



HAL
open science

Efficient improper learning for online logistic regression

Rémi Jézéquel, Pierre Gaillard, Alessandro Rudi

► **To cite this version:**

Rémi Jézéquel, Pierre Gaillard, Alessandro Rudi. Efficient improper learning for online logistic regression. 2020. hal-02510505v1

HAL Id: hal-02510505

<https://hal.science/hal-02510505v1>

Preprint submitted on 17 Mar 2020 (v1), last revised 2 Nov 2020 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

EFFICIENT IMPROPER LEARNING FOR ONLINE LOGISTIC REGRESSION

Rmi Jzquel

Pierre Gaillard

Alessandro Rudi

INRIA - Dpartement dInformatique de lcole Normale Suprieure
PSL Research University
Paris, France

ABSTRACT

We consider the setting of online logistic regression and consider the regret with respect to the ℓ_2 -ball of radius B . It is known (see [Hazan et al., 2014]) that any proper algorithm which has logarithmic regret in the number of samples (denoted n) necessarily suffers an exponential multiplicative constant in B . In this work, we design an efficient improper algorithm that avoids this exponential constant while preserving a logarithmic regret.

Indeed, [Foster et al., 2018] showed that the lower bound does not apply to improper algorithms and proposed a strategy based on exponential weights with prohibitive computational complexity. Our new algorithm based on regularized empirical risk minimization with surrogate losses satisfies a regret scaling as $O(B \log(Bn))$ with a per-round time-complexity of order $O(d^2)$.

1 Introduction

In online learning, a learner sequentially interacts with an environment and tries to learn based on data observed on the fly [Cesa-Bianchi and Lugosi, 2006, Hazan et al., 2016]. More formally, at each iteration $t \geq 1$, the learner receives some input x_t in some input space \mathcal{X} ; makes a prediction \hat{y}_t in a decision domain $\hat{\mathcal{Y}}$ and the environment reveals the output $y_t \in \mathcal{Y}$. The inputs x_t and the outputs y_t are sequentially chosen by the environment and can be arbitrary. No stochastic assumption (except boundedness) on the data sequence $(x_t, y_t)_{1 \leq t \leq n}$ is made. The accuracy of a prediction $\hat{y}_t \in \hat{\mathcal{Y}}$ at instant $t \geq 1$ for the outcome $y_t \in \mathcal{Y}$ is measured through a loss function $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \rightarrow \mathbb{R}$. The learner aims at minimizing his cumulative regret

$$R_n(f) = \sum_{t=1}^n \ell(\hat{y}_t, y_t) - \sum_{t=1}^n \ell(f(x_t), y_t), \quad (1)$$

uniformly over all functions f in a reference class of functions \mathcal{F} . All along this paper, we will consider the more specific setting of online logistic regression setting for binary classification. The latter corresponds to binary outputs $y_t \in \mathcal{Y} = \{-1, 1\}$, real decisions $\hat{y}_t \in \hat{\mathcal{Y}} = \mathbb{R}$, the logistic loss function $\ell : (\hat{y}_t, y_t) \mapsto \log(1 + e^{-y_t \hat{y}_t})$ and the reference class $\mathcal{F} = \{x \mapsto \theta^\top x; \theta \in \mathcal{B}(\mathbb{R}^d, B)\}$ of linear functions in the ℓ_2 -ball of radius $B > 0$.

Logistic regression, which dates back to [Berkson, 1944], has been widely studied in the past decades both in the statistical and online setting. It allows to estimate conditional probabilities and is heavily used in practice for multi-class and binary classification. Since the statistical literature is abundant, we highlight here only the key existing approaches for online logistic regression that are relevant for the present work. Using basic properties of the logistic loss, classical algorithms from Online Convex Optimization can be used to minimize the regret (1). On the one hand, remarking that the logistic loss is convex and Lipschitz, one may use Online Gradient Descent (OGD) of [Zinkevich, 2003], which guarantees a regret of order $O(B\sqrt{n})$. On the other hand, using that the logistic loss is e^{-B} -exp concave, one can use Online Newton Step (ONS) from [Hazan et al., 2007] which achieves a logarithmic regret of order $O(de^B \log(n))$.

In view of this results, one could wonder if obtaining a better dependence on the number of samples comes with an exponential deterioration on the multiplicative constant in B . [Hazan et al., 2014] considered this exact question

and showed that indeed any proper algorithm in the regime $n = O(e^B)$ has at least a worst-case regret of order $\Omega(B^{2/3}n^{1/3})$ for one dimensional inputs. Therefore any bound of the form $O(B \log(n))$ is impossible for proper algorithms. We recall that an algorithm is called proper if its prediction function $\hat{f}_t : \mathcal{X} \rightarrow \hat{\mathcal{Y}}$ is in the reference class \mathcal{F} . In other words, it means that for all $t \geq 1$, the prediction is of the form $\hat{y}_t = \hat{f}_t(x_t)$ with $f_t \in \mathcal{F}$ independent of x_t (i.e., the prediction function is linear in x_t in our case).

However, it was recently shown that this lower-bound does not apply to improper algorithms [Foster et al., 2018]. Indeed, based on the simple observation that the logistic loss is 1-mixable (see [Vovk, 1998] for the definition), they could apply the Vovks Aggregating Algorithm [Vovk, 1998] which leverage mixability to achieve a regret of order $O(\log(Bn))$. In particular, they showed that for online logistic regression improper algorithms can significantly outperform proper algorithms by proving a doubly-exponential improvement on the constant B . Yet, the complexity of their algorithm, while being polynomial in d and n is highly prohibitive making the algorithm infeasible in practice. Vovks Aggregating Algorithm is indeed based on a continuous version of the exponentially weighted average forecaster. To output a prediction one needs to approximate an integral over the d -dimensional ball which requires the use of MCMC approximations. Using the projected Langevin Monte Carlo sampler from [Bubeck et al., 2018], they record a computation time of $O(B^6 n^{12} (Bn + d)^{12})$.

This is the starting point of this work. Can we achieve similar performance in online logistic regression with practical computational complexity? Recently, some works attacked this question for logistic regression in the batch statistical setting with i.i.d. data only. [Marteau-Ferey et al., 2019] considered the classical regularized empirical risk minimizer (ERM). Though the latter is proper, using generalized self-concordance properties they could avoid the exponential constant in B under additional assumptions including a well-specified problem, capacity and source conditions. In parallel and independently of this work, [Mourtada and Gaïffas, 2019] have also designed a practical improper algorithm in the statistical setting based on ERM with an improper regularization using virtual data. They could provide an upper-bound on the excess risk in expectation of order $O((d + B^2)/n)$. However, they left open the question of achieving it in an online setting.

Contributions In this paper, we introduce a new practical improper algorithm, that we call AIOLI (Algorithmic efficient Improper Online Logistic regression), for online logistic regression. The latter is based on Follow The Regularized Leader (FTRL) [McMahan, 2011] with surrogate losses. AIOLI takes inspiration from the Azoury-Warmuth-Vovk forecaster (also named non-linear Ridge regression or AWW) from [Vovk, 2001] and [Azoury and Warmuth, 2001] which adds a non-proper penalty based on the next input x_t and from Online Newton Step [Hazan et al., 2007] which leverage the exp-concavity of logistic regression to achieve logarithmic regret. The per-round space and time complexity of AIOLI is of order $(O(nd^2 + n \log(n)))$ which is close to the one of ONS and greatly improves the ones of [Foster et al., 2018].

We provide in Theorem 1 an upper-bound on the regret of AIOLI of the order $O(dBR \log(BRn))$. This makes AIOLI provably better than any proper algorithm in the regime where $n = O(e^B)$. To illustrate our results, we provide simulations on synthetic data generated by the *adversarial* distribution of [Hazan et al., 2014] that show that, contrary to classical FTRL, the regret of AIOLI is indeed logarithmic. We summarize in Table 1 the rates and per-round computational complexities of the key-algorithms for logistic regression.

In addition to introducing AIOLI, we make two technical contributions that we believe to be of their own interests. Our first technical contribution is based on the simple observation that the logistic function $x \mapsto \log(1 + e^{-x})$ is *only* e^{-B} -exp concave on $[-B, B]$ when x is close to $-B$. For the rest of the range (typically $x \in [0, B]$), far better exp-concavity parameters (that we also refer to as *curvature*) may be achieved. Therefore, contrary to ONS which uses the worst-case value for the curvature, we consider quadratic approximations of the logistic loss with data-dependent curvature parameters. These approximations are used as surrogate losses minimized by AIOLI.

Our second technical contribution is to use an improper regularization that allows us to not pay the worst *curvature* but only the one for x close to 0. This regularization is inspired from the non-linear Ridge forecaster of [Azoury and Warmuth, 2001] and [Vovk, 2001]. Typically, when a new input x_t is observed by the learner, the latter can use it to regularize more in the direction of x_t . If the learner knew the next output y_t a perfect regularization would be to add the loss $\ell(f(x_t), y_t)$ when computing FTRL. Yet y_t is unknown and the learner must use a regularization independent of y_t . The non-linear Ridge forecaster consists in replacing y_t by 0. Instead, AIOLI regularizes by adding both $\ell(f(x_t), 1)$ and $\ell(f(x_t), -1)$ to the empirical loss to be minimized. The important phenomena is that the dominant regularization is $\ell(f(x_t), y_t)$ if $y_t \hat{\theta}_t^\top x_t \ll 0$, that is when the algorithm makes a large error. It is worth emphasizing that this regularization depends on the next input x_t and thus makes our algorithm improper. We believe this type of regularization to be new for online logistic regression and have significant interest to inspire future work.

Algorithm	OGD	ONS	[Foster et al., 2018]	AIOLI
Regret	$B\sqrt{n}$	$de^B \log(n)$	$d \log(Bn)$	$dB \log(Bn)$
Total complexity	nd	nd^3	$B^6 n^{12} (Bn + d)^{12}$	$nd^2 + n \log(n)$

Table 1: Regret bounds and computational complexities (in $O(\cdot)$) of relevant algorithms

Setting and notation We recall the setting and introduce the main notations that will be used all along the paper. Our framework is formalized as a sequential game between a learner and an environment. At each forecasting instance $t \geq 1$, the learner is given an input $x_t \in \mathcal{X} \subseteq \mathcal{B}(\mathbb{R}^d, R)$ for some radius $R > 0$ and dimension $d \geq 1$; chooses a vector $\hat{\theta}_t \in \mathbb{R}^d$ (possibly based on the current input x_t and on the past information $x_1, y_1, \dots, x_{t-1}, y_{t-1}$); and makes the prediction $\hat{y}_t = \hat{\theta}_t^\top x_t \in \mathbb{R}$. Then, the environment chooses $y_t \in \{-1, 1\}$; reveals it to the learner which incurs the loss $\ell_t(\hat{\theta}_t) = \ell(\hat{\theta}_t^\top x_t, y_t)$ where for all $\theta \in \mathbb{R}^d$,

$$\ell(\theta^\top x_t, y_t) = \log(1 + e^{-y_t \theta^\top x_t}).$$

Moreover, the gradients of the loss functions at the estimator will be denoted as $g_t = \nabla \ell_t(\hat{\theta}_t) \in \mathbb{R}^d$. We recall that the goal of the learner is to minimize the cumulative regret

$$R_n(\theta) = \sum_{t=1}^n \ell(\hat{\theta}_t^\top x_t, y_t) - \sum_{t=1}^n \ell(\theta^\top x_t, y_t),$$

uniformly over all $\theta \in \mathcal{B}(\mathbb{R}^d, B)$ and all possible sequences $(x_1, y_1), \dots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$.

