HDBRR: A Statistical Package for High Dimensional Ridge Regression without MCMC

Monroy-Castillo B. **Pérez-Elizalde S.*** **Pérez-Rodríguez P.*** **Crossa J.**
Colegio de Postgraduados* CIMMyT

Abstract

Ridge regression is a useful tool to deal with collinerity in the homoscedastic linear regression model, which provide biased estimators of the regression parameters with lower variance than the least square estimators. Evenmore, when the number of predictors (p) is much larger than the number of observations (n), ridge regression give us unique least square estimators by restringing the parametric space to the neighborhood of the origin. From the Bayesian point of view ridge regression results of assigning a Gaussian prior on the regression parameters and assuming they are conditionally independent. However, from both classical and Bayesian approaches the estimation of parameters is a highly demanding computational task, in the first one being an optimization problem and in the second one a high dimensional integration problem usually faced up through Markov Chain Monte Carlo (MCMC). The main drawback of MCMC is the practical impossibility of checking convergence to the posterior distribution, which is commonly very slow due to the large number of regression parameters. Here we propose a computational algorithm to obtain posterior estimates of regression parameters, variance components and predictions for the conventional ridge Regression model, based on a reparameterization of the model which allows us to obtain the marginal posterior means and variances by integrating out a nuisance parameter whose marginal posterior is defined on the open interval $(0, 1)$.

Keywords: Bayesian Methods, Regression, Variable Selection, Shrinkage, Ridge Regression, MCMC, R.

1. Introduction

Nowadays most research areas use massive quantities of information generated by the increasingly sophisticated computer equipment; for example, in genomics an increasing amount of data is available as new sequencing technologies appears. A lot of statistical models have been proposed in order to learn valuable information from data; however, even with the simplest models, the statisticians or data scientists have to deal with high dimensional inference problems which require millions of computation tasks. One of such models is the ridge regression, being a useful tool to deal with collinearity in the homoscedastic linear regression model by providing biased estimators of regression parameters with lower variance than the least square estimators. Even more, when the number of predictors (p) is much larger than the number of observations (n), ridge regression gives a unique least square estimator by restricting the parametric space.

From the Bayesian point of view, ridge regression results of assigning a Gaussian prior on the

regression parameters and assuming they are conditionally independent. However, since the Bayesian estimation of parameters is a high dimensional integration problem, it is also a highly demanding computational task which is usually faced up through Markov Chain Monte Carlo (MCMC), in particular Gibbs Sampling because the full posterior conditionals are available in closed form. The most successful MCMC option implemented in the R software is the package BGLR(Pérez and de los Campos 2016) , other non Bayesian R package options are penalized(Goeman *et al.* 2021) and ridge(Moritz *et al.* 2021).

The main drawback of MCMC in high dimensional settings is checking of convergence to the joint posterior distribution, which is commonly very slow due to the large number of regression parameters and the high correlations between successive samples from the conditional posteriors in the Gibbs sampling implementation of MCMC. As Rajaratnam and Sparks (2015) shows for the regression model, meanwhile the MCMC samples yield a good approximation of the posterior means of the regression parameters, their posterior variances and the posterior mean of the residual variance may be underestimated if the simulated Markov chain is not large enough; nevertheless, the length of the chain is an issue that still being an open research field. In this paper we propose a simple numerical method to estimate posterior means and variances of the parameters in the ridge regression model as a way to abandon the theoretical guarantees of MCMC methods. We use the SVD and the QR decompositions together with a reparameterization to get closed expressions of the conditional posteriors from where we obtain the marginal posterior means and variances by numerical integration on the open interval $(0, 1)$; furthermore, variable selection and prediction are straightforward consequences. The proposed method is implemented in R and allows to work within the big matrix framework by using storing and parallelization packages bigstatsr, bigparallelr and parallel.

2. Method and Materials

In the Bayesian approach to inference statistics we formally combine, through the Bayes rule, prior information and sample data to learn about unknown quantities of interest. The previous to data uncertainty about the parameter of interest θ is expressed by the prior distribution, the information about θ that comes from observed data is incorporated by the likelihood function and by the Bayes formula we obtain the posterior distribution of the parameter given the data (posterior distribution) (Lee 2012). However, calculating the posterior distribution is not always an easy task because integration is required and, even in low dimensional settings, Monte Carlo or numerical integration methods are needed. In this way, Gilks *et al.* (1996) introduced the MCMC (Markov Chain Monte Carlo) which provides a straightforward and intuitive way to both simulate values from an unknown distribution and use those simulated values to perform subsequent analyses (Speagle 2020). For more information about MCMC see Robert and Casella (2010); Andrieu *et al.* (2003). In this paper we do not use the sampling based approach to approximate de posterior distribution, instead we focus on a numerical approximation of the posterior of a nuisance parameter to integrate out it and to obtain numerical aproximations of posterior means and variances of regression parameters and predictions.

2.1. Model

Consider the linear model

$$y_i = \mathbf{x}_i'\boldsymbol{\beta} + \epsilon_i, \quad i = 1, \ldots, n.$$

where ϵ_i, $i = 1, \ldots, n$, is a Gaussian error with mean 0 and variance σ^2, $\boldsymbol{\beta} \subseteq \mathbb{R}^p$ is a vector of regression coefficients and $\mathbf{x}_i$, $i = 1, \ldots, n$, is p dimensional vector of observed without error covariates. Moreover, assume that $\text{cov}(\epsilon_i, \epsilon_j) = 0$, $i \neq j$. In matrix form the model is written as

$$\mathbf{y} \quad = \quad \mathbf{X}\boldsymbol{\beta} \; + \; \boldsymbol{\varepsilon}, \tag{1}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)'$ is the vector of *regression parameters*, $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)^T$ is the vector of random errors distributed as $\mathcal{N}_p(\mathbf{0}, \sigma^2\mathbf{I}_{nn})$. The $n \times p$ matrix $\mathbf{X}$ is called the *design matrix* and $\mathbf{y}$ is generally referred to as the vector *response variable*. Since the mean of $\mathbf{y}$, $\mathbf{X}\boldsymbol{\beta}$, is a linear combination of the columns of $\mathbf{X}$, the model in (1) is known as the *linear regression model*.

When the number observations is greater than the number of covariates, $n > p$, the best linear unbiased estimator of $\boldsymbol{\beta}$ is $\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$. However, when multicollinearity occurs, although the least squares estimators are unbiased, their variances are inflated because $(\mathbf{X}'\mathbf{X})^{-1}$ tends to be singular. Another scenario where the obtention of the least squares estimator is an ill-posed problem occurs when $p >> n$, which implies that $\hat{\boldsymbol{\beta}}$ is not unique. In both cases some sort of restriction of the parametric space or penalization is needed in order to have unique estimators with lower variance. By adding a degree of bias to the regression estimates, ridge regression gives an unique estimator of $\boldsymbol{\beta}$ with variance lower than the least squares estimator variance.

2.2. Ridge Regression

Hoerl (1962) and Hoerl and Kennard (1968) first suggested that to control the inflation and general instability associated with the least squares estimates we may use the same Tikhonov regularization for all the regression parameters, which gives the ridge estimator:

$$\hat{\boldsymbol{\beta}}^* \quad = \quad \left(\mathbf{X}'\mathbf{X} + k\mathbf{I}\right)^{-1}\left(\mathbf{X}'\mathbf{y}\right); \qquad k > 0, \tag{2}$$

where k is the ridge penalty parameter. Large values of k tend to reduce the magnitude of the estimated regression coefficients, leading to fewer effective model parameters Cannon (2009). See Hoerl and Kennard (1970b,a); Hoerl *et al.* (1975); van Wieringen (2015); Alheety and Kibria (2011); Yahya and Olaifa (2014) for more about ridge regression.

2.3. Bayesian inference for ridge regression

In Bayesian inference all the uncertainty about the unknown parameters $(\boldsymbol{\beta}, \sigma^2)$ is described by the join posterior distribution, which is obtained, through the Bayes rule, as the likelihood function, the joint density of $\mathbf{y}$ seen as a function of the parameters, times the prior. The prior is a probability density function we use to measure the uncertainty about $(\boldsymbol{\beta}, \sigma^2)$ before any data has been observed.

From model in (1) the likelihood function for the parameters $(\boldsymbol{\beta}, \sigma^2)$ is given by

$$L(\boldsymbol{\beta}, \sigma^2 \mid \mathbf{y}) \propto \left(\frac{1}{\sigma^2}\right)^{n/2} \exp\left\{-\frac{1}{2\sigma^2}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})^T(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})\right\}. \quad (3)$$

If the prior for each β_j, $j = 1, \ldots, p$, is Gaussian with mean 0 and variance σ^2_β and prior independence of $\boldsymbol{\beta}$ and σ^2 is assumed then the joint prior is of the form

$$\pi(\boldsymbol{\beta}, \sigma^2, \sigma^2_\beta) = \pi\left(\boldsymbol{\beta} \,|\, \sigma^2_\beta\right) \pi(\sigma^2)\, \pi(\sigma^2_\beta), \quad (4)$$

where,

$$\pi\left(\boldsymbol{\beta} \,|\, \sigma^2_\beta\right) = \mathcal{N}_p\left(\boldsymbol{\beta} \,|\, \mathbf{0}, \sigma^2_\beta \mathbf{I}_p\right)$$

and for the variance parameters we may assign conjugate priors (Reich and Ghosh 2019; Gelman *et al.* 2021); that is, the inverse gamma distributions

$$\begin{aligned} \pi\left(\sigma^2\right) &= \mathcal{IG}\left(\sigma^2 \left| \frac{n_0}{2}, \frac{n_0 s_0^2}{2}\right.\right) \\ \pi\left(\sigma^2_\beta\right) &= \mathcal{IG}\left(\sigma^2_\beta \left| \frac{p_0}{2}, \frac{p_0 d_0^2}{2}\right.\right), \end{aligned}$$

where n_0, s_0^2, p_0 and d_0^2 are known hyperparameters.The prior independence of the elementos of $\boldsymbol{\beta}$ given σ^2_β implies that marginally the distribution of $\boldsymbol{\beta}$ is multivariate t with p_0 degrees of freedom. Then, this hierarchical structure implies that regression parameters are not independent to each other, which seems to be an appropiated structure in presence of colineallity or when $p >> n$. The prior variance of the elements of $\boldsymbol{\beta}$ has also an interpretation in terms of regularization: for fixed σ^2, as σ^2_β tends to 0 the shrinkage to the prior mean increases, which means that large values of the parameters are penalized.

