Theoretical Computer Science 519 (2014) 4–28

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Adaptive and optimal online linear regression on ℓ^1-balls

Sébastien Gerchinovitz [a,*,1], Jia Yuan Yu [b]

[a] École Normale Supérieure, 45 rue d'Ulm, 75005 Paris, France
[b] IBM Research, Damastown Technology Campus, Dublin 15, Ireland

ARTICLE INFO

Keywords:
Online learning
Linear regression
Adaptive algorithms
Minimax regret

ABSTRACT

We consider the problem of online linear regression on individual sequences. The goal in this paper is for the forecaster to output sequential predictions which are, after T time rounds, almost as good as the ones output by the best linear predictor in a given ℓ^1-ball in $\mathbb{R}^d$. We consider both the cases where the dimension d is small and large relative to the time horizon T. We first present regret bounds with optimal dependencies on d, T, and on the sizes U, X and Y of the ℓ^1-ball, the input data and the observations. The minimax regret is shown to exhibit a regime transition around the point $d = \sqrt{T}UX/(2Y)$. Furthermore, we present efficient algorithms that are adaptive, i.e., that do not require the knowledge of U, X, Y, and T, but still achieve nearly optimal regret bounds.

1. Introduction

In this paper, we consider the problem of online linear regression against arbitrary sequences of input data and observations, with the objective of being competitive with respect to the best linear predictor in an ℓ^1-ball of arbitrary radius. This extends the task of convex aggregation. We consider both low- and high-dimensional input data. Indeed, in a large number of contemporary problems, the available data can be high-dimensional—the dimension of each data point is larger than the number of data points. Examples include analysis of DNA sequences, collaborative filtering, astronomical data analysis, and cross-country growth regression. In such high-dimensional problems, performing linear regression on an ℓ^1-ball of small diameter may be helpful if the best linear predictor is sparse. Our goal is, in both low and high dimensions, to provide online linear regression algorithms along with bounds on ℓ^1-balls that characterize their robustness to worst-case scenarios.

1.1. Setting

We consider the online version of linear regression, which unfolds as follows. First, the environment chooses a sequence of observations $(y_t)_{t\geqslant 1}$ in $\mathbb{R}$ and a sequence of input vectors $(\boldsymbol{x}_t)_{t\geqslant 1}$ in $\mathbb{R}^d$, both initially hidden from the forecaster. At each time instant $t \in \mathbb{N}^* = \{1, 2, \ldots\}$, the environment reveals the data $\boldsymbol{x}_t \in \mathbb{R}^d$; the forecaster then gives a prediction $\widehat{y}_t \in \mathbb{R}$; the environment in turn reveals the observation $y_t \in \mathbb{R}$; and finally, the forecaster incurs the square loss $(y_t - \widehat{y}_t)^2$. The dimension d can be either small or large relative to the number T of time steps: we consider both cases.

In the sequel, $\boldsymbol{u} \cdot \boldsymbol{v}$ denotes the standard inner product between $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$, and we set $\|\boldsymbol{u}\|_\infty \triangleq \max_{1\leqslant j\leqslant d} |u_j|$ and $\|\boldsymbol{u}\|_1 \triangleq \sum_{j=1}^d |u_j|$. The ℓ^1-ball of radius $U > 0$ is the following bounded subset of $\mathbb{R}^d$:

$$B_1(U) \triangleq \left\{\boldsymbol{u} \in \mathbb{R}^d \colon \|\boldsymbol{u}\|_1 \leqslant U\right\}.$$

* Corresponding author.
E-mail addresses: sebastien.gerchinovitz@ens.fr (S. Gerchinovitz), jiayuanyu@ie.ibm.com (J.Y. Yu).

[1] This research was carried out within the INRIA project CLASSIC hosted by École Normale Supérieure and CNRS.

0304-3975/$ – see front matter
http://dx.doi.org/10.1016/j.tcs.2013.09.024

Given a fixed radius $U > 0$ and a time horizon $T \geqslant 1$, the goal of the forecaster is to predict almost as well as the best linear forecaster in the reference set $\{\boldsymbol{x} \in \mathbb{R}^d \mapsto \boldsymbol{u} \cdot \boldsymbol{x} \in \mathbb{R} \colon \ \boldsymbol{u} \in B_1(U)\}$, i.e., to minimize the regret on $B_1(U)$ defined by

$$\sum_{t=1}^{T} (y_t - \widehat{y}_t)^2 - \min_{\boldsymbol{u} \in B_1(U)} \left\{ \sum_{t=1}^{T} (y_t - \boldsymbol{u} \cdot \boldsymbol{x}_t)^2 \right\}.$$

We shall present algorithms along with bounds on their regret that hold uniformly over all sequences[2] $(\boldsymbol{x}_t, y_t)_{1 \leqslant t \leqslant T}$ such that $\|\boldsymbol{x}_t\|_\infty \leqslant X$ and $|y_t| \leqslant Y$ for all $t = 1, \ldots, T$, where $X, Y > 0$. These regret bounds depend on four important quantities: U, X, Y, and T, which may be known or unknown to the forecaster.

1.2. Contributions and related works

In the next paragraphs we detail the main contributions of this paper in view of related works in online linear regression.

