2018 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 17–20, 2018, AALBORG, DENMARK

SPARSE BAYESIAN BINARY LOGISTIC REGRESSION USING THE SPLIT-AND-AUGMENTED GIBBS SAMPLER

Maxime Vono, Nicolas Dobigeon

University of Toulouse, INP-ENSEEIHT
IRIT, CNRS, Toulouse, France

Pierre Chainais

University of Lille, CNRS, Centrale Lille
UMR 9189 - CRIStAL, Lille, France

ABSTRACT

Logistic regression has been extensively used to perform classification in machine learning and signal/image processing. Bayesian formulations of this model with sparsity-inducing priors are particularly relevant when one is interested in drawing credibility intervals with few active coefficients. Along these lines, the derivation of efficient simulation-based methods is still an active research area because of the analytically challenging form of the binomial likelihood. This paper tackles the sparse Bayesian binary logistic regression problem by relying on the recent split-and-augmented Gibbs sampler (SPA). Contrary to usual data augmentation strategies, this Markov chain Monte Carlo (MCMC) algorithm scales in high dimension and divides the initial sampling problem into simpler ones. These sampling steps are then addressed with efficient state-of-the-art methods, namely proximal MCMC algorithms that can benefit from the recent closed-form expression of the proximal operator of the logistic cost function. SPA appears to be faster than efficient proximal MCMC algorithms and presents a reasonable computational cost compared to optimization-based methods with the advantage of producing credibility intervals. Experiments on handwritten digits classification problems illustrate the performances of the proposed approach.

Index Terms— Bayesian inference, data augmentation, logistic regression, Markov chain Monte Carlo, sparsity, variable splitting.

1. INTRODUCTION

Logistic regression, developed at least 60 years ago [1, 2], has been extensively studied [3–5] and used for classification problems over the past decades. For instance, this model has been succesfully resorted in machine learning [6], medicine [7] and social sciences [8]. In those areas, and especially in medicine, interpretation of logistic regression coefficients and credibility intervals are often required to take an important decision. Thus, under these considerations, one cannot rely on a deterministic formulation of the logistic regression and can use a probabilistic derivation of this model. Additionally, the number of attributes D (often called features) can be large with respect to (w.r.t.) the number of observations N and giving an interpretation to each attribute could be challenging. To cope with this problem, it is widely admitted to use a sparsity-inducing regularization [9] in order to take into account the most relevant attributes only.

Much research has been conducted to tackle the Bayesian logistic regression problem with simulation-based methods. The main issue is the analytically challenging form of the binomial likelihood (no common conjugate prior distributions can be proposed) which has been overcome in different ways. Thus, since the work of [10] on the probit model, analogue data augmentation techniques have been applied to re-write the logistic distribution. For instance, [11] represented this distribution as a normal-scale mixture and considered a Gaussian prior on the weights vector $\boldsymbol{\beta}$. Few years laters, this model was challenged and/or improved by surrogate data-augmentation schemes proposed in [12–14]. The majority of the above approaches considered a Gaussian prior distribution on $\boldsymbol{\beta}$ leading to normal posterior distributions. Although this type of prior distributions is convenient (well-understood properties, conjugancy, etc.), it restricts the use of other informative priors which can be non-smooth and/or even non-convex. Additionally, these approaches, by involving for some of them multiple layers of latent variables, could be complicated to implement and their computational cost in high-dimensional classification problems could be problematic. Recently, these issues have been unlocked by so-called proximal Markov chain Monte Carlo (MCMC) algorithms [15] which take advantage of convex analysis (proximal operators and Moreau-Yoshida envelopes) to build efficient sampling schemes from possibly non-smooth log-concave distributions scaling in large dimension. These sampling methods are special instances of Metropolis-Hastings algorithms based on the Langevin diffusion process which were improved in [16]. More recently, the connection between simulation-based algorithms and optimization has been strengthened by the so-called *split-and-augmented Gibbs sampler* (SPA) [17]. This algorithm stands for a general tool to conduct Bayesian inference that uses a "divide-and-conquer" strategy. Relying on variable splitting akin the alternating direction method of multipliers (ADMM), it divides the initial sampling tasks into simpler ones. Then, efficient existing MCMC algorithms can sample from each conditional probability distribution (e.g. data-augmentation or proximal MCMC schemes).

This paper addresses the sparse Bayesian binary logistic regression problem using this very recent approach and compares the performances of the latter with the ADMM and state-of-the-art proximal MCMC algorithms. To this purpose, Section 2 presents the sparse Bayesian binary logistic regression problem. Section 3 introduces SPA through its two main ingredients namely variable splitting and data augmentation. Its close relation with the ADMM is also discussed. Section 4 derives SPA for the sparse Bayesian logistic regression and details the implementation of each sampling step. Section 5 illustrates and compares the performances of SPA with ADMM and the proximal Moreau-Yoshida unadjusted Langevin algorithm (P-MYULA) proposed in [16]. Finally, Section 6 draws concluding remarks and possible extensions of this work.

