Robust Bounds for Classification via Selective Sampling

Nicolò Cesa-Bianchi CESA-BIANCHI@DSI.UNIMI.IT
DSI, Università degli Studi di Milano, Via Comelico 39, 20135 - Milano, Italy

Claudio Gentile CLAUDIO.GENTILE@UNINSUBRIA.IT
DICOM, Università dell'Insubria, Via Mazzini 5, 21100 - Varese, Italy

Francesco Orabona FORABONA@IDIAP.CH
Idiap, Rue Marconi 19, Case Postale 592, CH-1920 Martigny, Switzerland

Abstract

We introduce a new algorithm for binary classification in the selective sampling protocol. Our algorithm uses Regularized Least Squares (RLS) as base classifier, and for this reason it can be efficiently run in any RKHS. Unlike previous margin-based semi-supervised algorithms, our sampling condition hinges on a simultaneous upper bound on bias and variance of the RLS estimate under a simple linear label noise model. This fact allows us to prove performance bounds that hold for an arbitrary sequence of instances. In particular, we show that our sampling strategy approximates the margin of the Bayes optimal classifier to any desired accuracy ε by asking $\widetilde{\mathcal{O}}\big(d/\varepsilon^2\big)$ queries (in the RKHS case d is replaced by a suitable spectral quantity). While these are the standard rates in the fully supervised i.i.d. case, the best previously known result in our harder setting was $\widetilde{\mathcal{O}}\big(d^3/\varepsilon^4\big)$. Preliminary experiments show that some of our algorithms also exhibit a good practical performance.

1. Introduction

A practical variant of the standard (fully supervised) online learning protocol is a setting where, at each prediction step, the learner can abstain from observing the current label. If the learner observes the label, which he can do by issuing a *query*, then the label value can be used to improve future predictions. If the label is predicted but not queried, then the learner never knows whether his prediction was correct. Thus, only queried labels are observed while all others remain unknown. This protocol is often called selective sampling, and we interchangeably use *queried labels* and *sampled labels* to denote the labels observed by the learner.

Given a general online prediction technique, like regularized least squares (RLS), we are interested in controlling the predictive performance as the query rate goes from fully supervised (all labels are queried) to fully unsupervised (no label is queried). This is motivated by observing that, in a typical practical scenario, one might want to control the accuracy of predictions while imposing an upper bound on the query rate. In fact, the number of observed labels has usually a very direct influence on basic computational aspects of online learning algorithms, such as running time and storage requirements.

In this work we develop semi-supervised variants of RLS for binary classification. We analyze these variants under no assumptions on the mechanism generating the sequence of instances, while imposing a simple linear noise model for the conditional label distribution. Intuitively, our algorithms issue a query when a common upper bound on bias and variance of the current RLS estimate is larger than a given threshold. Conversely, when this upper bound gets small, we infer via a simple large deviation argument that the margin of the RLS estimate on the current instance is close enough to the margin of the Bayes optimal classifier. Hence the learner can safely avoid issuing a query on that step.

In order to summarize our results, assume for the sake of simplicity that the Bayes optimal classifier — which for us is a linear classifier $\boldsymbol{u}$— has unknown

Appearing in *Proceedings of the 26th International Conference on Machine Learning*, Montreal, Canada, 2009. Copyright 2009 by the author(s)/owner(s).

margin $|\boldsymbol{u}^\top \boldsymbol{x}_t| \ge \varepsilon > 0$ on all instances $\boldsymbol{x}_t \in \mathbb{R}^d$. Then, in our data model, the average (per-step) risk of the fully supervised RLS, asking $N_T = T$ labels in T steps, is known to converge to the average risk of the Bayes optimal classifier at rate $d\big(\varepsilon^2 T\big)^{-1}$ excluding logarithmic factors. In this work we show that, using our semi-supervised RLS variant, we can replace $N_T = T$ with any desired query bound $N_T = d\,T^\kappa$ (for $0 \le \kappa \le 1$) while achieving a convergence rate of order $d\big(\varepsilon^2 T\big)^{-1} + \big(\varepsilon^{2/\kappa} T\big)^{-1}$.

One might wonder whether these results could also be obtained just by running a standard RLS algorithm with a constant label sampling rate, say $T^{\kappa-1}$, independent of the sequence of instances. If we could prove for RLS an *instantaneous* regret bound like d/T then the answer would be affirmative. However, the lack of assumptions on the way instances are generated makes it hard to prove any nontrivial instantaneous regret bound.

If the margin ε is known, or, equivalently, our goal is to approximate the Bayes margin to some accuracy ε, then we show that the above strategies achieve, with high probability, any desired accuracy ε by querying only order of d/ε^2 labels (excluding logarithmic factors). Again, the reader should observe that this bound could not be obtained by, say, concentrating all queries on an initial phase of length $O(d/\varepsilon^2)$. In such a case, an obvious adversarial strategy would be to generate noninformative instances just in that phase.

In short, if we require online semi-supervised learning algorithms to work in worst-case scenarios we need to resort to nontrivial label sampling techniques.

We have run comparative experiments on both artificial and real-world medium-sized datasets. These experiments, though preliminary in nature, reveal the effectiveness of our sampling strategies even from a practical standpoint.