Our first contribution (Section 2) consists of a minimax analysis of online linear regression on ℓ^1-balls in the arbitrary sequence setting. We first provide a refined regret bound expressed in terms of Y, d, and a quantity $\kappa = \sqrt{T} U X / (2dY)$. This quantity κ is used to distinguish two regimes: we show a distinctive regime transition[3] at $\kappa = 1$ or $d = \sqrt{T} U X / (2Y)$. Namely, for $\kappa < 1$, the regret is of the order of $dY^2 \kappa$ (proportional to $\sqrt{T}$), whereas it is of the order of $dY^2 \ln \kappa$ (proportional to $\ln T$) for $\kappa > 1$.

The derivation of this regret bound partially relies on a Maurey-type argument used under various forms with i.i.d. data, e.g., in [1–4] (see also [5]). We adapt it in a straightforward way to the deterministic setting. Therefore, this is yet another technique that can be applied to both the stochastic and individual sequence settings.

Unsurprisingly, the refined regret bound mentioned above matches the optimal risk bounds for stochastic settings[4] [6,2] (see also [7]). Hence, linear regression is just as hard in the stochastic setting as in the arbitrary sequence setting. Using the standard online to batch conversion, we make the latter statement more precise by establishing a lower bound for all κ at least of the order of $\sqrt{\ln d}/d$. This lower bound extends those of [8,9], which only hold for small κ of the order of $1/d$.

The algorithm achieving our minimax regret bound is both computationally inefficient and non-adaptive (i.e., it requires prior knowledge of the quantities U, X, Y, and T that may be unknown in practice). Those two issues were first overcome by [10] via an automatic tuning termed *self-confident* (since the forecaster somehow trusts himself in tuning its parameters). They indeed proved that the self-confident p-norm algorithm with $p = 2 \ln d$ and tuned with U has a cumulative loss $\widehat{L}_T = \sum_{t=1}^{T} (y_t - \widehat{y}_t)^2$ bounded by

$$\begin{aligned} \widehat{L}_T &\leqslant L_T^* + 8UX\sqrt{(e \ln d)\, L_T^*} + (32e \ln d)\, U^2 X^2 \\ &\leqslant 8UXY\sqrt{eT \ln d} + (32e \ln d)\, U^2 X^2, \end{aligned}$$

where $L_T^* \triangleq \min_{\{\boldsymbol{u} \in \mathbb{R}^d \colon \ \|\boldsymbol{u}\|_1 \leqslant U\}} \sum_{t=1}^{T} (y_t - \boldsymbol{u} \cdot \boldsymbol{x}_t)^2 \leqslant TY^2$. This algorithm is efficient, and our lower bound in terms of κ shows that it is optimal up to logarithmic factors in the regime $\kappa \leqslant 1$ without prior knowledge of X, Y, and T.

Our second contribution (Section 3) is to show that similar adaptivity and efficiency properties can be obtained via exponential weighting. We consider a variant of the EG$^\pm$ algorithm [9]. The latter has a manageable computational complexity and our lower bound shows that it is nearly optimal in the regime $\kappa \leqslant 1$. However, the EG$^\pm$ algorithm requires prior knowledge of U, X, Y, and T. To overcome this adaptivity issue, we study a modification of the EG$^\pm$ algorithm that relies on the variance-based automatic tuning of [11]. The resulting algorithm—called *adaptive* EG$^\pm$ *algorithm*—can be applied to general convex and differentiable loss functions. When applied to the square loss, it yields an algorithm of the same computational complexity as the EG$^\pm$ algorithm that also achieves a nearly optimal regret but without needing to know X, Y, and T beforehand.

Our third contribution (Section 3.3) is a generic technique called *loss Lipschitzification*. It transforms the loss functions $\boldsymbol{u} \mapsto (y_t - \boldsymbol{u} \cdot \boldsymbol{x}_t)^2$ (or $\boldsymbol{u} \mapsto |y_t - \boldsymbol{u} \cdot \boldsymbol{x}_t|^\alpha$ if the predictions are scored with the α-loss for a real number $\alpha \geqslant 2$) into Lipschitz continuous functions. We illustrate this technique by applying the generic adaptive EG$^\pm$ algorithm to the modified loss functions. When the predictions are scored with the square loss, this yields an algorithm (the LEG algorithm) whose main regret term slightly improves on that derived for the adaptive EG$^\pm$ algorithm without Lipschtizification. The benefits of this technique are clearer for loss functions with higher curvature: if $\alpha > 2$, then the resulting regret bound roughly grows as U instead of a naive $U^{\alpha/2}$.

Finally, in Section 4, we provide a simple way to achieve minimax regret uniformly over all ℓ^1-balls $B_1(U)$ for $U > 0$. This method aggregates instances of an algorithm that requires prior knowledge of U. For the sake of simplicity, we assume

[2] Actually our results hold whether $(\boldsymbol{x}_t, y_t)_{t \geqslant 1}$ is generated by an oblivious environment or a non-oblivious opponent since we consider deterministic forecasters.

[3] In high dimensions (i.e., when $d > \omega T$, for some absolute constant $\omega > 0$), we do not observe this transition (cf. Fig. 1).

[4] For example, $(\boldsymbol{x}_t, y_t)_{1 \leqslant t \leqslant T}$ may be i.i.d., or $\boldsymbol{x}_t$ can be deterministic and $y_t = f(\boldsymbol{x}_t) + \varepsilon_t$ for an unknown function f and an i.i.d. sequence $(\varepsilon_t)_{1 \leqslant t \leqslant T}$ of Gaussian noise.