the leaves of a tree. The cost of misclassification is the tree distance between the two classes. In this case, a direct calculation reveals that $\sigma^2(C) = O(K)$. This is expected given our previous example, since a tree can be thought of as having a small number of large heterogeneous groups.

Overall, we see that in some cases where we can leverage the structure of the cost matrix, while in others we cannot. It is our intuition that just structure on the cost matrix is not sufficient to reduce the complexity of the problem, without corresponding structure on the weight matrix. When such a structure is present, we expect our method to be able to leverage it through the use of regularization, or using the set $\mathcal{W}$.

4.3. A safety guarantee

Robustness to model mismatch is a crucial concern, as the consequences of model mismatch can be quite catastrophic in selective sampling. Our algorithm learns over a biased subsample from the underlying distribution and when our model is incorrect, the error we minimize over this biased subsample may no longer reflect the error under the true distribution.[4] Importance weighted algorithms (Beygelzimer et al., 2009; 2010) do work with an unbiased distribution, but the extent of label complexity savings from these approaches in our setting—when minimizing a surrogate loss in a multiclass scenario—is not clear.

We now suggest a partial fix to model mismatch, by querying an additional N_T labels, whenever the algorithm was going to query N_T labels. The idea is to run an independent passive learning algorithm on a purely random subsample of size N_T. Let us denote this subsample by S and its size by N. This can be achieved, for instance, by also querying the label of x_{t+1} whenever our algorithm recommends to query x_t. We now run a low-regret algorithm on S and measure its cumulative prediction loss (in the surrogate loss (4)) on this subsample. Let us denote the iterates generated in this process by $\widehat{W}_t$. At the same time, we also measure the prediction loss of our active learner on the subsample. Now standard arguments (such as those used in the proof of Lemma 6 in the appendix) can be used to guarantee that with probability at least $1 - 4\delta \log T$

$$0 \leq \frac{1}{N}\sum_{i\in S}(\mathbb{E}\ell(\widehat{W}_i;(x,y)) - \min_{W\in\mathcal{W}} \mathbb{E}\ell(W;(x,y))$$
$$\leq \frac{R^1_\ell}{N} + c\left[\frac{d}{N\gamma_\ell}\log\left(\frac{R^2\gamma_\ell N}{\gamma}+1\right) + \frac{R\omega}{N\gamma_\ell}\log\frac{1}{\delta}\right],$$

where R^1_ℓ is the cumulative regret in the loss function ℓ of the iterates $\widehat{W}_i$ on the sample S. A similar claim can also be made for the active learning algorithm, replacing $\widehat{W}_i$ by W_i and R^1_ℓ by R^2_ℓ.

Based on these bounds, we now check the condition

$$\frac{R^2_\ell}{N} \geq \frac{R^1_\ell}{N} + c\left[\frac{d}{N\gamma_\ell}\log\left(\frac{R^2\gamma_\ell T}{\gamma}+1\right) + \frac{R\omega}{N\gamma_\ell}\log\frac{1}{\delta}\right].$$

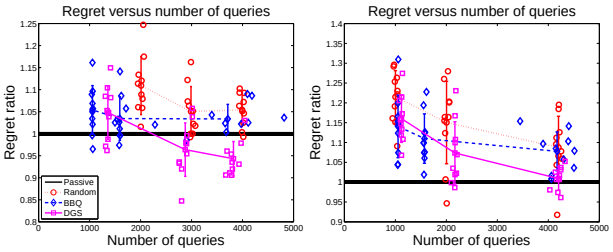
When this holds, we are guaranteed that the (average) expected risk of our active learning iterates is larger than that of the random subsampling approach. In that case, we pick the solution resulting from random subsampling. This guarantees that we never do worse than a constant factor of random subsampling but can still do much better when the model assumptions are correct. We note that this is not a safeguard specific to our method, and can be used with any sequential active learning algorithm. Of course, having better guarantees without our model assumptions is an active area of research.

