Recursive ridge regression using second-order stochastic algorithms

Antoine Godichon-Baggioni, Bruno Portier, Wei Lu

Recursive ridge regression using second-order stochastic algorithms

Antoine Godichon-Baggioni$^{(*)}$, Bruno Portier$^{(**)}$ and Wei Lu$^{(**)}$
$(*)$ Sorbonne-Université, Laboratoire de Probablités, Statistique et Modélisation„ 75005 Paris,
$(**)$ INSA Rouen Normandie, Laboratoire de Mathématiques de l'INSA, 76800 Saint Etienne du Rouvray,
antoine.godichon_baggioni@upmc.fr, bruno.portier,wei.lu@insa-rouen.fr

Abstract

Recursive second-order stochastic algorithms are presented for solving ridge regression problems in the linear and binary logistic case. The proposed algorithms allow us to update the estimates of ridge solution when the data arrive in continuous flow. We establish the almost sure convergence with rate of proposed algorithms. Numerical experiments on simulated and real-world data show the advantages of our algorithms compared to alternative methods.

Keywords: Ridge regression; stochastic optimization; stochastic Newton algorithm; recursive estimation; machine learning

1. Introduction

Ridge regression is a widely used method for multiple regression problems that uses an $l2$ penalty to shrink the coefficients of correlated predictors or to provide a solution to the high dimensional problem (the number of regressors is greater than the number of data). The method was first introduced by [13] in the context of linear regression and have been used in many fields of application where the problem of collinearity of the predictors was present [14, 19, 21]. This method was extended to the logistic regression framework by [27]. In [17], the authors showed how ridge estimators can be used in logistic regression to improve parameter estimates and reduce the error committed by other predictions.

More recently, this method has become a popular tool in the field of machine learning since the use of a Ridge penalty allows to automatically model large volumes of data without prior processing. For example, one can implement this method in a principal component regression (PCR) model [2]. Kernel ridge regression [1, 30] combines supervised classification and ridge regression. In interpretation of voltammetric signals, conjugation of ridge regression with Self-paced learning algorithm provides better performances than traditional models [11]. In the field of neural networks, ridge regression can be used to optimize extreme learning machines [20].

From a computational point of view, a ridge solution is easily obtained in the linear regression case, using the singular value decomposition method to find the optimal value of the penalty parameter (see e.g. [10]). However, for the logistic regression case, iterative algorithms must be used since the solution is not explicit. A ridge solution can be obtained via Newton Raphson methods [17]. A coordinate descent method has also been proposed in this context, which is much faster [8]. Nevertheless, the large volumes of data or the use of data arriving in continuous flow requires the use of recursive algorithms for the calculation of the estimates. The stochastic gradient algorithm [25] has naturally been proposed in this context (e.g. [26]) but sometimes it is not very efficient (see [3]).

In this work, we propose second-order stochastic algorithms to solve the Ridge regression problem in the framework of linear regression and binary logistic regression. More precisely, if X denotes a random vector of predictors of $\mathbb{R}^p$ and Y a real variable of interest or a discrete variable in $\{0, 1\}$, we propose stochastic Newton type algorithms to find the value of the couple $(\beta_0, \beta) \in \mathbb{R} \times \mathbb{R}^p$ which minimizes the convex function

$G : \mathbb{R}^{p+1} \longmapsto \mathbb{R}$ defined by

$$G(h_0, h) = \mathbb{E}\left[l(X, Y, h_0, h)\right] + \lambda \left\|h\right\|^2 =: \mathbb{E}\left[g(X, Y, h_0, h)\right], \tag{1}$$

where the penalty λ is a positive real number, and the function $l(X, Y, h_0, h)$ is equal to $(Y - h_0 - h^T X)^2$ for linear regression and $\log(1 + \exp(h_0 + h^T X) - (h_0 + h^T X)Y$ for logistic regression.

The difficulty in the construction of a second order stochastic algorithm lies in the estimation of the inverse of the Hessian matrix of the function to minimize. Indeed, it is necessary to be able to estimate this inverse in a recursive and efficient way. In some cases, it is possible to use the Riccati formula (cf [3]). A second difficulty comes from the penalty term which requires a recursive estimation of the identity matrix. Inspired by the work of [9] and [6], we propose an approach based on a double use of Riccati's formula to obtain asymptotically efficient estimators for Ridge type problems. In practice, these estimators give much better results than the usual online estimators taking into account only first order information of stochastic gradient type. Indeed, they enable to adapt the steps in each direction, which is of particular interest for ill-conditioned problems [5, 18].

The paper is organized as follows. In Section 2, we consider the case of the ridge linear regression while Section 3 concerns the case of ridge logistic regression. Section 4 is devoted to numerical experiments on simulated and real data. Proofs are postponed in Section 5.

2. Ridge linear regression

2.1. The centered case

Framework.. Let (X, Y) be a random vector lying in $\mathbb{R}^p \times \mathbb{R}$. Let us first suppose that X is a centered random vector, with a finite moment of order 2, and Y is a random variable with variance σ^2. The aim is to approximate Y by a linear function of the form $z^T X + z_0$ where the values of $z_0 \in \mathbb{R}$ and $z \in \mathbb{R}^p$ are obtained by minimizing the penalized least squares criterion defined by:

$$G(z_0, z) = \mathbb{E}\left[(Y - z^T X - z_0)^2\right] + \lambda \left\|z\right\|^2 \tag{2}$$

where λ is a strictly positive real parameter. It is easy to prove that the solution (β_0, β_λ) of this minimization problem is unique and given by $\beta_0 = \mathbb{E}\left[Y\right]$ and

$$\beta_\lambda = \left(\mathbb{E}\left[XX^T + \lambda I_p\right]\right)^{-1} \mathbb{E}\left[XY\right] = (\Gamma + \lambda I_p)^{-1} \mathbb{E}\left[XY\right] \tag{3}$$

where I_p is the identity matrix of order p and $\Gamma = \mathbb{E}\left[XX^T\right]$.

Remark 2.1. *If there exists $\beta_0 \in \mathbb{R}$ and $\beta \in \mathbb{R}^p$ such that $\mathbb{E}\left[Y|X\right] = \beta^T X + \beta_0$, then $\beta_\lambda = (\Gamma + \lambda I_p)^{-1}\Gamma\beta$. Moreover, if $\lambda = 0$ and the matrix Γ is positive definite, then $\beta_\lambda = \beta$.*

In the following, we will focus on the recursive estimation of the parameter β_λ using two approaches, the first one, by estimating directly β_λ from its expression given in (3), the second one, by using a stochastic Newton algorithm [5, 18] to minimize the function G given in (2).

Recursive estimation of β_λ.. Let us suppose that we have a sequence of independent and identically distributed random vectors $(X_n, Y_n)_{n\geq 1}$ with the same distribution as (X, Y). One can easily suggest a non-recursive estimator of parameter β_λ. We can indeed propose :

$$\beta_n = \left(\alpha I_p + \sum_{j=1}^{n} X_j X_j^T + n\lambda I_p\right)^{-1} \sum_{j=1}^{n} X_j Y_j = (S_n + n\lambda I_p)^{-1} \sum_{j=1}^{n} X_j Y_j$$

where for all $n \geq 1$, we set $S_n = \alpha I_p + \sum_{j=1}^{n} X_j X_j^T$ with $\alpha \geq 0$. When the parameter β exists, that is when $\mathbb{E}\left[Y|X\right] = \beta^T X + \beta_0$, its ordinary least squares estimator is given by :

$$\widetilde{\beta}_n = S_n^{-1} \sum_{j=1}^{n} X_j Y_j.$$

This estimator can be recursively calculated using the formula

$$\widetilde{\beta}_n = \widetilde{\beta}_{n-1} + S_n^{-1} X_n (Y_n - \widetilde{\beta}_{n-1}^T X_n), \tag{4}$$

with $\widetilde{\beta}_0 = 0$. In addition, matrix S_n can be recursively inverted thanks to the following Riccati inversion formula (also called Sherman-Morrison formula) [7]:

$$S_n^{-1} = S_{n-1}^{-1} - (1 + X_n^T S_{n-1}^{-1} X_n)^{-1} S_{n-1}^{-1} X_n X_n^T S_{n-1}^{-1} \tag{5}$$

where we set $S_0^{-1} = \alpha^{-1} I_p$ with $\alpha > 0$, to avoid invertibility problem. This algorithm coincides with the stochastic Newton algorithm associated with the problem of minimizing the function G in the case where $\lambda = 0$ [5]. It of course coincides with the recursive least squares algorithm.