1.1. Related work and comparison

Selective sampling is a well-known semi-supervised online learning setting. Pioneering works in this area are (Cohn et al., 1990) and (Freund et al., 1997). More recent related results focusing on linear classification problems include (Balcan et al., 2006; Balcan et al., 2007; Cavallanti et al., 2009; Cesa-Bianchi et al., 2006b; Dasgupta et al., 2008; Dasgupta et al., 2005; Strehl & Littman, 2008), although some of these works analyze batch rather than online protocols. Most previous studies consider the case when instances are drawn i.i.d. from a fixed distribution, exceptions being the worst-case analysis in (Cesa-Bianchi et al., 2006b) and the very recent analysis in the KWIK learning protocol (Strehl & Littman, 2008). Both of these papers use variants of RLS working on arbitrary instance sequences. The work (Cesa-Bianchi et al., 2006b) is completely worst case: the authors make no assumptions whatsoever on the mechanism generating labels and instances; however, they are unable prove bounds on the label query rate as we do here. The KWIK model of (Strehl & Littman, 2008) —see also the more general setup in (Li et al., 2008)— is closest to the setting considered in this paper. There the goal is to approximate the Bayes margin to within a given accuracy ε. The authors assume arbitrary sequences of instances and the same linear stochastic model for labels as the one considered here. A modification of the linear regressor in (Auer, 2002), combined with covering arguments, allows them to compete against an adaptive adversarial strategy for generating instances. Their algorithm, however, yields the significantly worse bound $\widetilde{\mathcal{O}}\big(d^3/\varepsilon^4\big)$ on the number of queries, and seems to work in the finite dimensional ($d < \infty$) case only. In contrast, our algorithms achieve the better query bound $\widetilde{\mathcal{O}}\big(d/\varepsilon^2\big)$ against oblivious adversaries. Moreover, our algorithms can be easily run in any infinite dimensional RKHS.

2. Preliminaries

In the selective sampling protocol for online binary classification, at each step $t = 1, 2, \ldots$ the learner receives an instance $\boldsymbol{x}_t \in \mathbb{R}^d$ and outputs a binary prediction for the associated unknown label $y_t \in \{-1, +1\}$. After each prediction the learner may observe the label y_t only by issuing a *query*. If no query is issued at time t, then y_t remains unknown. Since one expects the learner's performance to improve if more labels are observed, our goal is to trade off predictive accuracy against number of queries.

All results proven in this paper hold for any fixed individual sequence $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ of instances, under the sole assumption that $\|\boldsymbol{x}_t\| = 1$ for all $t \ge 1$. Given any such sequence, we assume the corresponding labels $y_t \in \{-1, +1\}$ are realizations of random variables Y_t such that $\mathbb{E}\, Y_t = \boldsymbol{u}^\top \boldsymbol{x}_t$ for all $t \ge 1$, where $\boldsymbol{u} \in \mathbb{R}^d$ is a fixed and unknown vector such that $\|\boldsymbol{u}\| = 1$. Note that $\mathrm{SGN}\big(\Delta_t\big)$, for $\Delta_t = \boldsymbol{u}^\top \boldsymbol{x}$, is the Bayes optimal classifier for this noise model.

We study selective sampling algorithms that use $\mathrm{SGN}\big(\widehat{\Delta}_t\big)$ to predict Y_t. The quantity $\widehat{\Delta}_t = \boldsymbol{w}_t^\top \boldsymbol{x}_t$ is a margin computed via the RLS estimate

$$\boldsymbol{w}_t = \big(I + S_{t-1}\, S_{t-1}^\top + \boldsymbol{x}_t \boldsymbol{x}_t^\top\big)^{-1} S_{t-1}\, \boldsymbol{Y}_{t-1} \qquad (1)$$

defined over the matrix $S_{t-1} = \big[\, \boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_{N_{t-1}} \big]$ of 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.

We are interested in simultaneously controlling the cumulative regret

$$R_T = \sum_{t=1}^{T} \Bigl(\mathbb{P}(Y_t\, \widehat{\Delta}_t < 0) - \mathbb{P}(Y_t\, \Delta_t < 0) \Bigr) \qquad (2)$$

and the number N_T of queried labels, uniformly over T.

Let $A_t = \bigl(I + S_{t-1}\, S_{t-1}^\top + \boldsymbol{x}_t\, \boldsymbol{x}_t^\top\bigr)$. We introduce the following relevant quantities:

$$B_t = \boldsymbol{u}^\top \bigl(I + x_t\, x_t^\top\bigr) A_t^{-1} \boldsymbol{x}_t \ , \qquad r_t = \boldsymbol{x}_t^\top A_t^{-1} \boldsymbol{x}_t$$
$$\boldsymbol{q}_t = S_{t-1}^\top A_t^{-1} \boldsymbol{x}_t \ , \qquad s_t = \bigl\| A_t^{-1} \boldsymbol{x}_t \bigr\| \ .$$

The following properties of the RLS estimate (1) have been proven in, e.g., (Cesa-Bianchi et al., 2006a).

