Better Algorithms for Selective Sampling

Francesco Orabona FRANCESCO@ORABONA.COM
Nicolò Cesa-Bianchi NICOLO.CESA-BIANCHI@UNIMI.IT
DSI, Università degli Studi di Milano, Italy

Abstract

We study online algorithms for selective sampling that use regularized least squares (RLS) as base classifier. These algorithms typically perform well in practice, and some of them have formal guarantees on their mistake and query rates. We refine and extend these guarantees in various ways, proposing algorithmic variants that exhibit better empirical behavior while enjoying performance guarantees under much more general conditions. We also show a simple way of coupling a generic gradient-based classifier with a specific RLS-based selective sampler, obtaining hybrid algorithms with combined performance guarantees.

1. Introduction

Online selective sampling is an active variant of online learning in which the learner is allowed to adaptively subsample the labels of an observed sequence of feature vectors. The learner's goal is to achieve a good trade-off between mistakes rate and number of sampled labels. This can viewed as an abstract protocol for interactive learning applications. For example, a system for categorizing stories in a newsfeed asks for human supervision whenever it feels that more training examples are needed to keep the desired accuracy.

Linear classifiers lend themselves well to selective sampling settings. The margin of the classifier on the current instance can be viewed as a measure of confidence for the classification of the instance's label. If this confidence is deemed too low, then the selective sampler queries the label and uses it, along with the instance, to adjust the underlying linear model. The selective sampler performance is evaluated both in terms of predictive accuracy over the entire sequence of examples and in terms of the overall number of sampled labels.

In this work we consider kernel-based linear classifiers that compute their margins using the regularized least squares (RLS) estimate. When used for selective sampling, these algorithms offer a good compromise between theoretical robustness and empirical performance. More specifically, we focus on two RLS-based selective samplers: BBQ (Cesa-Bianchi et al., 2009) and the single teacher version of the algorithm proposed in (Dekel et al., 2010), which we call DGS. As far as we know, these are the only two examples of efficient online algorithms that offer simultaneous guarantees on mistake and query rate without stringent assumptions on the data process (such as i.i.d. or linear separability). Both algorithms make the same stochastic assumption on the process generating labels: the Bayes classifier belongs to the RKHS induced by the kernel with which the algorithm is run. On the contrary, no assumptions are made on the generation of the instances sequence[1]. BBQ and DGS base their query decisions on a quantity, called r_t, which estimates the variance of the RLS margin at the current time step t. Intuitively, if the variance of the RLS linear estimate is high in the direction of the feature vector observed at time t, then the label of that vector gets sampled. BBQ compares r_t with a threshold polynomial in $1/t$, where the degree of the polynomial is a free parameter. The threshold used by DGS, instead, is the squared margin returned by the RLS estimate on the current instance divided by a logarithmic quantity. In both cases, the label is sampled if r_t is bigger than the threshold.

In this work we refine and extend the theoretical analysis of both algorithms, and provide empirical evidence about differences in their behavior. In particular:

1. While previous analyses could only handle unit

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

[1] Although the analyses of both algorithms work for adversarially chosen instance sequences, DGS assumptions are weaker, as they allow each instance to be adversarially chosen as a function of past label realizations.

norm Bayes classifiers, we remove this assumption and explicit the dependence of regret and queries on the unknown norm of the Bayes classifier. This is especially important when RLS is run in the RKHS induced by a universal kernel (Steinwart, 2001), since now the Bayes classifier can be any continuous function of the instances. As a consequence, we can virtually drop any assumption about the way labels stochastically depend on each instance.

2. Most selective samplers, including BBQ, have a parameter to control the sampling rate. The original version of DGS, instead, is parameterless. This makes the algorithm rather awkward to use. Indeed, in some experiments we observed that DGS keeps sampling all labels for a long initial stretch of the instance sequence. We thus introduced a tunable parameter in DGS sampling rule and derived new bounds on mistakes and queries that take this parameter into account (further modifications of the original rule were necessary in order to prove these bounds, we call DGS-Mod the DGS algorithm using the modified rule). Experiments show that DGS-Mod performs much better than DGS, and in some case better than all other baselines we considered.

3. We have improved the analysis of BBQ by deriving a tighter upper bound on the regret. The improvement is significant in the following sense. If the instance process is i.i.d. and satisfies the well-known Mammen-Tsybakov low noise condition, then —provided BBQ parameter is properly tuned— the instantaneous regret per sampled label vanishes at rate $N^{-\frac{1+\alpha}{2}}$, where α is the exponent in the low noise condition. This is asymptotically better than the previous rate and matches the rate achieved by DGS/DGS-Mod under the same assumptions.

4. In many situations, one would like to run a selective sampler using an underlying linear classifier different than RLS. For example, one might be interested in specific regularizers that promote certain patterns in the data, such as sparsity. Unfortunately, to the best of our knowledge, there are no simultaneous bounds on regret and queries for arbitrary online linear classifiers run in selective sampling mode. In this paper we show that one can combine any gradient-based online classifier based on strongly convex regularizers with the DGS-Mod algorithm and obtain a selective sampler with the same query rate as DGS-Mod and the same mistake bound as the online classifier (plus a constant term).

5. Our experiments reveal that both BBQ and DGS-Mod generally perform well when run with linear kernels. With nonlinear kernels, instead, DGS-Mod suffers badly while BBQ keeps a very good performance. This fact is apparently caused by a bad dependence of DGS/DGS-Mod query bound on the determinant of the Gram matrix. Finally, if data are generated according to the stochastic assumptions underlying both algorithms, then BBQ outperforms all other baselines.

2. Preliminaries

Selective sampling is a modification of the online learning protocol for binary classification. At each step $t = 1, 2, \dots$ the learner receives and instance $\boldsymbol{x} \in \mathbb{R}^d$ and outputs a binary prediction for the instance's binary label $y_t \in \{-1, +1\}$. After each prediction, the learner may observe the true label y_t only by querying for it. Hence, if no query is issued at time t, then y_t remains unknown. Since the learner's performance is expected to improve as more labels are observed, the goal in selective sampling is to trade off predictive accuracy against number of queries.