However, it will not be possible to do the same with β_n because of the additional term $n\lambda I_p$. To recursively calculate this estimator with the help of Riccati's formula, the matrix I_p must be estimated with an estimator of the form $n^{-1} \sum_{j=1}^n Z_j Z_j^T$ where (Z_j) are vectors of $\mathbb{R}^p$. One can imagine several ways to estimate the identity matrix. In particular, we can proceed as in [22, 9]. Let $e_1, e_2, \ldots, e_p$ be the p vectors of the canonical basis of $\mathbb{R}^p$. Note that $I_p = \sum_{j=1}^p e_j e_j^T$. Consider the sequence $(Z_n)_{n \geq 1}$ of vectors in $\mathbb{R}^p$ defined for all $n \geq 1$ by $Z_n = e_{(n \mod p)+1}$. We then have

$$I_p = \lim_{n \to \infty} \frac{p}{n} \sum_{j=1}^n Z_j Z_j^T \tag{6}$$

We then propose to estimate parameter β_λ by:

$$\widehat{\beta}_n = \left(S_n + p\lambda \sum_{j=1}^n Z_j Z_j^T \right)^{-1} \sum_{j=1}^n X_j Y_j \tag{7}$$

under suitable weak conditions and standard arguments, one can prove the convergence of $\widehat{\beta}_n$ to β_λ.

Theorem 1. *Assume that X admits a moment of order 4 and that XY admits a second order moment. Then,*

$$\left\| \widehat{\beta}_n - \beta_\lambda \right\|^2 = O\left(\frac{\ln \ln n}{n} \right) \quad a.s.$$

The proof is given in Section 5. We now focus on the practical implementation of the algorithm. Indeed, using two Riccati inversion formulas in a row [6], we can recursively compute $\widehat{\beta}_n$. For all $n \geq 1$, we set $W_n = Q_{n-1} + p\lambda Z_n Z_n^T$ and $Q_n = W_n + X_n X_n^T$ with $Q_0 = \alpha I_p$. We have for all $n \geq 0$,

$$S_n + p\lambda \sum_{j=1}^n Z_j Z_j^T = Q_n$$

The recursive estimation algorithm of the parameter β_λ is as follows:

$$\begin{aligned} W_n^{-1} &= Q_{n-1}^{-1} - \frac{p\lambda}{1 + Z_n^T Q_{n-1}^{-1} Z_n} Q_{n-1}^{-1} Z_n Z_n^T Q_{n-1}^{-1} \\ Q_n^{-1} &= W_n^{-1} - (1 + X_n^T W_n^{-1} X_n)^{-1} W_n^{-1} X_n X_n^T W_n^{-1} \\ \widehat{\beta}_n &= \widehat{\beta}_{n-1} + Q_n^{-1} \left(X_n (Y_n - \widehat{\beta}_{n-1}^T X_n) - \lambda p Z_n Z_n^T \widehat{\beta}_{n-1} \right) \end{aligned} \tag{8}$$

The stochastic Newton algorithm. . Note that the stochastic Newton algorithm associated with the problem of minimizing the function G is slightly different than the recursive calculation of $\widehat{\beta}_n$. Indeed, the stochastic Newton algorithm is defined by:

$$\widehat{\beta}_n^{SN} = \widehat{\beta}_{n-1}^{SN} + Q_n^{-1} \left(X_n (Y_n - X_n^T \widehat{\beta}_{n-1}^{SN}) - \lambda \widehat{\beta}_{n-1}^{SN} \right) \tag{9}$$

with $\widehat{\beta}_0^{SN}$ bounded.

Theorem 2. *Assume that there exists $\eta > 0$ such that X and Y admit respectively moment of order $4+4\eta$ and $2+2\eta$. Then the estimator proposed by stochastic Newton's algorithm defined by (9) satisfies*

$$\left\|\widehat{\beta}_n^{SN} - \beta_\lambda\right\|^2 = \mathcal{O}\left(\frac{\ln n}{n}\right) \text{ a.s.}$$

The proof is given in Section 5. The difference of hypothesis here is due to the fact that this result relies on the law of log iterated for martingales, while in Theorem 1, it relies on the law of the iterated logarithm for independent random variables.

2.2. The general case

Let us consider the case where the random vector X is not centered and denote its covariance

$$\Sigma := \mathbb{E}\left[(X - \mathbb{E}[X])(X - \mathbb{E}[X])^T\right].$$

The parameters $\beta_{0,\lambda} \in \mathbb{R}$ and $\beta_\lambda \in \mathbb{R}^p$ which minimize the penalized least squares criterion defined by (2) satisfy

$$\begin{aligned} \beta_\lambda &= (\Sigma + \lambda I_p)^{-1}\mathbb{E}\left[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])\right] \\ \beta_{0,\lambda} &= \mathbb{E}[Y] - \beta_\lambda \mathbb{E}[X]. \end{aligned}$$

Remark 2.2. *If there exists $\beta_0 \in \mathbb{R}$ and $\beta \in \mathbb{R}^p$ such that $\mathbb{E}[Y|X] = \beta_0 + \beta^T X$, then $\beta_\lambda = (\Sigma + \lambda I_p)^{-1}\Sigma\beta$ and $\beta_{0,\lambda} = \beta_0 + (\beta - \beta_\lambda)^T\mathbb{E}[X]$.*

As in the previous section, it is possible to recursively estimate β_λ. The algorithm then requires to estimate the covariance matrix Σ instead of the matrix Γ.To estimate the variance-covariance matrix Σ, we propose the estimator $n^{-1}C_n$ defined by :

$$C_n = \sum_{j=1}^{n}(X_j - \overline{X}_n)(X_j - \overline{X}_n)^T.$$

Remark that it is possible to recursively compute the matrix C_n as

$$C_n = C_{n-1} + \frac{n-1}{n}(X_n - \overline{X}_{n-1})(X_n - \overline{X}_{n-1})^T$$

with $C_0 = 0$. Then, the algorithm for estimating the parameter β_λ is recursively defined by

$$\begin{aligned} W_n^{-1} &= Q_{n-1}^{-1} - \frac{p\lambda}{1 + Z_n^T Q_{n-1}^{-1} Z_n} Q_{n-1}^{-1} Z_n Z_n^T Q_{n-1}^{-1} \\ Q_n^{-1} &= W_n^{-1} - (1 + \phi_n^T W_n^{-1}\phi_n)^{-1} W_n^{-1}\phi_n\phi_n^T W_n^{-1} \\ \widehat{\beta}_n &= \widehat{\beta}_{n-1} + Q_n^{-1}\left(\phi_n\left(\mathbf{Y}_n + \phi_n^T\widehat{\beta}_{n-1}\right) - \lambda p Z_n Z_n^T\widehat{\beta}_{n-1}\right) \qquad (10) \\ \widehat{\beta}_{n,0} &= \overline{Y}_n - \widehat{\beta}_n^T\overline{X}_n \qquad (11) \end{aligned}$$

with $Q_0 = \alpha I_p$, $\widehat{\beta}_0$ bounded, $\phi_n = \sqrt{(n-1)/n}\left(X_n - \overline{X}_{n-1}\right)$, $\mathbf{Y_n} = \sqrt{(n-1)/n}\left(Y_n - \overline{Y}_{n-1}\right)$ and $Z_n = e_{(n \mod p)+1}$. Remark that $\overline{X}_n$ and $\overline{Y}_n$ can of course be easily updated. The following theorem gives the almost sure rate of convergence of the estimates.

Theorem 3. *We suppose that there exists $\eta > 0$ such that X and Y admit respectively moment of order $4+4\eta$ and $2+2\eta$, then the estimators defined by (10) and (11) satisfy*

$$\left\|\hat{\beta}_n - \beta_\lambda\right\|^2 = \mathcal{O}\left(\frac{\ln n}{n}\right) \text{ a.s.} \qquad \text{and} \qquad \left\|\hat{\beta}_{n,0} - \beta_0\right\|^2 = \mathcal{O}\left(\frac{\ln n}{n}\right) \text{ a.s.}$$

The proof is given in Section 5.

Remark 2.3. *We can easily propose a stochastic Newton algorithm based on the same idea in the section 3.2.*

2.3. Remark on the recursive least squares estimator.

To close this section, let us consider the historical framework of Ridge regression which consists, starting from a sequence of N observations $(X_j, Y_j)_{1\leq j\leq N}$ of $\mathbb{R}^p \times \mathbb{R}$, in finding the value that minimizes the following criterion :

$$G(z_0, z) = \sum_{j=1}^{N} (Y_j - z_0 - z^T X_j)^2 + \lambda \|z\|^2 .$$

The ridge solution is then given by the couple $(\overline{Y}_N, \widehat{\beta}_N)$ with $\widehat{\beta}_N = (S_N + \lambda I_p)^{-1} \sum_{j=1}^{N} X_j Y_j$ and the computation and the search for the optimal value of the regularity parameter λ is done via the use of the singular value decomposition of the matrix S_N. We can however note that it is possible to compute recursively the estimator $\widehat{\beta}_N$ using the formulas defining the recursive least squares estimator (4)-(5) by choosing $\alpha = \lambda$ (cf. [15]). When p is large, this method of computation can be more efficient than the direct computation.