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

1. $\mathbb{E}\, \widehat{\Delta}_t = \Delta_t - B_t$;
2. $|B_t| \le s_t + r_t$;
3. $s_t \le \sqrt{r_t}$;
4. $\|\boldsymbol{q}_t\|^2 \le r_t$;
5. *For all $\varepsilon > 0$,*
   $$\mathbb{P}\Bigl(\bigl| \widehat{\Delta}_t + B_t - \Delta_t \bigr| \ge \varepsilon \Bigr) \le 2 \exp\left(- \frac{\varepsilon^2}{2\, \|\boldsymbol{q}_t\|^2} \right) \ ;$$
6. *If N_T is the total number of queries issued in the first T steps, then*
   $$\sum_{\substack{1 \le t \le T \\ Y_t \text{ queried}}} r_t \le \sum_{i=1}^{d} \ln(1 + \lambda_i) \le d \ln\left(1 + \frac{N_T}{d}\right)$$
   where λ_i is the i-th eigenvalue of the (Gram) matrix $S_T^\top S_T$ defined on the queried instances.

3. A new selective sampling algorithm

Our main theoretical result provides bounds on the cumulative regret and the number of queried labels for the selective sampling algorithm introduced in Figure 1. We call this algorithm the BBQ (Bound on Bias Query) algorithm. BBQ queries $\boldsymbol{x}_t$ whenever r_t is larger than a threshold vanishing as $t^{-\kappa}$, where $0 \le \kappa \le 1$ is an input parameter. This simple query condition builds on Property 5 of Lemma 1. This property shows that $\widehat{\Delta}_t$ is likely to be close to the margin Δ_t of the Bayes optimal predictor when both the bias B_t and the variance bound $\|\boldsymbol{q}_t\|^2$ are small. Since these quantities are both bounded by (functions of) r_t (see Properties 2, 3, and 4 of Lemma 1), this suggests that one can safely disregard Y_t when r_t is small.

Algorithm 1 The BBQ selective sampler

```
Parameters: $0 \le \kappa \le 1$
Initialization: Weight vector $\boldsymbol{w} = \boldsymbol{0}$
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 $\mathrm{SGN}(\boldsymbol{w}^\top \boldsymbol{x}_t)$;
  if $r_t > t^{-\kappa}$ then
    query label $y_t$,
    update $\boldsymbol{w}_t$ using $(\boldsymbol{x}_t, y_t)$ as in (1).
  end if
end for
```

According to our noise model, the label of $\boldsymbol{x}_t$ is harder to predict if $|\Delta_t|$ is small. For this reason, our regret bound is split into a cumulative regret on "big margin" steps t, where $|\Delta_t| \ge \varepsilon$, and "small margin" steps, where $|\Delta_t| < \varepsilon$. On one hand, we bound the regret on small margin steps simply by $\varepsilon\, T_\varepsilon$, where $T_\varepsilon = \bigl|\{1 \le t \le T \,:\, |\Delta_t| < \varepsilon\}\bigr|$. On the other hand, we show that the overall regret can be bounded in terms of the best possible choice of ε with no need for the algorithm to know this optimal value.

Theorem 1 *If BBQ is run with input $\kappa \in [0, 1]$ then its cumulative regret R_T after any number T of steps satisfies*

$$R_T \le \min_{0 < \varepsilon < 1} \Biggl(\varepsilon\, T_\varepsilon + (2 + e)\, \lceil 1/\kappa \rceil !\, \left(\frac{8}{\varepsilon^2}\right)^{1/\kappa} + \left(1 + \frac{2}{e}\right) \frac{8d}{\varepsilon^2} \ln\left(1 + \frac{N_T}{d}\right) \Biggr) \ .$$

The number of queried labels is $N_T = \mathcal{O}\left(d\, T^\kappa \ln T\right)$.

It is worth observing that the bounds presented here hold in the finite dimensional ($d < \infty$) case only. One can easily turn them to work in any RKHS after switching to an eigenvalue representation of the cumulative regret —e.g., by using the middle bound in Property 6 of Lemma 1 rather than the rightmost one, as we did in the proof below. This essentially corresponds to analyzing Algorithm 1 in a dual variable representation. A similar comment holds for Remark 1 and Theorem 2 below.

In the rest of the paper we denote by $\{\phi\}$ the indicator function of a Boolean predicate ϕ.

Proof: [of Theorem 1] Fix any $\varepsilon \in (0,1)$. As in (Cavallanti et al., 2009), we first observe that our label noise model allows us to upper bound the time-t regret $\mathbb{P}(Y_t\,\widehat{\Delta}_t < 0) - \mathbb{P}(Y_t\,\Delta_t < 0)$ as

$$\begin{aligned}\mathbb{P}(Y_t\,\widehat{\Delta}_t < 0) &- \mathbb{P}(Y_t\,\Delta_t < 0)\\ &\le \varepsilon\{|\Delta_t| < \varepsilon\} + \mathbb{P}\Big(\widehat{\Delta}_t\Delta_t \le 0,\, |\Delta_t| \ge \varepsilon\Big)\\ &\le \varepsilon\{|\Delta_t| < \varepsilon\} + \mathbb{P}\Big(\big|\widehat{\Delta}_t - \Delta_t\big| \ge \varepsilon\Big)\ .\end{aligned}$$

Hence the cumulative regret (2) can be split as follows:

$$R_T \le \varepsilon\,T_\varepsilon + \sum_{t=1}^{T} \mathbb{P}\Big(\big|\widehat{\Delta}_t - \Delta_t\big| \ge \varepsilon\Big)\ . \qquad (3)$$