2 Main contributions

This section gathers the main contributions of the present paper. Essentially, we introduce in Section 2.1 our new algorithm for online logistic regression. In Section 2.2, we prove the corresponding upper-bounds on the regret and we provide an efficient implementation in Section 2.3.

2.1 AIOLI : a new algorithm for online logistic regression

We introduce here and briefly describe a new algorithm AIOLI for online logistic regression. More details on the underlying ideas are provided in Section 3. AIOLI is based on FTRL which is applied on surrogate quadratic losses and with an additional improper regularization. It requires the knowledge of three hyper-parameters: a regularization parameter $\lambda > 0$, the diameter of the input space $R > 0$ and the diameter of the reference classe $B > 0$. At each forecasting instance $t \geq 1$, we first define the following quadratic approximations of the past losses for $1 \leq s < t$ that are defined by: for all $\theta \in \mathbb{R}^d$

$$\hat{\ell}_s(\theta) = \ell_s(\hat{\theta}_s) + g_s^\top (\theta - \hat{\theta}_s) + \frac{\eta_s}{2} (\theta - \hat{\theta}_s)^\top g_s g_s^\top (\theta - \hat{\theta}_s), \quad \text{with } \eta_s = \frac{e^{y_s \hat{y}_s}}{1 + BR}. \quad (2)$$

This approximation is discussed more in details in Section 3.1. The main point to be noticed is that the curvature parameters η_s are adapted to the predictions of the algorithms \hat{y}_s in contrast to ONS which uses the worst-case values e^{-B} for all $s \geq 1$.

Then, AIOLI computes the following estimator

$$\hat{\theta}_t = \operatorname{argmin}_{\theta \in \mathbb{R}^d} \left\{ \sum_{s=1}^{t-1} \hat{\ell}_s(\theta) + \ell(\theta^\top x_t, 1) + \ell(\theta^\top x_t, -1) + \lambda \|\theta\|^2 \right\} \quad (3)$$

and predicts $\hat{y}_t = \hat{\theta}_t^\top x_t$.

Remark that this algorithm is indeed improper since $\hat{\theta}_t$ depends on the next input x_t which implies a non-linear prediction $\hat{y}_t = \hat{\theta}_t^\top x_t$ (see Figure 1). We propose in Section 2.3 an efficient scheme to sequentially compute $\hat{\theta}_t$ with low computational and storage complexities.

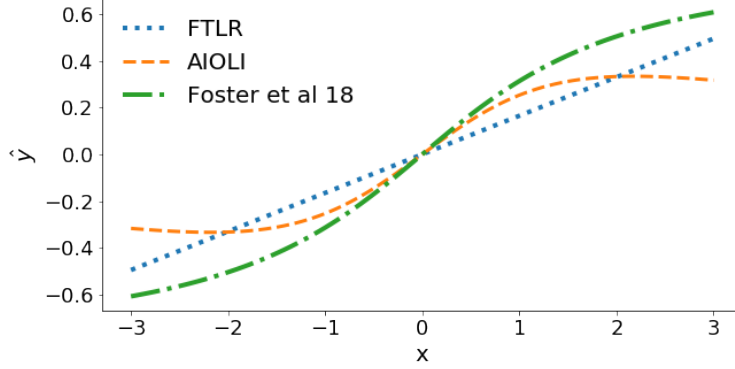


Figure 1: Example of prediction functions obtained by FTRL, AIOLI and the algorithm of [Foster et al., 2018].

2.2 Logarithmic upper-bound on the regret without exponential constants

We state now our main theoretical result which is an upper bound on the regret suffered by AIOLI .

Theorem 1. *Let $\lambda, R > 0$ and $d, n \geq 1$. Let $(x_1, y_1), \dots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$ be an arbitrary sequence of observations. AIOLI (as defined in Equation (3)) run with regularization parameter $\lambda > 0$ satisfies the following upper-bound on the regret*

$$R_n(\theta) \leq \lambda \|\theta\|^2 + d(1 + BR) \log \left(1 + \frac{nR^2}{8(1 + BR)\lambda} \right),$$

for all $\theta \in \mathcal{B}(\mathbb{R}^d, B)$. In particular, by choosing $\lambda = \frac{1}{B^2}$, it yields for all $\theta \in \mathcal{B}(\mathbb{R}^d, B)$

$$R_n(\theta) \leq d(1 + BR) \log \left(1 + \frac{nB^2R^2}{8(1 + BR)} \right) + 1. \quad (4)$$

This theorem is a consequence of the more general theorem 7 which is deferred to Appendix B. We only highlight below the key ingredients of the proof. Theorem 1 states that the regret of AIOLI is logarithmic in n with a multiplicative constant of order dB which is an exponential improvement in B over the one achieved by proper algorithms such as ONS [Hazan et al., 2007]. Yet, our regret upper-bound is weaker than the one of [Foster et al., 2018] which is of order $O(d \log(nB))$. Their algorithm however requires a prohibitive time complexity of order $O(B^6 n^{12} (Bn + d)^{12})$ through complex MCMC procedures. We leave for future work the question whether their regret is achievable by our algorithm or not.

Sketch of proof The proof of the theorem is based on two main steps: 1) we upper-bound the cumulative regret with the true losses with the cumulative regret using the quadratic surrogate losses; 2) we can then follow (with some adjustments) the analysis for online linear regression with squared loss of [Azoury and Warmuth, 2001] and [Vovk, 2001] (see also the proof of [Gaillard et al., 2018]). Fix $\theta \in \mathcal{B}(\mathbb{R}^d, B)$.

Step 1. The first step (i.e., the upper-bound of the regret with the surrogate regret) uses the key Lemma 5, which implies that the quadratic surrogate loss are lower-bounds on the logistic losses. That is,

$$\forall t \geq 1, \quad \widehat{\ell}_t(\theta) \leq \ell_t(\theta).$$

Using that by definition (see Equation (2)) we also have $\widehat{\ell}_t(\widehat{\theta}_t) = \ell_t(\widehat{\theta}_t)$ for all $t \geq 1$, this entails $\ell_t(\widehat{\theta}_t) - \ell_t(\theta) \leq \widehat{\ell}_t(\widehat{\theta}_t) - \widehat{\ell}_t(\theta)$, which implies

$$R_n(\theta) = \sum_{t=1}^n \ell_t(\widehat{\theta}_t) - \sum_{t=1}^n \ell_t(\theta) \leq \sum_{t=1}^n \widehat{\ell}_t(\widehat{\theta}_t) - \sum_{t=1}^n \widehat{\ell}_t(\theta) = \widehat{R}_n(\theta).$$

Step 2. Using that the surrogates losses $\widehat{\ell}_t$ are quadratic, the second part of the proof follows the one of [Gaillard et al., 2018] for online least square regression. After technical linear algebra computation, this leads to

$$\widehat{R}_n(\theta) \leq \sum_{t=1}^n (\theta_{t+1} - \widehat{\theta}_t)^\top A_t (\theta_{t+1} - \widehat{\theta}_t) - (\theta_t - \widehat{\theta}_t)^\top A_{t-1} (\theta_t - \widehat{\theta}_t),$$

where $A_t = \lambda I + \sum_{s=1}^t \frac{\eta_s}{2} g_s g_s^\top$ and we recall that $g_s = \nabla \widehat{\ell}_s(\widehat{\theta}_s)$ and $\eta_t = e^{y_t \widehat{y}_t} / (1 + BR)$. Using the definition of $\widehat{\theta}_t$, after some computations, we can upper-bound

$$(\theta_{t+1} - \widehat{\theta}_t)^\top A_t (\theta_{t+1} - \widehat{\theta}_t) - (\theta_t - \widehat{\theta}_t)^\top A_{t-1} (\theta_t - \widehat{\theta}_t) \leq -\frac{1}{2} g_t^\top A_t^{-1} g_t^{-y_t}.$$

Note that either g_t or $g_t^{-y_t}$ is small. More precisely, if η_t is exponentially small then this is also the case for $g_t^{-y_t}$ which is key to avoid the exponential constant. It should be put in comparison with the bound $g_t^\top A_t^{-1} g_t$ that one would have obtained with the FTRL algorithm. More precisely, we have the following relation $g_t^{-y_t} = -(1 + BR)\eta_t g_t$ which leads to

$$\widehat{R}_n(\theta) \leq (1 + BR) \sum_{t=1}^n \frac{\eta_t}{2} g_t^\top A_t^{-1} g_t.$$

This leaves us with a telescoping sum that finally provides the final regret upper-bound of the theorem. ■

2.3 Efficient Implementation

In this section, we show how to compute incrementally the proposed forecaster $\widehat{\theta}_t$, defined in (3). First, we defined the sufficient statistics used by AIOLI as

$$A_t = \lambda I + \frac{1}{2} \sum_{s=1}^t \eta_s g_s g_s^\top, \quad b_t = \frac{1}{2} \sum_{s=1}^t (\eta_s g_s^\top \widehat{\theta}_s - 1) g_s. \quad (5)$$

In the next lemma we characterize also $\widehat{\theta}_t$ in terms A_{t-1}, b_{t-1}, x_t .

Lemma 2 (Characterizing $\widehat{\theta}_t$ given $A_{t-1}, b_{t-1}, \theta_t, x_t$). *Using the notation above define*

$$W_t = L_{t-1}^{-1} (b_{t-1}, x_t) \in \mathbb{R}^{d \times 2},$$

where L_{t-1} is the Cholesky decomposition of A_{t-1} , i.e. the lower triangular matrix satisfying $A_{t-1} = L_{t-1} L_{t-1}^\top$ and $\omega_t \in \mathbb{R}^2$ is the solution of the following problem

$$\omega_t = \underset{\omega \in \mathbb{R}^{p_t}}{\operatorname{argmin}} \Omega_t(\omega), \quad \Omega_t(\omega) = \|\omega\|^2 - 2u_t^\top \omega + \log(1 + e^{-v_t^\top \omega}) + \log(1 + e^{v_t^\top \omega}), \quad (6)$$

where $p_t \in \{1, 2\}$ is the rank of the matrix W_t , $u_t = \Sigma_t^{1/2} U^\top e_1$, $v_t = \Sigma_t^{1/2} U^\top e_2$ with $\{U_t, \Sigma_t\}$ corresponding to the economic eigenvalue decomposition¹ of $W_t^\top W_t$ and $e_1 = (1, 0)$, $e_2 = (0, 1)$. Then

$$\widehat{\theta}_t = L_{t-1}^{-\top} W_t U_t \Sigma_t^{-1/2} \omega_t. \quad (7)$$

Computing $\widehat{\theta}_t$ given A_{t-1}, b_{t-1}, x_t therefore boils down to solving the two dimensional optimization problem in (6), for which we can use gradient descent, since Ω_t is smooth strongly convex with a small condition number depending only on R/λ , as proven in the next lemma.