3. Ridge logistic regression

3.1. Framework

Let X be a random vector of $\mathbb{R}^p$ and Y be a random variable in $\{0, 1\}$. We assume that the conditional distribution of $Y|X$ is a Bernoulli distribution. More precisely,

$$\mathcal{L}(Y|X) = \mathcal{B}\left(\pi(\beta_0 + \beta^T X)\right) \qquad \text{with} \qquad \pi(x) = \frac{\exp(x)}{1 + \exp(x)},$$

where $(\beta_0, \beta) = (\beta_0, \beta_1, \ldots, \beta_p)$ is a vector of $\mathbb{R}^{p+1}$ giving the vector of unknown parameters. It is easy to show that (β_0, β) minimizes the function G defined for all $z_0 \in \mathbb{R}$ and $z \in \mathbb{R}^p$ by:

$$G(z_0, z) = = \mathbb{E}\left[\log\left(1 + \exp(z_0 + z^T X)\right) - (z_0 + z^T X)Y\right].$$

Estimating the ridge logistic regression parameter consists in minimizing the penalized criterion defined for all $(z_0, z) \in \mathbb{R} \times \mathbb{R}^p$ by :

$$G_\lambda(z_0, z) = G(z_0, z) + \lambda \|z\|^2 = \mathbb{E}\left[g_\lambda(X, Y, (z_0, z))\right]. \tag{12}$$

where $\lambda > 0$ is the penalty term. Let us denote by β_λ the vector of $\mathbb{R}^{p+1}$ defined by:

$$\beta_\lambda = \underset{(z_0,z)\in\mathbb{R}^{p+1}}{\arg\min}\ G_\lambda(z_0, z)$$

Unlike the linear ridge regression, the parameter β_λ does not have an explicit expression. Remark that if X admits a first order moment, the functional G is differentiable with

$$\nabla_{z_0} G_\lambda(z_0, z) = \mathbb{E}\left[\pi\left(z_0 + z^T X\right) - Y\right] \quad \text{and} \quad \nabla_z G_\lambda(z_0, z) = \mathbb{E}\left[\pi(z_0 + z^T X)X - XY\right] + \lambda z.$$

For all $x \in \mathbb{R}$, let us denote $\alpha(x) = \pi(x)(1 - \pi(x))$. Let A_{p+1} be the square matrix of order $(p + 1)$ defined by :

$$A_{p+1} = \sum_{j=2}^{p+1} e_j e_j^T = I_{p+1} - e_1 e_1^T$$

where $e_1, \ldots, e_{p+1}$ represent the vectors of the canonical basis of $\mathbb{R}^{p+1}$. If X admits a second order moment, the Hessian matrix of G_λ is defined by:

$$H_\lambda(z_0, z) = \begin{pmatrix} \mathbb{E}\left[\alpha(z_0 + z^T X)\right] & \mathbb{E}\left[\alpha(z_0 + z^T X)X^T\right] \\ \mathbb{E}\left[\alpha(z_0 + z^T X)X\right] & \mathbb{E}\left[\alpha(z_0 + z^T X)XX^T\right] + \lambda I_p \end{pmatrix} = H(z_0, z) + \lambda\, A_{p+1}$$

where the matrix H represents the Hessian matrix of the function G. Note that the matrix H_λ is positive definite, and we can so consider in the sequel a stochastic Newton algorithm for estimating parameter β_λ.

3.2. The stochastic Newton algorithm

Let us suppose that we have a sequence of independent and identically distributed random vectors $(X_n, Y_n)_{n\geq 1}$ taking values in $\mathbb{R}^p \times \{0,1\}$, such that for any $n \geq 1$, $\mathcal{L}(Y_n|X_n) = \mathcal{B}\Big(\pi(\beta_0 + \beta^T X_n)\Big)$. For all $n \geq 1$, let us set $\phi_n = (1, X_n^T)^T$. In order to estimate the parameter β_λ minimizing the function G_λ, we propose a stochastic Newton algorithm of the form:

$$\widehat{\beta}_n = \widehat{\beta}_{n-1} - Q_n^{-1} \nabla g_\lambda(X_n, Y_n, \widehat{\beta}_{n-1})$$

where $\overline{Q}_n = n^{-1} Q_n$ is an estimator of the Hessian matrix $H_\lambda(\beta_\lambda) = H(\beta_\lambda) + \lambda A_{p+1}$. The difficulty is now to propose a recursive estimator of the matrix $H_\lambda^{-1} := H_\lambda\,(\beta_\lambda)^{-1}$. In order to use the ideas introduced in the previous section, we must first propose an estimator of the matrix A_{p+1} which allows the use of Riccati's formula to recursively compute its inverse.
Let $(Z_n)_{n\geq 1}$ be the sequence of vectors of $\mathbb{R}^{p+1}$ defined by $Z_n = c_\gamma n^{-\gamma} e_1$ with $c_\gamma > 0$ and $0 < \gamma < 1/4$ if $n \mod (p+1) = 0$, and $Z_n = e_{n \mod (p+1)+1}$ otherwise, where $(e_j)_{1\leq j\leq p+1}$ is the canonical basis of $\mathbb{R}^{p+1}$. Then $n^{-1}(p+1)\sum_{j=1}^n Z_j Z_j^T \underset{n\to\infty}{\longrightarrow} A_{p+1}$. Note that considering $c_\gamma n^{-\gamma}$ for the first coordinate is purely technical, and enables to verify that assumptions in [5] are satisfied, so as to obtain the rate of convergence of Newton estimates.

By taking the ideas introduced in the previous section and adapting them to this problem, we can then develop the following recursive Stochastic Newton algorithm to estimate β_λ.

$$\begin{aligned}
W_n^{-1} &= Q_{n-1}^{-1} - \frac{(p+1)\lambda}{1+\lambda(p+1)Z_n^T Q_{n-1}^{-1} Z_n} Q_{n-1}^{-1} Z_n Z_n^T Q_{n-1}^{-1} \\
Q_n^{-1} &= W_n^{-1} - \alpha(\widehat{\beta}_{n-1}^T \phi_n)(1+\alpha(\widehat{\beta}_{n-1}^T\phi_n)\phi_n^T W_n^{-1}\phi_n)^{-1} W_n^{-1}\phi_n\phi_n^T W_n^{-1} \\
\widehat{\beta}_n &= \widehat{\beta}_{n-1} + Q_n^{-1}\Big(\phi_n(Y_n - \pi(\widehat{\beta}_{n-1}^T\phi_n)) - \lambda A_{p+1}\widehat{\beta}_{n-1}\Big)
\end{aligned} \tag{13}$$

with $Q_0^{-1} = \alpha^{-1} I_{p+1}$ with $\alpha > 0$ and $\widehat{\beta}_0$ bounded. Let us make some comments about the different matrices involved in this algorithm. For all $n \geq 1$, we have $W_n = Q_{n-1} + \lambda(p+1)Z_n Z_n^T$ and $Q_n = W_n + \alpha(\widehat{\beta}_{n-1}^T\phi_n)\phi_n\phi_n^T$ with $Q_0 = \alpha I_{p+1}$. For all $n \geq 1$, we so have

$$Q_n = \alpha I_{p+1} + \sum_{j=1}^n \alpha(\widehat{\beta}_{j-1}^T\phi_j)\phi_j\phi_j^T + \lambda(p+1)\sum_{j=1}^n Z_j Z_j^T$$

Let us recall that $(p+1)n^{-1}\sum_{j=1}^n Z_j Z_j^T$ is an estimate of A_{p+1} while matrix $n^{-1}S_n$ with $S_n = \sum_{j=1}^n \alpha(\widehat{\beta}_{j-1}^T\phi_j)\phi_j\phi_j^T$ is an estimate of $H(\beta_\lambda)$. Therefore, $\overline{Q}_n := \frac{1}{n}Q_n$ is an estimator of H_λ.

3.3. Convergence results

We are now interested in the convergence rate of the estimator $\widehat{\beta}_n$. More precisely, the following theorem gives the almost sure rate of convergence as well as the asymptotic efficiency of the estimates.