2. PROBLEM FORMULATION

This section reviews the derivation of the sparse Bayesian logistic regression for binary classification problems. A possible extension

978-1-5386-5477-4/18/$31.00

of this binary classifier to multiclass classification problems is to consider a one-versus-all approach which is simpler and may be as relevant as the multi-class approach [18].

2.1. Sparse Bayesian binary logistic regression

The so-called sparse Bayesian binary logistic regression is recalled hereafter. Suppose that one observes binary responses $\mathbf{y} \in \mathbb{R}^N$ which are conditionally independent Bernoulli random variables with probability of success $h\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)$. The function h is the standard logistic function defined as for all $t \in \mathbb{R}$,

$$h(t) = \frac{\exp(t)}{1+\exp(t)}, \tag{1}$$

$\boldsymbol{\beta} \in \mathbb{R}^D$ represents the vector of regression coefficients to be estimated and for all $i \in \{1,\ldots,N\}$, $\mathbf{x}_i \in \mathbb{R}^D$ stands for the features vector associated to the i-th observation. Under this model, the likelihood has the form

$$p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{x}_{1:N}) = \prod_{i=1}^{N} h\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)^{y_i}\left(1-h\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)\right)^{1-y_i}. \tag{2}$$

Adopting a sparsity-inducing regularization in a Bayesian framework boils down to consider an appropriate informative prior distribution on the parameters vector $\boldsymbol{\beta}$. A common choice for this prior is the Laplacian (or double exponential) distribution used in [19, 20] for the Lasso problem and defined as

$$p(\boldsymbol{\beta}) \propto \exp\left[-\tau\left\|\boldsymbol{\beta}\right\|_1\right] \tag{3}$$

where $\tau > 0$ is an hyperparameter controlling the sparsity degree. This choice is strengthened by the well-justified sparsity-inducing properties of the Laplacian prior [21, 22] which were resorted in a lot of applications [23, 24]. The application of Bayes' rule leads to the posterior distribution

$$\begin{aligned} p(\boldsymbol{\beta}|\mathbf{y}) = \exp\Bigg[&-\tau\left\|\boldsymbol{\beta}\right\|_1 + \sum_{i=1}^{N} y_i \log\left\{h\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)\right\} \\ &+(1-y_i)\log\left\{1-h\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)\right\}\Bigg]. \end{aligned} \tag{4}$$

2.2. Related work

The posterior distribution (4) being log-concave with a gradient-Lipschitz smooth term $-\log p(\mathbf{y}|\boldsymbol{\beta})$ along with a closed-form expression of the ℓ_1-norm proximal operator (soft-thresholding operator), proximal MCMC algorithms and especially P-MYULA can be resorted to sample from it. However, considering directly the posterior in (4) within the same sampling step could slow down the convergence of the Markov chain associated to $\boldsymbol{\beta}$ towards its stationary distribution. To alleviate this issue, sampling from (4) will be conducted with SPA whose construction and properties are presented in Section 3.

3. SPLIT-AND-AUGMENTED GIBBS SAMPLER

In this section, SPA is resorted for the approximate sampling from the general distribution

$$\pi(\boldsymbol{\beta}) \propto \exp\left[-g(\boldsymbol{\beta}) - \sum_{i=1}^{N} f(\mathbf{k}_i^T\boldsymbol{\beta})\right] \tag{5}$$

where $f : \mathbb{R} \to \bar{\mathbb{R}}$, $g : \mathbb{R}^D \to \bar{\mathbb{R}}$ and $\mathbf{k}_i \in \mathbb{R}^D$ is the i-th line of the matrix $\mathbf{K} \in \mathbb{R}^{M\times D}$. Its key ingredients are presented, namely variable splitting and data augmentation. Additionally, the parallel between this MCMC algorithm and the ADMM is recalled.

3.1. Split-augmented distribution

In place of sampling from the target distribution (5) such as proximal MCMC algorithms [15, 16], SPA samples from another distribution $\pi_{\rho,\alpha}$ called split-augmented distribution.

Variable splitting – The construction of this distribution first relies on a variable splitting step which consists in introducing two splitting variable $\mathbf{z}_1 \in \mathbb{R}^M$ and $\mathbf{z}_2 \in \mathbb{R}^D$ leading to the so-called *split distribution* defined by

$$\begin{aligned} \pi_\rho \triangleq p(\boldsymbol{\beta}, \mathbf{z}_1, \mathbf{z}_2) \propto \exp&\left[-g(\mathbf{z}_2) - \sum_{i=1}^{N} f(z_{1,i})\right] \\ &\times \exp\left[-\frac{1}{2\rho^2}\left\|\mathbf{K}\boldsymbol{\beta}-\mathbf{z}_1\right\|_2^2 - \frac{1}{2\rho^2}\left\|\boldsymbol{\beta}-\mathbf{z}_2\right\|_2^2\right], \end{aligned} \tag{6}$$

where $z_{1,i}$ is the i-th component of $\mathbf{z}_1$. This joint distribution preludes the use of a Gibbs sampler to sample from each conditional distribution associated to $\boldsymbol{\beta}$, $\mathbf{z}_1$ and $\mathbf{z}_2$, respectively. Sampling from these conditional distributions instead of (5) might be easier because f and g will be dissociated. Although the split distribution is different from the target distribution (5), the marginal distribution of $\boldsymbol{\beta}$ under π_ρ coincides with (5) in a limiting case as pointed out by Theorem 1.