Given a RKHS $\mathcal{H}$ with feature map ϕ and inner product $\langle \cdot \, , \cdot \rangle$, we make the following assumptions on the data-generating process: the sequence $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots \in \mathbb{R}^d$ of instances is such that there exists $f \in \mathcal{H}$ for which $\big| \langle f \, , \phi(\boldsymbol{x}_t) \rangle \big| \le 1$ for all t. Moreover, we assume labels y_t are realizations of independent random variables Y_t such that $\mathbb{E}[Y_t \mid \boldsymbol{x}_t] = \langle f \, , \phi(\boldsymbol{x}_t) \rangle$. Hence $\textsc{sgn}\big(\langle f \, , \phi(\boldsymbol{x}_t) \rangle\big)$, is the Bayes optimal classification for this noise model. Note that we do not make any assumption on $\|f\|^2 = \langle f \, , f \rangle$. In particular, we assume that this quantity is not known to the algorithm.

Note that if $\mathcal{H}$ has a universal kernel (Steinwart, 2001), then any continuous function $g(\boldsymbol{x}_t) = \mathbb{E}[Y_t \mid \boldsymbol{x}_t]$ is well approximated by some $f \in \mathcal{H}$. Hence, with universal kernels our noise model becomes quite general. In fact it only requires the labels to be generated by a stochastic source with a probability density function, which must be continuous w.r.t. the input space. For simplicity, in the sequel we always use the linear kernel and write $\mathbb{E}[Y_t \mid \boldsymbol{x}_t] = \boldsymbol{u}^\top \boldsymbol{x}_t$ for some $\boldsymbol{u} \in \mathbb{R}^d$. Here $\boldsymbol{u} \in \mathbb{R}^d$ is the Bayes classifier of unknown norm $\|\boldsymbol{u}\|$ which satisfies $\big|\boldsymbol{u}^\top \boldsymbol{x}_t\big| \le 1$ for all t. We also use the notation $\Delta_t = \boldsymbol{u}^\top \boldsymbol{x}_t$.

The trade-off addressed by the learner concerns the simultaneous control of the number N_T of queried labels and the cumulative regret

$$R_T = \sum_{t=1}^{T} \Big(\mathbb{P}(Y_t \, \widehat{\Delta}_t < 0) - \mathbb{P}(Y_t \, \Delta_t < 0) \Big) \qquad (1)$$

uniformly over the number T of prediction steps. Following previous papers (Cesa-Bianchi et al., 2009; Dekel et al., 2010), our bounds can depend on the number of steps where the labels Y_t are close to being random. According to our noise model, this is captured by $\varepsilon T_\varepsilon$, where $T_\varepsilon = \bigl|\{1 \le t \le T \,:\, |\Delta_t| < \varepsilon\}\bigr|$.

We consider selective sampling algorithms that use $\mathrm{SGN}\bigl(\widehat{\Delta}_t\bigr)$ to predict y_t, where $\widehat{\Delta}_t \in \mathbb{R}$ is an estimate of Δ_t. Let Z_t be the indicator function of the event that y_t is queried at time t. Then the cumulative regret can be decomposed as follows (we often use $\{\,\cdot\,\}$ to denote the indicator function of an event).

Lemma 1. *(Dekel et al., 2010) For any $\varepsilon > 0$,*

$$R_T \le \varepsilon T_\varepsilon + \sum_{t=1}^T \bar{Z}_t\{\Delta_t\widehat{\Delta}_t < 0,\, |\Delta_t| > \varepsilon\} + \sum_{t=1}^T Z_t\{\Delta_t\widehat{\Delta}_t < 0,\, |\Delta_t| > \varepsilon\}|\Delta_t| \;.$$

A different regret decomposition, which we later use, is the one below here.

Lemma 2. *(Cesa-Bianchi et al., 2009)*

$$R_T \le \varepsilon T_\varepsilon + \sum_{t=1}^T \mathbb{P}\Bigl(\bigl|\widehat{\Delta}_t - \Delta_t\bigr| \ge \varepsilon\Bigr) \;.$$

Our algorithms predict with the margin $\widehat{\Delta}_t = \boldsymbol{w}_t^\top \boldsymbol{x}_t$, where $\boldsymbol{w}_t$ is computed via the familiar RLS estimate[2]

$$\boldsymbol{w}_t = \bigl(I + S_{t-1}S_{t-1}^\top + \boldsymbol{x}_t\boldsymbol{x}_t^\top\bigr)^{-1} S_{t-1}\boldsymbol{Y}_{t-1} \;. \qquad (2)$$

The random matrix $S_{t-1} = \bigl[\boldsymbol{x}'_1, \dots, \boldsymbol{x}'_{N_{t-1}}\bigr]$ contains the N_{t-1} queried instances up to time $t-1$. The random vector $\boldsymbol{Y}_{t-1} = \bigl(Y'_1, \dots, Y'_{N_{t-1}}\bigr)$ contains the observed labels (so that Y'_k is the label of $\boldsymbol{x}'_k$), and I is the $d \times d$ identity matrix (whenever it is possible, we avoid expressing our quantities using dual-variable notation).

Introduce the notation $A_t = \bigl(I + S_{t-1}S_{t-1}^\top\bigr)$, $B_t = \boldsymbol{u}^\top\bigl(I + x_t x_t^\top\bigr)(A_t + \boldsymbol{x}_t\boldsymbol{x}_t^\top)^{-1}\boldsymbol{x}_t$, $r_t = \boldsymbol{x}_t^\top(A_t + \boldsymbol{x}_t\boldsymbol{x}_t^\top)^{-1}\boldsymbol{x}_t$. The following are standard properties of the RLS estimate (2), see for example (Cesa-Bianchi et al., 2009).