Theorem 4. *Assume that there exists $\eta > 0$ such that X and Y admit moments of order $4+4\eta$ and $2+2\eta$, then the estimates defined by* (13) *satisfy*

$$\Big\|\widehat{\beta}_n - \beta_\lambda\Big\|^2 = \mathcal{O}\left(\frac{\ln n}{n}\right) \text{ a.s.}$$

In addition,

$$\sqrt{n}(\hat{\beta}_n - \beta_\lambda) \xrightarrow[n\to+\infty]{\mathcal{L}} \mathcal{N}(0, H_\lambda^{-1}\Sigma_\lambda H_\lambda^{-1}),$$

where $\Sigma_\lambda = \mathbb{E}\left[\nabla_h g_\lambda(X,Y,\theta)\nabla_h g_\lambda(X,Y,\theta)^T\right]$.

The proof is given in section 5.

4. Experiments

In this section, we study the performances of the second-order recursive algorithm (SR defined by (8)) for ridge linear regression, and the stochastic newton algorithm (SN defined by (9) and (13)) for both ridge linear regression and ridge logistic regression. We use the statistical software R [24] to carry out our numerical experiments. In these experiments, we compare our methods to the averaged implicit stochastic gradient descent (AI-SGD) proposed by [29] and available in the R-package `sgd`. As SR and SN algorithms, AI-SGD algorithm is also a recursive algorithm, but it is a first order method. However, it usually achieves superior results to classical stochastic gradient descent. When the data are regularized, we also compare our methods with the cyclical coordinate descent (CDD) proposed by [8] and available in the R-package `glmnet`. Note that CCD is an iterative method, not adapted to sequentially process data. Nevertheless, it is very efficient and widely used in practice. In this section, we first consider simulated data, and then some real data sets.

4.1. Experiments on simulated data

First, we illustrate the theoretical results with synthetic data. For both ridge linear regression and ridge logistic regression, we follow an example from [31]. We chose this model, since it considers a random vector whose covariance matrix has eigenvalues at very different scales. In this case first-order algorithms can be sensitive, so that it may be meaningful to use second-order algorithms.

4.1.1. Experimental model

We generated a standard Gaussian matrix $\tilde{\boldsymbol{X}} \in \mathbb{R}^{p\times N}$ with $N = 12000$ and $p = 200$. We computed the SVD (Singular Value Decomposition) of the matrix $\tilde{\boldsymbol{X}}$, i.e. we got $\tilde{\boldsymbol{X}} = U\tilde{D}V^T$. We then generated the data matrix $\boldsymbol{X}$ with $\boldsymbol{X} = \sqrt{N}UDV^T$, where $D = \text{diag}(1, 1/2^2, 1/3^2, \ldots, 1/p^2)$. We generated the outcome values for linear regression task from the model

$$Y = \boldsymbol{X}\beta + \varepsilon,$$

and the outcome values for logistic regression task from the model

$$Y = \text{sign}(\boldsymbol{X}\beta + \varepsilon),$$

where β is a realization of standard Gaussian random variable, which is the parameter to estimate and $\varepsilon \sim \mathcal{N}_N(0, 0.1I_N)$ is the random vector error.

4.1.2. Protocol

We consider three different values of regularization parameters λ: $\lambda = 1/N_T, 100/N_T$ or $1000/N_T$, where N_T is the training sample size. The value $\lambda = 1/N_T$ is the case chosen in [31]. The other values are chosen to show the performances of algorithms in cases where the criterion is over penalized. For each value of λ, we estimate β_λ using the three algorithms SR,SN and AI-SGD. Each data is split into a training sample of size 10000 and a test sample of size 2000. To compare the three algorithms, we compute the percentage of explained variance (EV) and the Root Mean Square Error (RMSE) for linear regression task, and the accuracies for logistic regression task. We carry out the experimentation on 50 samples before computing the mean and the standard deviation (sd) of EV (in %), RMSE and accuracies (in %).

4.1.3. Comparison of the different algorithms

Table 1 is concerned with the performances of the three methods for the linear regression task. Results show that SN and SR algorithms perform almost identically. In addition, their performances are better and more stable than the first order method AI-SGD, especially in the cases where λ is small, i.e the cases where the eigenvalues of the covariance matrix are not over harmonized.

λ	Method	Training EV (sd)	Test RMSE (sd)
10^{-4}	SR	96.39 (7.73)	0.103 (0.002)
	SN	96.39 (7.73)	0.103 (0.002)
	AI-SGD	94.02 (9.90)	0.144 (0.021)
0.01	SR	94.94 (6.86)	0.138 (0.016)
	SN	94.95 (6.85)	0.137 (0.016)
	AI-SGD	93.52 (8.44)	0.158 (0.025)
0.1	SR	80.77 (19.47)	0.240 (0.062)
	SN	80.78 (19.46)	0.240 (0.062)
	AI-SGD	80.29 (19.61)	0.245 (0.064)

Table 1: Mean and standard deviation of RMSE and EV of SR, SN and AI-SGD algorithms in the linear regression case.

Table 2 presents the performances of SN and AI-SGD algorithms for the logistic regression task. We can see that the second-order method SN works better than the first-order method AI-SGD on both training sample and test sample, with higher mean and lower sd of accuracies. Similarly to the linear case, the improvement is more obvious in the cases where λ is small.

λ	Method	Training Accuracy (sd)	Test Accuracy (sd)
10^{-4}	SN	93.55 (4.00)	93.45 (4.15)
	AI-SGD	88.85 (5.97)	88.90 (6.10)
0.01	SN	89.77 (5.71)	89.74 (5.88)
	AI-SGD	88.30 (6.31)	88.39 (6.36)
0.1	SN	86.49 (8.05)	86.45 (8.16)
	AI-SGD	86.33 (8.12)	86.29 (8.25)

Table 2: Mean and standard deviation of Accuracy of SN and AI-SGD algorithms in the logistic regression case.

4.2. Experiments on real data

We consider two well-known data sets : BUZZ data set for linear regression task and COVTYPE data set for logistic regression task. As in the previous paragraph, we compare our methods with AI-SGD, but also with CCD. To do so, we regularize all data sets, because CCD can only work on standardized data.

4.2.1. Presentation of the data sets

BUZZ data set was created by [16], which contains examples of buzz events from the social network Twitter. Their study focuses on the problem of predicting the level of activity related to a keyword without having prior knowledge of the underlying social network. The data set was also studied by [31]. There exist 77 predictors, with more than 40 strongly correlated predictors, leading to a good example for ridge regression. In [31], the authors split the data set into 466600 training samples (80%) and 116650 test samples (20%).

COVTYPE data set is a well-known data set, which was collected in 1998 by [4], and was widely studied (see for example [29, 31, 28]). Based on 581011 observations and 54 predictors, the objective of the initial study was to predict the cover type of the forests located in Roosevelt National Park. In our study, we will restrict ourselves to the most frequent modality of the variable to be predicted "covertype", namely "Spruce/Fir", which represents 48.8% of the observations. The "cover-type" variable thus becomes a binary variable, with the "fir" modality by 1, and the other modalities by 0. We split randomly the data into 464809 training samples (80%) and 116202 test samples (20%).

4.2.2. Results and comments

For real data, the choice of regularization parameter λ is of course important and in practice it is often chosen using a cross-validation step. In [28], authors suggest to set λ to $1/n$ with n the number of observations, which is in the range of the smallest values that would generally be applied in practice. In the Python-package $sklearn$ [23], the value of λ also defaults to $1/n$. However, when data are sequentially obtained, cross-validation approach is forbidden and n is unknown, but it is always possible to arbitrarily choose the value of

λ so that strong correlations are taken into account. We therefore decided to perform experiments with $\lambda = 1$ and $\lambda = 1/n$.

For BUZZ dataset, we employ ridge linear regression to predict the activity volume. Results obtained for BUZZ dataset, are stated in Table 3. The proposed algorithms perform better than AI-SGD on BUZZ with higher explained variance and lower RMSE on both training sample and test sample, but they perform slightly worse than CCD. However, it is no less important to remember that CCD is a iterative method, while our algorithms are recursive.

	Method	SN	SR	AI-SGD	CCD
$\lambda = 1/n$	Training EV(%)	93.49	93.48	92.02	93.45
	Test EV(%)	93.96	93.95	93.28	93.96
	Training RMSE	159.09	159.13	176.07	159.46
	Test RMSE	139.24	139.27	146.82	139.24
$\lambda = 1$	Training EV(%)	89.48	89.48	84.61	93.43
	Test EV(%)	90.70	90.79	86.58	94.04
	Training RMSE	202.15	202.15	244.49	159.80
	Test RMSE	172.72	172.72	207.46	138.23

Table 3: Performances on BUZZ data set. EV and RMSE of SR, SN, AI-SGD and CCD algorithms.

