

HDBRR: A Statistical Package for a High Dimension Ridge Regression without MCMC

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

Abstract

Ridge regression is a useful tool to deal with collinearity 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

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

where ϵ_i , $i = 1, \dots, n$, 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 , $i = 1, \dots, n$, 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

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad (1)$$

where $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)'$ is the vector of *regression parameters*, $\boldsymbol{\epsilon} = (\epsilon_1, \dots, \epsilon_n)'$ 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 $p \gg n$, 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:

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

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 joint 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 , $j = 1, \dots, p$, is Gaussian with mean 0 and variance σ_β^2 and prior independence of $\boldsymbol{\beta}$ and σ^2 is assumed then the joint prior is of the form

$$\pi(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2) = \pi(\boldsymbol{\beta} \mid \sigma_\beta^2) \pi(\sigma^2) \pi(\sigma_\beta^2), \quad (4)$$

where,

$$\pi(\boldsymbol{\beta} \mid \sigma_\beta^2) = \mathcal{N}_p(\boldsymbol{\beta} \mid \mathbf{0}, \sigma_\beta^2 \mathbf{I}_p)$$

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

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

where n_0, s_0^2, p_0 and d_0^2 are known hyperparameters. The prior independence of the elements of $\boldsymbol{\beta}$ given σ_β^2 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 appropriated structure in presence of colineallity or when $p \gg n$. The prior variance of the elements of $\boldsymbol{\beta}$ has also an interpretation in terms of regularization: for fixed σ^2 , as σ_β^2 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_β^2 . Thus, the prior expected values of σ^2 and σ_β^2 are

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

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

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

Then, by noting that

$$\mathbb{E} \left[\left(\sum_{j=1}^p x_{ij} \beta_j \right)^2 \middle| \sigma^2, \sigma_\beta^2 \right] = \text{Var} \left[\sum_{j=1}^p x_{ij} \beta_j \middle| \sigma^2, \sigma_\beta^2 \right] = \sigma_\beta^2 \sum_{j=1}^p x_{ij}^2$$

we have that

$$\mathbb{E}[\text{PVE}] = \mathbb{E}[\mathbb{E}[\text{PVE} | \sigma^2, \sigma_\beta^2]] = \frac{1}{s_0^2} \frac{p_0 d_0^2}{p_0 - 1} \sum_{i=1}^n \sum_{j=1}^p \frac{x_{ij}^2}{n}.$$

Let

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

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

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

therefore

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

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_β^2 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,

$$\pi(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2 | \mathbf{y}) \propto L(\boldsymbol{\beta}, \sigma^2 | \mathbf{y}) \pi(\boldsymbol{\beta}, \sigma^2, \sigma_\beta^2),$$

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

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

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

$$\pi(\beta | u, v, \mathbf{y}) = \mathcal{N}_p \left(\beta \mid \hat{\beta}(u), \frac{1}{v(1-u)}\Sigma_n(u) \right). \quad (6)$$

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

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

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

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

and variance

$$\mathbb{V}[\beta | u, \mathbf{y}] = \frac{S(u)}{(\nu - 2)(1 - u)}\Sigma_n(u), \quad \nu - 2 > 0.$$

Finally, the marginal posterior of u is obtained as

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

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

$$\mathbb{E}[\beta | \mathbf{y}] = \mathbb{E}[\mathbb{E}[\beta | u, \mathbf{y}]] = \int_0^1 \hat{\beta}(u) \pi(u | \mathbf{y}) du$$

and

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

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 σ_β^2 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,

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

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 $\lambda = \sigma^2/\sigma_\beta^2 = u/(1-u)$ is given by

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

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

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

and

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

2.5. Variable Selection

Suppose that the prior density for β is such that,

$$\pi(\beta_j | \sigma_\beta^2, \gamma_j) = (1 - \gamma_j)N(\beta_j | 0, \sigma_\beta^2) + \gamma_j N(\beta_j | 0, c_j^2 \sigma_\beta^2), \quad j = 1, \dots, p.$$