Lemma 3. *For each $t = 1, 2, \dots$ the following inequalities hold:*

1. *$\mathbb{E}\,\widehat{\Delta}_t = \Delta_t - B_t$, where $|B_t| \le \|\boldsymbol{u}\|\sqrt{r_t} + r_t$;*
2. *For all $\varepsilon > 0$,*
$$\mathbb{P}\Bigl(\bigl|\widehat{\Delta}_t + B_t - \Delta_t\bigr| \ge \varepsilon \,\Big|\, S_{t-1}\Bigr) \le 2\exp\left(-\frac{\varepsilon^2}{2r_t}\right) \;.$$
3. *If N_T is the total number of queries issued in the first T steps, then*
$$\sum_{t=1}^T Z_t r_t \le \ln|A_{T+1}| \le d\ln\left(1 + \frac{\sum_{i=1}^{N_T}\|\boldsymbol{x}_i\|^2}{d}\right) \;.$$

Using Lemma 3 we can prove the following.

Lemma 4. *For all $\varepsilon, p > 0$,*

$$\mathbb{P}\Bigl(\bigl|\widehat{\Delta}_t - \Delta_t\bigr| \ge \varepsilon \,\Big|\, S_{t-1}\Bigr) \le 2\exp\left(-\frac{\varepsilon^2}{8r_t}\right) + \exp\left(1 - \frac{\varepsilon^{2p}}{\bigl(4r_t\bigl(\|\boldsymbol{u}\|^2 + \varepsilon\bigr)\bigr)^p}\right) \;.$$

Proof. We expand the indicator of $\bigl|\widehat{\Delta}_t - \Delta_t\bigr| \ge \varepsilon$ by introducing the bias term B_t

$$\Bigl\{\bigl|\widehat{\Delta}_t - \Delta_t\bigr| \ge \varepsilon\Bigr\} \le \Bigl\{\bigl|\widehat{\Delta}_t + B_t - \Delta_t\bigr| \ge \frac{\varepsilon}{2}\Bigr\} + \Bigl\{|B_t| > \frac{\varepsilon}{2}\Bigr\} \;.$$

The first term is bounded with Lemma 3(2). For the second term, note that

$$\Bigl\{|B_t| > \frac{\varepsilon}{2}\Bigr\} \le \Bigl\{\|\boldsymbol{u}\|\sqrt{r_t} + r_t > \frac{\varepsilon}{2}\Bigr\} \le \left\{r_t > \frac{\varepsilon^2}{4\bigl(\|\boldsymbol{u}\|^2 + \varepsilon\bigr)}\right\} = \left\{r_t^p > \left(\frac{\varepsilon^2}{4\bigl(\|\boldsymbol{u}\|^2 + \varepsilon\bigr)}\right)^p\right\} \le \exp\left(1 - \frac{\varepsilon^{2p}}{\bigl(4r_t\bigl(\|\boldsymbol{u}\|^2 + \varepsilon\bigr)\bigr)^p}\right) \;.$$

The first inequality is obtained from Lemma 3(1). The last one uses $\{b < 1\} \le e^{1-b} \;\; \forall\, b$. □

3. A new bound for the BBQ algorithm

In this section we improve the regret bound of the BBQ selective sampler (Algorithm 1). The proof is based on applying techniques from (Dekel et al., 2010) to the original BBQ proof of (Cesa-Bianchi et al., 2009). Here we only show the main differences.

Theorem 1. *If BBQ is run with input $\kappa \in [0,1]$ then, after any number T of steps, $N_T \le T^\kappa \ln|A_{T+1}|$ with probability 1. Moreover, the cumulative regret satisfies*

$$R_T \le \min_{0<\varepsilon<1}\left(\varepsilon T_\varepsilon + \frac{8\bigl(\|\boldsymbol{u}\|^2 + 1\bigr)}{\varepsilon}\ln\left(\frac{5N_T}{\delta}\right)\ln|A_{T+1}| + 2\,\lceil 1/\kappa\rceil!\left(\frac{8}{\varepsilon^2}\right)^{1/\kappa} + e\left(\frac{4\bigl(\|\boldsymbol{u}\|^2 + \varepsilon\bigr)}{\varepsilon^2}\right)^{1/\kappa}\right)$$

with probability at least $1-\delta$ uniformly over T.

[2] Note that the sign of the prediction does not change if $\boldsymbol{x}_t\boldsymbol{x}_t^\top$ is removed from the expression in parenthesis.

Algorithm 1 The BBQ selective sampler

```
**Parameter:** $0 \le \kappa \le 1$
**Initialization:** Vector $\boldsymbol{w} = \boldsymbol{0}$, matrix $A_1 = I$
**for** each time step $t = 1, 2, \dots$ **do**
    Observe instance $\boldsymbol{x}_t \in \mathbb{R}^d$
    Predict label $y_t \in \{-1, +1\}$ with $\mathrm{SGN}(\boldsymbol{w}_t^\top \boldsymbol{x}_t)$
    **if** $r_t > t^{-\kappa}$ **then**
        Query label $y_t$
        $A_{t+1} = A_t + \boldsymbol{x}_t \boldsymbol{x}_t^\top$
        $\boldsymbol{w}_{t+1} = A_{t+1}^{-1}(A_t \boldsymbol{w}_t + y_t \boldsymbol{x}_t)$
    **else**
        $A_{t+1} = A_t, \boldsymbol{w}_{t+1} = \boldsymbol{w}_t$
    **end if**
**end for**
```

Note that when the dimension d is finite, Lemma 3(3) shows that N_T is of order $d\,T^\kappa\big(\ln T + \ln\ln T\big)$.