Due to the problem of estimation of the ridge regression parameters is ill-posed, prior elicitation is a critical step in Bayesian inference since the posterior is too sensitive to the assignation of values to the hyperparameters in the prior of $\boldsymbol{\beta}$. Here we use an approach closed to those proposed by Guan and Stephens (2011) and Pérez and de los Campos (2016) to model the initial knowledge of $\boldsymbol{\beta}, \sigma^2$ and σ^2_β. Thus, the prior expected values of σ^2 and σ^2_β are

$$\begin{aligned} \mathrm{E}\left[\sigma^2 \left| \frac{n_0}{2}, \frac{n_0 s_0^2}{2}\right.\right] &= \frac{n_0 s_0^2}{n_0 - 2}, \qquad n_0 > 2. \\ \mathrm{Var}\left[\sigma^2 \left| \frac{n_0}{2}, \frac{n_0 s_0^2}{2}\right.\right] &= \frac{n_0^2 s_0^4}{(n_0-2)^2(2n_0-8)}, \qquad n_0 > 4. \end{aligned}$$

Therefore, in order to have prior moments of first and second order finite for the variance components, the value of n_0 and p_0 should be at least 5. Then,we use 5 as the default value for n_0 and p_0 in the HDBRR package, but flat priors could be obtained as n_0 and p_0 tend to 0. To assign values to s_0^2 and d_0^2 we will use the prior expected value of the proportion of variance explained (PVE) by the model with respect to the residual variance, that is given by

$$\mathrm{PVE} = \frac{1}{n}\sum_{i=1}^{n} \frac{(\boldsymbol{x}_i'\boldsymbol{\beta})^2}{\sigma^2} = \frac{1}{n\sigma^2}\sum_{i=1}^{n}(\sum_{j=1}^{p} x_{ij}\beta_j)^2.$$

Then, by noting that

$$\mathrm{E}\left[\left(\sum_{j=1}^{p} x_{ij}\beta_j\right)^2 |\sigma^2, \sigma^2_\beta\right] \;=\; \mathrm{Var}\left[\sum_{j=1}^{p} x_{ij}\beta_j|\sigma^2, \sigma^2_\beta\right] = \sigma^2_\beta \sum_{j=1}^{p} x^2_{ij}$$

we have that

$$\mathrm{E[PVE]} \;=\; \mathrm{E[E[PVE}|\sigma^2, \sigma^2_\beta]] \;=\; \frac{1}{s_0^2}\frac{p_0 d_0^2}{p_0 - 1}\sum_{i=1}^{n}\sum_{j=1}^{p}\frac{x^2_{ij}}{n}.$$

Let

$$\mathrm{h} \;=\; \frac{\mathrm{E[PVE]}}{1+\mathrm{E[PVE]}} \;=\; \frac{p_0 d_0^2 \sum_{i=1}^{n}\sum_{j=1}^{p}\frac{x^2_{ij}}{n}}{s_0^2(p_0-1) + p_0 d_0^2 \sum_{i=1}^{n}\sum_{j=1}^{p}\frac{x^2_{ij}}{n}}.$$

the proportion of the total prior variance explained by the model. From this,

$$\frac{h}{1-h} \;=\; \frac{p_0 d_0^2 \sum_{i=1}^{n}\sum_{j=1}^{p}\frac{x^2_{ij}}{n}}{s_0^2(p_0-1)},$$

therefore

$$d_0^2 \;=\; \left(\frac{h}{1-h}\right)\frac{s_0^2(p_0-1)}{p_0 \sum_{i=1}^{n}\sum_{j=1}^{p}\frac{x^2_{ij}}{n}}.$$

The value s_0^2 may be interpreted as a prior guess about the residual variance.

In particular, when each covariate and the response have been centered about its sample mean

$$d_0^2 \;=\; \left(\frac{h}{1-h}\right)\frac{s_0^2(p_0-1)}{p_0 \sum_{j=1}^{p} s_j^2};$$

moreover, if the all the data are standardized and $s_0^2 = 1$ we have that $d_0^2 = \left(\frac{h}{1-h}\right)\frac{(p_0-1)}{p_0 p}$ which approaches to $1/p$ when $h = 0.5$ and p_0 is large. Also note that as h tends to 1 the prior for σ^2_β becomes flat but proper distribution.

Once the prior distribution has been assigned and the likelihood function defined, then the posterior distribution of the regression parameters is derived in what follows.

2.4. Posterior Distribution

By the Baye's Rule, the joint posterior is obtained as the product of likelihood function in (3) and the prior in (4). Thus,

$$\pi\left(\boldsymbol{\beta},\, \sigma^2,\, \sigma^2_{\boldsymbol{\beta}} \mid \mathbf{y}\right) \;\propto\; L\left(\boldsymbol{\beta},\, \sigma^2 |\mathbf{y}\right)\, \pi(\boldsymbol{\beta}, \sigma^2, \sigma^2_{\boldsymbol{\beta}}),$$

Now, cosiderer the transformations $v = 1/\sigma^2 + 1/\sigma^2_{\boldsymbol{\beta}}$ and $u = \sigma^2/\left(\sigma^2 + \sigma^2_{\boldsymbol{\beta}}\right)$; then, the joint posterior distribution of $(\boldsymbol{\beta},\, u,\, v)$ is given by

$$\begin{aligned}\pi(\boldsymbol{\beta},\, u,\, v \mid \mathbf{y}) \;\propto\; & v^{\frac{n+n_0+p+p_0}{2}-1} u^{\frac{p+p_0}{2}-1}(1-u)^{\frac{n+n_0}{2}-1} \\ & \times \exp\left\{-\frac{v(1-u)}{2}\left[\left(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}(u)\right)' \boldsymbol{\Sigma}_n^{-1}(u)\left(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}(u)\right) \,+\, SSE(u) \,+\, n_0 s_0^2\right]\right\} \\ & \times \exp\left\{-\frac{uv}{2}\left(\hat{\boldsymbol{\beta}}(u)\hat{\boldsymbol{\beta}}(u) \,+\, p_0 d_0^2\right)\right\},\end{aligned} \tag{5}$$

where $\boldsymbol{\Sigma}_n(u) = \left(\mathbf{X}'\mathbf{X} + \frac{u}{1-u}\mathbf{I}\right)^{-1}$, $\hat{\boldsymbol{\beta}}(u) = \boldsymbol{\Sigma}_n(u)\mathbf{X}'\mathbf{y}$, $\hat{\mathbf{y}}(u) = \mathbf{X}\hat{\boldsymbol{\beta}}(u)$ and $SSE(u) = (\mathbf{y} - \hat{\mathbf{y}}(u))'(\mathbf{y} - \hat{\mathbf{y}}(u))$. From (5), the full conditional posterior of $\boldsymbol{\beta}$ is

$$\pi(\boldsymbol{\beta} \,|\, u,\, v,\, \mathbf{y}) \;=\; \mathcal{N}_p\left(\boldsymbol{\beta} \;\middle|\; \hat{\boldsymbol{\beta}}(u),\, \frac{1}{v(1-u)}\boldsymbol{\Sigma}_n(u)\right). \tag{6}$$

Now, let $S(u) = (1-u)\,(SSE(u) + n_0 s_0^2) + u(\hat{\boldsymbol{\beta}}'(u)\hat{\boldsymbol{\beta}}(u) + p_0 d_0^2)$ and by the definition of conditional distribution of $\boldsymbol{\beta}$ given (u, v), we have that

$$\pi(v \,|\, u,\, \mathbf{y}) \;=\; \frac{\pi(\boldsymbol{\beta},\, v \,|\, u,\, \mathbf{y})}{\pi(\boldsymbol{\beta} \,|\, u,\, v,\, \mathbf{y})} \;=\; \mathcal{G}\left(v \;\middle|\; \frac{n + n_0 + p_0}{2},\, \frac{S(u)}{2}\right), \tag{7}$$

where $\pi(\boldsymbol{\beta},\, v \,|\, u,\, \mathbf{y})$, from (5), is a Normal-Gamma density. It follows that, the posterior distribution of $\boldsymbol{\beta}$ given u is t with $\nu = n + n_0 + p_0$ degrees of freedom, mean

$$\mathrm{E}[\boldsymbol{\beta} \,|\, u,\, \mathbf{y}] \;=\; \hat{\boldsymbol{\beta}}(u)$$

and variance

$$\mathrm{V}[\boldsymbol{\beta} \,|\, u,\, \mathbf{y}] \;=\; \frac{S(u)}{(\nu-2)(1-u)}\boldsymbol{\Sigma}_n(u), \qquad\qquad \nu - 2 > 0.$$