Where $\gamma_j \stackrel{iid}{\sim} \text{Bernoulli}(\phi_j)$ 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_β^2 and $c^2 \sigma_\beta^2$ are set “small and large”, respectively, so that $\mathcal{N}(0, \sigma_\beta^2)$ is concentrated around 0 and $\mathcal{N}(0, c_j^2 \sigma_\beta^2)$ 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_β^2 and $c_j^2 \sigma_\beta^2$ such that the pdf $\pi(\beta_j | \gamma_j = 0) = \mathcal{N}(\beta_j | 0, \sigma_\beta^2)$ is larger than the pdf $\pi(\beta_j | \gamma_j = 1) = \mathcal{N}(\beta_j | 0, c_j^2 \sigma_\beta^2)$ on the interval $(-\delta_j, \delta_j)$ (see Figure 1). This condition is satisfied for any σ_β and c_j such that

$$\frac{\log\left(\frac{c_j^2 \sigma_\beta^2}{\sigma_\beta^2}\right)}{\frac{1}{\sigma_\beta^2} - \frac{1}{c_j^2 \sigma_\beta^2}} \leq \delta_j^2. \quad (9)$$

Hence,

$$\delta_j = \sqrt{\frac{2c_j^2 \sigma_\beta^2 \log(c_j)}{c_j^2 - 1}}, \quad c_j > 1.$$

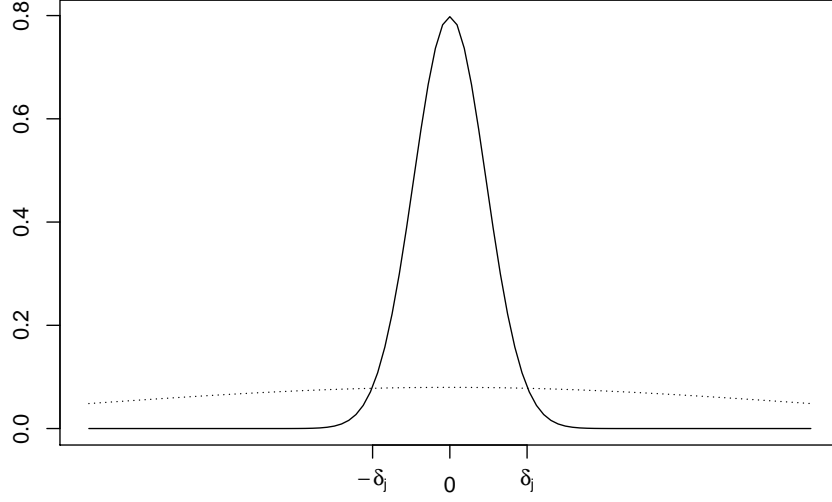


Figure 1: $\mathcal{N}(0, \sigma_\beta^2)$ and $\mathcal{N}(0, c_j^2 \sigma_\beta^2)$ densities. Intersection at δ_i

For the selection of c_j , note that it is equal to the ratio of $\pi(\beta_j = 0 | \gamma_j = 0)$ and $\pi(\beta_j = 0 | \gamma_j = 1)$; thus, c_j may be interpreted as the prior odds that X_j should be excluded of the model if β_j is too 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 $j = 1, \dots, p$, $\phi_j = \phi$ 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 $\gamma = (\gamma_1, \dots, \gamma_p)$ is given by

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

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

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

Then,

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

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

$$\pi(\boldsymbol{\gamma} \mid \mathbf{y}) = \pi(\gamma_j \mid \boldsymbol{\gamma}_{-j}, \mathbf{y}) \pi(\boldsymbol{\gamma}_{-j} \mid \mathbf{y}),$$

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

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

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

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

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

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

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 the 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:

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

where, $V_j = \left(\sum_{i=1}^n x_{ij}^2 + \frac{\tilde{u}}{1 - \tilde{u}} \right)^{-1}$ and $E_j = V_j \mathbf{X}_j' (\mathbf{y} - \mathbf{X}_{-j} \tilde{\boldsymbol{\beta}}_{-j})$ 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 to include the covariate X_j in the model is

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

2.6. Posterior Computation

The main feature of the parametrization 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 recursive methods. In fact, once the posterior distribution of u is calculated.