Proof. We use Lemma 1 when $Z_t = 1$ and Lemma 2 when $Z_t = 0$,

$$\begin{aligned}\sum_{t=1}^T & \mathbb{P}(Y_t\,\widehat{\Delta}_t < 0) - \mathbb{P}(Y_t\,\Delta_t < 0) \\ & \le \varepsilon\{|\Delta_t| < \varepsilon\} + \sum_{t\,:\,r_t \le t^{-\kappa}} \mathbb{P}\Big(\big|\widehat{\Delta}_t - \Delta_t\big| \ge \varepsilon\Big) \\ & + \sum_{t\,:\,r_t > t^{-\kappa}} \Big\{\widehat{\Delta}_t\Delta_t \le 0,\, |\Delta_t| \ge \varepsilon\Big\}\,|\Delta_t| \;.\end{aligned}$$

Proceeding as in (Dekel et al., 2010), we can write

$$\Big\{\widehat{\Delta}_t\Delta_t \le 0,\, |\Delta_t| \ge \varepsilon\Big\}\,|\Delta_t| \le \frac{(\Delta_t - \widehat{\Delta}_t)^2}{\varepsilon} \;.$$

We now use the fact that in BBQ S_t, and consequently N_t, are deterministic quantities for all t. From Lemma 4, overapproximating and setting $p = 1$, it follows that with probability at least $1 - \frac{\delta}{N_T}$ we have

$$\Big(\widehat{\Delta}_t - \Delta_t\Big)^2 \le 8(U^2 + 1) r_t \ln\left(\frac{(2+e)N_T}{\delta}\right) \;.$$

Hence we have that, with probability at least $1 - \delta$,

$$\begin{aligned}\sum_{t\,:\,r_t > t^{-\kappa}} & \Big\{\widehat{\Delta}_t\Delta_t \le 0,\, |\Delta_t| \ge \varepsilon\Big\}\,|\Delta_t| \\ & \le \frac{1}{\varepsilon} \sum_{t\,:\,r_t > t^{-\kappa}} 8(U^2 + 1) r_t \ln\left(\frac{(2+e)N_T}{\delta}\right) \\ & \le \frac{8\big(\|\boldsymbol{u}\|^2 + 1\big)}{\varepsilon} \ln\left(\frac{5N_T}{\delta}\right) \sum_{t\,:\,r_t > t^{-\kappa}} r_t \\ & \le \frac{8\big(\|\boldsymbol{u}\|^2 + 1\big)}{\varepsilon} \ln\left(\frac{5N_T}{\delta}\right) \ln|A_{T+1}|\end{aligned}$$

where in the last step we used Lemma 3(3). For the rounds when a label is not asked we use the following inequality from (Cesa-Bianchi et al., 2009),

$$\sum_{t=1}^T \exp\left(-a t^\kappa\right) \le \lceil 1/\kappa \rceil! \left(\frac{1}{a}\right)^{1/\kappa} \;.$$

Lemma 4 with $p = 1/\kappa$ now implies

$$\begin{aligned}\sum_{t\,:\,r_t \le t^{-\kappa}} & \mathbb{P}\Big(\big|\widehat{\Delta}_t - \Delta_t\big| \ge \varepsilon\Big) \\ & \le 2\lceil 1/\kappa \rceil! \left(\frac{8}{\varepsilon^2}\right)^{1/\kappa} + e\left(\frac{4\big(\|\boldsymbol{u}\|^2 + \varepsilon\big)}{\varepsilon^2}\right)^{1/\kappa} \;.\end{aligned}$$

The bound on the number N_T of queried labels is the same as in (Cesa-Bianchi et al., 2009). □

This bound, beside showing explicitly the dependency on the norm of $\boldsymbol{u}$, has a better dependency on ε in the regret bound, compared to the one proved in (Cesa-Bianchi et al., 2009). This allows us to prove an optimal rate in the i.i.d. case. In fact, consider the case when the instances $\boldsymbol{x}_t$ are i.i.d. random variables $\boldsymbol{X}_t$ with fixed but unknown distribution. We model the distribution of the instances around the hyperplane $\boldsymbol{u}^\top \boldsymbol{x} = 0$ using the popular *Mammen-Tsybakov low noise condition* (Tsybakov, 2004):

There exist $c > 0$ and $\alpha \ge 0$ such that

$$\mathbb{P}\Big(\big|\boldsymbol{u}^\top \boldsymbol{X}\big| < \varepsilon\Big) \le c\,\varepsilon^\alpha \qquad \text{for all } \varepsilon > 0.$$

Note that, when the noise exponent α is 0, this condition is vacuous.

It is easy to show that a proper choice of κ as a function of α gives that the regret of BBQ, when expressed in terms of the number N of queries, vanishes at rate $N^{-\frac{1+\alpha}{2}}$ excluding log factors. This is the same rate as the one obtained by the DGS algorithm under the same low noise condition, although DGS requires no tuning.

4. A modified DGS algorithm

In this section we propose and analyze a modification of the DGS selective sampler, called DGS-Mod (Algorithm 2), where we introduce a parameter $\alpha > 0$ in the query rule. This parameter, which appears in both regret and query bounds, allows us to trade off regret against queries in a smooth way. In addition to that, we also change the query rule to make the algorithm independent from the unknown norm of the Bayes optimal classifier $\boldsymbol{u}$ (which in previous analyses was assumed to be known and set to 1).

Algorithm 2 The DGS-Mod algorithm

```
  **Parameter:** $\alpha > 0$
  **Initialization:** Vector $\boldsymbol{w}_1 = \mathbf{0}$, matrix $A_1 = I$
  **for** each time step $t = 1, 2, \dots$ **do**
    Observe instance $\boldsymbol{x}_t \in \mathbb{R}^d$
    Set $\hat{\Delta}_t = \boldsymbol{w}_t^\top \boldsymbol{x}_t$
    Predict label $y_t \in \{-1, +1\}$ with $\textsc{sgn}(\hat{\Delta}_t)$
    $\theta_t^2 = 2\alpha\big(\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t\big) h_\delta(t) \log t$
    **if** $\hat{\Delta}_t^2 \le \theta_t^2$ **then**
      Query label $y_t$
      $\boldsymbol{w}_{t+1/2} = \boldsymbol{w}_t - \operatorname{sign}(\hat{\Delta}_t) \left| \frac{|\hat{\Delta}_t| - 1}{\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t} \right|_+ A_t^{-1} \boldsymbol{x}_t$
      $A_{t+1} = A_t + \boldsymbol{x}_t \boldsymbol{x}_t^\top$   and   $r_t = \boldsymbol{x}_t^\top A_{t+1}^{-1} \boldsymbol{x}_t$
      $\boldsymbol{w}_{t+1} = A_{t+1}^{-1}(A_t \boldsymbol{w}_{t+1/2} + y_t \boldsymbol{x}_t)$
    **else**
      $A_{t+1} = A_t, \boldsymbol{w}_{t+1} = \boldsymbol{w}_t$
    **end if**
  **end for**
```

