Methods and Tools for Bayesian Variable Selection and Model Averaging in Normal Linear Regression

Anabel Forte,

Department of Statistics and Operations research, University of Valencia

Gonzalo Garcia-Donato*

Department of Economics and Finance, University of Castilla-La Mancha

and Mark F.J. Steel

Department of Statistics, University of Warwick

Abstract

In this paper we briefly review the main methodological aspects concerned with the application of the Bayesian approach to model choice and model averaging in the context of variable selection in regression models. This includes prior elicitation, summaries of the posterior distribution and computational strategies. We then examine and compare various publicly available R-packages, summarizing and explaining the differences between packages and giving recommendations for applied users. We find that all packages reviewed (can) lead to very similar results, but there are potentially important differences in flexibility and efficiency of the packages.

Motivation

A very common problem in statistics is where several statistical models are proposed as plausible descriptions for certain observations $\boldsymbol{y}$ and the observed data are used to resolve the model uncertainty.

*Corresponding author: email: gonzalo.garciadonato@uclm.es

This problem is normally known as *model selection* or *model choice* if the aim is to select a single "best" model, but if the model uncertainty is to be formally reflected in the inferential process, we typically use *model averaging*, where inference on issues that are not model-specific (such as prediction or effects of covariates) is averaged over the set of models under consideration.

A particularly important model uncertainty problem in practice is *variable selection* where the proposed models share a common functional form (e.g. a normal linear regression model) but differ in which explanatory variables, from a given set, are included to explain the response. The focus in this paper will be on variable selection in the context of normal linear models, a problem frequently encountered in practice and formally introduced in Section .

Model uncertainty is a classic problem in statistics that has been scrutinized from many different perspectives. Hence, quite often, the main issues for practitioners are to decide which methodology to use and/or how to implement the methodology in practice. One appealing approach is based on the Bayesian paradigm and is considered by many the *formal* Bayesian answer to the problem. This approach is the one based on the posterior probabilities of the models under consideration and results in a coherent and complete analysis of the problem which provides answers to practical questions. For instance, a single model can be selected as that most supported by the data (the model with the highest posterior probability) or inference can be performed using the posterior model probabilities as weights, normally denoted by Bayesian model averaging (BMA). In this paper we describe how the formal Bayesian method can be implemented in `R` (R Core Team, 2015), analyzing the different packages that are currently available in CRAN (`cran.r-project.org`). Emphasis is placed on comparison but also on putting in perspective the details of the implementations.

As with any Bayesian method, the prior distribution for the unknown parameters needs to be specified. It is well known that this aspect is particularly critical in model uncertainty problem since results are potentially highly sensitive to the priors used (see e.g. Berger and Pericchi, 2001; Ley and Steel, 2009). In this paper, we pay special attention to the family of priors in the tradition started by Jeffreys, Zellner and Siow (Jeffreys, 1961; Zellner and Siow, 1980; Zellner, 1986) and continued by many other authors with important contributions during the last ten years. These types of priors, which we label *conventional*, are introduced in Section . Bayarri et al. (2012) have recently shown that conventional priors have a number of optimal properties that make them a very appealing choice for dealing with model uncertainty.

Bayesian variable selection in Linear Models

Consider a Gaussian response variable $\boldsymbol{y}$, of size n, assumed to be explained by an intercept and some subset of p possible explanatory variables with values grouped in the $n \times p$ matrix $\boldsymbol{X} = (x_1, \ldots, x_p)$. Throughout the paper we suppose that $n > p$ and that $\boldsymbol{X}$ is of full column rank (but see a comment on the problem with $p > n$ in the concluding section). We define a binary vector $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_p)^t$ where $\gamma_i = 1$ if x_i is included in the model M_γ and zero otherwise. This is the variable selection problem, a model uncertainty problem with the following 2^p competing models:

$$M_\gamma : \boldsymbol{y} = \alpha \mathbf{1}_n + \boldsymbol{X}_\gamma \boldsymbol{\beta}_\gamma + \boldsymbol{\varepsilon}, \tag{1}$$

where $\boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2 \boldsymbol{I}_n)$ and the $n \times p_\gamma$ (where $p_\gamma = \sum \gamma_i$, the number of covariates in M_γ) design matrices $\boldsymbol{X}_\gamma$ are all possible submatrices of $\boldsymbol{X}$. If we choose the null matrix for X_γ, corresponding to $\gamma = \mathbf{0}$, we obtain the null model with only the intercept

$$M_0 : \boldsymbol{y} = \alpha \mathbf{1}_n + \boldsymbol{\varepsilon}. \tag{2}$$

Without loss of generality, we assume that columns of $\boldsymbol{X}$ have been centered on their corresponding means, which makes the covariates orthogonal to the intercept, and gives the intercept an interpretation that is common to all models. The set of all competing models is called the model space and is denoted as $\mathcal{M}$.