Theorem 1 *[17, Theorem 1] Let* $p_\rho(\boldsymbol{\beta}) = \int \pi_\rho(\boldsymbol{\beta}, \mathbf{z}_1, \mathbf{z}_2)\mathrm{d}\mathbf{z}_1\mathrm{d}\mathbf{z}_2$. *Then, it follows*

$$\left\|\pi - p_\rho\right\|_{\mathrm{TV}} \xrightarrow[\rho^2\to 0]{} 0. \tag{7}$$

Data augmentation – In order to ensure that this variable splitting step will work, one might set ρ^2 to a small value, see Theorem 1. However, such a small value can lead to higher correlation between MCMC draws. One surrogate to improve these mixing properties is to consider a data augmentation scheme ensuring less interactions between MCMC samples [25]. Under the first splitting step, the introduction of auxiliary variables $\mathbf{u}_1 \in \mathbb{R}^M$ and $\mathbf{u}_2 \in \mathbb{R}^D$ leads to the so-called *split-augmented distribution*

$$\pi_{\rho,\alpha} \triangleq p(\boldsymbol{\beta}, \mathbf{z}_1, \mathbf{z}_2, \mathbf{u}_1, \mathbf{u}_2) \tag{8}$$

$$\begin{aligned} &\propto \exp\left[-g(\mathbf{z}_2) - \sum_{i=1}^{N} f(z_{1,i})\right] \\ &\times \exp\left[-\frac{1}{2\rho^2}\left\|\mathbf{K}\boldsymbol{\beta}-(\mathbf{z}_1-\mathbf{u}_1)\right\|_2^2 - \frac{1}{2\rho^2}\left\|\boldsymbol{\beta}-(\mathbf{z}_2-\mathbf{u}_2)\right\|_2^2\right] \\ &\times \exp\left[-\frac{1}{2\alpha^2}\left\|\mathbf{u}_1\right\|_2^2 - \frac{1}{2\alpha^2}\left\|\mathbf{u}_2\right\|_2^2\right]. \end{aligned} \tag{9}$$

Considering this joint distribution is relevant since the introduction of auxiliary variables $\mathbf{u}_1$ and $\mathbf{u}_2$ does not alter the split distribution π_ρ in (6). The proof derives from straightforward marginalization of $\mathbf{u}_1$ and $\mathbf{u}_2$ in (8). Note that the spirit of this data augmentation step differs from data augmentation schemes [11–14]. Thus, the latter introduce latent variables in order to simplify the sampling from (4). On the other hand, the data augmentation leading to the split-augmented distribution (8) is built from an approximation of (4), namely the split distribution (6), in order to improve the mixing properties of the Markov chain built with SPA.

Algorithm 1: SPA

Input: Functions f, g, operator $\mathbf{K}$, hyperparam. ρ^2, α^2, total nb of iterations T_{MC}, nb of burn-in iterations T_{bi}, initialization $\mathbf{z}_1^{(0)}$,$\mathbf{z}_2^{(0)}$,$\mathbf{u}_1^{(0)}$ & $\mathbf{u}_2^{(0)}$

for $t \leftarrow 1$ **to** T_{MC} **do**

% Drawing the variable of interest

Sample $\boldsymbol{\beta}^{(t)}$ according to $p\left(\boldsymbol{\beta}|\mathbf{z}_1^{(t-1)}, \mathbf{z}_2^{(t-1)}, \mathbf{u}_1^{(t-1)}, \mathbf{u}_2^{(t-1)}\right)$ (10) ;

% Drawing the splitting variables

Sample each component of $\mathbf{z}_1^{(t)}$ according to $p\left(z_{1,i}|\boldsymbol{\beta}^{(t)}, u_{1,i}^{(t-1)}\right)$ (11) ;

Sample $\mathbf{z}_2^{(t)}$ according to $p\left(\mathbf{z}_2|\boldsymbol{\beta}^{(t)}, \mathbf{u}_2^{(t-1)}\right)$ (12) ;

% Drawing the auxiliary variables

Sample $\mathbf{u}_1^{(t)}$ according to $p\left(\mathbf{u}_1|\boldsymbol{\beta}^{(t)}, \mathbf{z}_1^{(t)}\right)$ (13) ;

Sample $\mathbf{u}_2^{(t)}$ according to $p\left(\mathbf{u}_2|\boldsymbol{\beta}^{(t)}, \mathbf{z}_2^{(t)}\right)$ (14) ;

end

Output: Collection of samples $\left\{\boldsymbol{\beta}^{(t)}, \mathbf{z}_1^{(t)}, \mathbf{z}_2^{(t)}, \mathbf{u}_1^{(t)}, \mathbf{u}_2^{(t)}\right\}_{t=T_{\mathrm{bi}}+1}^{T_{\mathrm{MC}}}$ asymptotically distributed according to (8).

