An Efficient Bandit Algorithm for $\sqrt{T}$-Regret in Online Multiclass Prediction?

Jacob Abernethy[*]
Division of Computer Science
UC Berkeley
jake@cs.berkeley.edu

Alexander Rakhlin
Department of Statistics
University of Pennsylvania
rakhlin@wharton.upenn.edu

Consider a sequence of examples $(\mathbf{x}_t, y_t)$ for $t = 1, \ldots, T$ where $\mathbf{x}_t \in \mathbb{R}^n$ and $y_t \in [K]$, where the goal of a Learner is to predict the class y_t from the input $\mathbf{x}_t$. In the more common *full-information* setting, the Learner observes the true class y_t after making her prediction $\hat{y}_t$. In the present open problem, however, we will consider the so-called *bandit* setting: after predicting $\hat{y}_t$, the Learner is only told "correct" or "incorrect", her feedback being a single bit $\mathbf{1}[\hat{y}_t \neq y_t]$.

We assume that the Learner's hypothesis class is the set of K-tuples of vectors $W = \langle \mathbf{w}_1, \ldots, \mathbf{w}_K \rangle$ where $\mathbf{w}_i \in \mathbb{R}^n$ (we can think of W as the $K \times n$ hypothesis matrix). Given an instance $\mathbf{x}_t$, such a hypothesis will produce a K-tuple of "scores" $\langle \mathbf{w}_1 \cdot \mathbf{x}_t, \ldots, \mathbf{w}_K \cdot \mathbf{x}_t \rangle$, and the Learner's prediction will be the class with the largest score:

$$\hat{y}_t = \arg\max_{k \in [K]} \mathbf{w}_k \cdot \mathbf{x}_t.$$

While ideally we would like to minimize the 0-1 loss suffered by the Learner, for computational and other reasons it is preferable to consider convex loss functions. A natural choice used in Kakade et al. [2008] is the *multi-class hinge loss*:

$$\ell(W, (\mathbf{x}_t, y_t)) = \max_{k \in [K] \setminus \{y_t\}} [1 - \mathbf{w}_{y_t} \cdot \mathbf{x}_t + \mathbf{w}_k \cdot \mathbf{x}_t]_+,$$

where $W := \langle \mathbf{w}_1, \ldots, \mathbf{w}_k \rangle$. Other suitable loss functions $\ell(\cdot, \cdot)$ may also be used. The ultimate goal of the Learner is to minimize *regret*,

$$\text{Regret} := \sum_{t=1}^{T} \ell(W_t, (\mathbf{x}_t, y_t)) - \min_{W^*} \sum_{t=1}^{T} \ell(W^*, (\mathbf{x}_t, y_t)).$$

The *comparator* is typically restricted to a ball (in some norm) of a fixed diameter D, and the regret will depend on D (similar to a margin bound). We note that such a regret bound should hold for any sequence of examples; indeed, they may even be adversarially chosen.

In Kakade et al. [2008], the authors present an algorithm known as The Banditron, which modifies a full-information algorithm, the Multiclass Perceptron [Fink et al., 2006, Crammer and Singer, 2003], for the bandit setting. While the algorithm has many desirable properties, foremost among these is its efficiency, it is shown to have a regret bound on the order[1] $O(T^{2/3})$, which is known to be suboptimal.

[*]Supported by a Yahoo! PhD Fellowship, DARPA grant FA8750-05-2-0249, and NSF grant DMS-0707060.

[1]It is noted in Kakade et al. [2008], however, that in some "low-noise" cases this bound can be improved to $O(\sqrt{T})$.

Open Problem: *Does there exist an efficient multiclass learning algorithm for the bandit setting that achieves expected regret on the order of $O(\sqrt{T})$? The regret bound can be shown with respect to a different, yet reasonable, loss function.*

The Banditron uses a common trick for constructing bandit algorithms: with probability ϵ, randomly sample a class k and use this to construct an unbiased estimate of the true loss function, and with probability $(1 - \epsilon)$ predict according to the current hypothesis. This method, while appealing due to its simplicity, is doomed to lead to $O(T^{2/3})$ in many online learning problems. This is discussed in Dani and Hayes [2006] and a similar lower bound is shown in Cesa-Bianchi et al. [2006].

Given the latter observation, one might conjecture that $T^{2/3}$-regret is is the best possible for any algorithm. But indeed this is false: we can utilize the Exp4 algorithm of Auer et al. [2003] to obtain an $O(\sqrt{T \log T})$ regret for the $0 - 1$ loss. This reduction consists of discretizing the space of possible hypothesis matrices W and treating each point as an "expert". The discretization will lead to $O(T^{n/2})$ many points, yet the Exp4 regret bound depends only logarithmically on the number of such experts. The downside of this approach, of course, is its lack of efficiency.