Finally, the marginal posterior of u is obtained as

$$\begin{aligned}
\pi(u \,|\, \mathbf{y}) \;&=\; \frac{\pi(u,\, v \,|\, \mathbf{y})}{\pi(v \,|\, u,\, \mathbf{y})} \;=\; \frac{\pi(\boldsymbol{\beta},\, u,\, v \,|\, \mathbf{y}) / \pi(\boldsymbol{\beta} \,|\, u,\, v,\, \mathbf{y})}{\pi(v \,|\, u,\, \mathbf{y})} \\
&\propto\; u^{\frac{p+p_0}{2}-1}(1-u)^{\frac{n-p+n_0}{2}-1}\,|\boldsymbol{\Sigma}_n(u)|^{1/2} \\
&\quad \times \left[(1-u)(SSE(u) + n_0 s_0^2) \;+\; u\,\left(\hat{\boldsymbol{\beta}}'(u)\hat{\boldsymbol{\beta}}(u) + p_0 d_0^2\right)\right]^{-\frac{n+n_0+p_0}{2}}, u \in (0,1)
\end{aligned} \tag{8}$$

It is important to point out that the marginal moments $\pi(\boldsymbol{\beta} \mid \mathbf{y})$ can be obtained by theorem of total expectation. I such a way, the unconditional posterior mean and variance of $\boldsymbol{\beta}$ are, respectively,

$$\mathrm{E}[\boldsymbol{\beta} \,|\, \mathbf{y}] = \mathrm{E}\left[\mathrm{E}[\boldsymbol{\beta} \,|\, u,\, \mathbf{y}]\right] \;=\; \int_0^1 \hat{\boldsymbol{\beta}}(u)\pi(u \,|\, \mathbf{y})du$$

and

$$\begin{aligned}
\mathrm{V}[\boldsymbol{\beta} \,|\, \mathbf{y}] \;&=\; \mathrm{E}\left[\mathrm{V}[\boldsymbol{\beta} \,|\, u,\, \mathbf{y}]\right] + \mathrm{V}\left[\mathrm{E}[\boldsymbol{\beta} \,|\, u,\, \mathbf{y}]\right] \\
&=\; \int_0^1 \left(\frac{S(u)}{(\nu-2)(1-u)}\boldsymbol{\Sigma}_n(u) + \left(\hat{\boldsymbol{\beta}}(u) - \mathrm{E}[\boldsymbol{\beta} \,|\, \mathbf{y}]\right)^2\right)\pi(u \,|\, \mathbf{y})du.
\end{aligned}$$

Both integrals above may be evaluated numerically with accurate precision in most cases.

Marginal posterior distributions of variance components

The marginal distributions of σ^2 and $\sigma^2_{\boldsymbol{\beta}}$ are obtained from the joint distribution, $\pi(u,\, v \,|\, \mathbf{y}) = \pi(v \,|\, u, \mathbf{y})\pi(u \,|\, \mathbf{y})$; that is, using the change of variable formula,

$$\pi(\sigma^2,\, \sigma^2_{\boldsymbol{\beta}} \,|\, \mathbf{y}) \;=\; \pi(v(\sigma^2,\, \sigma^2_{\boldsymbol{\beta}}) \,|\, u\left(\sigma^2,\, \sigma^2_{\boldsymbol{\beta}}\right),\, \mathbf{y})\,\pi\left(u\left(\sigma^2,\, \sigma^2_{\boldsymbol{\beta}}\right) \,|\, \mathbf{y}\right)\left|\frac{\partial(u,v)}{\partial\left(\sigma^2, \sigma^2_{\boldsymbol{\beta}}\right)}\right.$$

However, the marginals can not be obtained in closed form, so numerical or Monte Carlo integration over $\mathbb{R}^+$ is needed. However, if only points estimates are needed, it is possible to get them using one dimensional integration over the interval $(0, 1)$. For example, the Bayesian estimator under square loss of the ridge parameter $\lambda = \sigma^2/\sigma^2_\beta = u/(1-u)$ is given by

$$\hat{\lambda} \;=\; \mathrm{E}[\lambda \,|\, \mathbf{y}] = \int_0^1 \frac{u}{1-u}\, \pi(u \,|\, \mathbf{y})\, \mathrm{d}u.$$

In the same way, the posterior means of σ^2_β and σ^2 are

$$\begin{aligned} \mathrm{E}\left[\sigma^2_\beta \,|\, \mathrm{y}\right] &= \mathrm{E}\left[\frac{1}{uv}\,\middle|\,\mathrm{y}\right] = \mathrm{E}\left[\frac{1}{u}\,\mathrm{E}\left[\frac{1}{v}\,\middle|\, u,\, \mathrm{y}\right]\right] \\ &= \frac{1}{n+n_0+p_0-1}\int_0^1 \frac{S(u)}{u}\, \pi(u \,|\, \mathbf{y})\, \mathrm{d}u. \end{aligned}$$

and

$$\mathrm{E}[\sigma^2 \,|\, \mathrm{y}] \;=\; \mathrm{E}\left[\frac{1}{v(1-u)}\,\middle|\,\mathrm{y}\right] \;=\; \frac{1}{n+n_0+p_0-1}\int_0^1 \frac{S(u)}{1-u}\, \pi(u \,|\, \mathbf{y})\, \mathrm{d}u.$$

2.5. Variable Selection

Suppose that the prior density for $\boldsymbol{\beta}$ is such that,

$$\pi\left(\beta_j \,|\, \sigma^2_\beta,\, \gamma_j\right) \;=\; (1-\gamma_j)\mathrm{N}\left(\beta_j \,|\, 0, \sigma^2_\beta\right) + \gamma_j \mathrm{N}\left(\beta_j \,|\, 0, c_j^2\sigma^2_\beta\right), \quad j = 1, \ldots, p.$$

Where $\gamma_j \overset{iid}{\sim} \text{Bernoulli}(\phi_j)$ is an indicator variable which is $\gamma_j = 1$ if the $j-th$ predictor variable is included in the model, in other case $\gamma_j = 0$. To use this hierarchical mixture setup for variable selection, the hyperparameters σ^2_β and $c^2\sigma^2_\beta$ are set "small and large", respectively, so that $\mathcal{N}(0, \sigma^2_\beta)$ is concentrated around 0 and $\mathcal{N}(0, c_j^2\sigma^2_\beta)$ is diffuse as in Figure 1.

If the data supports $\gamma_j = 0$ over $\gamma_j = 1$, then β_j is probably small enough so that X_j will not be needed in the model. Suppose a value $\delta_j > 0$ such that if $|\beta_j| < \delta_j$ it would be preferable to exclude X_j. The parameter δ_j should be chosen so that the posterior probability $\Pr(\gamma_j = 1 \,|\, \mathbf{y})$ must be higher for those values of β_j such that $|\beta_j| > \delta_j$ than for those in the neighborhood of 0. Before any data has been observed, δ_j may be fixed by choosing σ^2_β and $c_j^2\sigma^2_\beta$ such that the pdf $\pi\left(\beta_j \,|\, \gamma_j = 0\right) = \mathcal{N}(\beta_j \mid 0, \sigma^2_\beta)$ is larger than the pdf $\pi\left(\beta_j \,|\, \gamma_j = 1\right) = \mathcal{N}(\beta_j \mid 0, c_j^2\sigma^2_\beta)$ on the interval $(-\delta_{j\gamma}, \delta_{j\gamma})$ (see Figure 1). This condition is satisfied for any σ_β and c_j such that

$$\frac{\log\left(\frac{c_j^2\sigma^2_\beta}{\sigma^2_\beta}\right)}{\frac{1}{\sigma^2_\beta} - \frac{1}{c_j^2\sigma^2_\beta}} \;\leq\; \delta_j^2. \tag{9}$$

Hence,

$$\delta_j \;=\; \sqrt{\frac{2c_j^2\sigma^2_\beta \log\left(c_j\right)}{c_j^2 - 1}}, \quad c_j > 1.$$

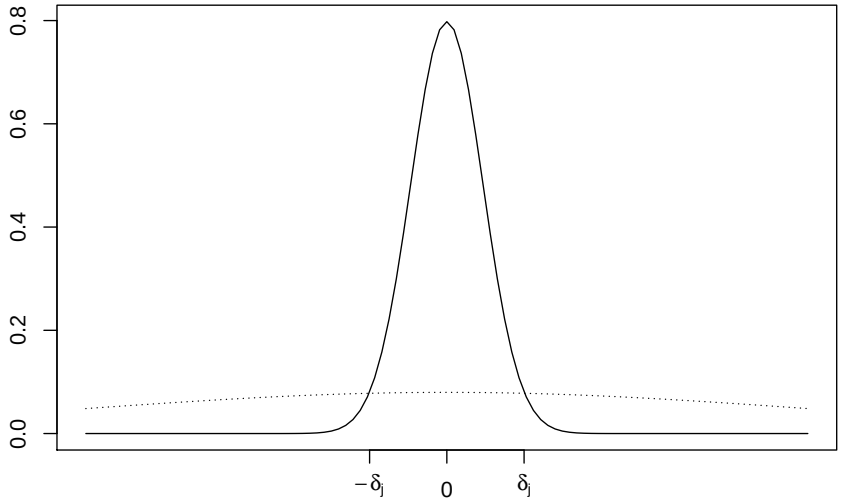


Figure 1: $\mathcal{N}(0, \sigma_\beta^2)$ and $\mathcal{N}(0, c_j^2\sigma_\beta^2)$ densities. Intersection at δ_i

For the selection of c_j, note that it is equal to the ratio of $\pi\left(\beta_j = 0 \,|\, \gamma_j = 0\right)$ and $\pi\left(\beta_j = 0 \,|\, \gamma_j = 1\right)$; thus, c_j may be interpreted as the prior odds that X_j should be excluded of the model if β_j is to small. Further explanation about the selection of c_j and δ_j can be found in George and McCulloch (1993). In what follows, we will assume that, for $j = 1, \ldots, p$, $\phi_j = \phi$ and $c_j = c$, which implies that $\delta_j = \delta$. In this way, the variable selection procedure is suitable for covariates in the same scale.