3.2. SPA algorithm

Sampling from such a distribution leads naturally to a special instance of Gibbs samplers, SPA, whose formulation is closely related to the ADMM main steps.

Gibbs sampler – SPA (see Algo. 1) considers the sampling from each conditional distribution associated to (8), that is

$$p(\boldsymbol{\beta}|\mathbf{z}_1, \mathbf{z}_2, \mathbf{u}_1, \mathbf{u}_2) \propto \exp\left[-\frac{1}{2\rho^2}\left\|\mathbf{K}\boldsymbol{\beta} - (\mathbf{z}_1 - \mathbf{u}_1)\right\|_2^2 - \frac{1}{2\rho^2}\left\|\boldsymbol{\beta} - (\mathbf{z}_2 - \mathbf{u}_2)\right\|_2^2\right] \tag{10}$$

$$p(z_{1,i}|\boldsymbol{\beta}, u_{1,i}) \propto \exp\left[-f(z_{1,i}) - \frac{1}{2\rho^2}\left(\mathbf{k}_i^T\boldsymbol{\beta} - (z_{1,i} - u_{1,i})\right)_2^2\right] \tag{11}$$

$$p(\mathbf{z_2}|\boldsymbol{\beta}, \mathbf{u}_2) \propto \exp\left[-g(\mathbf{z}_2) - \frac{1}{2\rho^2}\left\|\boldsymbol{\beta} - (\mathbf{z}_2 - \mathbf{u}_2)\right\|_2^2\right] \tag{12}$$

$$p(\mathbf{u_1}|\boldsymbol{\beta}, \mathbf{z}_1) \propto \exp\left[-\frac{1}{2\alpha^2}\left\|\mathbf{u}_1\right\|_2^2 - \frac{1}{2\rho^2}\left\|\mathbf{K}\boldsymbol{\beta} - (\mathbf{z}_1 - \mathbf{u}_1)\right\|_2^2\right] \tag{13}$$

$$p(\mathbf{u_2}|\boldsymbol{\beta}, \mathbf{z}_2) \propto \exp\left[-\frac{1}{2\alpha^2}\left\|\mathbf{u}_2\right\|_2^2 - \frac{1}{2\rho^2}\left\|\boldsymbol{\beta} - (\mathbf{z}_2 - \mathbf{u}_2)\right\|_2^2\right]. \tag{14}$$

As stated in [17], SPA can be viewed as a "divide-and-conquer" approach where the initial sampling difficulty is divided in different easier sampling steps. Thus, the conditional distributions associated to $\boldsymbol{\beta}$, $\mathbf{u}_1$ and $\mathbf{u}_2$ are Gaussian with diagonal covariance matrices for the last two distributions. Thereby, sampling from these distributions can be performed efficiently even in high dimension. On the other hand, sampling from the distributions (11) and (12) will depend on the form of functions f and g. For instance, the derivation of SPA for the sparse Bayesian binary logistic regression leads to log-concave conditional distributions where efficient proximal MCMC algorithms can be proposed as described in Section 4.2.

Parallel with ADMM – An interesting property of SPA is its close relation with the ADMM (see Algo. 2). Thus, computing the maximum a posteriori (MAP) estimates of each conditional distribution involved in SPA boils down to the ADMM main steps. Additionally, SPA and ADMM share a general framework that yields simpler sub-problems to be considered and can embed efficient algorithms at each step, see Section 4.2.

Algorithm 2: ADMM (scaled version)

Input: Functions f, g, operator $\mathbf{K}$, penalty parameter ρ^{-2}, initialization $t \leftarrow 0$ and $\mathbf{z}_1^{(0)}$,$\mathbf{z}_2^{(0)}$,$\mathbf{u}_1^{(0)}$,$\mathbf{u}_2^{(0)}$

while *stopping criterion not satisfied* **do**

% Minimization w.r.t. $\boldsymbol{\beta}$

$\boldsymbol{\beta}^{(t)} \in \arg\min_{\boldsymbol{\beta}} -\log p\left(\boldsymbol{\beta}|\mathbf{z}_1^{(t-1)}, \mathbf{z}_2^{(t-1)}, \mathbf{u}_1^{(t-1)}, \mathbf{u}_2^{(t-1)}\right)$;

% Minimization w.r.t. $\mathbf{z}_2$ and each component of $\mathbf{z}_1$

$z_{1,i}^{(t)} \in \arg\min_{z_{1,i}} -\log p\left(z_{1,i}|\boldsymbol{\beta}^{(t)}, u_{1,i}^{(t-1)}\right)$;

$\mathbf{z}_2^{(t)} \in \arg\min_{\mathbf{z}_2} -\log p\left(\mathbf{z}_2|\boldsymbol{\beta}^{(t)}, \mathbf{u}_2^{(t-1)}\right)$;

% Dual ascent

$\mathbf{u}_1^{(t)} = \mathbf{u}_1^{(t-1)} + \mathbf{K}\boldsymbol{\beta}^{(t)} - \mathbf{z}_1^{(t)}$;

$\mathbf{u}_2^{(t)} = \mathbf{u}_2^{(t-1)} + \boldsymbol{\beta}^{(t)} - \mathbf{z}_2^{(t)}$;

% Updating iterations counter

$t \leftarrow t + 1$;

end

Output: Approximate solution of the optimization problem $\hat{\boldsymbol{\beta}}$.