For COVTYPE dataset, we use ridge logistic regression to predict the variable "covertype". We can observe from Table 4 that the proposed method achieves the same accuracy as CCD on COVTYPE, which is remarkable as our method is recursive. Moreover, it provides higher accuracy than AI-SGD on both training sample and test sample.

	Method	SN	AI-SGD	CCD
$\lambda = 1/n$	Training Acc(%)	75.59	75.22	75.59
	Test Acc(%)	75.66	75.37	75.66
$\lambda = 1$	Training Acc(%)	69.82	69.77	69.82
	Test Acc(%)	69.80	69.72	69.80

Table 4: Performances on COVTYPE data set. Accuracy of SN, AI-SGD and CCD algorithms.

Conclusion

In this paper, recursive second-order algorithms for ridge linear regression and ridge logistic regression have been proposed. These algorithms are perfectly adapted to the context of machine learning since they allow an online update of the ridge solution from large volumes of data. The originality of the paper is that our methods not only take into account second order information, but also avoid the inverse matrix calculation, making it more appropriate for usage in an online context. Observe that this study does not focus on the choice of the ridge parameter λ. Anyway, for any fixed choice of λ, the proposed algorithms can achieve good and stable results, which is confirmed with experiments.

5. Proof

5.1. Proof of Theorem 1

We set $\overline{Q}_n = \frac{1}{n}Q_n = \frac{1}{n}\left(S_n + p\lambda\sum_{j=1}^n Z_j Z_j^T\right)$ and $H := \Gamma + \lambda I_p$, then we have, denoting by $\|.\|_F$ the Frobenius norm for matrices

$$\begin{aligned}\left\|\overline{Q}_n - H\right\|_F^2 &= \left\|\frac{1}{n}\left(\alpha I_p + \sum_{k=1}^n X_k X_k^T + p\lambda\sum_{j=1}^n Z_j Z_j^T\right) - \Gamma - \lambda I_p\right\|_F^2 \\ &\leq 3\left\|\frac{1}{n}\sum_{k=1}^n X_k X_k^T - \mathbb{E}\left[XX^T\right]\right\|_F^2 + 3\left\|\frac{p\lambda}{n}\sum_{k=1}^n Z_k Z_k^T - \lambda I_p\right\|_F^2 + 3\left\|\frac{\alpha}{n}I_p\right\|_F^2.\end{aligned}$$

It is obvious that

$$\left\|\frac{\alpha}{n}I_p\right\|^2 = \mathcal{O}\left(\frac{1}{n^2}\right).$$

Since X admits moment of order 4, by the law of the iterated logarithm [12],

$$\left\|\frac{1}{n}\sum_{k=1}^n X_k X_k^T - \mathbb{E}\left[XX^T\right]\right\|_F^2 = \mathcal{O}\left(\frac{\ln\ln n}{n}\right) \quad a.s.$$

In addition, denoting by $\lfloor.\rfloor$ the integer part function,

$$\left\|\frac{p\lambda}{n}\sum_{k=1}^n Z_k Z_k^T - \frac{n}{n}\lambda I_p\right\|_F^2 = \left\|\frac{p\lambda}{n}\sum_{k=\lfloor\frac{n}{p}\rfloor p+1}^n Z_k Z_k^T + \left(\frac{p}{n}\left\lfloor\frac{n}{p}\right\rfloor - \frac{n}{n}\right)\lambda I_p\right\|_F^2.$$

Moreover,

$$\frac{p^2\lambda^2}{n^2}\left\|\sum_{k=\lfloor\frac{n}{p}\rfloor p+1}^n Z_k Z_k^T\right\|_F^2 \leq \frac{p^2\lambda^2}{n^2}\left(n - \left\lfloor\frac{n}{p}\right\rfloor p\right)\sum_{k=\lfloor\frac{n}{p}\rfloor p+1}^n \left\|Z_k Z_k^T\right\|_F^2 \leq \frac{p^3\lambda^2}{n^2}\sum_{k=\lfloor\frac{n}{p}\rfloor p+1}^n \left\|Z_k Z_k^T\right\|_F^2 \leq \frac{p^4\lambda^2}{n^2},$$

and

$$\left\|\left(\frac{p}{n}\left\lfloor\frac{n}{p}\right\rfloor - \frac{n}{n}\right)\lambda I_p\right\|_F^2 \leq \left\|\frac{p}{n}\lambda I_p\right\|_F^2 \leq \frac{p^4\lambda^2}{n^2}.$$

We then have

$$\left\|\overline{Q}_n - H\right\|_F^2 = \mathcal{O}\left(\frac{\ln\ln n}{n}\right) \quad a.s. \tag{14}$$

Note that the difference between two inverse matrices can be written as

$$\overline{Q}_n^{-1} - H^{-1} = \overline{Q}_n^{-1}(H - \overline{Q}_n)H^{-1},$$

so that

$$\left\|\overline{Q}_n^{-1} - H^{-1}\right\|_{op}^2 \leq \left\|\overline{Q}_n^{-1}\right\|_{op}^2 \left\|H - \overline{Q}_n\right\|_{op}^2 \left\|H^{-1}\right\|_{op}^2.$$

We proved that $\left\|\overline{Q}_n - H\right\|_F^2 = \mathcal{O}\left(\frac{\ln\ln n}{n}\right)$ a.s. Furthermore, we have $\left\|\overline{Q}_n^{-1}\right\|_{op}^2 \leq \frac{1}{\lambda_{min}(\overline{Q}_n)^2}$ and $\left\|H^{-1}\right\|_{op}^2 \leq \frac{1}{\lambda_{min}(H)^2}$. It is obvious that $\lambda_{min}(H) \geq \lambda$, thus $\left\|\overline{H}^{-1}\right\|_{op}^2 \leq \frac{1}{\lambda^2} = \mathcal{O}(1)$. Moreover, $\overline{Q}_n$ converges to H, so that we have also $\left\|\overline{Q}_n^{-1}\right\|_{op}^2 = \mathcal{O}(1)$ a.s. To sum up, we have

$$\left\|\overline{Q}_n^{-1} - H^{-1}\right\|_{op}^2 = \mathcal{O}\left(\frac{\ln\ln n}{n}\right) \quad a.s.$$

In addition, since XY admits a moment of order 2, we have, by the law of the iterated logarithm,

$$n^{-1}\left\|\sum_{j=1}^{n} X_jY_j - \mathbb{E}[XY]\right\|^2 = \mathcal{O}\left(\frac{\ln\ln n}{n}\right) \quad a.s.$$

We now focus on the rate of convergence of $\hat{\beta}_n$. We have $\beta_\lambda = H^{-1}\mathbb{E}[XY]$. Therefore,

$$\begin{aligned}
\left\|\hat{\beta}_n - \beta_\lambda\right\|^2 &= \left\|\frac{1}{n}\overline{Q}_n^{-1}\sum_{j=1}^{n} X_jY_j - H^{-1}\mathbb{E}[XY]\right\|^2 \\
&= \left\|(\overline{Q}_n^{-1} - H^{-1})\frac{1}{n}\sum_{j=1}^{n} X_jY_j + H^{-1}\left(\frac{1}{n}\sum_{j=1}^{n} X_jY_j - \mathbb{E}[XY]\right)\right\|^2 \\
&\leq 2\left\|(\overline{Q}_n^{-1} - H^{-1})\frac{1}{n}\sum_{j=1}^{n} X_jY_j\right\|^2 + 2\left\|H^{-1}\left(\frac{1}{n}\sum_{j=1}^{n} X_jY_j - \mathbb{E}[XY]\right)\right\|^2 \\
&\leq 2\left\|\overline{Q}_n^{-1} - H^{-1}\right\|_{op}^2\left\|\frac{1}{n}\sum_{j=1}^{n} X_jY_j\right\|^2 + 2\left\|H^{-1}\right\|_{op}^2\left\|\frac{1}{n}\sum_{j=1}^{n} X_jY_j - \mathbb{E}[XY]\right\|^2 \\
&= \mathcal{O}\left(\frac{\ln\ln n}{n}\right) \quad a.s
\end{aligned}$$

which concludes the proof.

5.2. Proof of Theorem 2

The aim is to apply Theorem 3.3 in [5], thus we have to prove that the hypotheses (A1b), (A1c), (A2a), (A2b), (A2c), (H1), (H2a) and (H2b) in [5] are satisfied.