The joint posterior distribution of $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_p)$ is given by

$$\pi(\boldsymbol{\gamma} \,|\, \mathbf{y}) \;=\; \pi(\mathbf{y} \,|\, \boldsymbol{\gamma})\pi(\boldsymbol{\gamma})/\pi(\mathbf{y})$$

where the marginal model given γ in terms of the joint posterior of $\left(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2\right)$ is

$$\pi\left(\mathbf{y} \,|\, \boldsymbol{\gamma}\right) \;=\; \frac{L\left(\boldsymbol{\beta}, \sigma^2 \,|\, \mathbf{y}\right) \pi\left(\boldsymbol{\beta} \,|\, \sigma_\beta^2, \boldsymbol{\gamma}\right) \pi\left(\sigma^2\right) \pi\left(\sigma_\beta^2\right)}{\pi\left(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2 \,|\, \boldsymbol{\gamma}, \mathbf{y}\right)}.$$

Then,

$$\begin{aligned} \pi(\boldsymbol{\gamma} \,|\, \mathbf{y}) \;&=\; \frac{\pi(\boldsymbol{\gamma})\, L\left(\boldsymbol{\beta}, \sigma^2 \,|\, \mathbf{y}\right) \pi\left(\boldsymbol{\beta} \,|\, \sigma_\beta^2, \boldsymbol{\gamma}\right)\, \pi\left(\sigma^2\right) \pi\left(\sigma_\beta^2\right)}{\pi(\mathbf{y})\, \pi\left(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2 \,|\, \boldsymbol{\gamma}, \mathbf{y}\right)} \\ &\propto\; \frac{\pi(\boldsymbol{\gamma})\pi\left(\boldsymbol{\beta} \,\middle|\, \sigma_\beta^2, \boldsymbol{\gamma}\right)}{\pi\left(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2 \,\middle|\, \boldsymbol{\gamma}, \mathbf{y}\right)} \end{aligned} \tag{10}$$

where it should be noted that $\pi\left(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2 \,\middle|\, \boldsymbol{\gamma} = \mathbf{0}, \mathbf{y}\right)$ is the joint posterior given in (5). Since,

$$\pi(\boldsymbol{\gamma} \,|\, \mathbf{y}) \;=\; \pi\left(\gamma_j \,|\, \boldsymbol{\gamma}_{-j}, \mathbf{y}\right) \pi\left(\boldsymbol{\gamma}_{-j} \,|\, \mathbf{y}\right),$$

where $\boldsymbol{\theta}_{-j}$ is the subvector composed by all the elements of $\boldsymbol{\theta}$ except the j-th element θ_j. Then, it follows from (10) and prior independence that

$$\begin{aligned} \pi\left(\gamma_j \,|\, \boldsymbol{\gamma}_{-j}, \mathbf{y}\right) &= \frac{\pi(\boldsymbol{\gamma} \,|\, \mathbf{y})}{\pi\left(\boldsymbol{\gamma}_{-j} \,|\, \mathbf{y}\right)} \propto \frac{\pi\left(\gamma_j\right) \pi\left(\beta_j \mid \sigma_\beta^2, \gamma_j\right)}{\pi\left(\boldsymbol{\gamma}_{-j} \,|\, \mathbf{y}\right) \pi\left(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2 \,|\, \boldsymbol{\gamma}, \mathbf{y}\right)} \\ &\propto \frac{\phi^{\gamma_j}(1-\phi)^{1-\gamma_j} \pi\left(\beta_j \mid \sigma_\beta^2, \gamma_j\right)}{\pi\left(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2 \,|\, \boldsymbol{\gamma}, \mathbf{y}\right)}. \end{aligned}$$

Expressing the right hand side of the result above in terms of ($\boldsymbol{\beta}$, u, v) we have the following:

$$\begin{aligned} \pi(\gamma_j \,|\, \boldsymbol{\gamma}_{-j}, \mathbf{y}) &\propto \frac{\phi^{\gamma_j}(1-\phi)^{1-\gamma_j} \pi\left(\beta_j \,|\, u, v, \gamma_j\right)}{\pi\left(\boldsymbol{\beta}, u, v \,|\, \boldsymbol{\gamma}, \mathbf{y}\right)} \\ &\propto \frac{\phi^{\gamma_j}(1-\phi)^{1-\gamma_j} \pi\left(\beta_j \,|\, u, v, \gamma_j\right)}{\pi\left(\beta_j \,|\, \boldsymbol{\beta}_{-j}, u, v, \boldsymbol{\gamma}, \mathbf{y}\right) \pi\left(\boldsymbol{\beta}_{-j} \,|\, u, v, \boldsymbol{\gamma}, \mathbf{y}\right) \pi(v \,|\, u, \boldsymbol{\gamma}, \mathbf{y}) \pi(u \,|\, \boldsymbol{\gamma}, \mathbf{y})} \\ &\propto \frac{\phi^{\gamma_j}(1-\phi)^{1-\gamma_j} \pi\left(\beta_j \,|\, u, v, \gamma_j\right)}{\pi\left(\beta_j \,|\, \boldsymbol{\beta}_{-j}, u, v, \boldsymbol{\gamma}, \mathbf{y}\right) \pi(v \,|\, u, \boldsymbol{\gamma}, \mathbf{y}) \pi(u \,|\, \boldsymbol{\gamma}, \mathbf{y})}. \end{aligned}$$

Then, the conditional posterior probability of exclusion of the variable X_j is

$$\begin{aligned} \Pr\left(\gamma_j = 0 \,|\, \boldsymbol{\gamma}_{-j} = \mathbf{0}, \mathbf{y}\right) &= \pi\left(\gamma_j = 0 \,|\, \boldsymbol{\gamma}_{-j} = \mathbf{0}, \mathbf{y}\right) \\ &\propto \frac{(1-\phi)\pi\left(\beta_j \,|\, u, v, \gamma_j = 0\right)}{\pi\left(\beta_j \,|\, \boldsymbol{\beta}_{-j}, u, v, \boldsymbol{\gamma} = \mathbf{0}, \mathbf{y}\right) \pi(v \,|\, u, \boldsymbol{\gamma} = \mathbf{0}, \mathbf{y})\, \pi(u \,|\, \boldsymbol{\gamma} = \mathbf{0}, \mathbf{y})} \\ &= C_j \frac{(1-\phi)\pi\left(\beta_j \,|\, u, v, \gamma_j = 0\right)}{\pi\left(\beta_j \,|\, \boldsymbol{\beta}_{-j}, u, v, \boldsymbol{\gamma} = \mathbf{0}, \mathbf{y}\right)}. \end{aligned} \tag{11}$$

Note that probability exclusion (11) does not depend on the values of $\boldsymbol{\beta}$, u and v and should be equal to one when the conditional posterior mean of β_j is equal to the prior mean and $\beta_j = 0$. Moreover, since the prior and the conditional posterior are both normal, it can be verified that he proportionality constant is $C_j = (1-\phi)^{-1}\sqrt{\sum_{i=1}^n \frac{1-u}{u} x_{ij}^2 + 1}$. Thus, the probability of the j-th predictor to be included in the model given that the others have been excluded is:

$$\begin{aligned} \hat{p}_j = \Pr\left(\gamma_j = 1 \,|\, \boldsymbol{\gamma}_{-j} = \mathbf{0}, \mathbf{y}\right) &= 1 - \Pr\left(\gamma_j = 0 \,|\, \boldsymbol{\gamma}_{-j} = \mathbf{0}, \mathbf{y}\right) \\ &= 1 - \exp\left\{-\frac{\tilde{u}\tilde{v}}{2}\tilde{\beta}_j^2 - \frac{(1-\tilde{u})\tilde{v}}{2V_j}\left(\tilde{\beta}_j - E_j\right)^2\right\} \end{aligned} \tag{12}$$

where, $V_j = \left(\sum_{i=1}^n x_{ij}^2 + \frac{\tilde{u}}{1-\tilde{u}}\right)^{-1}$ and $E_j = V_j \mathrm{X}_j' \left(\mathbf{y} - \mathbf{X}_{-j}\tilde{\boldsymbol{\beta}}_{-j}\right)$ are the conditional posterior variance and mean of β_j. The quantities $\tilde{u}$,$\tilde{v}$ and $\tilde{\boldsymbol{\beta}}$ are arbitrary values u,v and $\boldsymbol{\beta}$, but

with high posterior density. From(12) we have that the conditional Bayes factor in favour of including the covariate X_j in the model is

$$BF_j = \frac{(1-\phi)\hat{p}_j}{(1-\hat{p}_j)\phi} = \frac{(1-\phi)}{\phi}\left(\exp\left\{\frac{\tilde{v}(1-\tilde{u})}{2}\left[\frac{\tilde{u}}{(1-\tilde{u})}\tilde{\beta}_j^2 + \frac{1}{V_j}\left(\tilde{\beta}_j - E_j\right)^2\right]\right\} - 1\right)$$

2.6. Posterior Computation

The main feature of the parameterization proposed above is that it enable to use standard matrix algebra to speed up the computation of the posterior distributions without using sampling techniques. This helps to significantly reduce computation time, avoiding slow sampling methods. In fact, once the posterior distribution of u is calculated all the estimations of interest are almost automatically available.