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, $s_1 \geq \dots \geq s_n$ of \mathbf{X} and the last $p - n$ columns are all vectors of zeros.

Hence

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

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

Similarly,

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

And

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

where \mathbf{P} is a diagonal matrix with $\mathbf{P}_{jj} = \frac{(1-u)\lambda_j}{(1-u)\lambda_j+u}$, $j = 1, \dots, n$.

So, substituting in (8) is obtained,

$$\begin{aligned}
\pi(u | \mathbf{y}) &\propto u^{\frac{p_0+p}{2}-1} (1-u)^{\frac{n_0+n}{2}-1} |(1-u)\mathbf{X}'\mathbf{X} + u\mathbf{I}|^{-1/2} [(1-u)(SSE(u) + n_0 s_0^2)]^{-\frac{n+n_0+p_0}{2}} \\
&\quad \times \left[1 + \frac{u}{1-u} \frac{\hat{\beta}'(u)\hat{\beta}(u) + p_0 d_0^2}{SSE(u) + n_0 s_0^2} \right]^{-\frac{n+n_0+p_0}{2}} \\
&\propto u^{\frac{p_0+p}{2}-1} (1-u)^{\frac{n_0+n}{2}-1} (1-u)^{-\frac{p}{2}} \left| \begin{array}{cc} \mathbf{\Lambda} + \frac{u}{1-u} \mathbf{I}_n & \mathbf{0}_{n,p-n} \\ \mathbf{0}_{p-n,p} & \frac{u}{1-u} \mathbf{I}_{p-n} \end{array} \right|^{1/2} \\
&\quad \times [(1-u)(SSE(u) + n_0 s_0^2)]^{-\frac{n+n_0+p_0}{2}} \left[1 + \frac{u}{1-u} \frac{\hat{\beta}'(u)\hat{\beta}(u) + p_0 d_0^2}{SSE(u) + n_0 s_0^2} \right]^{-\frac{n+n_0+p_0}{2}} \\
&\propto u^{\frac{p+p_0}{2}-1} (1-u)^{\frac{n+n_0}{2}-1} u^{\frac{n-p}{2}} \left(\prod_{j=1}^n (1-u)\lambda_j + u \right)^{-\frac{1}{2}} \\
&\quad \times [(1-u)(SSE(u) + n_0 s_0^2)]^{-\frac{n+n_0+p_0}{2}} \left[1 + \frac{u}{1-u} \frac{\hat{\beta}'(u)\hat{\beta}(u) + p_0 d_0^2}{SSE(u) + n_0 s_0^2} \right]^{-\frac{n+n_0+p_0}{2}}, \tag{13}
\end{aligned}$$

where

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

Note that the equation (13) is used when $n > p$. In other case, $u^{\frac{p+p_0}{2}-1}$ is replaced with $u^{\frac{n+p_0}{2}-1}$.

Also note that

$$\text{Cov}(\hat{\mathbf{y}}(u), \mathbf{y} | u) = \text{Cov}(\mathbf{U}\mathbf{P}\mathbf{U}'\mathbf{y}, \mathbf{y} | u) = \mathbf{U}\mathbf{P}\mathbf{U}'\text{Cov}(\mathbf{y}) = \sigma^2\mathbf{U}\mathbf{P}\mathbf{U}'$$

and

$$\text{Var}(\hat{\mathbf{y}}(u) | u) = \sigma^2\mathbf{U}\mathbf{P}\mathbf{U}',$$

hence

$$\text{cor}(\hat{y}_i, y_i) = \int_0^1 \frac{\sum_{j=1}^r u_{i,j}^2 \mathbf{P}_{jj}}{\sqrt{\sum_{j=1}^r u_{i,j}^2 \mathbf{P}_{jj}^2}} \pi(u | \mathbf{y}) du. \tag{15}$$