Lemma 3. *Let $\varepsilon, \gamma > 0, T \in \mathbb{N}$, let ω_t be the solution of (6) and let ω_t^T be defined recursively as*

$$\omega_t^i = \omega_{t-1}^i - \gamma \nabla \Omega_t(\omega_{t-1}^i), \quad \forall i \in \{1, \dots, T\}.$$

Then $\|\omega_t^T - \omega_t\| \leq \varepsilon$, when $\omega_t^0 = 0$ and γ, T are chosen as follows

$$\gamma = \frac{\lambda}{4\lambda + R^2}, \quad T \geq \left(4 + \frac{R^2}{\lambda}\right) \log \frac{Rt}{\varepsilon \sqrt{\lambda}}.$$

The efficient sequential implementation of $\widehat{\theta}_t$ reported in (1) is obtained by combining the different steps given by: the characterization $\widehat{\theta}_t$ (Lemma 2); the efficient solution of (6) (Lemma 3); and the fact that L_t can efficiently be updated online by doing a Cholesky rank 1 update. More details on the algorithm are provided in Algorithm 2 in Appendix D. The total computational cost is of order $O(nd^2 + n \log n)$ as proven in the next theorem. The proof relies on the facts that rank 1 Cholesky updates cost $O(d^2)$ and that the cost of $w = L^{-1}v$ with $L \in \mathbb{R}^{d \times d}$ triangular invertible and $v, w \in \mathbb{R}^d$ (i.e. the solution of a triangular linear system $Lw = v$) is $O(d^2)$ [Golub and Van Loan, 2012].

¹I.e., $W_t^\top W_t = U_t \Sigma_t U_t^\top$ with $U_t \in \mathbb{R}^{2 \times p_t}$ with p_t the rank of $W_t^\top W_t$, such that $U_t^\top U_t = I$ and $\Sigma_t \in \mathbb{R}^{p_t \times p_t}$ is diagonal and positive.

Parameters λ, T, n , constants B, R
initialize $L_0 = \lambda^{1/2}I, b_0 = 0, \theta_0 = 0$

for $t = 1, \dots, n$ **do**

 receive x_t
 compute $W_t, U_t, \Sigma_t, u_t, v_t$ using $L_{t-1}, b_{t-1}, x_t, \theta_t$, as specified in Lemma 2
 compute ω_t^T using T steps of gradient descent using u_t, v_t as specified in Lemma 3
 compute $\tilde{\theta}_t = L_t^{-\top} W_t U_t \Sigma_t^{-1/2} \omega_t^T$ and predicts $\hat{y}_t = \tilde{\theta}_t^\top x_t$
 receive y_t
 compute g_t using $\tilde{\theta}_t, y_t, x_t$
 compute L_t via rank 1 Cholesky update of L_{t-1} with vector $\sqrt{\eta_t/2}g_t$
 compute $b_t = b_{t-1} + (\eta_t g_t^\top \tilde{\theta}_t - 1)g_t$

end

Algorithm 1: AIOLI descriptive version (see Algorithm 2, Appendix D for a detailed version)

Theorem 4 (Efficient implementation). *Let $T, n \in \mathbb{N}$ and $\tilde{\theta}_t$ be the solution of Algorithm 2 at step t , with hyperparameter T . Choosing $T = \left\lceil \left(4 + \frac{R^2}{\lambda}\right) \log \left(\frac{3n^2 R^2}{\sqrt{\lambda}} \left(\frac{nR^2}{8\lambda} + B \right) \right) \right\rceil$ leads to a regret $\tilde{R}_n(\theta)$ for the forecaster $(\tilde{\theta}_t)_{t=1}^n$ bounded by*

$$\tilde{R}_n(\theta) \leq \lambda \|\theta\|^2 + d(1 + BR) \log \left(1 + \frac{nR^2}{8(1 + BR)\lambda} \right) + 1.$$

Moreover, Algorithm 2 has a total computational complexity

$$O \left(nd^2 + n \frac{R^2}{\lambda} \log \left[\frac{Rn}{\lambda} + B \right] \right).$$

To conclude, note that, when $\lambda = \frac{1}{B^2}$, the total computational complexity of Algorithm 2 is $O(nd^2 + n \log n)$.

3 Key ideas of the analysis

In this section, we present more in details the two main ideas of our analysis. We believe that they might be of independent technical interest for future work.

3.1 Quadratic approximations with adaptive curvature

The main historical approach to prove logarithmic regret for online logistic regression is based on the observation that the logistic losses $\ell_t : \theta \mapsto \ell(\theta^\top x_t, y_t)$ are α -exp-concave for some fixed exp-concavity parameter $\alpha > 0$. In other words, for all $t \geq 1$, the functions $\theta \mapsto \exp(-\alpha \ell_t(\theta))$ are convex. From [Hazan et al., 2016, Lemma 4.2], α -exp-concavity implies in particular that for all $\theta, \hat{\theta}_t \in \mathcal{B}(\mathbb{R}^d, B)$

$$\ell_t(\theta) \geq \ell_t(\hat{\theta}_t) + \nabla \ell_t(\hat{\theta}_t)^\top (\theta - \hat{\theta}_t) + \frac{\eta}{2} (\theta - \hat{\theta}_t)^\top \nabla \ell_t(\hat{\theta}_t) \nabla \ell_t(\hat{\theta}_t)^\top (\theta - \hat{\theta}_t) \quad (8)$$

where $\eta \leq \frac{1}{2} \min\{\frac{1}{4GB}, \alpha\}$, where G is an upper-bound of the ℓ_2 -norm of the gradients. We refer to η as the *curvature* constant. The above inequality provides a quadratic lower approximation of the logistic loss. It plays a crucial role in the analysis of ONS [Hazan et al., 2007] to provide a logarithmic regret upper-bound of order $O(\frac{1}{\eta} d \log(n))$. We can note that in this inequality, η is fixed for all $t \geq 1$ and independent of θ and $\hat{\theta}_t$. However, for the logistic loss, the best exp-concavity constant $\alpha > 0$ is of order e^{-BR} which leads to an undesirable exponential multiplicative constant.

Our idea is to replace the worst-case fixed $\eta > 0$ with a data adaptive constant η_t . To do so, we first remark that at time $t \geq 1$, the curvature constant is bad (i.e., of order e^{-BR}) when the prediction $\hat{y}_t = \hat{\theta}_t^\top x_t$ of the algorithm was significantly wrong. That is, when $y_t \hat{y}_t \approx -BR$. In contrast, if the algorithm predicted well the sign of the next outcome, i.e., if $y_t \hat{y}_t \geq 0$ then Inequality (8) holds with a much larger curvature constant greater than $(1 + BR)^{-1}$. Based on this high-level idea, we could prove Inequality (8) by replacing the fixed curvature $\eta > 0$ with

$$\eta_t = \frac{e^{y_t \hat{y}_t}}{1 + BR}. \quad (9)$$

The latter inequality yielded to our choice of surrogate quadratic approximations $\widehat{\ell}_t$ defined in Equation (2). This adaptive quadratic lower-approximation of the logistic loss is a direct consequence of the following technical lemma applied with $a = y_t \theta^\top x_t$, $b = y_t \widehat{y}_t$, and $C = BR$.

Lemma 5. *Let $C > 0$ and $f : x \in \mathbb{R} \mapsto \log(1 + e^{-x})$. Then, for all $a \in [-C, C]$ and $b \in \mathbb{R}$,*

$$f(a) \geq f(b) + f'(b)(a - b) + \frac{e^b}{2(1 + C)} f'(b)^2 (a - b)^2.$$

The proof is postponed to the supplementary material (see Appendix C).

3.2 Improper regularization

The other key ingredient of our analysis is to ensure that only the rounds where the curvature η_t (9) are large matter in the analysis. This is the role of our new improper regularization added in the definition (3) of $\widehat{\theta}_t$. The underlying idea is to add the possible next losses $\ell(\theta^\top x_t, 1)$ and $\ell(\theta^\top x_t, -1)$ to the minimization problem solved by AIOLI (see (3)).

We explain now the high-level idea why this regularization helps when η_t is small. We need to distinguish two cases. On the one hand, if the prediction is good, i.e., \widehat{y}_t and y_t have same signs. Then, $\eta_t \propto \exp(y_t \widehat{y}_t)$ is large and since the prediction is already good. Thus, the regularization does not hurt much. On the other hand, when \widehat{y}_t and y_t have opposite signs, the curvature parameter may be exponentially small. But, then the addition of $\ell_t(\theta^\top x_t, y_t)$ greatly improves the predictions of the algorithm in these cases, because the data point (x_t, y_t) was already included in the history when optimizing $\widehat{\theta}_t$ in (1). Moreover, the addition of the the wrong output $-y_t$ does not impact much the prediction since in \widehat{y}_t we have

$$\ell(\widehat{y}_t, -y_t) = \log(1 + e^{y_t \widehat{y}_t}) = \log(1 + (1 + BR)\eta_t)$$

which is small whenever η_t is small.

4 Extensions

4.1 Non-parametric setting

For the sake of simplicity, the analysis of the present paper was only carried out for finite dimensional logistic regression in \mathbb{R}^d . Yet, most of the results remain valid for Reproducing Kernel Hilbert Spaces (RKHS) \mathcal{H} (see [Aronszajn, 1950] for details on RKHS). Then, Theorem 1 holds by replacing the finite dimension $d \geq 1$ with the effective dimension

$$d_{\text{eff}}(\lambda) = \text{Tr}(K_{nn}(K_{nn} + \lambda I)^{-1}),$$

where the input matrix K_{nn} is defined as $(K_{nn})_{i,j} = x_i^\top x_j$. The regret is then of order $O(Bd_{\text{eff}}(B\lambda) + \lambda B^2)$. Note that the effective dimension is always upper-bounded by $d_{\text{eff}}(\lambda) \leq n/\lambda$, providing in the worst case, the regret upper-bound of order $O(B\sqrt{n})$ for well-chosen λ . Under the capacity condition, which is a classical assumption for kernels (see [Marteau-Ferey et al., 2019] for instance), better bounds on the effective dimension are provided which yield to faster regret rates.

In the case of RKHS, using standard kernel trick, the total computational complexity of the algorithm is then $O(n^3)$. The latter might be however prohibitive in large dimension. An interesting research direction is to investigate whether we can apply standard approximation techniques such as random features or Nystrm projection similarly to what [Calandriello et al., 2017] and [Jézéquel et al., 2019] did for exp-concave and square loss respectively. In particular, what is the trade-off between computational complexity and regret and what is the lowest complexity that still allows optimal regret?