We proceed by expanding the indicator of $\big|\widehat{\Delta}_t - \Delta_t\big| \ge \varepsilon$ with the introduction of the bias term B_t

$$\begin{aligned}\Big\{\big|\widehat{\Delta}_t - \Delta_t\big| \ge \varepsilon\Big\} &\le \Big\{\big|\widehat{\Delta}_t + B_t - \Delta_t\big| \ge \frac{\varepsilon}{2}\Big\}\\ &\quad + \Big\{|B_t| > \frac{\varepsilon}{2}\Big\}\ .\end{aligned}$$

Note that

$$\Big\{|B_t| > \frac{\varepsilon}{2}\Big\} \le \Big\{r_t > \frac{\varepsilon^2}{8}\Big\} \le e\exp\left(-\frac{\varepsilon^2}{8r_t}\right)$$

the first inequality deriving from a combination of Properties 2 and 3 in Lemma 1 and then overapproximating, whereas the second one uses $\{b<1\} \le e^{1-b}\ \forall\, b$. Moreover, by Properties 4 and 5 in Lemma 1, we have that

$$\mathbb{P}\Big(\big|\widehat{\Delta}_t + B_t - \Delta_t\big| \ge \frac{\varepsilon}{2}\Big) \le 2\exp\left(-\frac{\varepsilon^2}{8r_t}\right)\ .$$

We substitute this back into (3) and single out the steps where queries are issued. This gives

$$\begin{aligned}R_T \le \varepsilon\,T_\varepsilon &+ (2+e)\sum_{t\,:\,r_t \le t^{-\kappa}} \exp\left(-\frac{\varepsilon^2}{8r_t}\right)\\ &+ (2+e)\sum_{t\,:\,r_t > t^{-\kappa}} \exp\left(-\frac{\varepsilon^2}{8r_t}\right)\ .\end{aligned}$$

The second term is bounded as follows:

$$\begin{aligned}\sum_{t\,:\,r_t \le t^{-\kappa}} \exp\left(-\frac{\varepsilon^2}{8r_t}\right) &\le \sum_{t=1}^{T} \exp\left(-\frac{\varepsilon^2 t^\kappa}{8}\right)\\ &\le \int_0^\infty \exp\left(-\frac{\varepsilon^2 x^\kappa}{8}\right)dx = \frac{1}{\kappa}\Gamma(1/\kappa)\left(\frac{8}{\varepsilon^2}\right)^{1/\kappa}\end{aligned}$$

where $\Gamma(\cdot)$ is the Euler's Gamma function $\Gamma(x) = \int_0^\infty e^{-t}\,t^{x-1}\,dt$. We further bound $\frac{1}{\kappa}\Gamma(1/\kappa) \le \lceil 1/\kappa \rceil!$ using the monotonicity of Γ. For the third term we write

$$\begin{aligned}\sum_{t\,:\,r_t > t^{-\kappa}} \exp\left(-\frac{\varepsilon^2}{8r_t}\right) &\le \frac{8}{e\varepsilon^2}\sum_{t\,:\,r_t > t^{-\kappa}} r_t\\ &\le \frac{8d}{e\varepsilon^2}\ln\left(1 + \frac{N_T}{d}\right)\ .\end{aligned}$$

The first step uses the inequality $e^{-x} \le \frac{1}{ex}$ for $x > 0$, while the second step uses Property 6 in Lemma 1. Finally, in order to derive a bound on the number N_T of queried labels, we have

$$\begin{aligned}N_T \le \sum_{t\,:\,r_t > t^{-\kappa}} \frac{r_t}{t^{-\kappa}} &\le T^\kappa \sum_{t\,:\,r_t > t^{-\kappa}} r_t\\ &\le d\,T^\kappa \ln\left(1 + \frac{N_T}{d}\right)\end{aligned}$$

where for the last inequality we used, once more, Property 6 in Lemma 1. Hence, $N_T = \mathcal{O}\left(d\,T^\kappa \ln T\right)$, and this concludes the proof. □

It is important to observe that, if we disregard the margin term $\varepsilon\,T_\varepsilon$ (which is fully controlled by the adversary), the regret bound depends logarithmically on T for any constant $\kappa > 0$:

$$R_T \le \varepsilon\,T_\varepsilon + \mathcal{O}\left(\frac{1}{\varepsilon^{2/\kappa}} + \frac{d}{\varepsilon^2}\ln T\right)\ .$$

If κ is set to 1 then our bound on the number of queries N_T becomes vacuous, and the selective sampling algorithm essentially becomes fully supervised. This recovers the known regret bound for RLS in the fully supervised case, $R_T \le \varepsilon\,T_\varepsilon + \mathcal{O}\big((d\ln T)/\varepsilon^2\big)$.

Remark 1 *A randomized variant of BBQ exists that queries label y_t with independent probability $r_t^{(1-\kappa)/\kappa} \in [0,1]$. Through a similar bias-variance analysis as the one in Theorem 1 above, one can show that* in expectation *(over the internal randomization of this algorithm) the cumulative regret R_T is bounded by $\min_{0<\varepsilon<1}\left(\varepsilon\,T_\varepsilon + \mathcal{O}\left(\frac{L}{\varepsilon^{2/\kappa}}\right)\right)$ while the number of queried labels N_T is $\mathcal{O}\left(T^\kappa\,L^{1-\kappa}\right)$, being $L = d\ln T$. This bound is similar (though generally incomparable) to the one of Theorem 1.*

4. A parametric performance guarantee

In the proof of Theorem 1 the quantity ε acts as a threshold for the cumulative regret, which is split into a sum over steps t such that $|\Delta_t| < \varepsilon$ (where the regret