5. Numerical simulations

In this section, we describe results from some evaluation of our algorithms on synthetic data. We evaluated three query strategies: DGS, BBQ and Random. In all our experiments, we generated i.i.d. x's from a mixture of Gaussians distribution in $\mathbb{R}^{1000}$. We picked random vectors as the means for each Gaussian, in a way that ensured that the different clusters have a non-trivial overlap in order to ensure adequate noise in the classification problem (details in the supplement). We also set W^*_i to the corresponding Gaussian means, and generated labels y according to our noise model (1). We evaluated each query criterion for number of classes $K = 5$ and $K = 10$. For each criterion, we picked the parameters of the rule so that they query roughly the same number of points. Note that the DGS rule as stated is parameter-free, but we instead used the DGS-MOD version of Orabona & Cesa-Bianchi (2011), which allows a general multiplier on the RHS of the rule (13). All our algorithms used the multiclass logistic loss in the update rule (9). We used the 0/1 cost matrix in all our experiments.

[3] We suspect this is unavoidable using a GLM, unless the weight matrix W^* has a structure aligned with the cost matrix.

[4] This is also a problem with previous selective sampling approaches.



$K = 5$ $K = 10$

Figure 2. Plots showing the ratio of active to passive regret, as a function of the number of queries (see text).

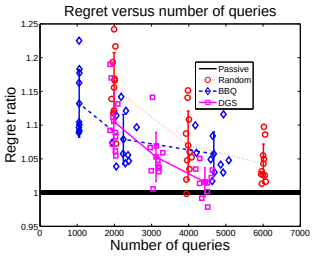


Figure 3. Plot showing the ratio of active to passive regret, as a function of the number of queries in a model mismatch scenario (see text for details). While the regret ratio does not increase by much, the actual regret was substantially higher than the correct model case both for active and passive.

Figure 2 shows the results of our simulations for $K = 5$ and $K = 10$. In each case, we had a total of $10,000$ data points. We have plotted the ratio of the cumulative regret from each approach to the regret attained by passive learning on all $10,000$ samples. The results are averaged over 20 trials. In the plots, we show the mean regret ratio and confidence intervals, at the point on the X-axis which is the mean of the number of queries with a particular parameter setting. We also plot the individual points to give the reader an idea of the spread in the number of queries as well as in the regret ratio[5]. We observe that DGS rule does the best in both the cases, beating even the passive learner with a smaller query complexity! We speculate that this is because training over fewer (but most informative) labeled samples is less prone to noise and yields better generalization for our methods. We also note that the strong performance of Random was somewhat surprising, even though DGS eventually outperforms it. We believe that this is due to the fact that our simulated data does not have a situation where there are only a few informative points close to the boundary. That is the kind of setting where a good active learning strategy stands to gain the most over random subsampling. Overall, we observe that our algorithms are indeed able to attain a small regret ratio, even at a subsampling level of 10% or 20%, which is certainly encouraging and in line with the theoretical results.

As remarked in Section 4, model mismatch can be a concern for our algorithms. To see the impact of this, we did an experiment where the probability of class i was proportional to $(x^T W_i^*)^2$, but we continued to use the multiclass logistic loss. Figure 3 shows the results of this experiment. While the relative regret is now closer to random subsampling, we are still doing no worse. This gives some reassurance about our robustness to model perturbations, and it would be interesting to do a detailed study in future work.

[5]Larger versions of these plots are in the supplement.

6. Discussion

In this paper, we present algorithms for selective sampling in cost-sensitive multiclass classification. Our algorithm and query criteria provide natural generalizations of previous works in the binary setting. We provide guarantees on the regret and label complexity of our approach, under probabilistic assumptions on the noise. We also introduce a notion of problem hardness in form of the multiclass Tsybakov condition, which provides a sufficient condition for active learning to gain over passive learning. Under this condition, our label complexity gains can be as large as exponential, which mirrors the binary case.

There are several interesting avenues for future work, some of which we outline here. As remarked earlier, our algorithm admits an arbitrary convex constraint set $\mathcal{W}$, which can be allowed to add information regarding the problem structure, such as group norms or low-rank structures (Harchaoui et al., 2012). It would be interesting to study the impact of this structure, both in theory and experiments. Another important direction is to understand how the probabilistic assumption can be relaxed further, without going to computationally intractable algorithms. On a more practical side, it seems natural to use approximations to speed up the computation of the quadratic form $x_t^T M_t^{-1} x_t$, which seems to be the most computationally expensive step for us.