4.2 Online-to-batch conversion

Even in the batch statistical setting, the lower-bound of [Hazan et al., 2014] holds for proper algorithms and few improper algorithms were introduced to avoid the statistical constant $O(e^B)$. Using the standard online-to-batch conversion [Helmbold and Warmuth, 1995], similarly to the algorithm of [Foster et al., 2018], our algorithm also provides an estimator with bounded excess risk in expectation. To do so, one can sample an index τ uniformly in $\{1, \dots, n\}$ and define the estimator \bar{f}_n defined for all $x \in \mathcal{X}$ by

$$\bar{f}_n(x) = \widehat{f}_\tau(x) \quad \text{with} \quad \widehat{f}_t(x) = \widehat{\theta}_t(x)^\top x, \quad 1 \leq t \leq T, \quad (10)$$

where $\widehat{\theta}_t(x)$ is the solution of the minimization problem (3) by substituting x_t with the new input $x \in \mathcal{X}$ and using $z_t = -\text{sign}(\theta_t^\top x)$. It is worth pointing out that $\bar{f}_n(x) \neq \widehat{\theta}_t^\top x$ is a non-linear function in x and is thus improper. The following corollary controls the excess-risk of \bar{f}_n in expectation. Its proof is standard but short and we recall it for the sake of completeness.

Corollary 6 (Online-to-batch conversion). *Let $n, d \geq 1$ and $B, R > 0$. Let ν be an unknown distribution over $\mathcal{B}(\mathbb{R}^d, R) \times \{-1, 1\}$ and $D_n = \{(x_i, y_i)\}_{1 \leq i \leq n}$ be i.i.d. sampled from ν . Then, the estimator \bar{f}_n defined in Equation (10) with $\lambda = 1/B^2$ satisfies*

$$\mathbb{E}[\ell(\bar{f}_n(X), Y)] - \inf_{\theta \in \mathcal{B}(\mathbb{R}^d, B)} \mathbb{E}[\ell(f(X), Y)] \leq \frac{1}{n} \left[d(1 + BR) \log \left(1 + \frac{nB^2R^2}{8(1 + BR)} \right) + 1 \right],$$

where $(X, Y) \sim \nu$ and the expectations are taken over (X, Y) , D_n and τ .

Proof Let us denote by R_n^b the upper-bound on the regret in the right-hand side of Equation (4). Then,

$$\begin{aligned} \mathbb{E}[\ell(\bar{f}_n(X), Y)] &= \mathbb{E}[\ell(\widehat{f}_\tau(X), Y)] = \mathbb{E}\left[\frac{1}{n} \sum_{t=1}^n \ell(\widehat{f}_t(X), Y)\right] \stackrel{(*)}{=} \mathbb{E}\left[\frac{1}{n} \sum_{t=1}^n \ell(\widehat{f}_t(x_t), y_t)\right] \\ &\leq \mathbb{E}\left[\frac{1}{n} \sum_{t=1}^n \ell(\theta^\top x_t, y_t)\right] + \frac{R_n^b}{n} \stackrel{(*)}{=} \mathbb{E}\left[\frac{1}{n} \sum_{t=1}^n \ell(\theta^\top X, Y)\right] + \frac{R_n^b}{n}, \end{aligned}$$

where the equalities (*) are because (X, Y) and (x_t, y_t) follow the same distribution and because \widehat{f}_t and θ are independent of (x_t, y_t) by definition. \square

Apart from [Foster et al., 2018], which is non-practical and also based on an online-to-batch conversion, we are only aware of the works of [Mourtada and Gaïffas, 2019] and [Marteau-Ferey et al., 2019] that improve the exponential constant $O(e^B)$ in the statistical setting. [Marteau-Ferey et al., 2019] make additional assumptions on the data distribution (self-concordance, well-specified model, capacity and source conditions). Their framework is hardly comparable to ours with constants that may be arbitrarily large in our setting. In contrast, the recent work of [Mourtada and Gaïffas, 2019] do provide an improper estimator that satisfies a result very similar to Corollary 6 with an expected bound on the excess risk of order $O(d + B^2R^2)$. Our upper-bound is slightly worse with an additional multiplicative factor $BR \log(BRn)$. The $\log n$ is due to the online setting in which it is optimal, see for instance the lower-bound of [Vovk, 2001]. Their estimator is based on an empirical regularized risk minimization (with the original losses) with an additional improper regularization using virtual data. They do not analyze the computational complexity but we believe it to be similar to ours. To conclude the comparison, note that in contrast to ours, their analysis relies on the self-concordance property of the logistic loss in contrast to ours.

5 Simulations

This section illustrates our theoretical results with synthetic experiments and compare the performance of three algorithms: FTRL with ℓ_2 -regularization and $\lambda = 1$, AIOLI, and the one of [Foster et al., 2018]. We sample the data points $(x_t, y_t) \in \mathbb{R} \times \{-1, 1\}$ according to the *adversarial* distributions designed by [Hazan et al., 2014] to prove the exponential lower bound for proper algorithms. We consider only the case $d = 1$ because the lower bound for proper algorithms already applies and the algorithm of [Foster et al., 2018] is practical in this case. Let $n \geq 1$, $B = \log(n)$, $\chi \in \{-1, 1\}$ and $\varepsilon = 0.01$, the data $(x_t, y_t)_{1 \leq t \leq n}$ are i.i.d. generated according to

$$(x_t, y_t) = \begin{cases} (1 - \frac{\sqrt{\varepsilon}}{2B}, 1) & \text{w.p. } \frac{\sqrt{\varepsilon}}{2B} + \chi \frac{\varepsilon}{B} \\ (\frac{\sqrt{\varepsilon}}{B}, -1) & \text{otherwise} \end{cases}.$$

The experiment is averaged over 10 simulations for $\chi = -1$ and 10 others for $\chi = 1$. We plot in Figure 2 the worst of these two average regrets obtained by each algorithm according to the value of n . The lower-bound of [Hazan et al., 2014] ensures that any proper algorithm has at least a regret of order $\Omega(n^{1/3})$ for these data. As expected, the regret of FTRL is polynomial in n (linear slope in log-log scale) while the ones of AIOLI and the algorithm of [Foster et al., 2018] are poly-logarithmic.

6 Conclusion and future work

To sum up, we designed a new efficient improper algorithm for online logistic regression. The latter only suffers logarithmic regret with much improved complexity compared to other existing methods. Some interesting questions are still remaining and left for future work.

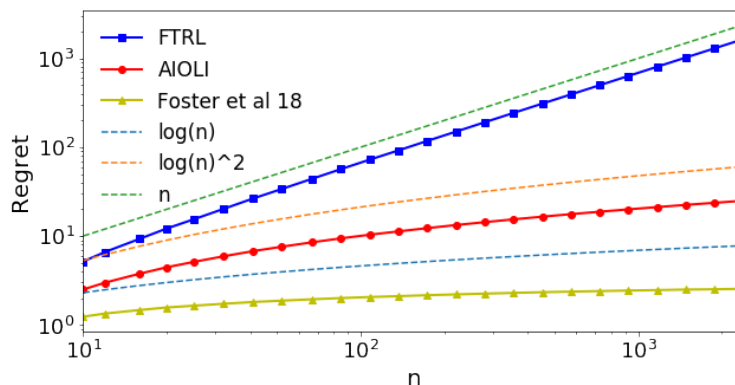


Figure 2: Regret of each algorithm according to number of sample n in log-log scale.

Our online-to-batch procedure only provides upper-bounds in expectation. Obtaining high-probability bounds is more challenging and universal conversion methods such as the one of [Mehta, 2016] may not work for improper procedures.

Another interesting direction for future research is the extension to multi-class classification. Our analysis strongly relies on binary outputs to produce the improper regularization and the extension to multi-class is not straightforward. The next step would then be to extend the results to other settings considered by [Foster et al., 2018] such as bandit multi-class learning or online multi-class boosting. More generally, it would be interesting to study what are the class of functions where adaptive *curvature* parameters and improper learning yield to improved guarantees.

Finally, as shown in Section 4.1, AIOLI may be applied to nonparametric logistic regression in RKHS. However, without any approximation schemes, the computational complexity may become prohibitive of order $O(n^3)$. Therefore, a possible line of research would be to study how much the performance of our algorithm would be affected by standard approximations techniques as Nystrm projections or random features.

References

- [Aronszajn, 1950] Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404.
- [Azoury and Warmuth, 2001] Azoury, K. S. and Warmuth, M. K. (2001). Relative loss bounds for on-line density estimation with the exponential family of distributions. *Machine Learning*, 43(3):211–246.
- [Berkson, 1944] Berkson, J. (1944). Application of the logistic function to bio-assay. *Journal of the American statistical association*, 39(227):357–365.
- [Bubeck et al., 2018] Bubeck, S., Eldan, R., and Lehec, J. (2018). Sampling from a log-concave distribution with projected langevin monte carlo. *Discrete & Computational Geometry*, 59(4):757–783.
- [Bubeck et al., 2015] Bubeck, S. et al. (2015). Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357.
- [Calandriello et al., 2017] Calandriello, D., Lazaric, A., and Valko, M. (2017). Efficient second-order online kernel learning with adaptive embedding. In *Neural Information Processing Systems*.
- [Cesa-Bianchi and Lugosi, 2006] Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, learning, and games*. Cambridge university press.
- [Foster et al., 2018] Foster, D. J., Kale, S., Luo, H., Mohri, M., and Sridharan, K. (2018). Logistic regression: The importance of being improper. *arXiv preprint arXiv:1803.09349*.
- [Gaillard et al., 2018] Gaillard, P., Gerchinovitz, S., Huard, M., and Stoltz, G. (2018). Uniform regret bounds over \mathbb{R}^d for the sequential linear regression problem with the square loss. *arXiv preprint arXiv:1805.11386*.
- [Golub and Van Loan, 2012] Golub, G. H. and Van Loan, C. F. (2012). *Matrix computations*, volume 3. JHU press.
- [Hazan et al., 2007] Hazan, E., Agarwal, A., and Kale, S. (2007). Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192.
- [Hazan et al., 2016] Hazan, E. et al. (2016). Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325.

- [Hazan et al., 2014] Hazan, E., Koren, T., and Levy, K. Y. (2014). Logistic regression: Tight bounds for stochastic and online optimization. In *Conference on Learning Theory*, pages 197–209.
- [Helmbold and Warmuth, 1995] Helmbold, D. P. and Warmuth, M. K. (1995). On weak learning. *Journal of Computer and System Sciences*, 50(3):551–573.
- [Jézéquel et al., 2019] Jézéquel, R., Gaillard, P., and Rudi, A. (2019). Efficient online learning with kernels for adversarial large scale problems. In *Advances in Neural Information Processing Systems*, pages 9427–9436.
- [Marteau-Ferey et al., 2019] Marteau-Ferey, U., Ostrovskii, D., Bach, F., and Rudi, A. (2019). Beyond least-squares: Fast rates for regularized empirical risk minimization through self-concordance. *arXiv preprint arXiv:1902.03046*.
- [McMahan, 2011] McMahan, H. B. (2011). Follow-the-regularized-leader and mirror descent: Equivalence theorems and ℓ_1 regularization. *Journal of Machine Learning Research*.
- [Mehta, 2016] Mehta, N. A. (2016). Fast rates with high probability in exp-concave statistical learning. *arXiv preprint arXiv:1605.01288*.
- [Mourtada and Gaïffas, 2019] Mourtada, J. and Gaïffas, S. (2019). An improper estimator with optimal excess risk in misspecified density estimation and logistic regression. *arXiv preprint arXiv:1912.10784*.
- [Vovk, 1998] Vovk, V. (1998). A game of prediction with expert advice. *Journal of Computer and System Sciences*, 56(2):153–173.
- [Vovk, 2001] Vovk, V. (2001). Competitive on-line statistics. *International Statistical Review*, 69(2):213–248.
- [Zinkevich, 2003] Zinkevich, M. (2003). Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936.