For ridge regression the degrees of freedom are commonly calculated by the trace of the matrix that transforms the vector of observations on the dependent variable into the ridge regression estimate of its expected value; that is:

$$edf = \text{E}[\text{tr}(\mathbf{U}\mathbf{P}\mathbf{U}' | \mathbf{y})] = \text{E}[\text{tr}(\mathbf{P} | \mathbf{y})] = \int_0^1 \sum_{j=1}^p \mathbf{P}_{jj} \pi(u | \mathbf{y}) du.$$

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 $\mathbf{X}' = \mathbf{Q}\mathbf{R}$, 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,

$$\begin{aligned} \Sigma_n(u) &= \left(\mathbf{X}'\mathbf{X} + \frac{u}{1-u} \mathbf{I}_p \right)^{-1} \\ &= \left(\mathbf{Q}\mathbf{R}\mathbf{R}'\mathbf{Q}' + \frac{u}{1-u} \mathbf{Q}\mathbf{Q}' \right)^{-1} \\ &= \mathbf{Q} \left(\mathbf{R}\mathbf{R}' + \frac{u}{1-u} \mathbf{I}_p \right)^{-1} \mathbf{Q}' \\ &= \mathbf{Q} \begin{bmatrix} \left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u} \mathbf{I}_n \right)^{-1} & \mathbf{0}_{n,p-n} \\ \mathbf{0}_{p-n,n} & \frac{1-u}{u} \mathbf{I}_{p-n} \mathbf{Q}' \end{bmatrix}. \end{aligned} \quad (16)$$

In the same way,

$$\begin{aligned} \hat{\beta}(u) &= \Sigma_n(u) \mathbf{X}'\mathbf{y} \\ &= \mathbf{Q} \left(\mathbf{R}\mathbf{R}' + \frac{u}{1-u} \mathbf{I}_p \right)^{-1} \mathbf{Q}' \mathbf{Q}\mathbf{R}\mathbf{y} \\ &= \mathbf{Q} \left(\mathbf{R}\mathbf{R}' + \frac{u}{1-u} \mathbf{I}_p \right)^{-1} \mathbf{R}\mathbf{y} \\ &= \mathbf{Q}_1 \left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u} \mathbf{I}_n \right)^{-1} \mathbf{R}_1\mathbf{y}, \end{aligned} \quad (17)$$

and

$$\begin{aligned} \hat{\mathbf{y}}(u) &= \mathbf{X}\hat{\beta}(u) \\ &= \mathbf{R}'\mathbf{Q}'\mathbf{Q} \left(\mathbf{R}\mathbf{R}' + \frac{u}{1-u} \mathbf{I}_p \right)^{-1} \mathbf{R}\mathbf{y} \\ &= \mathbf{R}_1' \left(\mathbf{R}_1\mathbf{R}_1' + \frac{u}{1-u} \mathbf{I}_n \right)^{-1} \mathbf{R}_1\mathbf{y}. \end{aligned} \quad (18)$$

Hence, by plugging (16) and (17) in (13) is obtained another way to calculate the marginal posterior of u , this is,

$$\begin{aligned} \pi(u | \mathbf{y}) &\propto u^{\frac{p_0+p}{2}-1} (1-u)^{\frac{n_0+n}{2}-1} u^{\frac{n-p}{2}} |(1-u)\mathbf{R}_1\mathbf{R}_1' + u\mathbf{I}_n|^{-1/2} \\ &\quad \times [(1-u)(SSE(u) + n_0 s_0^2)]^{-\frac{n+n_0+p_0}{2}} \left[1 + \frac{u}{1-u} \frac{\hat{\beta}'(u)\hat{\beta}(u) + p_0 d_0^2}{SSE(u) + n_0 s_0^2} \right]^{-\frac{n+n_0+p_0}{2}} \end{aligned} \quad (19)$$

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

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

and the variance

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

Therefore it is possible to calculate the correlation as in the equation (15).