Algorithm 2 The parametric BBQ selective sampler

```
**Parameters:** $0 < \varepsilon, \delta < 1$
**Initialization:** weight vector $\boldsymbol{w} = \boldsymbol{0}$
**for** each time step $t = 1, 2, \ldots$ **do**
  observe instance $\boldsymbol{x}_t \in \mathbb{R}^d$;
  predict label $y_t \in \{-1, +1\}$ with $\textsc{sgn}(\boldsymbol{w}^\top \boldsymbol{x}_t)$
  **if** $\big[\varepsilon - r_t - s_t\big]_+ < \|\boldsymbol{q}_t\| \sqrt{2 \ln \frac{t(t+1)}{2\delta}}$ **then**
    query label $y_t$
    update $\boldsymbol{w}_t$ using $(\boldsymbol{x}_t, y_t)$ as in (1)
  **end if**
**end for**
```

grows by less than ε) and a sum over the remaining steps. Most technicalities in the proof are due to the fact that the final bound depends on the optimal choice of this ε, which the algorithm need not know. On the other hand, if a specific value for ε is provided in input to the algorithm, then the cumulative regret over steps t such that $|\Delta_t| \geq \varepsilon$ can be bounded by *any constant* $\delta > 0$ using only order of $(d/\varepsilon^2)\ln(T/\delta)$ queries. In particular, when $\min_t |\Delta_t| \geq \varepsilon$, the above logarithmic bound implies that the per-step regret vanishes *exponentially fast* as a function of the number of queries.

As we stated in the introduction, this result cannot be obtained as an easy consequence of known results, due to the adversarial nature of the instance sequence.

We now develop the above argument for a practically motivated variant of our BBQ selective sampler. Let us disregard for a moment the bias term B_t. In order to guarantee that $\big|\widehat{\Delta}_t - \Delta_t\big| \leq \varepsilon$ holds when no query is issued, it is enough to observe that Property 5 of Lemma 1 implies that $\big|\widehat{\Delta}_t - \Delta_t\big| \leq \sqrt{2 r_t \ln \frac{2}{\delta}}$ with probability at least $1 - \delta$. This immediately delivers a rule prescribing that no query be issued at time t when $\sqrt{2 r_t \ln \frac{2}{\delta}} \leq \varepsilon$. A slightly more involved condition, one that better exploits the inequalities of Lemma 1, allows us to obtain a significantly improved practical performance. This results in the algorithm described in Figure 2. The algorithm, called Parametric BBQ, takes in input two parameters ε and δ, and issues a query at time t whenever[1]

$$\big[\varepsilon - r_t - s_t\big]_+ < \|\boldsymbol{q}_t\| \sqrt{2 \ln \frac{2t(t+1)}{\delta}} \ . \qquad (4)$$

Theorem 2 *If Parametric BBQ is run with input $\varepsilon, \delta \in (0,1)$ then:*

1. *with probability at least $1 - \delta$, $\big|\widehat{\Delta}_t - \Delta_t\big| \leq \varepsilon$ holds on all time steps t when no query is issued;*
2. *the number N_T of queries issued after any number T of steps is bounded as*

$$N_T = \mathcal{O}\left(\frac{d}{\varepsilon^2}\left(\ln \frac{T}{\delta}\right) \ln \frac{\ln(T/\delta)}{\varepsilon}\right) \ .$$

This theorem has been phrased so as to make it easier to compare to a corresponding result in (Strehl & Littman, 2008) for the KWIK ("Knows What It Knows") framework. In that paper, the authors use a modification of Auer's upper confidence linear regression algorithm for associative reinforcement learning (Auer, 2002). This modification allows them to compete against any adaptive adversarial strategy generating instance vectors $\boldsymbol{x}_t$, but it yields the significantly worse bound $\widetilde{\mathcal{O}}\big(d^3/\varepsilon^4\big)$ on N_T (in the KWIK setting N_T is the number of times the prediction algorithm answers "I don't know"). Besides, their strategy seems to work in the finite dimensional ($d < \infty$) case only. In contrast, Parametric BBQ works against an oblivious adversary only, but it has the better bound $N_T = \widetilde{\mathcal{O}}\big(d/\varepsilon^2\big)$, with the $\widetilde{\mathcal{O}}$ notation hiding a mild (logarithmic) dependence on T. Moreover, Parametric BBQ can be readily run in infinite ($d = \infty$) dimensional RKHS —recall the comment before the proof of Theorem 1. In fact, this is a quite important feature: the real-world experiments of Section 5 needed kernels in order to either attain a good empirical performance (on Adult) or use a reasonable amount of computational resources (on RCV1).