A Notation

In this section, we recall and define useful notations that will be used all along the proofs. At each round $t \geq 1$, we recall that the forecaster is given an input $x_t \in \mathcal{X} \subset \mathcal{B}(\mathbb{R}^d, R)$; chooses a prediction $\hat{\theta}_t \in \mathbb{R}^d$; forms the prediction $\hat{y}_t = \hat{\theta}_t^\top x_t$; and observes the outcome $y_t \in \{-1, 1\}$. The loss of a parameter $\theta \in \mathbb{R}^d$ at time $t \geq 1$ is measured by $\ell_t(\theta) = \ell(\theta^\top x_t, y_t) = \log(1 + e^{-y_t \theta^\top x_t})$.

We also define for all $t \geq 1$, all $\theta \in \mathbb{R}^d$ and $y \in \{-1, 1\}$:

- the loss suffered by θ if the outcome was y : $\ell_t^y(\theta) = \log(1 + e^{-y \theta^\top x_t})$
- the gradient of the loss in $\hat{\theta}_t$ if the outcome was y : $g_t^y = \nabla \ell_t^y(\hat{\theta}_t)$
- the curvature if the outcome was y : $\eta_t^y = \frac{e^{y \hat{\theta}_t^\top x_t}}{1 + B R}$
- the quadratic surrogate losses if the outcome was y :
 $\hat{\ell}_t^y(\theta) = \ell_t^y(\hat{\theta}_t) + g_t^{y^\top}(\theta - \hat{\theta}_t) + \frac{\eta_t^y}{2}(\theta - \hat{\theta}_t)^\top g_t^y g_t^{y^\top}(\theta - \hat{\theta}_t)$
- the corresponding loss, surrogate loss, gradient, and curvature for the true outcome \hat{y}_t :
 $\ell_t = \ell_t^{\hat{y}_t}$, $\hat{\ell}_t = \hat{\ell}_t^{\hat{y}_t}$, $g_t = g_t^{\hat{y}_t}$, $\eta_t = \eta_t^{\hat{y}_t}$
- the regularized cumulative loss and cumulative surrogate loss respectively:

$$L_t(\theta) = \sum_{s=1}^t \ell_s(\theta) + \lambda \|\theta\|^2, \quad \hat{L}_t(\theta) = \sum_{s=1}^t \hat{\ell}_s(\theta) + \lambda \|\theta\|^2$$

With these notations, we defined θ_t and $\bar{\theta}_t$ as:

$$\theta_t = \operatorname{argmin}_{\theta \in \mathbb{R}^d} \hat{L}_{t-1}(\theta), \quad \text{and} \quad \bar{\theta}_t = \operatorname{argmin}_{\theta \in \mathbb{R}^d} \left\{ \hat{L}_{t-1}(\theta) + \ell_t^1(\theta) + \ell_t^{-1}(\theta) \right\}. \quad (11)$$

B Proof of the main theorem

Theorem 7. *Let $\varepsilon, \lambda, R > 0$ and $d, n \geq 1$. Let $(x_1, y_1), \dots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$ be an arbitrary sequence of observations. Define $\hat{\theta}_t$ for $t \geq 1$ as in Equation (11) with regularization parameter $\lambda > 0$. Then, any estimator $\hat{\theta}_t$ which verifies for all $t \geq 1$, $\|\hat{\theta}_t - \bar{\theta}_t\| \leq \varepsilon$ satisfies the following upper-bound on the regret*

$$R_n(\theta) \leq \lambda \|\theta\|^2 + d(1 + BR) \log \left(1 + \frac{nR^2}{8(1 + BR)\lambda} \right) + 3nR \left(\frac{nR^2}{8\lambda} + B \right) \varepsilon.$$

Proof Let $\theta \in \mathcal{B}(\mathbb{R}^d, B)$. Let us first upper-bound the regret $R_n(\theta)$ by the regret using the surrogate losses. Applying Lemma 5 with $a = y_t \theta^\top x_t \in [-BR, BR]$ and $b \in y_t \hat{\theta}_t^\top x_t \in \mathbb{R}$, we have for all $t \geq 1$:

$$\ell_t(\theta) \geq \hat{\ell}_t(\theta).$$

Together with $\ell_t(\hat{\theta}_t) = \hat{\ell}_t(\hat{\theta}_t)$, it yields that the regret on the true loss is upper-bounded by the regret on the quadratic approximations

$$R_n(\theta) = \sum_{t=1}^n \ell_t(\hat{\theta}_t) - \sum_{t=1}^n \ell_t(\theta) \leq \sum_{t=1}^n \hat{\ell}_t(\hat{\theta}_t) - \sum_{t=1}^n \hat{\ell}_t(\theta) = \hat{R}_n(\theta). \quad (12)$$

Now, we are left with analyzing a quadratic problem. We can thus follow in the main lines the proof of [Gaillard et al., 2018] for online least squares. By definition of $\theta_{n+1} = \operatorname{argmin}_{\theta \in \mathbb{R}^d} \hat{L}_n(\theta)$, $\hat{L}_n(\theta_{n+1}) \leq \hat{L}_n(\theta)$, which can be written as

$$\sum_{t=1}^n \hat{\ell}_t(\theta_{n+1}) - \sum_{t=1}^n \hat{\ell}_t(\theta) \leq \lambda \|\theta\|^2 - \lambda \|\theta_{n+1}\|^2.$$

Now, the regret can be upper-bounded as

$$\begin{aligned} \hat{R}_n(\theta) &\leq \lambda \|\theta\|^2 + \sum_{t=1}^n \hat{\ell}_t(\hat{\theta}_t) - \sum_{t=1}^n \hat{\ell}_t(\theta_{n+1}) - \lambda \|\theta_{n+1}\|^2 \\ &= \lambda \|\theta\|^2 + \sum_{t=1}^n \left[\hat{\ell}_t(\hat{\theta}_t) + \hat{L}_{t-1}(\theta_t) - \hat{L}_t(\theta_{t+1}) \right]. \end{aligned} \quad (13)$$

With a little bit of abuse, we call the terms inside the sum on the right the *instant regrets*. Grouping the terms of same degrees in the quadratic approximation,

$$\begin{aligned}\widehat{\ell}_t(\theta) &= \ell_t(\widehat{\theta}_t) + g_t^\top (\theta - \widehat{\theta}_t) + \frac{\eta_t}{2} (\theta - \widehat{\theta}_t) g_t g_t^\top (\theta - \widehat{\theta}_t) \\ &= \ell_t(\widehat{\theta}_t) - g_t^\top \widehat{\theta}_t + \frac{\eta_t}{2} \widehat{\theta}_t^\top g_t g_t^\top \widehat{\theta}_t + g_t^\top \theta - \eta_t \theta^\top g_t g_t^\top \widehat{\theta}_t + \frac{\eta_t}{2} \theta^\top g_t g_t^\top \theta \\ &= c_t^* - 2b_t^{*\top} \theta + \frac{\eta_t}{2} \theta^\top g_t g_t^\top \theta\end{aligned}$$

with $c_t^* = \ell_t(\widehat{\theta}_t) - g_t^\top \widehat{\theta}_t + \frac{\eta_t}{2} \widehat{\theta}_t^\top g_t g_t^\top \widehat{\theta}_t$ and $b_t^* = \frac{1}{2}(-g_t + \eta_t(\widehat{\theta}_t^\top g_t)g_t)$. Similarly, we can write the cumulative loss as

$$\widehat{L}_t(\theta) = \underbrace{\sum_{s=1}^t c_s^*}_{c_t} - 2 \underbrace{\left(\sum_{s=1}^t b_s^* \right)^\top}_{b_t} \theta + \theta^\top \underbrace{\left(\lambda I + \sum_{s=1}^t \frac{\eta_s}{2} g_s g_s^\top \right)}_{A_t} \theta. \quad (14)$$

The minimum of this quadratic, reached in $\theta_{t+1} = A_t^{-1} b_t$, is

$$\widehat{L}_t(\theta_{t+1}) = c_t - 2 \underbrace{b_t^\top A_t^{-1}}_{\theta_{t+1}} A_t \theta_{t+1} + \theta_{t+1}^\top A_t \theta_{t+1} = c_t - \theta_{t+1}^\top A_t \theta_{t+1}.$$

We can write now the instant regret at t as

$$\begin{aligned}\widehat{\ell}_t(\widehat{\theta}_t) + \widehat{L}_{t-1}(\theta_t) - \widehat{L}_t(\theta_{t+1}) &= \widehat{\ell}_t(\widehat{\theta}_t) - c_t^* + \theta_{t+1}^\top A_t \theta_{t+1} - \theta_t^\top A_{t-1} \theta_t \\ &= g_t^\top \widehat{\theta}_t - \frac{\eta_t}{2} \widehat{\theta}_t^\top g_t g_t^\top \widehat{\theta}_t + \theta_{t+1}^\top A_t \theta_{t+1} - \theta_t^\top A_{t-1} \theta_t \\ &= g_t^\top \widehat{\theta}_t - \widehat{\theta}_t^\top (A_t - A_{t-1}) \widehat{\theta}_t + \theta_{t+1}^\top A_t \theta_{t+1} - \theta_t^\top A_{t-1} \theta_t.\end{aligned} \quad (15)$$

The oracle θ_{t+1} minimizes the quadratic function \widehat{L}_t with Hessian $2A_t$. Thus, performing one newton step from $\widehat{\theta}_t$ gives

$$\begin{aligned}\theta_{t+1} &= \widehat{\theta}_t - \frac{1}{2} A_t^{-1} \nabla \widehat{L}_t(\widehat{\theta}_t) \\ &= \widehat{\theta}_t - \frac{1}{2} A_t^{-1} \left[\nabla \widehat{L}_{t-1}(\widehat{\theta}_t) + g_t \right] \\ &= \widehat{\theta}_t + \frac{1}{2} A_t^{-1} g_t^{-y_t} - \frac{1}{2} A_t^{-1} \left[\nabla \widehat{L}_{t-1}(\widehat{\theta}_t) + g_t + g_t^{-y_t} \right]\end{aligned} \quad (16)$$

Similarly, we have

$$\begin{aligned}\theta_t &= \widehat{\theta}_t - \frac{1}{2} A_{t-1}^{-1} \nabla \widehat{L}_{t-1}(\widehat{\theta}_t) \\ &= \widehat{\theta}_t + \frac{1}{2} A_{t-1}^{-1} (g_t + g_t^{-y_t}) - \frac{1}{2} A_{t-1}^{-1} \left[\nabla \widehat{L}_{t-1}(\widehat{\theta}_t) + g_t + g_t^{-y_t} \right].\end{aligned} \quad (17)$$