4. SPARSE BAYESIAN BINARY LOGISTIC REGRESSION WITH SPA

This section derives SPA and discusses implementation details for the considered sparse Bayesian binary logistic regression problem.

4.1. Applying SPA

In the particular case where the responses $\mathbf{y}$ are binary and take values in $\{-1, 1\}^N$, the posterior distribution defined in (4) becomes

$$p(\boldsymbol{\beta}|\mathbf{y}) \propto \exp\left[-\tau\left\|\boldsymbol{\beta}\right\|_1 - \sum_{i=1}^{N}\log\left\{1 + \exp\left(-y_i\mathbf{x}_i^T\boldsymbol{\beta}\right)\right\}\right]. \tag{15}$$

This posterior distribution involves two terms that can be identified with the general target distribution defined in (5), namely

$$\forall\boldsymbol{\beta} \in \mathbb{R}^D, \quad f(\mathbf{k}_i^T\boldsymbol{\beta}) = \log\left\{1 + \exp\left(-\mathbf{k}_i^T\boldsymbol{\beta}\right)\right\} \tag{16}$$

$$\forall\boldsymbol{\beta} \in \mathbb{R}^D, \quad g(\boldsymbol{\beta}) = \tau\left\|\boldsymbol{\beta}\right\|_1, \tag{17}$$

where $\mathbf{k}_i = y_i\mathbf{x}_i$. Thereby, the matrix $\mathbf{K} \in \mathbb{R}^{N\times D}$ is defined by

$$\mathbf{K} = \mathbf{D_y}\mathbf{X} \tag{18}$$

where $\mathbf{D_y}$ is the $N \times N$ diagonal matrix with y_i as i-th diagonal element and $\mathbf{X}$ is the observation matrix associated to the features.

4.2. Implementation details

As noted in Section 3.2, only the conditional distributions associated to $\mathbf{z}_2$ and to each component of $\mathbf{z}_1$ are not standard. More precisely, the conditional distribution associated to each component $z_{1,i}$, $i \in \{1,\ldots,N\}$ has the form

$$p(z_{1,i}|\boldsymbol{\beta}, u_{1,i}) \propto \exp\left[-\log\left\{1+\exp\left(-z_{1,i}\right)\right\} - \frac{1}{2\rho^2}\left(\mathbf{k}_i^T\boldsymbol{\beta} - (z_{1,i}-u_{1,i})\right)^2\right]. \quad (19)$$

Instead of using further data augmentation strategies [11] to sample from (11), SPA leads naturally to N independent sampling steps which can be processed in parallel. Interestingly, the log-concavity of these N distributions enables to consider P-MYULA [16]. This algorithm relies on the proximal operator of a possible non-smooth function. Here, the logistic loss function $t \to \log\left\{1+\exp\left(-t\right)\right\}$ is differentiable on $\mathbb{R}$ and its closed-form proximity operator has been proposed recently in [26]. The formulation of the latter involves the generalized W-Lambert function introduced in [27, 28] as detailed hereafter.

Theorem 2 *[26, Proposition 2] Let $\lambda \in]0,+\infty[$ and $f_{\log} : t \to \log\left\{1+\exp\left(-t\right)\right\}$. Then*

$$\forall t \in \mathbb{R}, \quad \operatorname{prox}^{\lambda}_{f_{\log}}(t) = t + W_{\exp(-t)}(\lambda\exp(-t)), \quad (20)$$

where $\operatorname{prox}^{\lambda}_{f_{\log}}(t) = \arg\min_s \left\{\frac{1}{2\lambda}(t-s)^2 + f_{\log}(s)\right\}$.

The generalized W-Lambert function can be efficiently evaluated as described in [28] and its C++ implementation is available online.

The conditional distribution associated to $\mathbf{z}_2$ in (12) becomes

$$p(\mathbf{z_2}|\boldsymbol{\beta}, \mathbf{u}_2) \propto \exp\left[-\tau\left\|\mathbf{z}_2\right\|_1 - \frac{1}{2\rho^2}\left\|\boldsymbol{\beta} - (\mathbf{z}_2 - \mathbf{u}_2)\right\|_2^2\right]. \quad (21)$$

Since the proximal operator associated to the ℓ_1-norm is the well-known soft-thresholding operator, P-MYULA can also be applied to sample from this conditional distribution.

5. EXPERIMENTS

This section reports and compares results obtained with SPA, P-MYULA and ADMM on binary classification problems. Additionally, in order to apply the proposed approach to a challenging multiclass classification problem, the one-versus-all approach [18] is proposed as it involves the training of binary classifiers. All the results were obtained using MATLAB, on a computer equipped with an Intel Xeon 3.70 GHz processor with 16.0 GB of RAM.