Let $h_\delta(t) = 4 \sum_{s=1}^{t-1} Z_s r_s + 36 \ln(t/\delta)$, the DGS-Mod strategy (Algorithm 2) decides whether to query by comparing the square of the current margin to the threshold $\theta_t^2 = 2\alpha\big(\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t\big) h_\delta(t) \ln t$. This strategy is motivated by the following Lemmas, that prove that in this way the regret on rounds when no label is asked sum to a constant factor that depends on the ratio between the unknown squared norm of $\boldsymbol{u}$ and α.

Lemma 5. *(Dekel et al., 2010) With probability at least $1 - \delta$, $(\Delta_t - \hat{\Delta}_t)^2 \le \big(\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t\big)\big(\|\boldsymbol{u}\|^2 + h_\delta(t)\big)$ holds simultaneously for all $t \ge 1$.*

Lemma 6. *With probability at least $1 - \delta$ uniformly over T we have*

$$\sum_{t=1}^{T} \bar{Z}_t \left\{ \Delta_t \widehat{\Delta}_t < 0 \right\} \le 1 + \frac{2}{3} \exp\left(\frac{1}{\alpha} \left(\frac{\|\boldsymbol{u}\|^2}{24} + 1 \right) \right) .$$

Proof. Using Lemma 5, for each $t \ge 2$ we can write

$$\begin{aligned} &\bar{Z}_t \left\{ \Delta_t \widehat{\Delta}_t < 0 \right\} \le \left\{ (\Delta_t - \widehat{\Delta}_t)^2 > \widehat{\Delta}_t^2,\, \widehat{\Delta}_t^2 > \theta_t^2 \right\} \\ &\le \left\{ \boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t \left(\|\boldsymbol{u}\|^2 + h_\delta(t) \right) > \theta_t^2 \right\} \\ &= \left\{ \frac{\|\boldsymbol{u}\|^2 + h_\delta(t)}{\alpha h_\delta(t)} > 2 \ln t \right\} \le \frac{1}{t^2} \exp\left(\frac{\|\boldsymbol{u}\|^2 + h_\delta(t)}{\alpha h_\delta(t)} \right) \\ &\le \frac{1}{t^2} \exp\left(\frac{1}{\alpha} \left(\frac{\|\boldsymbol{u}\|^2}{36 \ln(2/\delta)} + 1 \right) \right) . \end{aligned}$$

Since the above chain of inequalities holds for all $t \ge 2$ with probability at least $1 - \delta$, by summing over $t = 1, \dots, T$ we obtain the desired result. □

We can now prove the regret bound for DGS-Mod and its bound on the number of queries.

Theorem 2. *After any number T of steps, with probability at least $1 - \delta$, the cumulative regret R_T of the modified DGS algorithm run with input $\alpha > 0$ satisfies*

$$\begin{aligned} R_T \le \min_{0 < \varepsilon < 1} \Bigg(& 1 + \varepsilon T_\varepsilon + \frac{2}{3} \exp\left[\frac{1}{\alpha} \left(\frac{\|\boldsymbol{u}\|^2}{24} + 1 \right) \right] \\ & + \frac{1}{\varepsilon} \left(2 \|\boldsymbol{u}\|^2 + 8 \ln |A_{T+1}| + 144 \ln \frac{T}{\delta} \right) \Bigg) \end{aligned}$$

and for $X \ge \max_t \|\boldsymbol{x}_t\|$ the number N_T of queries satisfies

$$\begin{aligned} N_T \le 1 + T_\varepsilon + & \frac{4(1 + X^2)}{\varepsilon^2} \ln |A_{T+1}| \\ & \times \left[\|\boldsymbol{u}\|^2 + \big(1 + 2\alpha \ln T\big) \left(4 \ln |A_{T+1}| + 36 \ln \frac{T}{\delta} \right) \right] . \end{aligned}$$

Proof. By Lemma 1, in order to bound R_T is enough bounding $\sum_{t=1}^{T} \bar{Z}_t \left\{ \Delta_t \widehat{\Delta}_t < 0,\, |\Delta_t| > \varepsilon \right\}$ via Lemma 6, and then using the bound

$$\begin{aligned} & \sum_{t=1}^{T} Z_t \{ \Delta_t \widehat{\Delta}_t < 0,\, |\Delta_t| > \varepsilon \} |\Delta_t^2| \\ & \qquad \le \frac{1}{\varepsilon} \left(2 \|\boldsymbol{u}\|^2 + 8 \ln |A_{T+1}| + 144 \ln \frac{T}{\delta} \right) \end{aligned}$$

obtained via an obvious modification of the corresponding result in (Dekel et al., 2010). In order to bound N_T, define $\beta_t = \frac{\varepsilon^2}{4} \frac{2\alpha \ln t}{1 + 2\alpha \ln t}$. Then

$$\begin{aligned} Z_t &= Z_t \left\{ \theta_t^2 < \beta_t \right\} + Z_t \left\{ \theta_t^2 \ge \beta_t \right\} \\ &= Z_t \left\{ \hat{\Delta}_t^2 \le \theta_t^2, \theta_t^2 < \beta_t \right\} + Z_t \left\{ \theta_t^2 \ge \beta_t \right\} . \qquad (3) \end{aligned}$$