In the same way the effective degrees of freedom are

$$\begin{aligned} edf &= \mathbb{E} \left[\text{tr} \left(\mathbf{R}'_1 \left(\mathbf{R}_1 \mathbf{R}'_1 + \frac{u}{1-u} \mathbf{I}_n \right)^{-1} \mathbf{R}_1 \right) \middle| \mathbf{y} \right] \\ &= \mathbb{E} \left[\text{tr} \left(\mathbf{R}_1 \mathbf{R}'_1 \left(\mathbf{R}_1 \mathbf{R}'_1 + \frac{u}{1-u} \mathbf{I}_n \right)^{-1} \right) \middle| \mathbf{y} \right] \\ &= \int_0^1 \left[\text{tr} \left(\mathbf{R}_1 \mathbf{R}'_1 \left(\mathbf{R}_1 \mathbf{R}'_1 + \frac{u}{1-u} \mathbf{I}_n \right)^{-1} \right) \right] \pi(u | \mathbf{y}) du. \end{aligned} \quad (20)$$

Therefore the (edf) can be approximated as,

$$\widehat{edf} = \left[\text{tr} \left(\mathbf{R}_1 \mathbf{R}'_1 \left(\mathbf{R}_1 \mathbf{R}'_1 + \frac{\hat{u}}{1-\hat{u}} \mathbf{I}_n \right)^{-1} \right) \right], \quad (21)$$

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 \mathbf{u} will be used to obtain the values described above. That is, instead of using a vector \mathbf{u} , the posterior mode of \mathbf{u} will be taken.

3. Results

With this information we can build the package **HDBRR** (Pérez-Elizalde *et al.* 2021) in R, the HDBRR function is,

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)
```

also we have a second function, the matop function, this help us to compute the SVD or QR decomposition, the matop form is

Box 1b: List of arguments of the matop function

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

3.1. Application Example

For this, we will use a database in the package “phenowheat”, this contain data from a balanced, four-way multiparental 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 cultivars were crossed pair-wise following the scheme $[(\text{Neodur} \times \text{Claudio}) \times (\text{Colosseo} \times \text{Rascon/Tarro})]$, i.e., NCCR) to produce two-way F_1 NCCR hybrids. These four-way F_1 hybrids were advanced through single-seed descent and bulked in the F_8 generation. The final NCCR population includes 338 recombinant inbred lines (RILs) (Milner *et al.* 2015). This population is representative of segregating populations commonly used in wheat breeding, where the three- and four-way cross scheme is increasingly adopted to generate wider genotypic variance in segregating populations as compared with the traditional biparental cross (Crossa *et al.* 2016).

The final number of SNPs included in the NCCR linkage map was 7594. The markers were centered and standardized. Phenotypic evaluation of the NCCR population was performed during two growing seasons (2010-2011 and 2011-2012) in locations in the Po Valley representative of the target environments where durum wheat is grown: 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 338 RILs, the four parents, and the five control genotypes were evaluated in an α -lattice incomplete-block experimental design; a 19 by 19 α -lattice design with two replications was considered in each environment. The four traits included in this study were GY (Mg ha⁻¹), HD (d), GWT (g 1000 kernels⁻¹), and GVW (kg hL⁻¹) where GY is the grain yield, HD the heading data, GWT the 1000-kernel weight and GVW the grain volume weight. The phenotypes of the four traits used for data analysis were the best linear unbiased estimates

after recovering the incomplete block information (i.e., adjusting for the random incomplete block effect) in each environment and across environments (Crossa *et al.* 2016).

The next code shows the HDBRR function with this data, y is obtained with the lmer function, this allows us to find the BLUP of the line with respect to the environment.

Box 1c: Example with dataset "phenowheat"