5.1. Experimental design

In order to assess the performances of the proposed approach, the sparse Bayesian binary logistic regression problem defined in Section 4 applied to handwritten digits classification is considered. Two well-known and often-studied datasets are resorted namely MNIST and USPS whose characteristics are recalled in Table 1. Such datasets have been chosen as i) they involve a set of roughly similar binary classification problems where performances (e.g. computational cost and number of iterations) can be averaged and ii) the number of features D is important ($> 10^2$). Note that this value takes into account the intercept which was not penalized as prescribed in [6, Section 4.4.4]. Six binary classification problems (three for each dataset) are considered. In addition, a possible extension (one-versus-all approach) of the proposed approach has been used to tackle the multiclass classification problem on the MNIST dataset. Thus, such an approach has been performed by training ten independent binary classifiers and then by choosing the classifier which outputs the largest value. All results associated to binary classification problems were obtained and averaged over three 5-fold cross-validation procedures. Due to more demanding computational costs, results associated to the MNIST one-versus-all experiment were obtained and averaged over one 5-fold cross-validation procedure. The simulation-based algorithms SPA and P-MYULA are considered and compared to the deterministic counterpart of SPA namely ADMM, see Section 3.2. The number of burn-in iterations have been set to $T_{\text{bi}} = 200$ for SPA and to $T_{\text{bi}} = 95200$ (due to slower mixing properties) for P-MYULA. For each MCMC method, 4800 samples obtained after the burn-in period were used. For the different algorithms, the regularization parameter has been fixed to $\tau = 1$ (tuned by cross-validation with ADMM).

The hyperparameters associated to SPA have been set to $(\rho, \alpha) = (3, 1)$. The choice $\rho = 3$ is a trade-off between the short computational time and the good classification scores. Sampling from (19) and (21) has been conducted by P-MYULA with paramters $(\lambda, \gamma) = (\rho^2, \rho^2/4)$ as recommended in [16]. ADMM (similarly to SPA) was implemented using the proximal operator defined in Theorem 2 with a fixed penalty parameter $\mu = \tau/50$. ADMM was run until the stopping criterion

$$\frac{\left\|\boldsymbol{\beta}^{(t)} - \boldsymbol{\beta}^{(t-1)}\right\|_2}{\left\|\boldsymbol{\beta}^{(t-1)}\right\|_2} \leq \delta \quad (22)$$

was satisfied ($\delta = 0.01$). Improved versions of the ADMM have been proposed recently but this version has been found sufficient to get an idea of the gap between state-of-the-art simulation and vanilla optimization-based methods.

Table 1. Datasets considered in the experiments.

	# observations N	**# features** D	**# classes**
MNIST	60000	785	10
USPS	9298	257	10

5.2. Performance results

Table 2 shows the average classification score obtained over the three (resp. one) 5-fold cross-validation procedure for each algorithm and binary classification experiment (resp. MNIST one-versus-all experiment) detailed in Section 5.1. Concerning the simulation-based algorithms, the minimum mean square error (MMSE) estimator of $\boldsymbol{\beta}$ has been used to compute the probabilities of belonging to each class and take a decision. The standard deviation (over the different folds) associated to the binary and one-versus-all classification problems was of the order 10^{-1} (%) for the different algorithms. The latter share roughly similar classification performances. Of course, we are aware that these classification results are far from the 99.79% score [29] made by neural networks on MNIST multiclass classification problem. Nevertheless, the results obtained by the proposed approach permit to compare state-of-the-art simulation-based methods on a challenging classification problem where i) the weights in $\boldsymbol{\beta}$ are interpretable and ii) credibility intervals can be drawn.

Table 2. Average classification score (%) over the cross-validation procedure for each experiment and algorithm.

	ADMM	P-MYULA	SPA
MNIST 1-vs-7	99.53	99.44	99.47
USPS 1-vs-7	99.18	99.06	99.11
MNIST 4-vs-6	99.06	98.88	99.12
USPS 4-vs-6	96.21	95.30	96.49
MNIST 3-vs-5	96.10	95.58	95.76
USPS 3-vs-5	97.83	97.08	97.47
MNIST one-vs-all	91.49	90.97	90.35

Table 3. Average number of iterations and computational time over the cross-validation procedure for the different algorithms (for the binary classification problem associated to MNIST only).

	# iterations	time (s)	time ($\times$ ADMM)
ADMM	113	19	1
P-MYULA	10^5	5022	264
SPA	5×10^3	695	37

The main differences between the methods implemented are twofold. First, the number of iterations and thereby the computational time associated to each method can widely differ, see Table 3. Second, the results given by SPA and P-MYULA carry also credibility intervals contrary to ADMM. Thus, ADMM only provides a point estimate for the parameter $\boldsymbol{\beta}$ (corresponding to the MAP estimator) whereas the proposed SPA offers a comprehensive description of the solution through an approximation of the posterior distribution. For instance, these confidence information are discussed in Section 5.3 and could be used to identify challenging observations to classify correctly.