Assuming that one of the models in $\mathcal{M}$ is the true model, the posterior probability of any model is

$$Pr(M_{\gamma^*} \mid \boldsymbol{y}) = \frac{m_{\gamma*}(\boldsymbol{y}) Pr(M_{\gamma^*})}{\sum_\gamma m_\gamma(\boldsymbol{y}) Pr(M_\gamma)}, \tag{3}$$

where $Pr(M_\gamma)$ is the prior probability of M_γ and m_γ is the integrated likelihood with respect to the prior π_γ:

$$m_\gamma(\boldsymbol{y}) = \int f_\gamma(\boldsymbol{y} \mid \boldsymbol{\beta}_\gamma, \alpha, \sigma)\, \pi_\gamma(\boldsymbol{\beta}_\gamma, \alpha, \sigma^2)\, d\boldsymbol{\beta}_\gamma\, d\alpha\, d\sigma^2, \tag{4}$$

also called the (prior) marginal likelihood. Note that, for $\gamma = \mathbf{0}$ this integrated likelihood becomes:

$$m_0(\boldsymbol{y}) = \int f_0(\boldsymbol{y} \mid \alpha, \sigma)\, \pi_0(\alpha, \sigma^2)\, d\alpha\, d\sigma^2, \tag{5}$$

An alternative expression for (3) is based on the Bayes factors:

$$Pr(M_{\gamma^*} \mid \boldsymbol{y}) = \frac{B_{\gamma^*}(\boldsymbol{y})Pr(M_{\gamma^*})}{\sum_\gamma B_\gamma(\boldsymbol{y})Pr(M_\gamma)}, \tag{6}$$

where B_γ is the Bayes factor of M_γ with respect to a fixed model, say M_0 (without any loss of generality) and hence $B_\gamma = m_\gamma/m_0$ and $B_0 = 1$.

The prior on the model parameters implicitly assigns posterior point mass at zero for those regression coefficients that are not included in M_γ, which automatically induces sparsity.

As stated in the introduction, we are mainly interested in software that implements the formal Bayesian answer which implies that we use the posterior distribution in (3). Even with this important characteristic in common there could be substantial differences between R-packages (leaving aside for the moment details on programming and the interface used) due to the following three aspects:

- the priors that the package accommodates, that is, $\pi_\gamma(\boldsymbol{\beta}_\gamma, \alpha, \sigma^2)$ and $Pr(M_\gamma)$,
- the tools provided to summarize the posterior distribution and obtain model averaged inference,
- the numerical methods implemented to compute the posterior distribution.

We now succinctly review the main methodological proposals for the above issues, emphasizing particularly how these features are implemented in the different R packages.

Prior Specification

The two inputs that are needed to obtain the posterior distribution are π_γ and $Pr(M_\gamma)$: the 2^p prior distributions for the parameters within each model and the prior distribution over the model space, respectively.

Without loss of generality, the prior distributions π_γ can be expressed as

$$\pi_\gamma(\boldsymbol{\beta}_\gamma, \alpha, \sigma^2) = \pi_\gamma(\boldsymbol{\beta}_\gamma \mid \alpha, \sigma^2)\pi_\gamma(\alpha, \sigma^2).$$

In this paper, the focus is on a family of prior distributions that has been named “conventional" or “g-prior" (Fernández et al., 2001; Berger and Pericchi, 2001). Under this approach, the standard Jeffreys' prior is used for the parameters that are common to all models

$$\pi_\gamma(\alpha, \sigma^2) = \sigma^{-2} \tag{7}$$

and

$$\pi_\gamma(\boldsymbol{\beta}_\gamma \mid \alpha, \sigma^2) = \int N_{p_\gamma}(\boldsymbol{\beta}_\gamma \mid \mathbf{0}, g\sigma^2(\boldsymbol{X}_\gamma^t \boldsymbol{X}_\gamma)^{-1}).$$

That is, either a normal (if g is fixed) or a continuous mixture of a normal distribution (if g acts as random hyper parameter) centered on zero (“by reasons of similarity”, Jeffreys, 1961) and scaled by $\sigma^2(\boldsymbol{X}_\gamma^t \boldsymbol{X}_\gamma)^{-1}$ (“a matrix suggested by the form of the information matrix”, Zellner and Siow, 1980) times a factor g, normally called a “g-prior”. Recent research has shown that such conventional priors possess a number of optimal properties that can be extended by putting specific priors on the hyperparameter g. Among these properties are invariance under affine transformations of the covariates, several types of predictive matching and consistency (for details see Bayarri et al., 2012).