```
data("phenowheat")
mod <- lmer(pheno$HD~pheno$env+(1|pheno$Line))
y <- unlist(ranef(mod))
n <- length(y)
X <- scale(X, scale=F)
fitall <- HDBRR(y, X/sqrt(ncol(X)), intercept = FALSE, corpred = "eb", c = 100)
summary(fitall)
plot(fitall)
```

The code in the Box 1c return the results for fitall, this shows us the coefficients obtained. If we have $p > 250$, the object shows only 250 coefficients.

Box 1c: Structure of the object fitall returned by HDBRR (after running the code in Box 1c)

```
> 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
```

In the same way, the object HDBRR returns a list of 21 elements, included betahat ($\hat{\beta}$), yhat (\hat{y}), sigsqhat ($\hat{\sigma}^2$) and sigbsqhat ($\hat{\sigma}_{\beta}^2$).

Box 1d: Structure of the object returned by HDBRR (after running the code in Box 1c)

```
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] 0.116 0.117 0.119 ...
 ..- attr(*,"names") = chr [1:200] "84.1344\%" ...
 $ uhat         : num 0.273
 $ umode        : num 0.261
 $ whichNa      : int(0)
 $ phat         : num [1:7594] 0.52 0.52 0.521 ...
 $ delta        : num 7.96
 $ edf          : num 140
 $ corr         : num [1:338] 0.804 0.818 0.793 ...
 $ y            : num [1:338] 0.0228 -1.5949 2.7807 ...
 $ intercept    : num 0
```

Now, we have the option for a summary, this function returns the **estimate**, **standard deviation**, **SNR** and $2 \log BF$ for the coefficients, in the principal function we have the argument `c`, when `c = NULL` then the summary don't have the last column ($2 \log BF$). Even more, we have the arguments `all.coef` and `crit`, when we have `all.coef = TRUE` then summary returns all coefficients, but when we have `all.coef = FALSE` (the default value) summary only returns the coefficients that $\log(\text{bayes factor}) > \text{crit}$ where the default value for `crit` is $\log(4)$. the summary returns the **ridge parameter** (λ) and the **effective degrees of freedom** (`edf`) too, when `c = NULL`, `edf = NULL`.

The form of the summary is in the Box 1e.

Box 1e: Summary of the object returned by HDBRR (after running the code in Box 1c) using the default values