Remark 2 *The bound on the number of queried labels in Theorem 2 is optimal up to logarithmic factors. In fact, it is possible to prove that there exists a sequence $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ of instances and a number $\varepsilon_0 > 0$ such that: for all $\varepsilon \leq \varepsilon_0$ and for any learning algorithm that issues $N = \mathcal{O}(d/\varepsilon^2)$ queries there exists a target vector $\boldsymbol{u} \in \mathbb{R}^d$ and a time step $t = \Omega\big(d/\varepsilon^2\big)$ for which the estimate $\widehat{\Delta}_t$ computed by the algorithm for $\Delta_t = \boldsymbol{u}^\top \boldsymbol{x}_t$ has the property $\mathbb{P}\big(|\widehat{\Delta}_t - \Delta_t| > \varepsilon\big) = \Omega\big(1\big)$. Hence, at least $\Omega\big(d/\varepsilon^2\big)$ queries are needed to learn any target hyperplane with arbitrarily small accuracy and arbitrarily high confidence.*

Proof: [Theorem 2] Let $\mathcal{I} \subseteq \{1, \ldots, T\}$ be the set of time steps when a query is issued. Then, using Property 2 of Lemma 1 we can write

$$\begin{aligned}
&\sum_{t \notin \mathcal{I}} \Big\{\big|\widehat{\Delta}_t - \Delta_t\big| > \varepsilon\Big\} \\
&\qquad \leq \sum_{t \notin \mathcal{I}} \Big\{\big|\widehat{\Delta}_t + B_t - \Delta_t\big| > \varepsilon - |B_t|\Big\} \\
&\qquad \leq \sum_{t \notin \mathcal{I}} \Big\{\big|\widehat{\Delta}_t + B_t - \Delta_t\big| > [\varepsilon - r_t - s_t]_+\Big\} \ .
\end{aligned}$$

[1] Here and throughout $[x]_+ = \max\{0, x\}$.

We first take expectations on both sides, and then apply Property 5 of Lemma 1 along with condition (4) rewritten as follows

$$2\exp\left(-\frac{\left([\varepsilon - r_t - s_t]_+\right)^2}{2\,\|\boldsymbol{q}_t\|^2}\right) \le \frac{\delta}{t(t+1)} \ .$$

This gives

$$\begin{aligned}
&\sum_{t\notin\mathcal{I}} \mathbb{P}\Big(\big|\widehat{\Delta}_t - \Delta_t\big| > \varepsilon\Big) \\
&\qquad \le \sum_{t\notin\mathcal{I}} \mathbb{P}\Big(\big|\widehat{\Delta}_t + B_t - \Delta_t\big| > [\varepsilon - r_t - s_t]_+\Big) \\
&\qquad \le \sum_{t\notin\mathcal{I}} 2\exp\left(-\frac{\left(|\varepsilon - r_t - s_t|_+\right)^2}{2\,\|\boldsymbol{q}_t\|^2}\right) \\
&\qquad \le \sum_{t\notin\mathcal{I}} \frac{\delta}{t(t+1)} \le \sum_{t=1}^{\infty} \frac{\delta}{t(t+1)} = \delta \ .
\end{aligned}$$

In order to derive a bound on the number N_T of queried labels, we proceed as follows. For every step $t \in \mathcal{I}$ in which a query was issued we can write

$$\begin{aligned}
\varepsilon - r_t - \sqrt{r_t} &\le \varepsilon - r_t - s_t \le \big[\varepsilon - r_t - s_t\big]_+ \\
&\le \|\boldsymbol{q}_t\| \sqrt{2\ln\frac{2t(t+1)}{\delta}} \le \sqrt{2\,r_t \ln\frac{2t(t+1)}{\delta}}
\end{aligned}$$

where we used Properties 3 and 4 of Lemma 1. Solving for r_t and overapproximating we obtain

$$r_t \ge \frac{\varepsilon^2}{2\varepsilon + \left(1 + \sqrt{2\ln\frac{2t(t+1)}{\delta}}\right)^2} \ . \qquad (5)$$

Similarly to the proof of Theorem 1, we then write

$$N_T \min_{t\in\mathcal{I}} r_t \le \sum_{t\in\mathcal{I}} r_t \le d\ln\left(1 + \frac{N_T}{d}\right) \ .$$

Using (5) we get $N_T = \mathcal{O}\left(\frac{d}{\varepsilon^2}\left(\ln\frac{T}{\delta}\right)\ln\frac{\ln(T/\delta)}{\varepsilon}\right)$. □

5. Experiments

In this section we report on preliminary experiments with the Parametric BBQ algorithm. The first test is a synthetic experiment to validate the model. We generated 10,000 random examples on the unit circle in $\mathbb{R}^2$. The labels of these examples were generated according to our noise model (see Section 2) using a randomly selected hyperplane $\boldsymbol{u}$ with unit norm. We then set $\delta = 0.1$ and analyzed the behavior of the algorithm with various settings of $\varepsilon > 0$ and using a linear kernel. In Figure 1 the jagged blue line represents the maximum error over the example sequence, i.e., $\max_t\big|\widehat{\Delta}_t - \Delta_t\big|$. (Although we stopped the plot at $\varepsilon = 1$, note that the maximum error is dominated by $|\widehat{\Delta}_t|$, which can be of the order of $\sqrt{N_t}$.) As predicted by Theorem 2, the maximum error remains below the straight line $y = \varepsilon$ (the maximum error predicted by the theory). In the same plot, the decreasing green line shows the number of queried labels, which closely follows the curve ε^{-2} predicted by the theory.

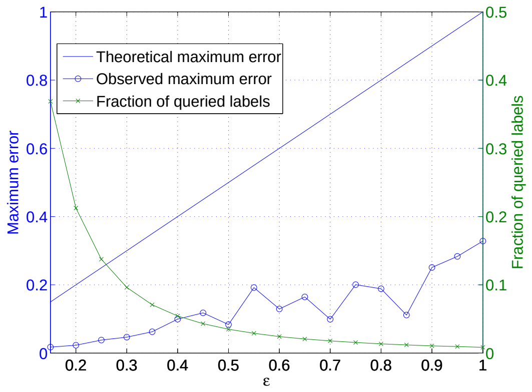

Figure 1. Maximum error (jagged blue line) and number of queried labels (decreasing green line) on a synthetic dataset for Parametric BBQ with $\delta = 0.1$ and $0.15 \le \varepsilon \le 1$. The straight blue line is the theoretical upper bound on the maximum error provided by the theory.