Consider the first term in (3). Using Lemma 1,

$$|\Delta_t| \le |\hat{\Delta}_t| + \sqrt{\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t \big(\|\boldsymbol{u}\|^2 + h_\delta(t) \big)}$$

holds for each $t \ge 2$ with probability at least $1 - \delta$. Hence, using $(a + b)^2 \le 2a^2 + 2b^2$ we have that

$$\begin{aligned} & \left\{ \hat{\Delta}_t^2 \le \theta_t^2, \theta_t^2 < \beta_t \right\} \\ & \le \left\{ \Delta_t^2 \le 2\theta_t^2 + 2\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t \big(\|\boldsymbol{u}\|^2 + h_\delta(t) \big), \theta_t^2 < \beta_t \right\} \\ & = \left\{ \Delta_t^2 \le 2\theta_t^2 \frac{1 + 2\alpha \ln t}{2\alpha \ln t} + 2 \|\boldsymbol{u}\|^2 \boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t, \theta_t^2 < \beta_t \right\} \\ & \le \left\{ \Delta_t^2 \le 2\beta_t \frac{1 + 2\alpha \ln t}{2\alpha \ln t} + 2 \|\boldsymbol{u}\|^2 \boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t \right\} \\ & = \left\{ \Delta_t^2 \le \frac{\varepsilon^2}{2} + 2 \|\boldsymbol{u}\|^2 \boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t \right\} \\ & \le \left\{ \Delta_t^2 \le \varepsilon^2 \right\} + \left\{ 2 \|\boldsymbol{u}\|^2 \boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t > \frac{\varepsilon^2}{2} \right\} \\ & \le \left\{ \Delta_t^2 \le \varepsilon^2 \right\} + \frac{4(1 + \|\boldsymbol{x}_t\|^2) \|\boldsymbol{u}\|^2 r_t}{\varepsilon^2} \end{aligned}$$

holds for each $t \ge 2$ with probability at least $1 - \delta$. In the last inequality we used $\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t \le (1 + \|\boldsymbol{x}_t\|^2) r_t$. For the second term in (3) we have

$$\begin{aligned} \{\theta_t^2 \ge \beta_t\} &= \{4\big(\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t\big)(1 + 2\alpha \ln t) h_\delta(t) \ge \varepsilon^2\} \\ &\le \frac{4}{\varepsilon^2}\big(\boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t\big)(1 + 2\alpha \ln t) h_\delta(t) \\ &\le \frac{4 r_t}{\varepsilon^2}(1 + X^2)(1 + 2\alpha \ln T) h_\delta(T) \ . \end{aligned}$$

Summing r_t over $t = 1, \dots, T$ and using $h_\delta(T) \le 4 \ln |A_{T+1}| + 36 \ln \frac{T}{\delta}$ concludes the proof. □

Comparing DGS-Mod and DGS bound, we can see that the independence on the norm of $\boldsymbol{u}$ comes at the cost of an additional constant term in the regret bound, and term $1 + 2\alpha \ln T$ multiplying the second part of the bound on the number of queries. We now compare the bounds for BBQ and DGS-Mod. Both regret bounds have a term that takes into account the rounds in which no label is asked, and depends on the unknown squared norm of the Bayes classifier $\boldsymbol{u}$. In BBQ this term has a polynomial dependence on $\|\boldsymbol{u}\|$, whereas in DGS-Mod this dependence is exponential. However, the main difference between the two algorithms is in the query bounds. Whereas in BBQ the query bound is deterministic and controlled by T^κ, DGS-Mod has a probabilistic query bound which depends, through T_ε, on the number of small margin instances —a far less controllable quantity. Moreover, DGS-Mod query bound contains the term $\ln^2 |A_{T+1}|$. This can grow fast when the RKHS is infinite-dimensional, as in the case of universal kernels. Indeed, in Section 6 we observe a super-linear growth of this term in case of Gaussian kernels, and a corresponding bad empirical behavior of DGS-Mod.

5. A hybrid algorithm

In this section we address the question of designing selective samplers whose underlying online classifier is not based on RLS. For example, consider the gradient-based online classifier that predicts the label of $\boldsymbol{x}_t \in \mathbb{R}^d$ with $\textsc{sgn}(\boldsymbol{v}_t^\top \boldsymbol{x}_t)$, where $\boldsymbol{v}_1 = \boldsymbol{0}$ and $\boldsymbol{v}_{t+1} = \nabla f^*\big(\nabla f(\boldsymbol{v}_t) - y_t \boldsymbol{x}_t\big)$. Here f is any differentiable function which is strongly convex w.r.t. a norm $\|\cdot\|$. The function f acts as a regularizer, promoting specific structures in the data sequence such as sparsity. One can derive randomized selective samplers from gradient-based classifiers using the technique of (Cesa-Bianchi et al., 2006). However, via this technique one only bounds the regret and not the number of queries. We now show a way of combining the DGS-Mod algorithm with any gradient-based online classifier so that the resulting selective sampler enjoys a simultaneous bound on regret and number of queries.

Algorithm 3 The hybrid DGS selective sampler

```
**Parameters:** $\alpha > 0$ (used by DGS-Mod)
**Initialization:** Weight vectors $\boldsymbol{v}_1 = \boldsymbol{0}$, $\boldsymbol{w}_1 = \boldsymbol{0}$
**for** each time step $t = 1, 2, \dots$ **do**
  Observe instance $\boldsymbol{x}_t \in \mathbb{R}^d$
  Set $\hat{\Delta}_t = \boldsymbol{w}_t^\top \boldsymbol{x}_t$
  **if** $\hat{\Delta}_t^2 \le \theta_t^2$ **then**
    Predict label $y_t \in \{-1, +1\}$ with $\textsc{sgn}(\boldsymbol{v}_t^\top \boldsymbol{x}_t)$
    Query label $y_t$
    Update $\boldsymbol{w}_t$ using the DGS-Mod algorithm
    $\boldsymbol{v}_{t+1} = \nabla f^*\big(\nabla f(\boldsymbol{v}_t) - y_t \boldsymbol{x}_t\big)$
  **else**
    Predict label $y_t \in \{-1, +1\}$ with $\textsc{sgn}(\hat{\Delta}_t)$
  **end if**
**end for**
```

Consider the hybrid DGS (Algorithm 3). This selective sampler runs in parallel the gradient-based classifier and the DGS-Mod algorithm. If DGS-Mod issues a query on the current instance, then the prediction of the gradient-based classifier is used; otherwise, the prediction of DGS-Mod is used. When a query is issued, both algorithms use the label to make an update.