```
> summary(fitall)
```

Call:

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

Coefficients:

	Estimate	Std. dev	SNR	2ln(BF)	
X1187	2.448331	0.7079706	3.458238	1.733250	
X1189	-3.477229	0.5949053	-5.845014	3.105383	*
X1190	-3.411983	0.5887797	-5.795009	3.020875	*
X1191	-3.437302	0.6060903	-5.671272	3.056947	*
X1192	-3.578898	0.6424279	-5.570895	3.252233	*
X1193	-3.595580	0.6084415	-5.909492	3.273802	*
X1194	-3.548588	0.5963829	-5.950185	3.206101	*
X1195	-3.581506	0.6196334	-5.780039	3.257444	*
X1196	-3.376780	0.5930485	-5.693935	2.971416	*
X1197	-3.444590	0.6027798	-5.714507	3.063753	*
X1198	-3.443699	0.6013025	-5.727066	3.062512	*
X1199	-3.395415	0.5896655	-5.758205	2.995950	*
X1200	2.524250	0.7140561	3.535086	1.811552	
X1205	-3.623269	0.6607038	-5.483955	3.312073	*
X1931	-2.484760	0.6701858	-3.707569	1.845033	
X1932	-2.422271	0.6321472	-3.831815	1.774233	
X1933	-2.417706	0.6402461	-3.776214	1.769846	
X1934	-2.452866	0.6641450	-3.693269	1.807623	
X1935	-2.120806	0.7404535	-2.864199	1.433109	
X1936	-2.752589	0.7502858	-3.668721	2.168280	*
X1937	-2.437252	0.6639373	-3.670907	1.790789	
X1938	-2.572225	0.6453258	-3.985932	1.946329	
X1950	2.164019	0.7499884	2.885403	1.447961	

Signif. codes: 10 '***' 6 '**' 2 '*' 0 ' ' '

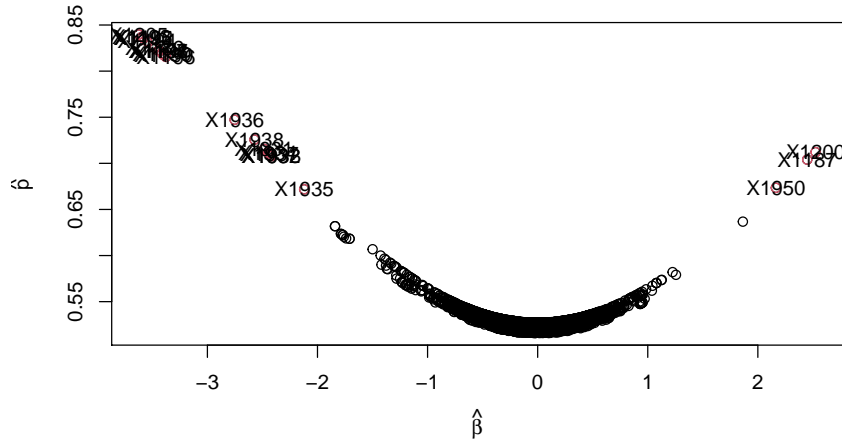
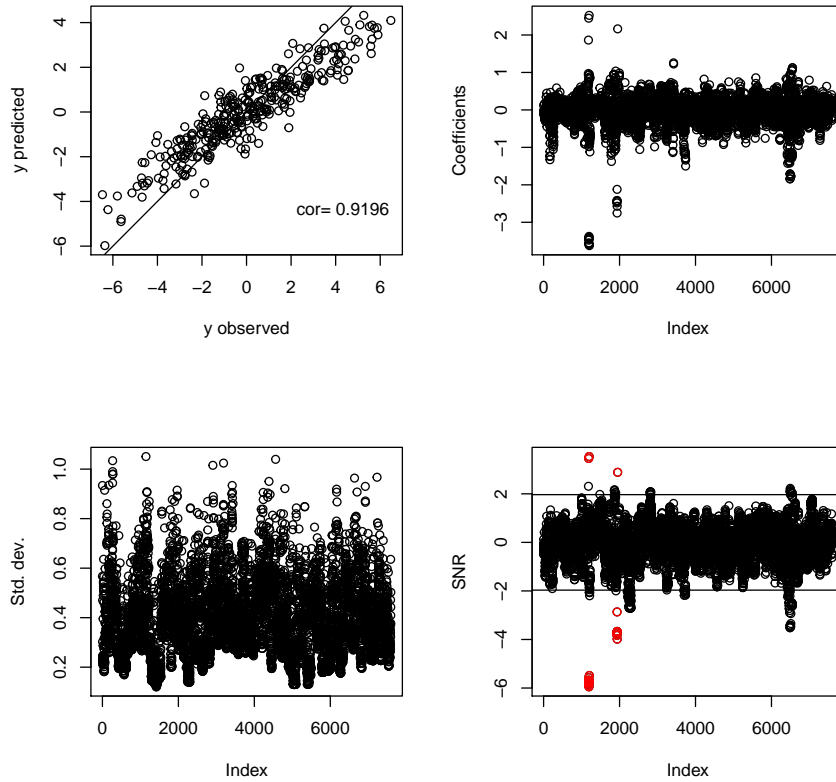
Ridge parameter: 0.3754

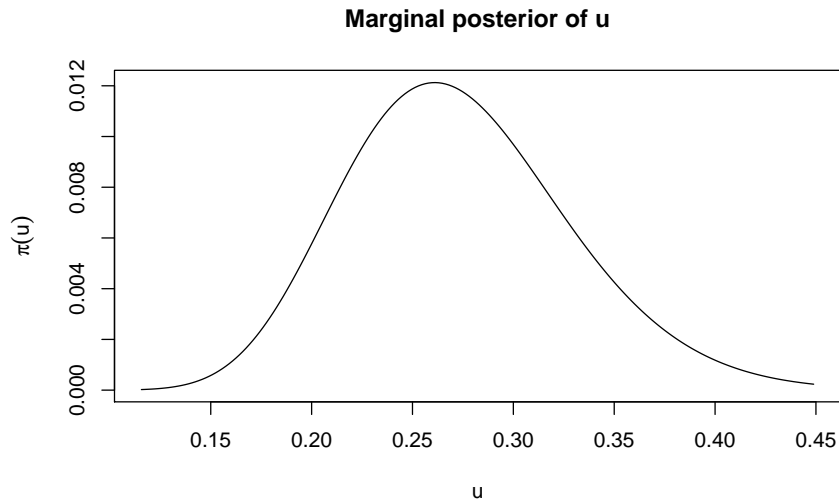
Effective degrees of freedom: 140.3917

The other function implemented for a package HDBRR is `plot`, this returns the variable selection graphs, this use the argument `crit = log(4)` as default value. The graphs “y observed vs y predicted”, “coefficients”, “Std.dev.” and “SNR” are included too. Finally we have the graph of “Marginal posterior of u”. Then with `plot(fitall)` we obtain Figures 2, 3 and 4.

Finally we have the option `predict` for a HDBRR object, this returns the vector of predicted values.

One option to study the results obtained with the package is to use the cross validation, in

Figure 2: Estimate coefficients vs selection probability of x_i .Figure 3: y observed vs y predicted, Coefficients, td.dev. and SNR.