Reorganizing the terms in the two previous equations leads to

$$g_t = 2 \left[A_t \widehat{\theta}_t - A_t \theta_{t+1} - A_{t-1} \widehat{\theta}_t + A_{t-1} \theta_t \right].$$

Substituting in the instant regret (15), this entails

$$\begin{aligned}\widehat{\ell}_t(\widehat{\theta}_t) + \widehat{L}_{t-1}(\theta_t) - \widehat{L}_t(\theta_{t+1}) &= 2\widehat{\theta}_t^\top A_t \widehat{\theta}_t - 2\theta_{t+1}^\top A_t \widehat{\theta}_t - 2\widehat{\theta}_t^\top A_{t-1} \widehat{\theta}_t + 2\theta_t^\top A_{t-1} \widehat{\theta}_t \\ &\quad - \widehat{\theta}_t^\top A_t \widehat{\theta}_t + \widehat{\theta}_t^\top A_{t-1} \widehat{\theta}_t + \theta_{t+1}^\top A_t \theta_{t+1} - \theta_t^\top A_{t-1} \theta_t \\ &= \theta_{t+1}^\top A_t \theta_{t+1} - 2\theta_{t+1}^\top A_t \widehat{\theta}_t + \widehat{\theta}_t^\top A_t \widehat{\theta}_t - \theta_t^\top A_{t-1} \theta_t + 2\theta_t^\top A_{t-1} \widehat{\theta}_t - \widehat{\theta}_t^\top A_{t-1} \widehat{\theta}_t \\ &= (\theta_{t+1} - \widehat{\theta}_t)^\top A_t (\theta_{t+1} - \widehat{\theta}_t) - (\theta_t - \widehat{\theta}_t)^\top A_{t-1} (\theta_t - \widehat{\theta}_t)\end{aligned} \quad (18)$$

Rewriting equations (16) and (17), we have

$$\begin{aligned}2A_t(\theta_{t+1} - \widehat{\theta}_t) &= g_t^{-y_t} - \delta_t \\ 2A_{t-1}(\theta_t - \widehat{\theta}_t) &= g_t^{-y_t} + g_t - \delta_t\end{aligned}$$

with $\delta_t = \nabla \widehat{L}_{t-1}(\widehat{\theta}_t) + g_t + g_t^{-y_t}$.

Subtracting the first equation to the second, we can write the instant regret as a variance term and an optimization error term,

$$\widehat{\ell}_t(\widehat{\theta}_t) + \widehat{L}_{t-1}(\theta_t) - \widehat{L}_t(\theta_{t+1}) = Z_t + \Omega_t \quad (19)$$

where

$$Z_t = \frac{1}{4} g_t^{-y_t \top} A_t^{-1} g_t^{-y_t} - \frac{1}{4} (g_t + g_t^{-y_t}) A_{t-1}^{-1} (g_t + g_t^{-y_t})$$

and

$$\Omega_t = \frac{1}{4} [-2g_t^{-y_t} A_t^{-1} \delta_t + \delta_t A_t^{-1} \delta_t + 2(g_t + g_t^{-y_t}) A_{t-1}^{-1} \delta_t - \delta_t A_{t-1}^{-1} \delta_t].$$

B.1 Upper-bound of the variance term Z_t

Let us focus on bounding the term Z_t . Developing the terms and using the fact that $A_{t-1} \leq A_t$, we have

$$\begin{aligned} Z_t &= \frac{1}{4} (g_t + g_t^{-y_t} - g_t)^\top A_t^{-1} (g_t + g_t^{-y_t} - g_t) - \frac{1}{4} (g_t + g_t^{-y_t})^\top A_{t-1}^{-1} (g_t + g_t^{-y_t}) \\ &= \frac{1}{4} g_t^\top A_t^{-1} g_t - \frac{1}{2} g_t^\top A_t^{-1} (g_t + g_t^{-y_t}) + \frac{1}{4} (g_t + g_t^{-y_t})^\top A_t^{-1} (g_t + g_t^{-y_t}) \\ &\quad - \frac{1}{4} (g_t + g_t^{-y_t})^\top A_{t-1}^{-1} (g_t + g_t^{-y_t}) \\ &\leq \frac{1}{4} g_t^\top A_t^{-1} g_t - \frac{1}{2} g_t^\top A_t^{-1} (g_t + g_t^{-y_t}) \\ &= -\frac{1}{4} g_t^\top A_t^{-1} g_t - \frac{1}{2} g_t^\top A_t^{-1} g_t^{-y_t} \\ &\leq -\frac{1}{2} g_t^\top A_t^{-1} g_t^{-y_t}. \end{aligned}$$

Using the definition of the logistic function, we can relate g_t and $g_t^{-y_t}$,

$$g_t^{-y_t} = \frac{y_t x_t}{1 + e^{-y_t \widehat{\theta}_t^\top x_t}} = e^{y_t \widehat{\theta}_t^\top x_t} \frac{y_t x_t}{1 + e^{y_t \widehat{\theta}_t^\top x_t}} = -(1 + BR) \eta_t g_t, \quad (20)$$

which implies,

$$Z_t \leq (1 + BR) \frac{\eta_t}{2} g_t^\top A_t^{-1} g_t.$$

Summing over $t = 1, \dots, n$, the sum telescopes thanks to Lemma 10, we obtain

$$\sum_{t=1}^n Z_t \leq (1 + BR) \sum_{k=1}^d \log \left(1 + \frac{\lambda_k(C_n)}{\lambda} \right), \quad (21)$$

where $C_n = \frac{1}{2} \sum_{t=1}^n \eta_t g_t g_t^\top$ and $\lambda_k(C_n)$ is the k largest eigenvalue of C_n .

It only remains now to upper-bound the eigenvalues of C_n , which we do now. Recalling that $g_t = -y_t x_t / (1 + \exp(y_t \widehat{\theta}_t^\top x_t))$, we have

$$\frac{\eta_t}{2} g_t g_t^\top = \frac{1}{2(1 + BR)} \frac{e^{y_t \widehat{\theta}_t^\top x_t}}{(1 + e^{y_t \widehat{\theta}_t^\top x_t})^2} x_t x_t^\top \leq \frac{1}{8(1 + BR)} x_t x_t^\top,$$

where for the inequality, we used that $x/(1+x)^2 \leq 1/4$ for $x \geq 0$. Therefore, $\lambda_k(C_n) \leq nR^2/(8(1 + BR))$ for all $k \geq 1$.

$$\sum_{t=1}^n Z_t \leq d(1 + BR) \log \left(1 + \frac{nR^2}{8(1 + BR)\lambda} \right) \quad (22)$$

B.2 Upper-bound on the optimization error Ω_t

It remains to bound the the approximation term Ω_t .

$$\begin{aligned}\Omega_t &= \frac{1}{4}[2(g_t + g_t^{-y_t})A_{t-1}^{-1}\delta_t - 2g_t^{-y_t}A_t^{-1}\delta_t + \underbrace{\delta_t A_t^{-1}\delta_t - \delta_t A_{t-1}^{-1}\delta_t}_{\leq 0}] \\ &\leq \frac{1}{2\lambda} [2\|g_t^{-y_t}\| + \|g_t\|] \|\delta_t\| \\ &\leq \frac{3R}{2\lambda} \|\delta_t\|.\end{aligned}$$

The last inequality is due to $\|g_t^y\| \leq R$ for all $\|x_t\| \leq R$ and $y \in \{-1, 1\}$. By definition of $\bar{\theta}_t \in \operatorname{argmin}_{\theta \in \mathbb{R}^d} \{\widehat{L}_{t-1}(\theta) + \ell_t(\theta) + \ell_t^{-y_t}(\theta)\}$, we have $\nabla \widehat{L}_{t-1}(\bar{\theta}_t) + \nabla \ell_t(\bar{\theta}_t) + \nabla \ell_t^{-y_t}(\bar{\theta}_t) = 0$. Note also that $g_t^y = \nabla \ell_t^y(\widehat{\theta}_t)$ for all $y \in \{-1, 1\}$. So δ_t may be rewritten as

$$\delta_t = \nabla \widehat{L}_{t-1}(\widehat{\theta}_t) + \nabla \ell_t(\widehat{\theta}_t) + \nabla \ell_t^{-y_t}(\widehat{\theta}_t) - \nabla \widehat{L}_{t-1}(\bar{\theta}_t) - \nabla \ell_t(\bar{\theta}_t) - \nabla \ell_t^{-y_t}(\bar{\theta}_t)$$

Using that $\nabla \ell_t$ and $\nabla \widehat{\ell}_t$ are $R^2/4$ -Lipschitz (for $\nabla \widehat{\ell}_t$ remark that $\|\nabla^2 \widehat{\ell}_t(\theta)\| = \|\eta_t g_t g_t^\top\| \leq R^2/4$), we have

$$\|\delta_t\| \leq \left[\frac{(t+1)R^2}{4} + 2\lambda B \right] \|\widehat{\theta}_t - \bar{\theta}_t\|. \quad (23)$$

Summing over t leads to

$$\sum_{t=1}^n \Omega_t \leq 3nR \left(\frac{nR^2}{8\lambda} + B \right) \varepsilon. \quad (24)$$

B.3 Conclusion of the proof

Using inequalities (12), (13) and (19), we have

$$R_n(\theta) \leq \widehat{R}_n(\theta) \leq \lambda \|\theta\|^2 + \sum_{t=1}^n Z_t + \sum_{t=1}^n \Omega_t$$

Finally, inequalities (21) and (24) concludes the proof. \square

C Lemmas

Proof of Lemma 5. Let $C > 0$. First, note that for all $x \in \mathbb{R}$, $f'(x) = -(1 + \exp(x))^{-1}$. To prove Lemma 5, we need to show that for $\alpha = (1 + C)^{-1}$, we have for all $a \in [-C, C]$ and $b \in \mathbb{R}$

$$\log(1 + e^{-a}) \geq \log(1 + e^{-b}) - \frac{1}{1 + e^b}(a - b) + \frac{\alpha}{2} \frac{e^b}{(1 + e^b)^2}(a - b)^2.$$

To do so, we fix $b \in \mathbb{R}$ and we define the function ξ as

$$\xi(a) = \log(1 + e^{-a}) - \log(1 + e^{-b}) + \frac{1}{1 + e^b}(a - b) - \frac{\alpha}{2} \frac{e^b}{(1 + e^b)^2}(a - b)^2, \quad -C \leq a \leq C$$

It remains to show that ξ is non-negative on $[-C, C]$. Because $\xi(b) = 0$, it suffices to prove

$$\xi'(a) \begin{cases} \leq 0 & \text{for } a \leq b \\ \geq 0 & \text{for } a \geq b \end{cases} \quad (25)$$

First, after some computation, differentiating ξ leads to

$$\xi'(a) = -\frac{1}{1 + e^a} + \frac{1}{1 + e^b} - \frac{\alpha e^b}{(1 + e^b)^2}(a - b),$$

which can also be rewritten as

$$\xi'(a) = \frac{e^a - e^b}{(1 + e^a)(1 + e^b)} - \alpha \frac{e^b}{(1 + e^b)^2}(a - b).$$

Reorganizing the terms gives the following equation

$$(1 + e^a)(1 + e^b)e^{-b}\xi'(a) = e^{a-b} - 1 - \alpha \frac{1 + e^a}{1 + e^b}(a - b).$$

Therefore, (25) holds true as soon as

$$\alpha \leq \frac{e^{a-b} - 1}{a - b} \frac{1 + e^b}{1 + e^a},$$

with the convention $(e^0 - 1)/0 = 1$. The latter is satisfied by Lemma 8 below, because $\alpha = (1 + C)^{-1} \leq (1 + |a|)^{-1}$ for all $a \in [-C, C]$. \square

Lemma 8. For all $a, b \in \mathbb{R}$,

$$\frac{1}{1 + |a|} \leq \frac{e^{a-b} - 1}{a - b} \frac{1 + e^b}{1 + e^a}.$$

Proof Define the function $h : \mathbb{R}^2 \mapsto \mathbb{R}$ that corresponds to the right-hand side of the inequality

$$h(a, b) = \frac{e^{a-b} - 1}{a - b} \frac{1 + e^b}{1 + e^a}, \quad (a, b) \in \mathbb{R}^2.$$

It is worth pointing out that even h is normally not defined for $a = b$, setting $h(a, a) = 1$ makes it well defined and infinitely differentiable on \mathbb{R}^2 .