Acknowledgements

The author wishes to thank Siva Balakrishnan, Miro Dudík, Dean Foster and John Langford for helpful comments on an earlier draft, and Karthik Sridharan, Nicolò Cesa-Bianchi, Claudio Gentile and Francesco Orabona for helpful discussions.

References

Beygelzimer, A., Dasgupta, S., and Langford, J. Importance weighted active learning. In *ICML*, 2009.

Beygelzimer, A., Hsu, D., Langford, J., and Zhang, T. Agnostic active learning without constraints. In *NIPS*, 2010.

Castro, R.M. and Nowak, R.D. Minimax bounds for active learning. *Information Theory, IEEE Transactions on*, 54(5):2339 –2353, 2008.

Cesa-Bianchi, N., Gentile, C., and Orabona, F. Robust bounds for classification via selective sampling. In *ICML*, pp. 121–128, 2009.

Clarkson, K. L. and Woodruff, D. P. Numerical linear algebra in the streaming model. In *STOC*, 2009.

Cohn, D., Atlas, L., and Ladner, R. Improving generalization with active learning. *Machine Learning*, 15:201–221, 1994.

Daniely, A., Sabato, S., Ben-David, S., and Shalev-Shwartz, S. Multiclass learnability and the erm principle. *Journal of Machine Learning Research - Proceedings Track*, 19:207–232, 2011.

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

Dekel, O., Gentile, C., and Sridharan, K. Robust selective sampling from single and multiple teachers. In *COLT*, pp. 346–358, 2010.

Filippi, S., Cappe, O., Garivier, A., and Szepesvari, C. Parametric bandits: The generalized linear case. *NIPS*, 2010.

Freedman, D. A. On tail probabilities for martingales. *The Annals of Probability*, 3(1):100–118, February 1975.

Gentile, C. and Orabona, F. On multilabel classification and ranking with partial feedback. In *NIPS*, 2012.

Halko, N., Martinsson, P. G., and Tropp, J. A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, 53(2):217–288, 2011.

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

Harchaoui, Z., Douze, M., Paulin, M., Dudik, M., and Malick, J. Large-scale image classification with trace-norm regularization. In *CVPR*, pp. 3386 – 3393, 2012.

Hazan, E., Agarwal, A., and Kale, S. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69:169–192, 2007.

Jain, P. and Kapoor, A. Active learning for large multi-class problems. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 762 –769, 2009.

Joshi, A.J., Porikli, F., and Papanikolopoulos, N.P. Scalable active learning for multiclass image classification. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(11):2259 –2273, 2012.

Kakade, S. M. and Tewari, A. On the generalization ability of online strongly convex programming algorithms. In *Advances in Neural Information Processing Systems 21*, 2009.

Kalai, A. and Vempala, S. Efficient algorithms for online decision problems. *J. Comput. Syst. Sci.*, 71 (3):291–307, 2005.

Lauritzen, S. L. *Graphical Models*. Oxford University Press, Oxford, 1996.

Luo, T., Kramer, K., Goldgof, D. B., Hall, L. O., Samson, S., Remsen, A., and Hopkins, T. Active learning to recognize multiple types of plankton. *Journal of Machine Learning Research*, 6:589–613, 2005.

Orabona, F. and Cesa-Bianchi, N. Better algorithms for selective sampling. In *ICML*, pp. 433–440, 2011.

Roth, D. and Small, K. Active learning with perceptron for structured output. In *ICML 2006 Workshop on Learning in Structured Output Spaces*, 2006.

Shalev-Shwartz, Shai. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2), 2012.

Tsybakov, A. B. Optimal aggregation of classifiers in statistical learning. *Ann. Statist.*, 32:135–166, 2004.

Yan, R., Yang, Jie, and Hauptmann, A. Automatically labeling video data using multi-class active learning. In *ICCV*, 2003.