Using the framework of (Orabona & Crammer, 2010), we get that any gradient-based classifier, using a β-strongly convex function f such that $f(a\boldsymbol{v}) \le a^2 f(\boldsymbol{v})$ for all $a \in \mathbb{R}^+$, has a mistake bound

$$M_T < \inf_{\boldsymbol{v} \in \mathbb{R}^d} \left(L_T(\boldsymbol{v}) + X \sqrt{\frac{2 f(\boldsymbol{v})}{\beta} T} \right)$$

on any sequence $(\boldsymbol{x}_t, y_t), \dots, (\boldsymbol{x}_T, y_T) \in \mathbb{R}^d \times \{-1, +1\}$ of examples. Here $L_T(\boldsymbol{v}) = \sum_{t=1}^T \ell_t(\boldsymbol{v})$ is the cumulative hinge loss of $\boldsymbol{v}$ and $\max_t \|\boldsymbol{x}_t\|_* \le X$. This can be used to prove the following result (proof omitted).

Theorem 3. *If the hybrid DGS algorithm is run on an arbitrary sequence of instances with labels generated according to the model of Section 2, then with probability at least $1 - \delta$ the number M_T of mistakes satisfies*

$$M_T \le L_T(\boldsymbol{u}) + X \sqrt{\frac{2 f(\boldsymbol{u})}{\beta} N_T} + \mathcal{O}(1)$$

where $\boldsymbol{u} \in \mathbb{R}^d$ defines the Bayes classifier. N_T is the number of queries issued on the same sequence by the DGS-Mod algorithm, which also bounds the number of queries made by the hybrid DGS algorithm.

This bound differs from the previous ones (Theorem 1 and 2) in a few aspects. There, the probability of mak-

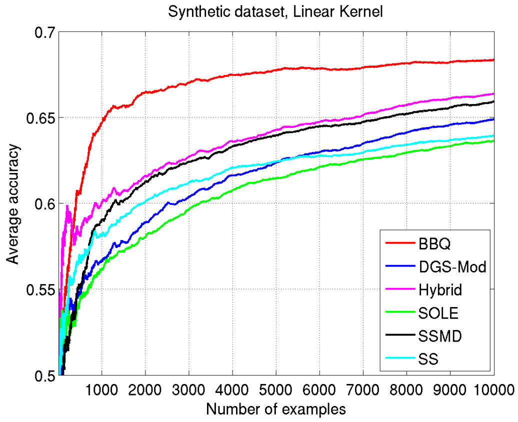

Figure 1. Accuracy against number of examples on the synthetic dataset when the query rate is ∼0.135 (averages over 5 random permutations).

ing a mistake is bounded as a function of the probability of $\boldsymbol{u}$ making a mistake. Here, we compare mistakes of the algorithm to hinge loss of $\boldsymbol{u}$. In the bound, the only term that depends on time is $\sqrt{N_T}$, which is small when DGS-Mod issues few queries. Moreover, this term is proportional to $X\sqrt{f(\boldsymbol{u})}$. So, if the regularizer f matches the properties of $\boldsymbol{u}$ and the data, through the dual norm $\|\cdot\|_*$, we can expect fewer mistakes. We see this experimentally in Section 6.

6. Experiments

In this section we compare the *average accuracy* (total number of correct online classifications divided by the number of examples) of BBQ and DGS-Mod against two baselines, both using RLS as base classifier. We use synthetic and real-world datasets with linear and Gaussian kernels. Our baselines are: SOLE from (Cesa-Bianchi et al., 2006), SS and SSMD from (Cavallanti et al., 2008). All algorithms have a tunable parameter to trade-off performance vs. number of queried labels.

Synthetic dataset with linear kernel. In our first experiment we use a synthetic dataset with 10,000 examples drawn from a uniform distribution in 100 dimensions, with instances $\boldsymbol{x}_t$ such that $\|\boldsymbol{x}_t\|_\infty \leq 1$. The hyperplane $\boldsymbol{u}$ generating the labels has all coefficients set to zero but the first two, which are chosen at random under the constraint $\|\boldsymbol{u}\| = 1$. The labels are generated stochastically according to the noise model described in Section 2. Instances with margin bigger than one in absolute value are discarded. In this experiment, we also test the hybrid algorithm of Section 5 using a p-norm Perceptron (Gentile, 2003) as gradient-based classifier. Cross-validation is used to choose p. Figure 1 shows average accuracy vs. number of examples when the parameter of each algorithm is chosen so that the query rate is ∼0.135. In this case the best algorithm is BBQ, while DGS-Mod exhibits a much lower accuracy. The hybrid algorithm, which takes advantage of the sparse $\boldsymbol{u}$, performs also quite well, although not as well as BBQ. Figure 2 shows the trade-off between average accuracy and fraction of queried labels. The behavior over a large spectrum of query rates is qualitatively similar to the one observed in Figure 1. In the following experiments the Hybrid algorithm is omitted because its performance remains very close to that of DGS-Mod.

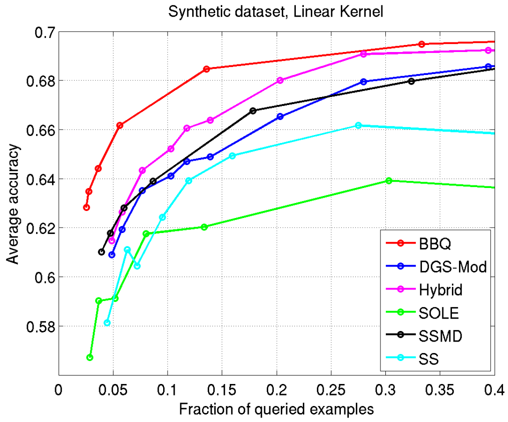

Figure 2. Accuracy against fraction of queried labels on the synthetic dataset (averages over 5 random permutations).