Let $a \geq b$. Then $(1 + e^b)/(1 + e^a) \geq e^{b-a}$, which implies

$$h(a, b) \geq \frac{(e^{a-b} - 1)e^{b-a}}{(a - b)} \geq \frac{e^{b-a} - 1}{b - a} \geq \frac{1}{1 + a - b},$$

where the last inequality is because $(e^x - 1)/x \geq (1 - x)^{-1}$ for all $x \leq 0$.

Otherwise, let $a \leq b$. Then $(1 + e^b)/(1 + e^a) \geq 1$, which entails

$$h(a, b) \geq \frac{e^{a-b} - 1}{(a - b)} \geq \frac{1}{1 + b - a}$$

Combining the two cases $a \leq b$ and $a \geq b$ together, we get

$$h(a, b) \geq \frac{1}{1 + |a - b|}. \quad (26)$$

Now we show that $\operatorname{argmin}_{b \in \mathbb{R}} h(a, b)$ contains a value between 0 and a , i.e., in $[0, a]$ if $a \geq 0$ or in $[a, 0]$ otherwise. Rewriting the function h as follows,

$$h(a, b) = \frac{1}{1 + e^a} \frac{e^{a-b} - 1}{a - b} + \frac{1}{1 + e^{-a}} \frac{e^{b-a} - 1}{b - a}, \quad (27)$$

it is clear that $h(a, b) = h(-a, -b)$. We can therefore suppose without loss of generality that a is non-negative. Indeed, if $b^*(a) \in \operatorname{argmin}_b h(a, b)$ then $-b^*(a) \in \operatorname{argmin}_b h(-a, b)$. A further look at Equation (27) shows also that h is convex in its second argument by convexity of the function $x \mapsto (e^x - 1)/x$ and stability of convex functions by composition with affine transformations and non-negative weighted sum.

To finish the proof, we will show that the derivative of $\partial h(a, b)/\partial b$ is non-negative for $b \rightarrow a$ and non-positive for $b = 0$. Convexity of h in its second argument will then conclude. Using $a \geq 0$, some computations (omitted here) lead to

$$\frac{\partial}{\partial b} h(a, b) = \frac{(b - a + 1)e^{a-b} - 1 + (b - a - 1)e^b + e^a}{(a - b)^2(1 + e^a)}.$$

Developping the first terms of the exponential series gives

$$(b - a + 1)e^{a-b} - 1 + (b - a - 1)e^b + e^a = \frac{1}{2}(e^a - 1)(a - b)^2 + o_{b \rightarrow a}((a - b)^2).$$

Therefore,

$$\lim_{b \rightarrow a} \frac{\partial}{\partial b} h(a, b) = \frac{1}{2} \frac{e^a - 1}{e^a + 1} \geq 0.$$

We note also that if $a = 0$ then $\frac{\partial}{\partial b} h(0, 0) = 0$. Now, if $a > 0$, using $(2 - x)e^x \leq 2 + x$ for all $x \geq 0$, we have

$$\frac{\partial}{\partial b} h(a, 0) = (2 - a)e^a - (a + 2) \leq 0$$

By convexity of the function $b \mapsto h(a, b)$, we conclude that $\operatorname{argmin}_{b \in \mathbb{R}} h(a, b) \in [0, a]$. Combined with Inequality (26) concludes the proof of the lemma. \square

The following Lemma is a standard result of online matrix theory (Lemma 11.11 of [Cesa-Bianchi and Lugosi, 2006]).

Lemma 9. *Let $V \in \mathbb{R}^{d \times d}$ be an invertible matrix, $u \in \mathbb{R}^d$ and $U = V - uu^\top$. Then,*

$$u^\top V^{-1} u = 1 - \frac{\det(U)}{\det(V)}.$$

Lemma 10. *If $C_n = \sum_{t=1}^n \frac{\eta_t}{2} g_t g_t^\top$ and $A_n = C_n + \lambda I$ then*

$$\sum_{t=1}^n \frac{\eta_t}{2} g_t^\top A_t^{-1} g_t = \sum_{k=1}^d \log \left(1 + \frac{\lambda_k(C_n)}{\lambda} \right)$$

where $\lambda_k(C_n)$ is the k largest eigenvalue of C_n .

Proof Remarking that $A_t = A_{t-1} + \frac{\eta_t}{2} g_t g_t^\top$ and applying lemma 9 we have

$$\frac{\eta_t}{2} g_t^\top A_t^{-1} g_t = 1 - \frac{\det(A_{t-1}/\lambda)}{\det(A_t/\lambda)}$$

We use now that $1 - u \leq \log(1/u)$ for $u > 0$ which yields

$$\frac{\eta_t}{2} g_t^\top A_t^{-1} g_t \leq \log \frac{\det(A_{t-1}/\lambda)}{\det(A_t/\lambda)}$$

Summing over $t = 1, \dots, n$, using $A_0 = \lambda I$ and $A_n = C_n + \lambda I$ with $C_n = \sum_{t=1}^n \frac{\eta_t}{2} g_t g_t^\top$, we get

$$\begin{aligned} \sum_{t=1}^n \frac{\eta_t}{2} g_t^\top A_t^{-1} g_t &\leq \log \left(\det \left(I + \frac{C_n}{\lambda} \right) \right) \\ &= \sum_{k=1}^d \log \left(1 + \frac{\lambda_k(C_n)}{\lambda} \right) \end{aligned}$$

\square

D Efficient implementation of AIOLI

Proof of Lemma 2 Given the definition of $\hat{\theta}_t$, in (3), using the notation in Appendix A and Eq. (14)

$$\begin{aligned} \hat{\theta}_t &= \operatorname{argmin}_{\theta \in \mathbb{R}^d} \widehat{L}_{t-1}(\theta) + \log(1 + e^{-\theta^\top x_t}) + \log(1 + e^{\theta^\top x_t}) \\ &= \operatorname{argmin}_{\theta \in \mathbb{R}^d} \theta^\top A_{t-1} \theta - 2\theta^\top b_{t-1} + \log(1 + \cosh(\theta^\top x_t)). \end{aligned}$$

Since A_{t-1} is invertible by construction and $A_{t-1} = L_{t-1} L_{t-1}^\top$, where L_{t-1} is lower triangular and the unique Cholesky decomposition of A_{t-1} [Golub and Van Loan, 2012], we can define the following equivalent problem, by the substitution $r = L_{t-1}^\top \theta$

$$r_t = \operatorname{argmin}_{r \in \mathbb{R}^d} r^\top r - 2r^\top \tilde{u}_t + \log(1 + \cosh(r^\top \tilde{v}_t)), \quad \tilde{u}_t = L_{t-1}^{-1} b_{t-1}, \quad \tilde{v}_t = L_{t-1}^{-1} x_t,$$

$$L_0 = \lambda^{-1/2}I, \tilde{b}_0 = 0, \tilde{\theta}_0 = 0$$

for $t = 1, \dots, n$ **do**

receives x_t

$$W_t = L_t^{-1}(b_{t-1}, x_t), \{U_t, \Sigma_t\} = \text{eigen-dec}(W_t^\top W_t), u_t = \Sigma_t^{1/2}U_t^\top e_1, v_t = \Sigma_t^{1/2}U_t^\top e_2$$

$$\omega_t^0 = 0$$

for $i = 1, \dots, T$ **do**

$$\omega_t^i = \omega_t^{i-1} - \frac{\lambda}{4\lambda + R^2} \left[2\omega_t^{i-1} - 2u_t - (1 + e^{v_t^\top \omega_t^{i-1}})^{-1}v_t + (1 + e^{-v_t^\top \omega_t^{i-1}})^{-1}v_t \right]$$

end

$$\tilde{\theta}_t = L_t^{-\top} (W_t (U_t \Sigma_t^{-1/2} \omega_t^T))$$

predict $\hat{y}_t = \tilde{\theta}_t^\top x_t$

receives y_t

$$g_t = -(1 + e^{y_t \tilde{\theta}_t^\top x_t})^{-1} y_t x_t, \quad \eta_t = \frac{e^{y_t \tilde{\theta}_t^\top x_t}}{1 + B R}$$

$$L_t = \text{chol-update}(L_{t-1}, \sqrt{\eta_t/2}g_t), \quad b_t = b_{t-1} + (\eta_t g_t^\top \tilde{\theta}_t - 1)g_t$$

end

Algorithm 2: AIOLI detailed version. Here eigen-dec corresponds to economic-eigendecomposition of a symmetric matrix and chol-update to the rank 1 Cholesky update [Golub and Van Loan, 2012]

and in particular $r_t = L_t^\top \tilde{\theta}_t$. Now note that any $r \in \mathbb{R}^d$ can be always written as $r = W_t q + \mu$ with $W_t = (\tilde{u}_t, \tilde{v}_t) \in \mathbb{R}^{d \times 2}$ for some $q \in \mathbb{R}^2$ and $\mu \in (\text{span } W_t)^\perp$, then $r_t = W_t q_t + \mu_t$ for q_t, μ_t defined as

$$\begin{aligned} (q_t, \mu_t) &= \underset{q \in \mathbb{R}^2, \mu \in (\text{span } W_t)^\perp}{\text{argmin}} \|W_t q + \mu\|^2 - 2\tilde{u}_t^\top (W_t q + \mu) + \log(1 + \cosh(\tilde{v}_t^\top (W_t q + \mu))) \\ &= \underset{q \in \mathbb{R}^2, \mu \in (\text{span } W_t)^\perp}{\text{argmin}} \|W_t q\|^2 + \|\mu\|^2 - 2\tilde{u}_t^\top W_t q + \log(1 + \cosh(\tilde{v}_t^\top W_t q)), \end{aligned}$$

where in the last inequality we use the fact that $\tilde{u}_t = W_t e_1$, $\tilde{v}_t = W_t e_2$ and $W_t^\top \mu = 0$, by construction. Now the solution of the problem above is given by $\mu_t = 0$ and q_t as

$$q_t = \underset{q \in \mathbb{R}^2}{\text{argmin}} \|W_t q\|^2 - 2\tilde{u}_t^\top W_t q + \log(1 + \cosh(\tilde{v}_t^\top W_t q)).$$

Now that in the problem above q is always applied to W_t , so in the case that W_t is not full rank then all the solutions of the form $q_t = q_t^0 + \zeta$ with $\zeta \in (\text{span } W_t^\top W_t)^\perp$ are admissible and leading to the same r_t . Then we can restrict the problem above as

$$q_t = \underset{q \in \text{span } W_t^\top W_t}{\text{argmin}} \|W_t q\|^2 - 2\tilde{u}_t^\top W_t q + \log(1 + \cosh(\tilde{v}_t^\top W_t q)).$$

To conclude, take the economic eigenvalue decomposition of $W_t^\top W_t$, i.e., $W_t^\top W_t = U_t \Sigma_t U_t^\top$ with $U_t \in \mathbb{R}^{2 \times p_t}$ with p_t the rank of $W_t^\top W_t$, such that $U_t^\top U_t = I$ and $\Sigma_t \in \mathbb{R}^{p_t \times p_t}$ is diagonal and positive [Golub and Van Loan, 2012].