There have been a few results that have broken the $\sqrt{T}$ regret boundary for bandit problems in the adversarial setting. First, Auer et al. [2003] exhibited an efficient algorithm for the so-called multi-armed bandit problem. Later, Dani et al. [2008] showed the first $O(\sqrt{T})$ bandit algorithm that worked in a much more general setting in which the decision/hypothesis set is convex and the loss functions are linear, known as *online linear optimization*, although the algorithm is not efficient in all cases. An efficient algorithm was later found in Abernethy et al. [2008]. These algorithms share a number of important components:

- A randomized sampling scheme which simultaneously explores and exploits
- The sampling scheme is coupled with a carefully constructed unbiased estimate of the true loss function
- The learning algorithm is specifically designed to handle high-variance loss function estimates. This is required because, as the algorithm becomes increasingly certain about its decisions, it will need to spend less

time exploring, leading to estimates with high variance. (The results of Abernethy et al. [2008] emphasize the use of heavier regularization in "regions" of the hypothesis space where the variance grows.)

One would like to generalize these recent results to the multiclass problem, but it is not immediately clear how this can be achieved. Given a current hypothesis matrix $W = \langle \mathbf{w}_1, \ldots, \mathbf{w}_k \rangle$ and an instance $\mathbf{x}_t$, what is a natural sampling scheme on the K classes that performs "simultaneous explore/exploit"?

We briefly present an approach that we attempted, unsuccessfully, and which may provide some insights to the curious reader. Given $W = \langle \mathbf{w}_1, \ldots, \mathbf{w}_k \rangle$ and an instance $\mathbf{x}_t$, we can define a distribution over the classes $[K]$ as

$$P(k) = p_k(W, \mathbf{x}_t) := \frac{\exp(\alpha \mathbf{w}_k \cdot \mathbf{x}_t)}{\sum_{j \in [K]} \exp(\alpha \mathbf{w}_j \cdot \mathbf{x}_t)}$$

where α is some parameter. A natural loss function is the log loss,

$$\begin{aligned} \ell(W, (\mathbf{x}, y)) &:= \frac{-1}{\alpha} \log \left(\frac{\exp(\alpha \mathbf{w}_y \cdot \mathbf{x})}{\sum_{j \in [K]} \exp(\alpha \mathbf{w}_j \cdot \mathbf{x})} \right) \\ &= -\mathbf{w}_y \cdot \mathbf{x} + \frac{1}{\alpha} \log \sum_{j \in [K]} e^{\alpha \mathbf{w}_j \cdot \mathbf{x}} \end{aligned}$$

From this sampling scheme, we may now construct an unbiased estimate of the gradient of the true loss function. If we sample $I \in [K]$ according to the proposed distribution $p(W, \mathbf{x}_t)$, then the gradient estimate with respect to $\mathbf{w}_k$, i.e. the kth row of W, can be defined as

$$\hat{\nabla}_k \ell = \begin{cases} 0, & \text{if } k \neq I \\ \mathbf{x}_t \left(1 - \frac{\mathbf{1}[k = y_t]}{p_k(W, \mathbf{x}_t)} \right) & \text{if } k = I. \end{cases}$$

This gradient estimate can be used to perform a parameter update, as is commonly done in online learning, and a regret bound can be achieved. Unfortunately, the above estimate scales with the inverse of $p_k(W, \mathbf{x})$, which will need to be $O(T^{-1/2})$ in some cases to ensure low regret. One would like to control this, but it is not clear how this can be done with currently known techniques, and it is here where we believe a significant difficulty lies.

References

J. Abernethy, E. Hazan, and A. Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *Proceedings of the 21st Annual Conference on Learning Theory (COLT)*, 2008.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM JOURNAL ON COMPUTING*, 32(1):48–77, 2003.

N. Cesa-Bianchi, G. Lugosi, and G. Stoltz. Regret minimization under partial monitoring. *MATHEMATICS OF OPERATIONS RESEARCH*, 31(3):562, 2006.

K. Crammer and Y. Singer. Ultraconservative online algorithms for multiclass problems. *The Journal of Machine Learning Research*, 3:951–991, 2003.

V. Dani and T. P. Hayes. Robbing the bandit: Less regret in online geometric optimization against an adaptive adversary. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 937–943. ACM New York, NY, USA, 2006.

V. Dani, T. Hayes, and S. M. Kakade. The price of bandit information for online optimization. *Advances in Neural Information Processing Systems*, 20, 2008.

M. Fink, S. Shalev-Shwartz, Y. Singer, and S. Ullman. Online multiclass learning by interclass hypothesis sharing. In *Proceedings of the 23rd international conference on Machine learning*, pages 313–320. ACM New York, NY, USA, 2006.

S. M. Kakade, S. Shalev-Shwartz, and A. Tewari. Efficient bandit algorithms for online multiclass prediction. In *Proceedings of the 25th international conference on Machine learning*, pages 440–447. ACM New York, NY, USA, 2008.