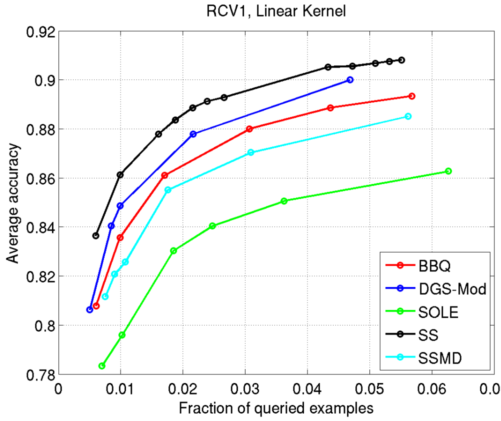

Figure 3. Accuracy against fraction of queried labels for the task of classifying the most frequent category of RCV1 (linear kernel, averages over 5 random permutations).

Real-world data with linear kernel. Our second experiment uses the first 40,000 newswire stories from the Reuters Corpus Volume 1 dataset (RCV1). Each

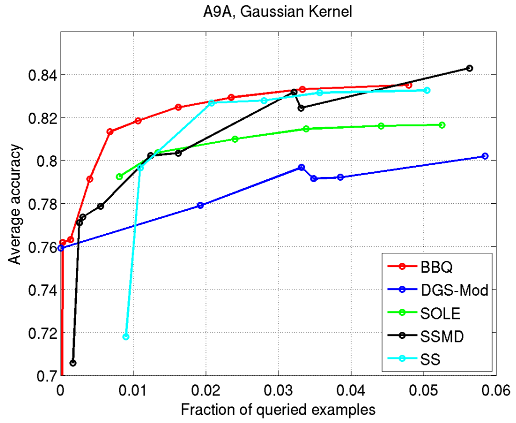

Figure 4. Accuracy against fraction of queried labels for a9a (Gaussian kernel, averages over 5 random permutations).

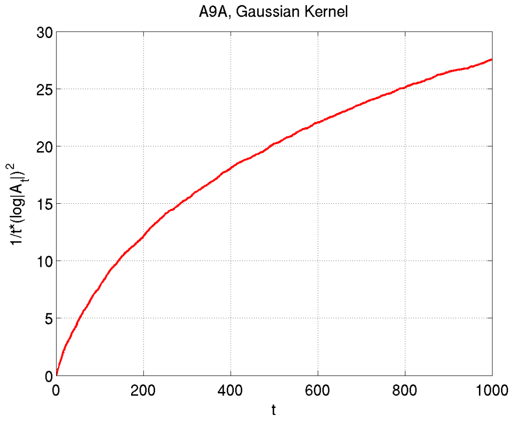

Figure 5. Behavior of $\frac{1}{t}\left(\ln|A_t|\right)^2$ as a function of t in a run of DGS-Mod.

newsstory of this corpus is tagged with one or more labels from a set of 102 categories. A standard TF-IDF bag-of-words encoding is used to obtain 138,860 features. We use a linear kernel and train one-vs-all based on the most frequent class. Figure 3 shows the average accuracy vs. the average fraction of queried labels. In this experiment the best performer is SS, with BBQ and DGS-Mod performing similarly well.

Real-world data with Gaussian kernel. For our third and last experiment we use a9a[3], a subset of the census-income (Adult) database with 32,561 binary-labeled examples and 123 features. We use a Gaussian kernel with $\sigma^2 = 12.5$, chosen by cross-validation. Here the performance of DGS-Mod is extremely poor, while BBQ performs best —see Figure 4. In an attempt to explain DGS-Mod behavior, we plot in Figure 5 the value of $\frac{1}{t}\left(\ln|A_t|\right)^2$ against the number of the examples on a given permutation of data. The term $\left(\ln|A_{T+1}|\right)^2$ appears in the query bound of DGS/DGS-Mod —see Theorem 2. While in the linear case this quantity is $\mathcal{O}(\ln T)$, in a generic RKHS it can grow much faster. On this dataset, we observe that $\left(\ln|A_t|\right)^2$ is superlinear in t, implying that DGS/DGS-Mod query bound becomes vacuous. In fact, in this case DGS queries at all time steps, while DGS-Mod does not perform well no matter how the parameter α is chosen.

7. Conclusions

In this work we derived improved regret bounds for RLS-based algorithms in the online selective sampling setting. We also conducted experiments to test the behavior of our algorithms against other RLS-based samplers. We plan to extend our empirical comparison by including algorithms developed under different assumptions on the data process, such as the recent work (Beygelzimer et al., 2010).

Acknowledgments

The authors acknowledge partial support by the PASCAL2 NoE under EC grant FP7-216886.

References

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

Cavallanti, G., Cesa-Bianchi, N., and Gentile, C. Linear classification and selective sampling under low noise conditions. In *NIPS*, pp. 249–256, 2008.

Cesa-Bianchi, N., Gentile, C., and Zaniboni, L. Worst-case analysis of selective sampling for linear classification. *JMLR*, 7:1025–1230, 2006.

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

Dekel, O., Gentile, C., and Sridharan, K. Robust selective sampling from single and multiple teachers. In *COLT*. MIT Press, 2010.

Gentile, C. The robustness of the p-norm algorithms. *Machine Learning*, 53(3):265–299, 2003.

Orabona, F. and Crammer, K. New adaptive algorithms for online classification. In *NIPS*, pp. 1840–1848. 2010.

Steinwart, I. On the influence of the kernel on the consistency of support vector machines. *JMLR*, 2:67–93, 2001.

Tsybakov, A. Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 32(1):135–166, 2004.

[3] `www.csie.ntu.edu.tw/~cjlin/libsvmtools/`