Table 3 presents the average number of iterations performed by each algorithm and their average computational time (in seconds and w.r.t. ADMM). Only the results associated to the MNIST binary classification problems are depicted, conclusions with the USPS dataset are similar. Note that the number of iterations and thereby the computational time of ADMM was adapted at each experiment contrary to simulation-based methods. P-MYULA appears to be slower than SPA (mainly due to slower mixing) which, by embedding the former, improved its computational time by a factor of about 7. Additionally, the latter has a reasonable computational cost compared to vanilla ADMM: it is only 37 times slower. This corresponds to the price to pay to get credibility intervals which can be decisive in a lot of applications (e.g. medicine).

5.3. Sparsity and credibility intervals

Sparsity **–** As introduced in Section 1, interpreting too many active weights of $\boldsymbol{\beta}$ could be challenging and of little interest. Thus sparsity is induced in the weights vector $\boldsymbol{\beta}$ by the Laplacian prior distribution (3). Figure 1 (left) shows an example of the MMSE estimator of a weights vector learned by SPA from MNIST data in the one-versus-all experiment. The active coefficients (dark blue & red) appear to correspond to the contours of the average target label (here the label 2) which is coherent with the classification task. The sparsity of this weight vector is illustrated by the histogram of its coefficients (right), as promoted by the sparsity-inducing Laplacian prior (3).

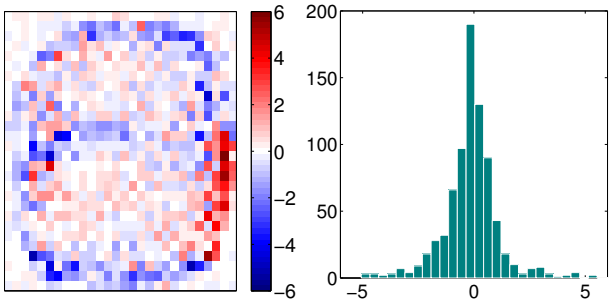

Fig. 1. MNIST one-vs-all experiment: Example of the MMSE estimator of a weight vectors $\boldsymbol{\beta}$ learned by SPA for the 2-vs-all binary classifier (left) and its associated histogram (right).

Credibility intervals **–** More importantly, the Bayesian inference conducted by simulation-based algorithms permits to draw credibility intervals on each feature and thereby on the output of each binary classifier. Such intervals could be used to detect the observations where the classification made by the models could be wrong. For instance, on the MNIST one-versus-all experiment, the percentage of potentially missclassified digits identified by SPA under 90% credibility intervals was of the order 21.98%. By removing these digits with uncertain decision, the classification score increased to 98.36% ($+8$ percentage points) assessing that under 90% credibility intervals, SPA is able to deliver decisions with high accuracy.

Figure 2 shows some observations detected as potentially misclassified by the analysis of 90% credibility intervals associated to SPA. As the one-versus-all approach cannot deliver joint estimated probabilities of belonging to each class, we cannot interpret the credibility intervals associated to the ten binary classifiers outputs similarly to the ones drawn from a multiclass classifier. Nevertheless, the resulting 90% intervals were used to point out the second and third most probable labels. Typical uncertain cases are depicted in Figure 2. For each of these labels, the probability that the associated binary classifier gave a larger response than the winning predicted label has been indicated (in blue). This probability was calculated empirically using the 4800 samples associated to each classifier. As a benefit, the proposed approach is able to propose a choice of potentially more or less credible alternative choices to the main output of the classifier.

6. CONCLUSION

This paper tackles the probabilistic inference of the sparse Bayesian binary logistic regression problem by relying on the recent split-and-augmented Gibbs sampler. The method relies on P-MYULA along with the recently proposed proximal operator of the logistic cost function, enabling new routes toward fast and efficient sampling schemes for regularized Bayesian logistic regression. Such a problem applied to handwritten digits classification can be solved efficiently with SPA with a reasonable computational cost compared to vanilla ADMM. In particular, the resulting Bayesian credibility intervals can be used to identify particularly uncertain decisions, in contrast with optimization-based methods. Such decisions could be for instance revised by conducting further analysis, training another classifier or asking for an expert choice. Future works could include the scaling of the proposed approach for big data settings where the datasets might not fit on a single machine as pointed out in [30].

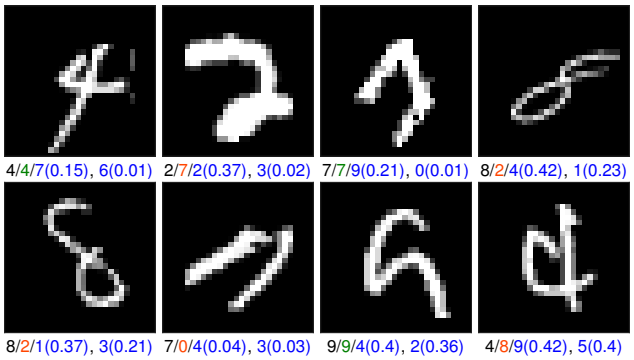

Fig. 2. MNIST one-vs-all experiment: Example of 8 handwritten digits identified as possibly missclassified by SPA (under 90% credibility intervals). The true label (black), the predicted one (green for correct decisions and orange for wrong ones), the second and third most probable labels (blue) and their respective weight (blue) are depicted at the bottom of each sub-figure.