Posterior computation through SVD decomposition

Let $\mathbf{UDV}'$ the full singular value decomposition (SVD) of the matrix of covariates $\mathbf{X}$ in the linear model (1). Note that $\mathbf{V} = [\mathbf{V}_1, \mathbf{V}_2]$, where $\mathbf{V}_1$ and $\mathbf{V}_2$ are orthonormals and $\mathbf{D} = [\mathbf{S}, \mathbf{0}]$ is a rectangular diagonal matrix, where $\mathbf{S}$ is the diagonal matrix of size $n \times n$ of positive singular values, $s_1 \geq \cdots \geq s_n$ of $\mathbf{X}$ and the last $p - n$ columns are all vectors of zeros.
Hence

$$\begin{aligned}
\boldsymbol{\Sigma}_n(u) &= \left(\mathbf{X}'\mathbf{X} + \frac{u}{1-u}\mathbf{I}_p\right)^{-1} \\
&= \left(\mathbf{V}\mathbf{D}'\mathbf{D}\mathbf{V}' + \frac{u}{1-u}\mathbf{V}\mathbf{V}'\right)^{-1} \\
&= \mathbf{V}\left(\begin{bmatrix} \boldsymbol{\Lambda} & \mathbf{0}_{n,p-n} \\ \mathbf{0}_{p-n,n} & \mathbf{0}_{p-n,p-n} \end{bmatrix} + \frac{u}{1-u}\mathbf{I}_p\right)^{-1}\mathbf{V}' \\
&= \mathbf{V}\begin{bmatrix} \boldsymbol{\Lambda} + \frac{u}{1-u}\mathbf{I}_n & \mathbf{0}_{n,p-n} \\ \mathbf{0}_{p-n,n} & \frac{u}{1-u}\mathbf{I}_{p-n} \end{bmatrix}^{-1}\mathbf{V}',
\end{aligned}$$

where $\boldsymbol{\Lambda} = \mathbf{S}'\mathbf{S} = \text{diag}(\lambda_1, \ldots, \lambda_n)$ is diagonal matrix whose elements are the eigenvalues of $\mathbf{X}\mathbf{X}'$.
Similarly,

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_u &= \boldsymbol{\Sigma}_u\mathbf{X}'\mathbf{y} \\
&= \mathbf{V}\begin{bmatrix} \boldsymbol{\Lambda} + \frac{u}{1-u}\mathbf{I}_n & \mathbf{0}_{n,p-n} \\ \mathbf{0}_{p-n,n} & \frac{u}{1-u}\mathbf{I}_{p-n} \end{bmatrix}^{-1}\mathbf{V}'\mathbf{V}\mathbf{D}'\mathbf{U}'\mathbf{y} \\
&= \mathbf{V}_1\left(\boldsymbol{\Lambda} + \frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{S}\mathbf{U}'\mathbf{y}.
\end{aligned}$$

And

$$\begin{aligned}\hat{\mathbf{y}}(u) &= \mathbf{X}\hat{\boldsymbol{\beta}}(u)\\ &= \mathbf{UDV}'\mathbf{V}_1\left(\boldsymbol{\Lambda}+\frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{SU}'\mathbf{y}\\ &= \mathbf{US}\left(\boldsymbol{\Lambda}+\frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{S}'\mathbf{U}'\mathbf{y}\\ &= \mathbf{UPU}'\mathbf{y},\end{aligned}$$

where $\mathbf{P}$ is a diagonal matrix with $\mathbf{P}_{jj}=\frac{(1-u)\lambda_j}{(1-u)\lambda_j+u}$, $j=1,\ldots,n$.
Thus, for $p>n$, substituting in (8) the covariance matrix by its SVD decomposition we have,

$$\begin{aligned}\pi(u\,|\,\mathbf{y}) &\propto u^{\frac{p_0+p}{2}-1}(1-u)^{\frac{n_0+n-p}{2}-1}|\mathbf{X}'\mathbf{X}+\frac{u}{1-u}\mathbf{I}|^{-1/2}[(1-u)(SSE(u)+n_0s_0^2)]^{-\frac{n+n_0+p_0}{2}}\\ &\quad\times\left[1+\frac{u}{1-u}\frac{\hat{\boldsymbol{\beta}}'(u)\hat{\boldsymbol{\beta}}(u)+p_0d_0^2}{SSE(u)+n_0s_0^2}\right]^{-\frac{n+n_0+p_0}{2}}\\ &\propto u^{\frac{p_0+p}{2}-1}(1-u)^{\frac{n_0+n-p}{2}-1}\left|\begin{matrix}\boldsymbol{\Lambda}+\frac{u}{1-u}\mathbf{I}_n & \mathbf{0}_{n,p-n}\\ \mathbf{0}_{p-n,p} & \frac{u}{1-u}\mathbf{I}_{p-n}\end{matrix}\right|^{-1/2}\\ &\quad\times[(1-u)(SSE(u)+n_0s_0^2)]^{-\frac{n+n_0+p_0}{2}}\left[1+\frac{u}{1-u}\frac{\hat{\boldsymbol{\beta}}'(u)\hat{\boldsymbol{\beta}}(u)+p_0d_0^2}{SSE(u)+n_0s_0^2}\right]^{-\frac{n+n_0+p_0}{2}}\\ &\propto u^{\frac{p+p_0}{2}-1}(1-u)^{\frac{n+n_0}{2}-1}u^{\frac{n-p}{2}}\left(\prod_{j=1}^{n}(1-u)\lambda_j+u\right)^{-\frac{1}{2}} \qquad (13)\\ &\quad\times[(1-u)(SSE(u)+n_0s_0^2)]^{-\frac{n+n_0+p_0}{2}}\left[1+\frac{u}{1-u}\frac{\hat{\boldsymbol{\beta}}'(u)\hat{\boldsymbol{\beta}}(u)+p_0d_0^2}{SSE(u)+n_0s_0^2}\right]^{-\frac{n+n_0+p_0}{2}},\end{aligned}$$

where

$$\begin{aligned}SSE(u) &= (\mathbf{y}-\hat{\mathbf{y}}(u))'(\mathbf{y}-\hat{\mathbf{y}}(u))\\ &= (\mathbf{y}-\mathbf{UPU}'\mathbf{y})'(\mathbf{y}-\mathbf{UPU}'\mathbf{y}) \qquad (14)\\ &= \mathbf{y}'\mathbf{U}(\mathbf{I}-\mathbf{P})^2\mathbf{U}'\mathbf{y}.\end{aligned}$$

In general,the marginal posterior of u is

$$\begin{aligned}\pi(u\,|\,\mathbf{y}) &\propto u^{\frac{p+p_0}{2}-1}(1-u)^{\frac{n+n_0}{2}-1}u^{\frac{n-p}{2}I(p>n)}\left(\prod_{j=1}^{\min(n,p)}(1-u)\lambda_j+u\right)^{-\frac{1}{2}} \qquad (15)\\ &\quad\times[(1-u)(SSE(u)+n_0s_0^2)]^{-\frac{n+n_0+p_0}{2}}\left[1+\frac{u}{1-u}\frac{\hat{\boldsymbol{\beta}}'(u)\hat{\boldsymbol{\beta}}(u)+p_0d_0^2}{SSE(u)+n_0s_0^2}\right]^{-\frac{n+n_0+p_0}{2}}\end{aligned}$$

Also note that

$$\mathrm{Cov}\left(\hat{\mathbf{y}}(u),\mathbf{y}\,|\,u\right) = \mathrm{Cov}\left(\mathbf{UPU}'\mathbf{y},\mathbf{y}\,|\,u\right) = \mathbf{UPU}'\mathrm{Cov}(\mathbf{y}) = \sigma^2\mathbf{UPU}'$$

and

$$\text{Var}\left(\hat{\mathbf{y}}(u)\,|\,u\right) \;=\; \sigma^2\mathbf{UPU}',$$

hence

$$\text{cor}\left(\hat{\mathrm{y}}_i,\mathrm{y}_i\right) \;=\; \int_0^1 \frac{\sum_{j=1}^r u_{i,j}^2\mathbf{P}_{jj}}{\sqrt{\sum_{j=1}^r u_{i,j}^2\mathbf{P}_{jj}^2}}\,\pi(u\,|\,\mathbf{y})\,du. \tag{16}$$

For ridge regression the efcetive degrees of freedom may be calculated as the expected value trace of the matrix $\mathbf{UPU}'$; that is:

$$edf \;=\; \text{E}\left[\text{tr}(\mathbf{UPU}'\,|\,\mathbf{y})\right] \;=\; \text{E}\left[\text{tr}(\mathbf{P}\,|\,\mathbf{y})\right] \;=\; \int_0^1\sum_{j=1}^p \mathbf{P}_{jj}\pi(u\,|\,\mathbf{y})\,\text{d}u.$$

A naive but useful approach to calculate the quantities above may be to plug in the posterior mode of u. Thus, for example, the effective degrees of freedom may be approximated by

$$\widehat{edf} \;=\; \sum_{j=1}^p \frac{(1-\hat{u})\,\lambda_j}{(1-\hat{u})\,\lambda_j+\hat{u}},$$

where $\hat{u}=\max\arg\pi(u\,|\,\mathbf{y})$. Obviously, when there is no shrinkage $\hat{u}=0$ and $edf=\text{rank}(\mathbf{X})$.