Verification of (A1b). We have for all $h \in \mathbb{R}^p$,

$$g(X,Y,h) = \frac{1}{2}((Y - X^Th)^2 + \lambda\left\|h\right\|^2),$$

so that

$$\begin{aligned}
\nabla g(X,Y,h) &= -X(Y - X^Th) + \lambda h \\
&= -XY + XX^T\beta + XX^T(h-\beta) + \lambda(h-\beta) + \lambda\beta.
\end{aligned}$$

As X and ϵ admit moments of order $4+4\eta$ and $2+2\eta$, we have

$$\begin{aligned}
\mathbb{E}\left[\left\|\nabla g(X,Y,h)\right\|^{2+2\eta}\right] &\leq 5^{1+2\eta}\mathbb{E}\left[\left\|XY\right\|^{2+2\eta}\right] + 5^{1+2\eta}\mathbb{E}\left[\left\|X\right\|^{4+4\eta}\right]\left\|h-\beta\right\|^{2+2\eta} \\
&\quad + 5^{1+2\eta}(\lambda\left\|h-\beta\right\|)^{2+2\eta} + 5^{1+2\eta}(\lambda\left\|\beta\right\|)^{2+2\eta} + \mathbb{E}\left[\left\|X\right\|^{4+4\eta}\right]\mathbb{E}\left[\left\|\beta\right\|^{2+2\eta}\right] \\
&\leq C_\eta(1 + \left\|h-\beta\right\|^{2+2\eta}).
\end{aligned}$$

Then hypothesis (A1b) is satisfied.

Verification of (A1c). For all $h \in \mathbb{R}^p$,

$$\begin{aligned}
\Sigma(h) &:= \mathbb{E}\left[\nabla g(X,Y,h)\nabla g(X,Y,h)^T\right] \\
&= \mathbb{E}[(Y - X^Th)^2XX^T] + \lambda^2hh^T - 2\lambda\mathbb{E}[h^TX(Y - X^Th)] \\
&= \mathbb{E}[(Y - X^T\beta)^2XX^T] + \mathbb{E}[((X^T(h-\beta))^2XX^T] - 2\lambda\mathbb{E}[h^TX(Y - X^T\beta)] \\
&\quad - 2\lambda\mathbb{E}[h^TX(Y - X^T(h-\beta))] + \lambda^2hh^T.
\end{aligned}$$

As X admits a moment of order 4 and $(Y - X^{\beta})$ admits a moment of order 2, the function Σ is continuous on β. Hypothesis (A1c) is then satisfied.

Verification of (A2a). For all $h \in \mathbb{R}^p$, we have

$$\left\|\nabla^2 G(h)\right\|_{op} = \left\|\mathbb{E}\left[XX^T\right] + \lambda I_p\right\|_{op} \leq \mathbb{E}\left[\|X\|^2\right] + \lambda.$$

Hypothesis (A2a) is then satisfied.

Verification of (A2b) and (A2c). We have

$$\nabla^2 G(h) = \mathbb{E}\left[XX^T\right] + \lambda I_p.$$

Since $\mathbb{E}\left[XX^T\right]$ is non-negative, $\nabla^2 G(h)$ is positive definite and Hypothesis (A2b) is then satisfied. In addition, $\nabla^2 G(.)$ is 0-Lipschitz, and (A2c) is so satisfied.

Verification of (H2b). With the help of equality (14), one directly has, denoting $\overline{Q}_n := \frac{1}{n} Q_n$,

$$\left\|\overline{Q}_n - H\right\|_F^2 = O\left(\frac{\ln \ln n}{n}\right) \quad a.s$$

and hypothesis (H2b) is so satisfied, which concludes the proof, i.e all the hypotheses are satisfied, and according to [5], we obtain the conclusion.

5.3. Proof of Theorem 3

We set $\overline{Q}_n := \frac{1}{n} Q_n = \frac{1}{n}\left(C_n + \alpha I_p + p\lambda \sum_{j=1}^n Z_j Z_j^T\right)$ and $H = (\Sigma + \lambda I_p)$, then we have

$$\begin{aligned}
\left\|\overline{Q}_n - H\right\|_F^2 &= \left\|\frac{1}{n}\left(\alpha I_p + C_n + p\lambda \sum_{j=1}^n Z_j Z_j^T\right) - \Sigma - \lambda I_p\right\|_F^2 \\
&\leq 3\left\|\frac{1}{n} C_n - \Sigma\right\|_F^2 + 3\left\|\frac{p\lambda}{n}\sum_{k=1}^n Z_k Z_k^T - \lambda I_p\right\|_F^2 + 3\left\|\frac{\alpha}{n} I_p\right\|_F^2.
\end{aligned}$$

We have already checked that (see the proof of Theorem 1)

$$\left\|\frac{\alpha}{n} I_p\right\|_F^2 = \mathcal{O}\left(\frac{1}{n^2}\right) \qquad \text{and} \qquad \left\|\frac{p\lambda}{n}\sum_{k=1}^n Z_k Z_k^T - \lambda I_p\right\|_F^2 = O\left(\frac{1}{n^2}\right).$$

In addition, we have

$$\begin{aligned}
\left\|\frac{1}{n} C_n - \Sigma\right\|_F^2 &\leq 2\left\|\frac{1}{n}\sum_{k=1}^n \left(X_k - \mathbb{E}\left[X\right]\right)\left(X_k - \mathbb{E}\left[X\right]\right)^T - \Sigma\right\|_F^2 + 2\left\|\left(\overline{X}_n - \mathbb{E}\left[X\right]\right)\left(\overline{X}_n - \mathbb{E}\left[X\right]\right)^T\right\|_F^2 \\
&= 2\left\|\frac{1}{n}\sum_{k=1}^n \left(X_k - \mathbb{E}\left[X\right]\right)\left(X_k - \mathbb{E}\left[X\right]\right)^T - \Sigma\right\|_F^2 + 2\left\|\overline{X}_n - \mathbb{E}\left[X\right]\right\|^4.
\end{aligned}$$

Then, with the help of the law of the iterated logarithm (applied twice), one has

$$\left\|\frac{1}{n} C_n - \Sigma\right\|_F^2 = O\left(\frac{\ln \ln n}{n}\right) \quad a.s \qquad \text{and} \qquad \left\|\overline{Q}_n - H\right\|_F^2 = O\left(\frac{\ln \ln n}{n}\right) \quad a.s. \tag{15}$$

Thus,

$$\left\|\hat{\beta}_n - \beta_\lambda\right\|^2 = \left\|\frac{1}{n}\overline{Q}_n^{-1}\sum_{j=1}^n (X_j - \overline{X}_n)(Y_j - \overline{Y}_n) - H^{-1}\mathbb{E}\left[(X - \mathbb{E}\left[X\right])(Y - \mathbb{E}\left[Y\right])\right]\right\|^2$$
$$\leq 2\left\|\left(\overline{Q}_n^{-1} - H^{-1}\right)\frac{1}{n}\sum_{j=1}^n (X_j - \overline{X}_n)(Y_j - \overline{Y}_n)\right\|^2 \Bigg\} =: T_1$$
$$+ 2\left\|H^{-1}\left(\frac{1}{n}\sum_{j=1}^n (X_j - \overline{X}_n)(Y_j - \overline{Y}_n) - \mathbb{E}\left[(X - \mathbb{E}\left[X\right])(Y - \mathbb{E}\left[Y\right])\right]\right)\right\|^2 \Bigg\} =: T_2$$

Moreover, by the law of the iterated logarithm,

$$T_2 \leq 2\left\|H^{-1}\right\|_{op}^2 \left\|\frac{1}{n}\sum_{j=1}^n (X_j - \overline{X}_n)(Y_j - \overline{Y}_n) - \mathbb{E}\left[(X - \mathbb{E}\left[X\right])(Y - \mathbb{E}\left[Y\right])\right]\right\|^2$$
$$\leq 4\left\|H^{-1}\right\|_{op}^2 \left(\left\|\frac{1}{n}\sum_{j=1}^n X_j Y_j - \mathbb{E}[XY]\right\|^2 + \left\|\overline{X}_n\overline{Y}_n - \mathbb{E}[X]\mathbb{E}[Y]\right\|^2\right)$$
$$= O\left(\frac{\ln\ln n}{n}\right) \quad a.s.$$

In addition, thanks to equation (15),

$$T_1 \leq 2\left\|\overline{Q}_n^{-1} - H^{-1}\right\|_F^2 \left\|\frac{1}{n}\sum_{j=1}^n (X_j - \overline{X}_n)(Y_j - \overline{Y}_n)\right\|^2 = O\left(\frac{\ln\ln n}{n}\right) \quad a.s.$$

We now focus on the rate of convergence of $\widehat{\beta}_{n,0}$. We have $\widehat{\beta}_{n,0} = \overline{Y}_n - \widehat{\beta}_n^T\overline{X}_n$, and $\beta_{0,\lambda} = \mathbb{E}\left[Y\right] - \beta_\lambda\mathbb{E}\left[X\right]$. Therefore, by the law of the iterated logarithm

$$\left\|\widehat{\beta}_{n,0} - \beta_{0,\lambda}\right\|^2 = \left\|\overline{Y}_n - \widehat{\beta}_n^T\overline{X}_n - \mathbb{E}\left[Y\right] - \beta_\lambda\mathbb{E}\left[X\right]\right\|^2$$
$$\leq 2\left\|\overline{Y}_n - \mathbb{E}\left[Y\right]\right\|^2 + 2\left\|\widehat{\beta}_n^T\overline{X}_n - \beta_\lambda\mathbb{E}\left[X\right]\right\|^2$$
$$\leq 2\left\|\overline{Y}_n - \mathbb{E}\left[Y\right]\right\|^2 + 4\left\|\widehat{\beta}_n^T - \beta_\lambda\right\|^2\left\|\overline{X}_n\right\|^2 + 4\left\|\beta_\lambda\right\|^2\left\|\mathbb{E}\left[X\right] - \overline{X}_n\right\|^2$$
$$= \mathcal{O}\left(\frac{\ln\ln n}{n}\right) \quad a.s,$$

which concludes the proof.