This initial test reveals that the algorithm is dramatically underconfident, i.e., it is a lot more precise than it thinks. Moreover, the actual error is rather insensitive to the choice of ε. In order to leverage on this, we ran the remaining tests using Parametric BBQ with a more extreme setting of parameters. Namely, we changed the query condition (the "if" condition in Algorithm 2) to

$$\big[1 - r_t - s_t\big]_+ < \|\boldsymbol{q}_t\| \sqrt{2\ln\frac{2}{\delta}} \qquad \text{for} \quad 0 < \delta < 1 \ .$$

This amounts to setting the desired error to a default value of $\varepsilon = 1$ while making the number of queried labels independent of T.

With the above setting, we compared Parametric BBQ to the second-order version of the label-efficient classifier (SOLE) of (Cesa-Bianchi et al., 2006b). This is a mistake-driven RLS algorithm that queries the label of the current instance with probability $1/(1 + b\,|\widehat{\Delta}_t|)$, where $b > 0$ is a parameter and $\widehat{\Delta}_t$ is the RLS margin. The other baseline algorithm is a vanilla sampler (called Random in the plots) that asks labels at random with constant probability $0 < p < 1$. Recall that SOLE does not come with a guaranteed bound on the

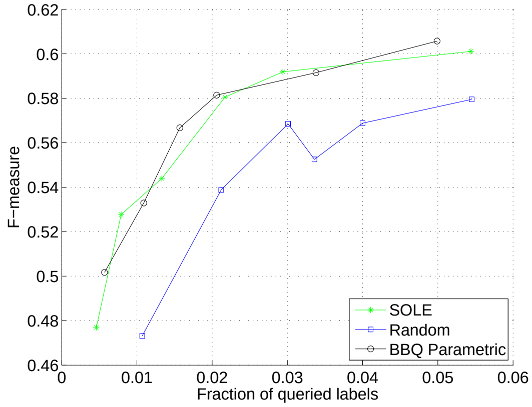

Figure 2. F-measure against fraction of queried labels on the a9a dataset (32,561 examples in random order). The plotted curves are averages over 10 random shuffles.

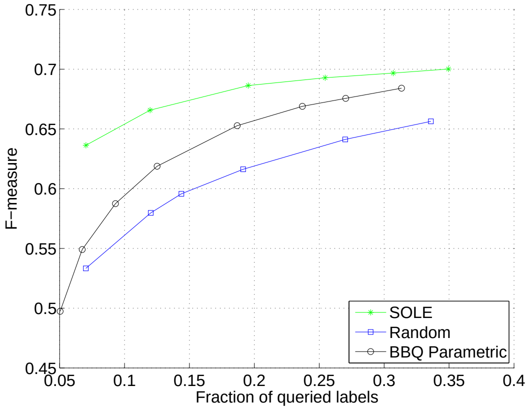

Figure 3. F-measure against fraction of queried labels averaged over the 50 most frequent categories of RCV1 (first 40,000 examples in chronological order).

number of queried labels. Random, on the other hand, has the simple expectation bound $\mathbb{E}[N_T] = p\,T$.

For each algorithm we plot the F-measure (harmonic mean of precision and recall) against the fraction of queried labels. We control the fraction of queried labels by changing the parameters of the three algorithms (δ for Parametric BBQ, b for SOLE, and p for Random).

For the first real-world experiment we chose a9a[2], a subset of the census-income (Adult) database with 32,561 binary-labeled examples and 123 features. In order to bring all algorithms to a reasonable performance level, we used a Gaussian kernel with $\sigma^2 = 12.5$. The plots (Figure 2) show that less than 6% queries are enough for the three algorithms to saturate their performance. In the whole query range Parametric BBQ is consistently slightly better than SOLE, while Random has the worst performance.

For our second real-world experiment we used the first 40,000 newswire stories in chronological order from the Reuters Corpus Volume 1 dataset (RCV1). Each newsstory of this corpus is tagged with one or more labels from a set of 102 classes. A standard TF-IDF bag-of-words encoding was used to obtain 138,860 features. We considered the 50 most populated classes and trained 50 classifiers one-vs-all using a linear kernel. Earlier experiments, such as those reported in (Cesa-Bianchi et al., 2006b), show that RLS-based algorithms perform best on RCV1 when run in a mistake driven fashion. For this reason, on this dataset we used a mistake-driven variant of Parametric BBQ, storing a queried label only when it is wrongly predicted. Figure 3 shows the (macro)average F-measure plotted against the average fraction of queried labels, where averages are computed over the 50 classifiers. Here the algorithms need over 35% of labels to saturate. Moreover, Parametric BBQ performs worse than SOLE, although still better than Random.