Posterior Computation through QR decomposition

Another procedure to compute the posterior distribution is given by the QR factorization of $\mathbf{X}'$. This is, let $\mathbf{X}'=\mathbf{QR}$, where $\mathbf{Q}=[\mathbf{Q}_1,\mathbf{Q}_2]$ is a $p\times p$ orthonormal matrix and $\mathbf{R}=[\mathbf{R}_1',\mathbf{R}_2']'$ is a $p\times n$ upper triangular matrix, where the entries of the $(p-n)\times n$ matrix $\mathbf{R}_2$ are all zeros. Thus, proceeding in the same way as with SVD decomposition,

$$\begin{aligned}
\boldsymbol{\Sigma}_n(u) &= \left(\mathbf{X}'\mathbf{X}+\frac{u}{1-u}\mathbf{I}_p\right)^{-1} \\
&= \left(\mathbf{QRR}'\mathbf{Q}'+\frac{u}{1-u}\mathbf{QQ}'\right)^{-1} \\
&= \mathbf{Q}\left(\mathbf{RR}'+\frac{u}{1-u}\mathbf{I}_p\right)^{-1}\mathbf{Q}' \\
&= \mathbf{Q}\begin{bmatrix}\left(\mathbf{R}_1\mathbf{R}_1'+\frac{u}{1-u}\mathbf{I}_n\right)^{-1} & \mathbf{0}_{n,p-n}\\ \mathbf{0}_{p-n,n} & \frac{1-u}{u}\mathbf{I}_{p-n}\mathbf{Q}'\end{bmatrix}.
\end{aligned} \tag{17}$$

In the same way,

$$\begin{aligned}
\hat{\boldsymbol{\beta}}(u) &= \boldsymbol{\Sigma}_n(u)\mathbf{X}'\mathbf{y} \\
&= \mathbf{Q}\left(\mathbf{RR}'+\frac{u}{1-u}\mathbf{I}_p\right)^{-1}\mathbf{Q}'\mathbf{QRy} \\
&= \mathbf{Q}\left(\mathbf{RR}'+\frac{u}{1-u}\mathbf{I}_p\right)^{-1}\mathbf{Ry} \\
&= \mathbf{Q}_1\left(\mathbf{R}_1\mathbf{R}_1'+\frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{R}_1\mathbf{y},
\end{aligned} \tag{18}$$

and

$$\begin{aligned}\hat{\mathbf{y}}(u) &= \mathbf{X}\hat{\boldsymbol{\beta}}(u)\\ &= \mathbf{R}'\mathbf{Q}'\mathbf{Q}\left(\mathbf{R}\mathbf{R}' + \frac{u}{1-u}\mathbf{I}_p\right)^{-1}\mathbf{R}\mathbf{y}\\ &= \mathbf{R}_1'\left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{R}_1\mathbf{y}.\end{aligned} \tag{19}$$

Hence, by plugging (17) and (18) in (13) is obtained another way to calculate the marginal posterior of u, this is,

$$\begin{aligned}\pi(u\,|\,\mathbf{y}) \;\propto\; & u^{\frac{p_0+p}{2}-1}(1-u)^{\frac{n_0+n}{2}-1}\left|(1-u)\mathbf{R}_1\mathbf{R}_1' + u\mathbf{I}_n\right|^{-1/2}\\ & \times[(1-u)(SSE(u)+n_0s_0^2)]^{-\frac{n+n_0+p_0}{2}}\left[1+\frac{u}{1-u}\frac{\hat{\boldsymbol{\beta}}'(u)\hat{\boldsymbol{\beta}}(u)+p_0d_0^2}{SSE(u)+n_0s_0^2}\right]^{-\frac{n+n_0+p_0}{2}}\end{aligned} \tag{20}$$

when $n > p$. The other form, $u^{\frac{p+p_0}{2}-1}$ is replaced by $u^{\frac{n+p_0}{2}-1}$. Note that the covariance is

$$\mathrm{Cov}\,(\hat{\mathbf{y}}(u), \mathbf{y}\,|\,u) = \sigma^2\mathbf{R}_1'\left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{R}_1.$$

and the variance

$$\mathrm{Var}\,(\hat{\mathbf{y}}(u)\,|\,u) = \sigma^2\mathbf{R}_1'\left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{R}_1\left[\mathbf{R}_1'\left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{R}_1\right]'.$$

Therefore it is possible to calculate the correlation as in the equation (16).
In the same way the effective degrees of freedom are

$$\begin{aligned}edf &= \mathrm{E}\left[\mathrm{tr}\left(\mathbf{R}_1'\left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u}\mathbf{I}_n\right)^{-1}\mathbf{R}_1\right)\middle|\,\mathbf{y}\right]\\ &= \mathrm{E}\left[\mathrm{tr}\left(\mathbf{R}_1\mathbf{R}_1'\left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u}\mathbf{I}_n\right)^{-1}\right)\middle|\,\mathbf{y}\right]\\ &= \int_0^1\left[\mathrm{tr}\left(\mathbf{R}_1\mathbf{R}_1'\left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u}\mathbf{I}_n\right)^{-1}\right)\right]\pi(u\,|\,\mathbf{y})\,\mathrm{d}u.\end{aligned} \tag{21}$$

Therefore the (edf) can be approximated as,

$$\widehat{edf} = \left[\mathrm{tr}\left(\mathbf{R}_1\mathbf{R}_1'\left(\mathbf{R}_1\mathbf{R}_1' + \frac{\hat{u}}{1-\hat{u}}\mathbf{I}_n\right)^{-1}\right)\right], \tag{22}$$

where $\hat{u} = \max\arg\pi(u\,|\,\mathbf{y})$.
Since manipulating high dimensional inverse matrices is complicated and expensive, then when we want to work with the QR method, the posterior mode of u will be used to obtain the values described above. That is, instead of using a vector u, the posterior mode of u will be taken.

3. Results

An package named HDBRR was built in R for implementing the methodology described in the previous sections. This package is available from CRAN at https://CRAN.R-project.org/package=HDBRR. The main function in the package is also named HDBRR and its parameters with their default values are given in the Box 1a.

Box 1a: List of arguments of the HDBRR function

```
HDBRR(y, X, n0 = 5, p0 = 5, s20 = NULL, d20 = NULL, h = 0.5,
    intercept = TRUE, vpapp = TRUE, npts = NULL, c = NULL,
    corpred = NULL, method = c("svd","qr"), bigmat = TRUE, ncores = 2)
```

Inputs

Box 1a displays a list of the main arguments of the **HDBRR** function implemented in the HDBRR package, a short description follows:

- `y, X` are the data vector, generally referred to as the *response variable* vector, `NA`'s allowed, and the design matrix of dimension $n \times p$ respectively.
- `n0, p0` are the numerators of the shape parameters `n0/2` and `p0/2` of the Inverse Gamma prior assigned to the residual variance and the shape parameter of the Inverse Gamma prior assigned to the regression coefficients variance, respectively. The default value for `n0/2` and `p0/2` hyperparameter is `5`.
- `s20, d20, h` are the numerators of scale parameters `(n0s20)/2` and `(p0d20)/2` of the Inverse Gamma prior assigned to the residual variance and the scale parameter of the Inverse Gamma prior assigned to the regression coefficients variance respectively. By default s20 and d20 are NULL which means that these parameters will be set at runtime internally, where `h` is a dimensionless shrinkage coefficient, `0<h<1`. If `h` $\to$ `0` then we have greater shrinkage, that is, $\beta \to$ `0`. If `h` $\to$ `1` then we will have less shrinkage.
- `intercept` Logical argument. Default value is `TRUE`. A model with intercept is fitted.
- `vpapp` Logical argument. Default value is `TRUE`. Computes an approximation of the predictive variance.
- `npts, c` number (integer) of points used to numerical evaluation of the density of u. The default value for the `npts` parameter is `200` and `c` is the ratio of Gaussian densities (Spike/Slab) in the prior mixture for variable selection.
- `corpred` method to compute the correlation between the pairs of predicted and observed. There are two available methods, Empirical Bayes (`"eb"`) and Bayesian method (`"b"`). The default value for the parameter `corpred` is `NULL`. If the value is `NULL` then the output values `corr` and `edf` will be `NULL`.
- `method` Options for the posterior computation. There are two methods available: `"qr"` decomposition of `X*t(X)` and `"svd"` decomposition of matrix `X`. The default value for the method is SVD decomposition.

- `bigmat, ncores` use of the bigstatsr package and number of the cores for computation respectively. The default value for the `ncores` is `2`. The number of cores can be detected with detectCores() and may be used in macOS and Linux systems.

The **values returned** by the **HDBRR** function are listed in Table 1.

Value	
\$betahat $\left(\widehat{\boldsymbol{\beta}}\right)$	(numeric, p) Vector with the beta estimates.
\$yhat $(\widehat{\mathbf{y}})$	(numeric, n) Vector with the y's estimates.
\$sdyhat	(numeric, n) Vector with the standard deviations of the predicted values.
\$sdpred	(numeric, n) Vector with the standard deviation of predicted variances.
\$varb	(numeric, p) Vector with the beta's variance.
\$sigsqhat $(\widehat{\sigma}^2)$	(numeric) estimated residual variance. The "Empirical Bayes" method is used.
\$sigbsqhat $\left(\widehat{\sigma}^2_\beta\right)$	(numeric) estimated variance of beta. The "Empirical Bayes" method is used.
\$u	(numeric, npts) Vector with the values of u.
\$postu $\pi(u \mid \mathbf{y})$	(numeric, npts) Vector with the values of the u posterior.
\$uhat $(\hat{u})$	(numeric) estimated u.
\$umode	(numeric) posterior mode of u.
\$whichNa	(integer) number of NA's in **y**.
\$phat $(\widehat{\boldsymbol{p}})$	(numeric, p) Vector selection probability of x_i.
\$delta (δ)	(numeric) Value used in the variable selection.
\$edf	(numeric) Value of the effective degrees of freedom for regression.
\$corr	(numeric, n) Vector of the correlation between $\hat{y}_i$ and y_i.

Table 1: Values returned by the HDBRR function implemented in the HDBRR package with R software.

Also, the **matop** function was implemented in the HDBRR package, it calculates the SVD and QR decompositions of an **X** matrix (specially if **X** is high dimensional) with the bigstatsr package. The function with its arguments is:

Box 1b: List of arguments of the matop function

```
matop(y, X, method = c("svd", "qr"), bigmat = TRUE)
```

The arguments `y, X, method` and `bigmat` of the **matop** function are the same of the **HDBRR** function. The **values returned** by the **matop** function are listed in Table 2.