Figure 4: Marginal posterior of u .

this case we use a K-Fold cross validation with $K = 10$, and we obtain the results in the Table 1.

	MSE	Correlation
Fold = 1	4.7367	0.6752
Fold = 2	5.9848	0.3813
Fold = 3	3.533	0.5907
Fold = 4	4.772	0.6583
Fold = 5	3.9884	0.6657
Fold = 6	5.8835	0.5838
Fold = 7	3.4608	0.5146
Fold = 8	3.6765	0.6828
Fold = 9	5.1121	0.553
Fold = 10	2.7146	0.6892
Pooled	4.3862	0.5995

Table 1: K-Fold cross validation with $K=10$

4. Conclusions

In the present work, we can note that it is possible to work high-dimensional ridge regression without the need to use MCMC methods, in addition, the implementation was done in the R software, which allowed us to observe that the computational cost was reduced and with this the time decreased. The results obtained with the implemented package showed that the fit was “good”. In addition, this implementation gives the opportunity to guarantee the convergence of the posterior distribution, moreover, it gives us an idea of what this is like. The small MSE in the cross validation indicates good predictions are obtained. Despite

everything, more studies can be carried out with the data set, it should be remembered that some data may be sensitive to the chosen prior distributions, so it is recommended to carry out a sensitivity study with the given hyperparameters, in this case they were used the default arguments.

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.
- Crossa J, de los Campos G, Maccaferri M, Tuberosa R, Burgueño J, Pérez-Rodríguez P (2016). “Extending the Marker x Environment Interaction Model for Genomic-Enabled Prediction and Genome-Wide Association Analysis in Durum Wheat.” *Crop Science*, **56**, 2193–2209.
- 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 (Lasso 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>.
- Pérez-Elizalde S, Monroy-Castillo B, Pérez-Rodríguez P (2021). *HDBRR: High Dimensional Bayesian Ridge Regression without MCMC, URL*. R package version 1.1.0, URL <https://CRAN.R-project.org/package=HDBRR>.
- 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