Now we consider the substitution $\omega = \Sigma_t^{1/2} U_t^\top q$, whose inverse is $q = U_t \Sigma_t^{-1/2} \omega$ since $q \in \text{span } W_t^\top W_t$ and $U_t U_t^\top$ is the projection matrix whose span is exactly $\text{span } W_t^\top W_t$, i. e. $U_t U_t^\top q = q$ for any $q \in \text{span } W_t^\top W_t$, which leads to the equivalent problem

$$\omega_t = \underset{\omega \in \mathbb{R}^{p_t}}{\text{argmin}} \omega^\top \omega - 2u_t^\top \omega + \log(1 + \cosh(v_t^\top \omega)),$$

where

$$\begin{aligned} u_t &= \Sigma_t^{-1/2} U_t^\top W_t^\top \tilde{u}_t = \Sigma_t^{-1/2} U_t^\top W_t^\top W_t e_1 = \Sigma_t^{1/2} U_t^\top e_1, \\ v_t &= \Sigma_t^{-1/2} U_t^\top W_t^\top \tilde{v}_t = \Sigma_t^{-1/2} U_t^\top W_t^\top W_t e_2 = \Sigma_t^{1/2} U_t^\top e_2. \end{aligned}$$

Note that in particular $\omega_t = \Sigma_t^{1/2} U_t q_t$ and $q_t = U_t \Sigma_t^{-1/2} \omega_t$. Then

$$\hat{\theta}_t = L_{t-1}^{-\top} r_t = L_{t-1}^{-\top} W_t q_t = L_{t-1}^{-\top} W_t U_t \Sigma_t^{-1/2} \omega_t.$$

□

Proof of Lemma 3 Since Ω_t is smooth and strongly convex, we can apply standard results on gradient descent (see for example Theorem 3.10 of [Bubeck et al., 2015]), obtaining

$$\|\omega_t^T - \omega_t\| \leq e^{-T/(2\kappa_t)} \|\omega_t^0 - \omega_t\|,$$

when gradient descent is used with step-size $\gamma = 1/\beta_t$ and where $\kappa_t = \beta_t/\alpha_t$ with α_t a lower bound of the strong convexity constant of Ω_t and β_t an upper bound the Lipschitz constant of $\nabla\Omega_t$. Note indeed that if Ω_t is α -strongly convex for some α , it will be also α' -strongly convex, for any $0 < \alpha' \leq \alpha$; moreover if $\nabla\Omega_t$ is β -Lipschitz for some β , it will be also β' -Lipschitz, for any $\beta' \geq \beta$; for more details see Chapter 3.4 of [Bubeck et al., 2015]. Now, by construction $\alpha_t = 1$, indeed $\Omega_t(\omega) - \|\omega\|^2$ is still a convex problem. Moreover, for any $\omega, \omega' \in \mathbb{R}^2$, by the mean value theorem applied to the function $g : [0, 1] \rightarrow \mathbb{R}^2$ defined as $g(r) = \nabla\Omega(\omega + r(\omega' - \omega))$, there exists a $q \in \mathbb{R}^2$ such that

$$\nabla\Omega_t(\omega) - \nabla\Omega_t(\omega') = \nabla^2\Omega_t(q)(\omega' - \omega).$$

This implies that $\|\nabla\Omega_t(\omega) - \nabla\Omega_t(\omega')\| \leq \sup_q \|\nabla^2\Omega_t(q)\| \|\omega' - \omega\|$ so the Lipschitz constant of $\nabla\Omega_t$ is upper bounded by $\beta_t = \sup_q \|\nabla^2\Omega_t(q)\|$. The Hessian of Ω_t is defined as

$$\nabla^2\Omega_t(\omega) = 2I + \frac{1}{1 + \cosh(v_t^\top \omega)} v_t v_t^\top,$$

then

$$\sup_q \|\nabla^2\Omega_t(q)\| \leq 2 + \|v_t\|^2 \sup_w \frac{1}{1 + \cosh(v_t^\top w)} \leq 2 + \frac{\|v_t\|^2}{2}.$$

To conclude, note that v_t in Thm. 2 is defined as $v_t = (W_t^\top W_t)^{1/2} e_2$ with $e_2 = (0, 1)$, $W_t = L_{t-1}^{-1}(b_{t-1}, x_t)$, L_{t-1} the lower triangular Cholesky decomposition of A_{t-1} (i.e. $A_{t-1} = L_{t-1} L_{t-1}^\top$) and A_{t-1}, b_{t-1} defined in Eq. (5). Then

$$\|v_t\|^2 = v_t^\top v_t = e_2^\top U_t \Sigma_t U_t^\top e_2 = e_2^\top W_t^\top W_t e_2 = x_t^\top L_{t-1}^{-1} L_{t-1}^{-1} x_t = x_t^\top A_{t-1}^{-1} x_t.$$

So $\|v_t\|^2 \leq \|x_t\|^2 \|A_{t-1}^{-1}\| \leq R^2/\lambda$, since $\|x_t\| \leq R$ by assumption and $A_{t-1} \succeq \lambda I$ by construction. Finally $\beta_t = 2 + \|v_t\|^2/2 \leq 2 + R^2/(2\lambda)$ and $\alpha_t = 1$, then $\kappa_t \leq 2 + R^2/(2\lambda)$ and $\gamma_t = 1/(2 + R^2/(2\lambda))$. We have

$$\|\omega_t^T - \omega_t\| \leq \exp(-T/(2\kappa_t) + \log \|\cdot\|) \leq \exp(-T/(4 + R^2/\lambda) + \log \|\omega_t^0 - \omega_t\|).$$

To quantify $\|\omega_t^0 - \omega_t\|$ we need a bound for $\|\omega_t\|$. Note that, since Ω_t is smooth and convex, ω_t is characterized by $\nabla\Omega_t(\omega_t) = 0$, i.e. $2\omega_t - 2u_t - (1 + e^{v_t^\top \omega_t})^{-1} v_t + (1 + e^{-v_t^\top \omega_t})^{-1} v_t = 0$, from which

$$\|\omega_t\| \leq \|u_t\| + \sup_q \left| (1 + e^{-v_t^\top q})^{-1} - (1 + e^{v_t^\top q})^{-1} \right| \|v_t\|/2 \leq \|u_t\| + \|v_t\|/2.$$

Analogously to the case of v_t , by definition of u_t , we have $\|u_t\|^2 = e_1^\top W_t^\top W_t e_1 = b_{t-1}^\top A_{t-1}^{-1} b_{t-1}$, then $\|u_t\|^2 \leq \|b_{t-1}\|^2 \|A_{t-1}^{-1}\| \leq \|b_{t-1}\|^2/\lambda$. Now we need a bound for b_{t-1} . Note that for any $s \in \{1, \dots, t-1\}$, we have $g_s = \nabla \ell_s(\hat{\theta}_s) = -(1 + e^{y_s \hat{\theta}_s^\top x_s})^{-1} y_s x_s$, moreover $\eta_s = e^{y_s \hat{\theta}_s^\top x_s} / (1 + BR)$ and

$$(\eta_s g_s^\top \hat{\theta}_s - 1) g_s = \frac{y_s x_s^\top \hat{\theta}_s}{2 + 2 \cosh(y_s x_s^\top \hat{\theta}_s)} (1 + BR)^{-1} y_s x_s + \frac{1}{1 + e^{y_s x_s^\top \hat{\theta}_s}} y_s x_s.$$

Since $\sup_z |z/(2 + 2 \cosh(z))| \leq 1$, $y_s \in \{-1, 1\}$ and $\|x_s\| \leq R$ by assumption, we have $\|(\eta_s g_s^\top \hat{\theta}_s - 1) g_s\| \leq ((1 + BR)^{-1} + 1)R \leq 2R$, then

$$\|b_{t-1}\| = \frac{1}{2} \sum_{s=1}^{t-1} \|(\eta_s g_s^\top \hat{\theta}_s - 1) g_s\| \leq (t-1)R. \quad (28)$$

To conclude, $\|u_t\| \leq R\lambda^{-1/2}(t-1)$, $\|\omega_t\| \leq \|u_t\| + \|v_t\|/2 \leq R\lambda^{-1/2}t$. By choosing $\omega_t^0 = 0$, then $P_t = \|\omega_t\| \leq R\lambda^{-1/2}t$ and so $\|\omega_t^T - \omega_t\| \leq \varepsilon$, when choosing $T \geq (4 + R^2/\lambda) \log(R\lambda^{-1/2}t/\varepsilon)$. \square

Proof of Theorem 4 We first analyze the cost of one iteration of Algorithm 1 (which is detailed in Algorithm 2 presented above). Note that at each step t , the cost of the gradient descent algorithm performed to compute ω_t^T is the number of iterations T , since we are solving a p_t -dimensional problem, with $p_t \in \{1, 2\}$. The two most expensive operation performed at step t (excluding gradient descent) are the solution of triangular linear systems of dimensions $d \times d$ when computing $L_{t-1}^{-1}v$ or $L_{t-1}^\top v$ for some vector $v \in \mathbb{R}^d$, which costs $O(d^2)$ (this operation is performed 4 times). The other expensive operation is the rank 1 Cholesky update of L_{t-1} with the vector $\sqrt{\eta_t/2}g_t$, which costs $O(d^2)$ [Golub and Van Loan, 2012], indeed the eigendecomposition is performed on the matrix $W_t^\top W_t$ which is 2×2 . By repeating such operation for n steps, we obtain a total cost of

$$O(nd^2 + nT).$$

The upper-bound on the regret is a direct consequence of Theorem 7 and Lemma 3. \square