Summary, plot and predict functions

For variable selection and visualization of results for class "HDBRR", the package contains the following functions.

(a) SVD

Value	
$y	(numeric, n) Data vector. NAs allowed.
$X	Design matrix of dimension $n \times p$.
$D	Vector containing the singular values of $\mathbf{X}$ of length $\min(n, p)$.
$L	A matrix whose columns contain the left singular vectors of $\mathbf{X}$.
$R	A matrix whose columns contain the right singular vectors of $\mathbf{X}$.
$ev	A vector containing the square of D.
$Ly	The cross-product between the matrix L and vector y.
$n	Number of rows of X.
$p	Number of columns of X.

(b) QR

Value	
$y	(numeric, n) Data vector. NAs allowed.
$X	Design matrix of dimension $n \times p$.
$R	An upper triangular matrix of dimension $p \times p$.
$n	Number of rows of X.
$p	Number of columns of X.
$Q	A matrix of dimension $n \times p$.

Table 2: Values returned by the HDBRR function when the (a) SVD and (b) QR decompositions are used.

Box 1c: List of arguments of the summary, plot and predict functions

```
summary(object, all.coef = FALSE, crit = log(4), ...)

plot(x, crit = log(4), ...)

predict(object,  ...)
```

The **inputs** for these functions are:

- `object` An HDBRR object generated by a call to HDBRR.
- `all.coef` Logical value. If `all.coef = TRUE`, estimations for all the regression parameters are returned; otherwise, only for those whose `2*log(bayes factor)>crit`.
- `crit` Numerical value. The lower bound of the log Bayes factor in favor to include a variable in the model. The default value for `crit` is `log(4)`.
- `...` Additional arguments to be passed to or from other methods.
- `x` An HDBRR object, typically generated by a call to HDBRR (for the `print.HDBRR` and `plot.HDBRR` functions) or an object of class `summary.HDBRR` (for the `print.summary.HDBRR` function).

The **returned values** for the **summary** is a list of estimated coefficients (`Estimate`), their standard deviations (`Std. dev`), the signal-to-noise ratio (`SNR`) and twice the conditional log Bayes Factor en favor to include each covariate (`2ln(BF)`). When `all.coef = TRUE`, then estimations for all ridge regression parameters will be returned, otherwise only for those whose `2ln(BF)>crit`. The default value for `crit` is `log(4)`. The interpretation of `2ln(BF)` is in the "Kass-Raftery" scale (**?**), where

2 log(Bayes Factor)	**Bayes Factor**	**Evidence against H_0**
$(-\infty,0)$	$[0,1)$	Negative
$[0,2)$	$[1,3)$	Weak
$[2,6)$	$[3,20)$	Positive
$[6,10)$	$[20,150)$	Strong
$[10,+\infty)$	$[150,+\infty)$	Very Strong

Table 3: Bayes Factor interpretation using "Kass-Raftery" scale.

Also, the `summary` function returns the estimation for the ridge parameter ($\hat{\lambda}$) and the **effective degrees of freedom** (edf). When the `corpred` argument is `NULL`, then the `edf` value will be `NULL`. If the `c` value is `NULL`, then the Bayes Factor will not be calculated.

The `plot` function returns three graphs: 1) the plot of $\hat{\beta}$ vs $\hat{p}$, 2) a plot with 4 subplots, **predicted values vs observed values**, **coefficients**, **Std. dev.** (for coefficients) and **SNR** and 3) the plot of the **"Marginal Posterior of u"**. CAMBIAR en la siguiente versión

la gráfica 2) de modo que los cuatro gráficos se produzcan individualmente, como en lm.

The `predict` function returns a vector with the predicted values ($\hat{\mathbf{y}}$).

Dataset

The HDBRR package comes with a dataset named **"phenowheat"**, which contains data from a balanced four-way multi-parental cross population from four elite durum wheat cultivars (Neodur, Claudio, Colosseo, and Rascon/Tarro) that were chosen as diverse contributors of different alleles of agronomic relevance. The final population includes 338 recombinant inbred lines (RILs) and the final number of SNPs included in the linkage map was 7594, which were centered and standardized. Phenotypic evaluation of the population was performed during two growing seasons (2010-2011 and 2011-2012) in locations in the Po Valley: Cadriano in the 2010-2011 growing season (Cad11) and the 2011-2012 growing season (Cad12); Poggio Renatico in the 2010-2011 growing season (Pr11) and Argelato in the 2011-2012 growing season (Arg12). The four traits included in this study where grain yield GY (Mg ha^{-1}), heading data HD (d) , 1000-kernel weight GWT (g 1000 kernels^{-1}) and grain volume weight GVW (kg hL^{-1}) . For a detailed description of the data set see Milner *et al.* (2015).

Box 2: Loading the phenowheat data set included in HDBRR

```
library(HDBRR)
data(phenowheat)
```

3.1. Application Example

In order to illustrate the use of the HDBRR package with other real data and to show the outputs obtained, the methods summary, predict and plot are shortly explained

Consider the "phenowheat" data set included in HDBRR. The data are loaded as in the Box 2. In order to obtain the y vector, the lmer function is used. This allow us to find the best linear unbiased predictions of the HD trait after adjusting for the environment (see Box 3b part #1#).

Box 3b: Example with dataset "phenowheat"

```
#1# mod <- lmer(pheno$HD~pheno$env+(1|pheno$Line))
y <- unlist(ranef(mod))

#2# n <- length(y)
X <- scale(X, scale=F)
fitall <- HDBRR(y, X/sqrt(ncol(X)), intercept = FALSE, corpred = "eb",
c = 100)

#3# fitall
summary(fitall, crit = 0)
plot(fitall, crit = 0)
predict(fitall)
```

In second part (#2#) the fit is done by calling the **HDBRR** function where the covariates are scaled to avoid extremely small regression coefficients. In the third part (#3#) the results for `fitall` are returned. If $p > 250$, then only 250 coefficients are displayed.

In Box 3c the structure of the object returned by **HDBRR** function is shown. For this case 250 coefficients were returned, but in the Box only 14 coefficients were printed.

Box 3c: Structure of the object `fitall` returned by HDBRR (after running the code in Box 3b part #3#)

```
> fitall

Call:
HDBRR(y = y, X = X/sqrt(ncol(X)), intercept = F, c = 100, corpred = "eb")

Coefficients:
       X1        X2        X3        X4        X5        X6        X7
-0.069050  0.117750 -0.075723 -0.115526 -0.006621 -0.111297 -0.062642
       X8        X9       X10       X11       X12       X13       X14
-0.088641 -0.096674 -0.114091 -0.063993 -0.082015 -0.114091 -0.107933

 ... 7580 coefficients was omitted
```

The object HDBRR is a list of 21 elements, included betahat ($\hat{\boldsymbol{\beta}}$), yhat ($\hat{\mathbf{y}}$), sigsqhat ($\hat{\sigma}^2$) and sigbsqhat ($\hat{\sigma}^2_{\beta}$). In the `Box 3d` there are some elements contained by an object of the class **HDBRR** .

Box 3d: Structure of the object returned by HDBRR (after running the code in Box 3b)

```
str(fm)
List of 21
 $ betahat      : num [1:7594] -0.06905 0.11775 0.07572 ...
 $ yhat         : num [1:338] -0.394 -0.52 2.876 ...
 $ sdyhat       : num [1:338] 0.643 0.648 0.619 ...
 $ sdpred       : num [1:338] 1.71 1.72 1.7 ...
 $ varb         : num [1:7594] 0.3219 0.8271 0.0511 ...
 $ sigsqhat     : num 2.52
 $ sigbsqhat    : num 6.88
 $ u            : num [1:200] 0.116 0.117 0.119 ...
 $ postu        : Named num [1:200] 2.22e-05 2.73e-05 ...
 ..- attr(*,"names") = chr [1:200]  "84.1344\%" ...
 $ uhat         : num 0.273
 $ umode        : num 0.261
 $ whichNa      : int(0)
 $ phat         : num [1:7594] 3.53e-04 1.02e-03 ...
 $ delta        : num 7.96
 $ edf          : num 140
 $ corr         : num [1:338] 0.804 0.818 0.793 ...
```

The **summary** function was explained above. Here, the `crit` argument was set equal to 0 because the probability of inclusion in very low (the default value for this argument is `log4`).

Box 3e: Summary of the object returned by HDBRR (after running the code in Box 3b part #3#).

```
> summary(fitall, crit = 0)

Call:
HDBRR(y = y, X = X/sqrt(ncol(X)), intercept = FALSE, c = 100,
    corpred = "eb")

Coefficients:
       Estimate  Std. dev       SNR   2ln(BF)
X1189 -3.477229 0.5949053 -5.845014 0.7370653
X1190 -3.411983 0.5887797 -5.795009 0.6232837
X1191 -3.437302 0.6060903 -5.671272 0.6674882
X1192 -3.578898 0.6424279 -5.570895 0.9133567
X1193 -3.595580 0.6084415 -5.909492 0.9422146
X1194 -3.548588 0.5963829 -5.950185 0.8609486
X1195 -3.581506 0.6196334 -5.780039 0.9178934
X1196 -3.376780 0.5930485 -5.693935 0.5616503
X1197 -3.444590 0.6027798 -5.714507 0.6802414
X1198 -3.443699 0.6013025 -5.727066 0.6786374
X1199 -3.395415 0.5896655 -5.758205 0.5942924
X1205 -3.623269 0.6607038 -5.483955 0.9899897
-----
Signif. codes: 10 '***' 6 '**' 2 '*' 0 ' '

Ridge parameter: 0.3754
Effective degrees of freedom: 140.3917
```

Note that the estimation of the ridge parameter returned is $\hat{\lambda} = 0.3754$. The estimated effective degrees of freedom (*edf*) are equal to 140.3917.