7. REFERENCES

[1] D. R. Cox, "The regression analysis of binary sequences," *J. Roy. Stat. Soc. Ser. B*, vol. 20, no. 2, pp. 215–242, 1958.

[2] S. H. Walker and D. B. Duncan, "Estimation of the probability of an event as a function of several independent variables," *Biometrika*, vol. 54, no. 1-2, pp. 167–179, 1967.

[3] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes," in *Adv. in Neural Information Process. Systems*, 2002, pp. 841–848.

[4] A. Agresti, *Logistic Regression*. Wiley-Blackwell, 2003, ch. 5, pp. 165–210.

[5] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant, *Applied logistic regression*. John Wiley & Sons, 2013, vol. 398.

[6] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2001.

[7] J. C. Marshall, D. J. Cook, N. Christou, G. R. Bernard, C. L. Sprung, and W. J. B. Sibbald, "Multiple organ dysfunction score: a reliable descriptor of a complex clinical outcome." *Crit. Care. Med.*, vol. 23 10, pp. 1638–52, 1995.

[8] H.-L. Chuang, "High school youths' dropout and re-enrollment behavior," *Economics of Education Review*, vol. 16, no. 2, pp. 171 – 186, 1997.

[9] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties," *Found. Trends Mach. Learn.*, vol. 4, no. 1, pp. 1–106, Jan. 2012.

[10] J. H. Albert and S. Chib, "Bayesian analysis of binary and polychotomous response data," *J. Amer. Stat. Assoc.*, vol. 88, no. 422, pp. 669–679, 1993.

[11] C. C. Holmes and L. Held, "Bayesian auxiliary variable models for binary and multinomial regression," *Bayesian Anal.*, vol. 1, no. 1, pp. 145–168, 03 2006.

[12] S. Frühwirth-Schnatter and R. Frühwirth, *Data Augmentation and MCMC for Binary and Multinomial Logit Models*, 2010, pp. 111–132.

[13] R. B. Gramacy and N. G. Polson, "Simulation-based regularized logistic regression," *Bayesian Anal.*, vol. 7, no. 3, pp. 567–590, Sept. 2012.

[14] N. G. Polson, J. G. Scott, and J. Windle, "Bayesian inference for logistic models using Pólya-Gamma latent variables," *J. Amer. Stat. Assoc.*, vol. 108, no. 504, pp. 1339–1349, 2013.

[15] M. Pereyra, "Proximal Markov chain Monte Carlo algorithms," *Stat. Comput.*, vol. 26, no. 4, pp. 745–760, July 2016.

[16] A. Durmus, E. Moulines, and M. Pereyra, "Efficient Bayesian computation by proximal Markov chain Monte Carlo: When Langevin meets Moreau," *SIAM J. Imag. Sci.*, vol. 11, no. 1, pp. 473–506, 2018.

[17] M. Vono, P. Chainais, and N. Dobigeon, "Split-and-augmented Gibbs sampler - Application to large-scale inference problems," *submitted*, 2018. [Online]. Available: https://arxiv.org/abs/1804.05809/

[18] R. Rifkin and A. Klautau, "In defense of one-vs-all classification," *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, Dec. 2004.

[19] T. Park and G. Casella, "The Bayesian lasso," *J. Amer. Stat. Assoc.*, vol. 103, no. 482, pp. 681–686, 2008.

[20] M. A. T. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1150–1159, Sept. 2003.

[21] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization," *Proc. Nat. Academy of Science*, vol. 100, no. 5, pp. 2197–2202, 2003.

[22] A. Y. Ng, "Feature selection, L_1 vs. L_2 regularization, and rotational invariance," in *Proc. Int. Conf. Machine Learning (ICML)*, 2004.

[23] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607–609, 1996.

[24] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

[25] D. A. van Dyk and X.-L. Meng, "The art of data augmentation," *J. Comput. Graph. Stat.*, vol. 10, no. 1, pp. 1–50, 2001.

[26] L. M. Briceno-Arias, G. Chierchia, E. Chouzenoux, and J.-C. Pesquet, "A random block-coordinate Douglas-Rachford splitting method with low computational complexity for binary logistic regression," *submitted*, Dec. 2017. [Online].

[27] A. Maignan and T. C. Scott, "Fleshing out the generalized Lambert W function," *ACM Commun. Comput. Algebra*, vol. 50, no. 2, pp. 45–60, Aug. 2016.

[28] I. Mezö, , and A. Baricz, "On the generalization of the Lambert W function," *Trans. Amer. Math. Soc.*, vol. 369, pp. 7917–7934, 2017.

[29] L. Wan, M. Zeiler, S. Zhang, Y. L. Cun, and R. Fergus, "Regularization of neural networks using DropConnect," in *Proc. Int. Conf. Machine Learning (ICML)*, vol. 28, no. 3, 2013, pp. 1058–1066.

[30] R. Bardenet, A. Doucet, and C. Holmes, "On Markov chain Monte Carlo methods for tall data," *J. Mach. Learn. Res.*, May 2017.