5.4. Proof of Theorem 4

The aim here is to apply Theorem 3.3 in [5]. Similar to the proof of Theorem 2, we have to prove that the hypotheses (A1b), (A1c), (A2a), (A2b), (A2c), (H1), (H2a) and (H2b) in [5] are satisfied. Compared to the previous proof, the main difficulty here is to verify (H1).

Verification of (A1b). Since $|Y| \leq 1$ and the function π is also bounded by 1, we have

$$\left\|\nabla g_\lambda(\phi, Y, \widetilde{z})\right\| \leq \left(|Y| + |\pi(\widetilde{z}^T\phi)|\right)\left\|\phi\right\| + \lambda\left\|z\right\| \leq 2\|\phi\| + \lambda\left\|z - \beta\right\| + \lambda\|\beta\|$$
$$\leq 2\left\|\phi\right\| + \lambda\left\|\widetilde{z} - \widetilde{\beta}\right\| + \lambda\left\|\beta\right\|,$$

where $\phi = (1, X^T)^T$, $\widetilde{z} = (z_0, z^T)^T$ and $\widetilde{\beta} = (\beta_0, \beta^T)^T$. Then hypothesis (A1b) is satisfied since X admits moment of $2 + 2\eta$.

Verification of (A1c). For all $\tilde{z} \in \mathbb{R}^{p+1}$,

$$\Sigma(\widetilde{z}) := \mathbb{E}\left[\nabla g_\lambda(\phi, Y, \widetilde{z}) \nabla g_\lambda(\phi, Y, \widetilde{z})^T\right] = \mathbb{E}\left[(Y - \pi(\tilde{z}^T\phi))^2 \phi\phi^T\right] + zz^T - 2\lambda\mathbb{E}\left[z^T\phi(Y - \pi(\tilde{z}^T\phi))\right],$$

Since π is continuous and bounded, since Y is bounded, and since ϕ admits a second order moment, the function $\Sigma(.)$ is continuous on $\mathbb{R}^{p+1}$. Hypothesis (A1c) is then satisfied.

Verification of (A2a). We have $\pi(1-\pi) \leq \frac{1}{4}$, and therefore for all $\widetilde{z} \in \mathbb{R}^{p+1}$, we have

$$\left\|\nabla^2 G_\lambda(\widetilde{z})\right\|_{op} \leq \frac{1}{4}\mathbb{E}\left[\|\phi\|^2\right] + \lambda.$$

Therefore, hypothesis (A2a) is satisfied.

Verification of (A2b). We have for all $z' \in \mathbb{R}^{p+1}$,

$$\nabla^2 G_\lambda(z') = \mathbb{E}\left[\pi\left((z')^T\phi\right)\left(1 - \pi\left((z')^T\phi\right)\right)\phi\phi^T\right] + \lambda(I_{p+1} - e_1 e_1^T),$$

thus for all $\widetilde{z} = (z_0, z^T)^T \in \mathbb{R}^{p+1}$ such that $\widetilde{z} \neq 0$, we have

$$\widetilde{z}^T \nabla^2 G_\lambda(z')\widetilde{z} = \widetilde{z}^T \mathbb{E}\left[\pi\left((z')^T\phi\right)\left(1 - \pi\left((z')^T\phi\right)\right)\phi\phi^T\right]\widetilde{z} + \|z\|^2.$$

If $z \neq 0$, it is obvious that $\widetilde{z}^T \nabla^2 G_\lambda(z')\widetilde{z} > 0$. Otherwise we have $z_0 \neq 0$, and $\tilde{z} = (z_0, 0)^T$, so that

$$\widetilde{z}^T \nabla^2 G_\lambda(z')\widetilde{z} = \mathbb{E}\left[\pi\left((z')^T\phi\right)\left(1 - \pi\left((z')^T\phi\right)\right)\right]\|z_0\|^2 > 0.$$

Thus $\nabla^2 G_\lambda(z')$ is positive definite and hypothesis (A2b) is so satisfied.

Verification of (A2c). For all $x \in \mathbb{R}$, we have

$$\pi'(x) = \frac{\exp(x)(\exp(x)+1) - \exp(x)^2}{(1+\exp(x))^2} = \frac{\exp(x)}{(1+\exp(x))^2} = \frac{1}{4cosh(x/2)^2} \leq \frac{1}{4}.$$

Thus, for all x, x', $|\pi(x) - \pi(x')| \leq \frac{1}{4}|x - x'|$. In addition, we have

$$|(\pi(1-\pi))'(x))| = |(\pi'(x)(1 - 2\pi(x))| \leq |(\pi'(x)|.$$

Thanks to the Cauchy-Schwarz inequality, we have for all $\tilde{z} \in \mathbb{R}^{p+1}$,

$$\begin{aligned}\left\|\nabla^2 G_\lambda(\widetilde{z}) - \nabla^2 G_\lambda(\widetilde{\beta})\right\|_{op} &\leq \mathbb{E}\left[|(\pi(1-\pi))(\widetilde{z}^T\phi) - (\pi(1-\pi))(\widetilde{\beta}^T\phi)|\,\|\phi\|^2\right] \\ &\leq \frac{1}{4}\mathbb{E}\left[\|\phi\|^3\right]\left\|\widetilde{z} - \widetilde{\beta}\right\|.\end{aligned}$$

The Hessian is then Lipschitz and hypothesis (A2c) is so satisfied.

Verification of (H1). Let us denote $\overline{Q}_n = \frac{1}{n}Q_n$ and remark that

$$\lambda_{\min}\left(\overline{Q}_n\right) \geq \min\left\{\frac{\alpha}{n} + \frac{\lambda(p+1)}{n}\left\lfloor\frac{n}{p+1}\right\rfloor, \frac{\alpha}{n} + \frac{\lambda(p+1)^{1-2\gamma}}{n}\sum_{k=1}^{\lfloor n/(p+1)\rfloor} k^{-2\gamma}c_\gamma^2\right\}.$$

Since we are interested by the asymptotic behavior of the smallest eigenvalue, let us suppose from now that $n > p+1$. First, remark that

$$\frac{\lambda(p+1)}{n}\left\lfloor\frac{n}{p+1}\right\rfloor \geq \lambda\frac{n-p}{n} \xrightarrow[n\to+\infty]{} \lambda.$$

In addition,

$$\sum_{k=1}^{\lfloor n/(p+1)\rfloor} k^{-2\gamma}c_\gamma^2 \geq c_\gamma^2 \int_1^{\lfloor n/(p+1)\rfloor+1} t^{-2\gamma}dt = \frac{c_\gamma^2}{1-2\gamma}\left(\left(\lfloor n/(p+1)\rfloor+1\right)^{1-2\gamma}-1\right).$$

Thus,

$$\lambda_{max}(\overline{Q}_n^{-1}) = O\left(n^{2\gamma}\right).$$

As $\alpha_k \leq \frac{1}{4}$, we have

$$\lambda_{max}(\overline{Q}_n) \leq \lambda_{max}\left(\frac{\alpha I_{p+1}}{n} + \frac{1}{n}\sum_{k=1}^{n}\frac{1}{4}\phi_k\phi_k^T + \frac{(p+1)\lambda}{n}\sum_{k=1}^{n} Z_k Z_k^T\right).$$

Note that since $\gamma < 1/2$,

$$\frac{\lambda(p+1)}{n}\sum_{k=1}^{\lfloor n/(p+1)\rfloor} k^{-2\gamma}c_\gamma^2 = O\left(n^{-2\gamma}\right) \tag{16}$$

and one can so check that

$$\left\|\frac{(p+1)\lambda}{n}\sum_{k=1}^{n} Z_k Z_k^T - \lambda A_{p+1}\right\|_F^2 = O\left(n^{-2\gamma}\right). \tag{17}$$

By the strong law of large numbers, we have

$$\frac{1}{n}\left(\alpha I_{p+1} + \sum_{k=1}^{n}\frac{1}{4}\phi_k\phi_k^T\right) \xrightarrow[n\to+\infty]{a.s.} \frac{1}{4}\mathbb{E}\left[\phi\phi^T\right].$$

Thus hypothesis (H1) is satisfied and according to Theorem 3.1 in [5], $\widehat{\beta}_n$ converges almost surely to $\widetilde{\beta}$.