Similarly, the `plot` function was discussed above and for this example the results are shown in Figure 2. For variable selection, in the `plot` function the value of the `crit` argument was set to 0, such that in the estimated coefficients vs selection probability plot, the variable names that appear are those for which $2\log(\text{BF}) > 0$ (see Figure **??** (a)).

Figure 2 (b) is the marginal posterior of u.

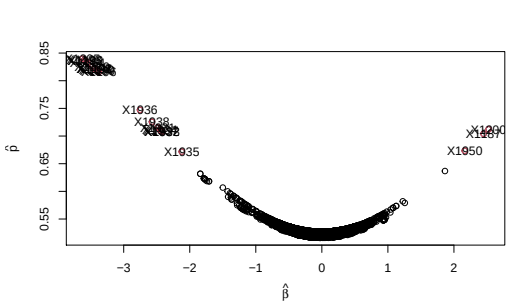
The `predict` function applied to the object HDBRR `fitall` returns the predictions of the model, which are the mean of the posterior predictive distribution. See `Box 3f`

Box 3f: Predict of the object returned by HDBRR (after running the code in Box 3b part #3#).

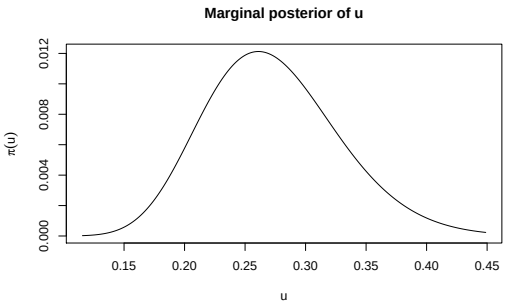
```
> predict(fitall)

  [1] -0.39403 -0.52017  2.87568  1.51151  0.11663  0.70171  0.12358  ...
 [10] -2.15284  0.62729 -2.25787  2.89244  1.41203  0.50038 -0.44047  ...
 [19]  1.90388 -2.76213 -1.91036  2.99180  3.89731  1.10541  1.47814 ...
 [28]  0.38303 -1.05772 -0.47791 -0.91379 -1.00456 -0.85098 -1.25591  ...
```

Finally, the Figure 2 (c) has 4 subplots, predicted values vs observed values, coefficients, Std. dev. (for coefficients) and SNR. The first subplot (predicted values vs observed values) include the Spearman correlation, for this case this value is 0.9196.

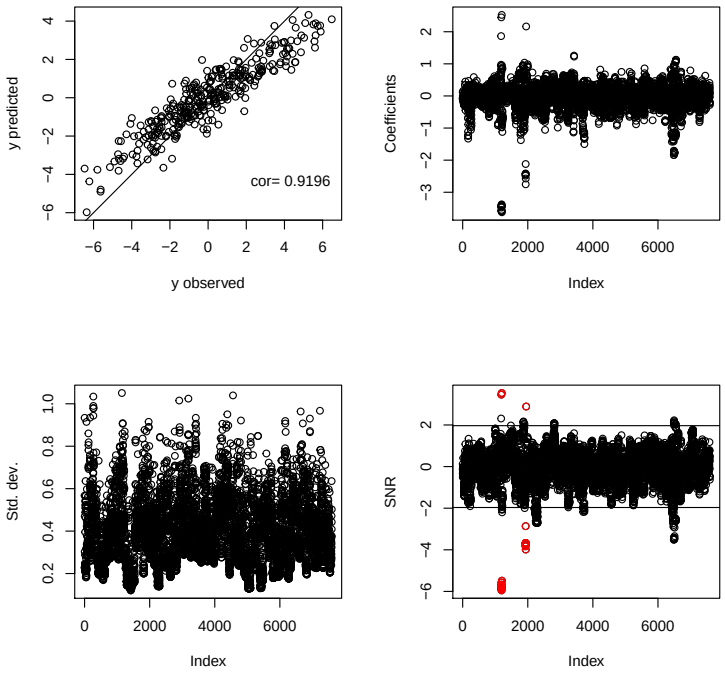
It is mentioned that the ability of MCMCs to sample from modern-day high-dimensional posteriors has been limited by a widespread perception that these chains typically experience serious convergence problems. Figure 2 (b) shows evidence on the advantage of the HDBRR over MCMC, the posterior distribution is always calculated.



(a) Estimated coefficients vs selection probability of x_i.



(b) Marginal posterior of u.



(c) **y** observed vs **y** predicted and their correlation, Coefficients, td.dev. and SNR.

Figure 2: Plots returned for the object `fitall` produced by the **HDBRR** function.

4. Conclusions

We propose a computational method to make Bayesian inference for high-dimensional ridge regression without using MCMC methods. Posterior means and variances of regression parameters, variance components and predictions for the conventional ridge Regression model are obtained by using a convenient reparameterization. The problem is reduced to numerical integration on the open interval $(0, 1)$ to get rid of a nuisance parameter, after SVD or QR decomposition of the matrix $\mathbf{X}'\mathbf{X}$. The method is implemented in the R package HDBRR, which allows us also to make also variable selection and prediction without appealing the theoretical guarantees of MCMC methods. The results of cross validation shown that the proposed method has a performance in computation time and accuracy at least as good as the results obtained by using MCMC methods.

References

Alheety M, Kibria BMG (2011). "Choosing ridge Parameters in the Linear regression Model with AR(1) Error: A comparative Simulation Study." *International Journal of Statistics and Economics*, **7**(A11).

Andrieu C, de Freitas N, Doucet A, Jordan MI (2003). "An Introduction to MCMC for Machine Learning." *Machine Learning*, **50**(1), 5–43.

Cannon A (2009). "Negative ridge regression parameters for improving the covariance structure of multivariate linear downscaling models." *International Journal of Climatology*, **29**(5), 761–769.

Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, BRubin D (2021). "Bayesian Data Analysis." Electronic. URL https://users.aalto.fi/~ave/BDA3.pdf.

George EI, McCulloch RE (1993). "Variable Selection Via Gibbs Sampling." *Journal of the American Statistical Association*, **88**(423), 881–889.

Gilks WR, Richardson S, Spiegelhalter D (1996). *Markov Chain Monte Carlo in Practice*. CHAPMAN & HALL/CRC.

Goeman J, Meijer R, Chaturvedi N, Lueder M (2021). *penalized: L1 (Laso and Fused Lasso) and L2 (Ridge) Penalized Estimation in GLMs and in the Cox Model*. URL https://cran.r-project.org/web/packages/penalized/index.html.

Guan Y, Stephens M (2011). "Bayesian Variable Selection Regression for Genome-Wide Association Studies and other Large-Scale problems." *The Annals of Applied Statistics*, **5**, 1780–1815.

Hoerl AE (1962). "Application of ridge analysis to regression problems." *Chemical Engineering Progress*, **58**, 54–59.

Hoerl AE, Kennard RW (1968). "On regression analysis and biased estimation." *Technometrics*, **10**, 422–423.

Hoerl AE, Kennard RW (1970a). "Ridge Regression: Applications to Nonorthogonal Problems." *Technometrics*, **12**(1), 69–82.

Hoerl AE, Kennard RW (1970b). "Ridge regression: Biased estimation for nonorthogonal problems." *Technometrics*, **12**(1), 55–67.

Hoerl AE, Kennard RW, Baldwin KF (1975). "Ridge regression: Some simulation." *Communications in Statistics*, **4**, 105–123.

Lee PM (2012). *Bayesian Statistics An Introduction*. Fourth edition. Wiley.

Milner SG, Maccaferri M, Huang BE, Mantovani P, Massi A, Frascaroli E, Tuberosa R, Salvi S (2015). "A multiparental cross population for mapping QTL for agronomic traits in durum wheat (Triticum turgidum ssp. durum)." *Plant Biotechnology Journal*, **14**, 735–748.

Moritz S, Cule E, Frankowski D (2021). *ridge: Ridge Regression with Automatic Selection of the Penalty Parameter*. URL https://CRAN.R-project.org/package=ridge.

Pérez P, de los Campos G (2016). *BGLR: A Statistical Package for Whole Genome Regression and Prediction*. R package version 1.0.8, URL https://CRAN.R-project.org/package=BGLR.

Rajaratnam B, Sparks D (2015). "MCMC-Based Inference in the Era of Big Data: A Fundamental Analysis of the Convergence Complexity of High-Dimensional Chains." 1508.00947.

Reich BJ, Ghosh SK (2019). *Bayesian Statistical Methods*. A CHAPMAN & HALL BOOK.

Robert CP, Casella G (2010). *Introducing Monte Carlo Methods with R*. Springer.

Speagle JS (2020). "A Conceptual Introduction to Markov Chain Monte Carlo Methods."

van Wieringen WN (2015). "Lecture notes on ridge regression."

Yahya WB, Olaifa JB (2014). "A note on ridge regression modeling techniques." *Electronic Journal of Applied Statistical Analysis*, **07**(02), 343–361.

Affiliation:

Sergio Pérez-Elizalde
Socio Economía Estadística e Informática
Colegio de Postgraduados, México
E-mail: sergiop@colpos.mx

Blanca Monroy-Castillo
Socio Economía Estadística e Informática
Colegio de Postgraduados, México
E-mail: blancamonroy.96@gmail.com

Paulino Pérez-Rodríguez
Socio Economía Estadística e Informática
Colegio de Postgraduados, México
E-mail: perpdgo@colpos.mx