Since SOLE and Parametric BBQ are both based on the mistake-driven RLS classifier, any difference of performance is due to their different query conditions: SOLE is margin-based, while Parametric BBQ uses $\|\boldsymbol{q}_t\|$ and related quantities. Note that, unlike the margin, $\boldsymbol{q}_t$ *does not* depend on the queried labels, but only on the correlation between their corresponding instances. This fact, which helped us a lot in the analysis of BBQ, could make a crucial difference between domains like RCV1 (where instances are extremely sparse) and Adult (where instances are relatively dense). More experimental work is needed in order to settle this conjecture.

6. Conclusions and ongoing research

We have introduced a new family of online algorithms, the BBQ family, for selective sampling under (oblivious) adversarial environments. These algorithms naturally interpolate between fully supervised and fully unsupervised learning scenarios. A parametric variant (Parametric BBQ) of our basic algorithm is designed to work in a weakened KWIK framework (Li et al., 2008; Strehl & Littman, 2008) with improved bounds on the number of queried labels.

We have made preliminary experiments. First, we validated the theory on an artificially generated dataset. Second, we compared a variant of Parametric BBQ to algorithms with similar guarantees, with encouraging results.

[2] www.csie.ntu.edu.tw/∼cjlin/libsvmtools/

A few issues we are currently working on are the following. First, we are trying to see if a sharper analysis of BBQ exists which allows one to prove a regret bound of the form $\varepsilon T_\varepsilon + \frac{d \ln T}{\varepsilon^2}$ when $N_T = \mathcal{O}(d \ln T)$. This bound would be a worst-case analog of the bound Cavallanti et al. (2009) have obtained in an i.i.d. setting. This improvement is likely to require refined bounds on bias and variance of our estimators. Moreover, we would like to see if it is possible either to remove the $\ln T$ dependence on the bound on N_T in Theorem 2 or to make Parametric BBQ work in adaptive adversarial environments (presumably at the cost of looser bounds on N_T). In fact, it is currently unclear to us how a direct covering argument could be applied in Theorem 2 which avoids the need for a conditionally independent structure of the involved random variables.

On the experimental side, we are planning to perform a more thorough empirical investigation using additional datasets. In particular, since our algorithms can also be viewed as memory bounded procedures, we would like to see how they perform when compared to budget-based algorithms, such as those in (Weston et al., 2005; Dekel et al., 2007; Cavallanti et al., 2007; Orabona et al., 2008).

Finally, since our algorithms can be easily adapted to solve regression tasks, we are planning to test the BBQ family on standard regression benchmarks.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments. The third author is supported by the DIRAC project under EC grant FP6-0027787. All authors acknowledge partial support by the PASCAL2 NoE under EC grant FP7-216886. This publication only reflects the authors' views.

References

Auer, P. (2002). Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, *3*, 397–422.

Balcan, M., Beygelzimer, A., & Langford, J. (2006). Agnostic active learning. *Proc. of the 23rd International Conference on Machine Learning* (pp. 65–72).

Balcan, M., Broder, A., & Zhang, T. (2007). Margin-based active learning. *Proceedings of the 20th Annual Conference on Learning Theory* (pp. 35–50).

Cavallanti, G., Cesa-Bianchi, N., & Gentile, C. (2007). Tracking the best hyperplane with a simple budget Perceptron. *Machine Learning*, *69*, 143–167.

Cavallanti, G., Cesa-Bianchi, N., & Gentile, C. (2009). Linear classification and selective sampling under low noise conditions. In *Advances in Neural Information Processing Systems 21* (pp. 249–256).

Cesa-Bianchi, N., Gentile, C., & Zaniboni, L. (2006a). Incremental algorithms for hierarchical classification. *Journal of Machine Learning Research*, *7*, 31–54.

Cesa-Bianchi, N., Gentile, C., & Zaniboni, L. (2006b). Worst-case analysis of selective sampling for linear classification. *Journal of Machine Learning Research*, *7*, 1025–1230.

Cohn, R., Atlas, L., & Ladner, R. (1990). Training connectionist networks with queries and selective sampling. In *Advances in Neural Information Processing Systems 2* (pp. 566–573).

Dasgupta, S., Hsu, D., & Monteleoni, C. (2008). A general agnostic active learning algorithm. In *Advances in Neural Information Processing Systems 21* (pp. 353–360).

Dasgupta, S., Kalai, A. T., & Monteleoni, C. (2005). Analysis of perceptron-based active learning. *Proceedings of the 18th Annual Conference on Learning Theory* (pp. 249–263).

Dekel, O., Shalev-Shwartz, S., & Singer, Y. (2007). The Forgetron: A kernel-based Perceptron on a budget. *SIAM Journal on Computing*, *37*, 1342–1372.

Freund, Y., Seung, S., Shamir, E., & Tishby, N. (1997). Selective sampling using the query by committee algorithm. *Machine Learning*, *28*, 133–168.

Li, L., Littman, M., & Walsh, T. (2008). Knows what it knows: a framework for self-aware learning. *Proceedings of the 25th International Conference on Machine Learning* (pp. 568–575).

Orabona, F., Keshet, J., & Caputo, B. (2008). The Projectron: a bounded kernel-based Perceptron. *Proceedings of the 25th International Conference on Machine Learning* (pp. 720–727).

Strehl, A., & Littman, M. (2008). Online linear regression and its application to model-based reinforcement learning. In *Advances in Neural Information Processing Systems 20* (pp. 631–638).

Weston, J., Bordes, A., & Bottou, L. (2005). Online (and offline) on an even tighter budget. *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics* (pp. 413–420).