Verification of (H2a). Let us consider $B_n := \sum_{j=1}^{n}\alpha(\widehat{\beta}_{j-1}^T\phi_j)\phi_j\phi_j^T$, which can be written as

$$B_n = \sum_{k=1}^{n}\nabla^2 G(\hat{\beta}_{k-1}) + \sum_{k=1}^{n}\psi_k,$$

where $G(\widetilde{z}) = \mathbb{E}\left[\log(1+\exp(\widetilde{z}^T\phi)) - \widetilde{z}^T\phi Y\right]$ and $\psi_k = a_k - \nabla^2 G(\hat{\theta}_k)$ with $a_k := \alpha(\widehat{\beta}_{k-1}^T\phi_k)\phi_k\phi_k^T$. Note that (ψ_k) is a sequence of martingale differences with respect to the filtration $(\mathcal{F}_n)$ with $\mathcal{F}_n = \sigma\left((\phi_1, Y_1), ..., (\phi_n, Y_n)\right)$. Besides,

$$\mathbb{E}\left[\|\psi_k\|_F^2 \,|\mathcal{F}_{k-1}\right] \leq \mathbb{E}\left[\|a_k\|_F^2 \,|\mathcal{F}_{k-1}\right] \leq \frac{1}{16}\mathbb{E}\left[\|\phi\|^4\right],$$

where $\|.\|_F$ is the Frobenius norm. Thanks to the law of large numbers for martingales, we have that for all $\delta > 0$,

$$\left\|\frac{1}{n}\sum_{k=1}^{n}\Psi_k\right\|_F^2 = o\left(\frac{\ln n^{1+\delta}}{n}\right) \quad a.s. \tag{18}$$

Remark that the function $h \mapsto \nabla^2 G(h)$ is continuous. Moreover, since $\hat{\beta}_n$ converges almost surely to $\tilde{\beta}$, we have by continuity

$$\left\|\frac{1}{n}\sum_{k=1}^{n}\nabla^2 G(\hat{\beta}_{k-1}) - \nabla^2 G(\tilde{\beta})\right\|_F \xrightarrow[n\to+\infty]{a.s.} 0.$$

Then, coupled with equality (17), it comes that $\overline{Q}_n$ converges almost surely to $\nabla_\lambda^2 G\left(\tilde{\beta}\right)$ and according to Theorem 3.2 in [5],

$$\left\|\hat{\beta}_n - \beta_\lambda\right\|^2 = \mathcal{O}\left(\frac{\ln n}{n}\right) \quad a.s.$$

Verification of (H2b). Since the function $h \mapsto \nabla^2 G(h)$ is $\frac{1}{4}\mathbb{E}[\|\phi\|^3]$-Lipschitz, there exists a positive random variable B such that

$$\left\| \frac{1}{n}\sum_{k=1}^{n} \nabla^2 G(\hat{\beta}_{k-1}) - \nabla^2 G(\beta_\lambda) \right\|_F \leq \frac{B}{4n}\mathbb{E}\left[\|\phi\|^3\right] \sum_{k=1}^{n} \frac{\sqrt{\log k}}{\sqrt{k}} = \mathcal{O}\left(\sqrt{\frac{\ln n}{n}}\right) \quad a.s.$$

Then, coupled with (18), it comes that

$$\left\| \frac{1}{n} B_n - \nabla^2 G\left(\tilde{\beta}\right) \right\|_F^2 = o\left(\frac{(\ln n)^{1+\delta}}{n}\right) \quad a.s,$$

and coupled with equality (17), we obtain

$$\left\| \overline{Q}_n - \nabla^2_\lambda G\left(\tilde{\beta}\right) \right\|_F^2 = O\left(n^{-2\gamma}\right).$$

All the hypotheses are satisfied, and according to [5], we obtain the conclusion.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] An, S., Liu, W., and Venkatesh, S. (2007). Face recognition using kernel ridge regression. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7. IEEE.

[2] Baye, M. R. and Parker, D. F. (1984). Combining ridge and principal component regression: a money demand illustration. *Communications in Statistics-Theory and Methods*, 13(2):197–205.

[3] Bercu, B., Godichon, A., and Portier, B. (2020). An efficient stochastic newton algorithm for parameter estimation in logistic regressions. *SIAM Journal on Control and Optimization*, 58(1):348–367.

[4] Blackard, J. A. and Dean, D. J. (1999). Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables. *Computers and electronics in agriculture*, 24(3):131–151.

[5] Boyer, C. and Godichon-Baggioni, A. (2020). On the asymptotic rate of convergence of stochastic newton algorithms and their weighted averaged versions. *arXiv preprint arXiv:2011.09706*.

[6] Cénac, P., Godichon-Baggioni, A., and Portier, B. (2020). An efficient averaged stochastic gauss-newton algorithm for estimating parameters of non linear regressions models. *arXiv preprint arXiv:2006.12920*.

[7] Duflo, M. (2013). *Random iterative models*, volume 34. Springer Science & Business Media.

[8] Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1.

[9] Gadat, S., Bercu, B., Bigot, J., and Siviero, E. (2021). A stochastic gauss-newton algorithm for regularized semi-discrete optimal transport.

[10] Golub, G. H., Heath, M., and Wahba, G. (1979). Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223.

[11] Górski, Ł. and Jakubowska, M. (2019). Ridge regression with self-paced learning algorithm in interpretation of voltammetric signals. *Chemometrics and Intelligent Laboratory Systems*, 191:73–81.

[12] Hartman, P. and Wintner, A. (1941). On the law of the iterated logarithm. *American Journal of Mathematics*, 63(1):169–176.

[13] Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

[14] Inman, J. R. (1975). Resistivity inversion with ridge regression. *Geophysics*, 40(5):798–817.

[15] Ismail, M. and Principe, J. (1996). Equivalence between rls algorithms and the ridge regression technique. In *Conference Record of The Thirtieth Asilomar Conference on Signals, Systems and Computers*, pages 1083–1087. IEEE.

[16] Kawala, F., Douzal-Chouakria, A., Gaussier, E., and Dimert, E. (2013). Prédictions d'activité dans les réseaux sociaux en ligne. In *4ième conférence sur les modèles et l'analyse des réseaux: Approches mathématiques et informatiques*, page 16.

[17] Le Cessie, S. and Van Houwelingen, J. C. (1992). Ridge estimators in logistic regression. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 41(1):191–201.

[18] Leluc, R. and Portier, F. (2020). Asymptotic optimality of conditioned stochastic gradient descent. *arXiv preprint arXiv:2006.02745*.

[19] Marquardt, D. W. and Snee, R. D. (1975). Ridge regression in practice. *The American Statistician*, 29(1):3–20.

[20] Neumann, K. and Steil, J. J. (2013). Optimizing extreme learning machines via ridge regression and batch intrinsic plasticity. *Neurocomputing*, 102:23–30.

[21] Newell, G. and Lee, B. (1981). Ridge regression: an alternative to multiple linear regression for highly correlated data. *Journal of Food Science*, 46(3):968–969.

[22] Ngia, L. S. and Sjoberg, J. (2000). Efficient training of neural nets for nonlinear adaptive filtering using a recursive levenberg-marquardt algorithm. *IEEE Transactions on Signal Processing*, 48(7):1915–1927.

[23] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

[24] R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

[25] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407.

[26] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1985). Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science.

[27] Schaefer, R., Roi, L., and Wolfe, R. (1984). A ridge logistic estimator. *Communications in Statistics-Theory and Methods*, 13(1):99–113.

[28] Schmidt, M., Le Roux, N., and Bach, F. (2017). Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1):83–112.

[29] Toulis, P., Tran, D., and Airoldi, E. (2016). Towards stability and optimality in stochastic gradient descent. In *Artificial Intelligence and Statistics*, pages 1290–1298. PMLR.

[30] Vovk, V. (2013). Kernel ridge regression. In *Empirical inference*, pages 105–116. Springer.

[31] Zhang, Z., Zhou, S., Li, D., and Yang, T. (2020). Gradient preconditioned mini-batch sgd for ridge regression. *Neurocomputing*, 413:284–293.