The specification of g has inspired many interesting studies in the literature. Of these, we have collected the most popular ones in Table 1.

Related with the conventional priors is the proposal by Raftery (1995) which is inspired by asymptotically reproducing the popular Bayesian Information Criterion (Schwarz, 1978). Raftery

Proposal	Reference	Name	Label
Constant g			
$g = n$	Zellner (1986); Kass and Wasserman (1995)	Unit Information prior (UIP)	C1
$g = p^2$	Foster and George (1994)	Risk inflation criterion prior (RIC)	C2
$g = \max\{n, p^2\}$	Fernández et al. (2001)	Benchmark prior (BRIC)	C3
$g = \log(n)$	Fernández et al. (2001)	Hannan-Quinn (HQ)	C4
$g_\gamma = \hat{g}_\gamma$	Liang et al. (2008)	Local Empirical Bayes (EBL)	C5
Random g			
$g \sim IGa(1/2, n/2)$	Jeffreys (1961); Zellner and Siow (1980, 1984)	Cauchy prior (JZS)	R1
$g \mid a \sim \pi(g) \propto (1+g)^{-a/2}$	Liang et al. (2008)	hyper-g	R2
$g \mid a \sim \pi(g) \propto (1+g/n)^{-a/2}$	Liang et al. (2008)	hyper-g/n	R3
$g \sim \pi(g) \propto (1+g)^{-3/2}$, $g > \frac{1+n}{p_\gamma+1} - 1$	Bayarri et al. (2012)	Robust prior	R4

Table 1: Specific proposals for the hyperparameter g in the literature. Column "Label" will be used as convenient reference to particular proposals throughout the paper. For the priors on g, $a > 2$ ensures a proper prior.

(1995) proposes using the same covariance matrix as the Unit Information Prior (see Table 1) but with mean the maximum likelihood estimator $\hat{\boldsymbol{\beta}}_\gamma$ (instead of the zero mean of the conventional prior).

Other priors specifically used in model uncertainty problems are the spike and slab priors, which assume that the components of $\boldsymbol{\beta}$ are independent, each having a mixture of two distributions: one highly concentrated on zero (the spike) and the other one quite disperse (the slab). There are two different developments of this idea in the literature. In the original proposal by Mitchell and Beauchamp (1988) the spike is a degenerate distribution at zero so this fits with what we have called the formal approach. The proposal by George and McCulloch (1993) in which the spike is a continuous distribution with a small variance also received a lot of attention, perhaps for computational advantages. In this implementation there is no posterior distribution over the model space as every model smaller than the full model has zero probability.

With respect to the priors over the model space $\mathcal{M}$, a very popular starting point is

$$Pr(M_\gamma \mid \theta) = \theta^{p_\gamma}(1-\theta)^{p-p_\gamma}, \tag{8}$$

where p_γ is the number of covariates in M_γ, and the hyperparameter $\theta \in (0,1)$ has the interpretation of the common probability that a given variable is included (independently of all others).

Among the most popular default choices for θ are

- Fixed $\theta = 1/2$, which assigns equal prior probability to each model, i.e $Pr(M_\gamma) = 1/2^p$;
- Random $\theta \sim \text{Unif}(0,1)$, giving equal probability to each possible number of covariates or model size.

Of course many other choices for θ – both fixed and random– have been considered in the literature. In general, fixed values of θ have been shown to perform poorly in controlling for multiplicity (the occurrence of spurious explanatory variables as a consequence of performing a large number of tests) and can lead to rather informative priors. This issue can be avoided by using random distributions for θ as, for instance, the second proposal above that has been studied in Scott and Berger (2010). Additionally, Ley and Steel (2009) consider the use of $\theta \sim$Beta$(1,b)$ which results in a binomial-beta prior for the number of covariates in the model or the model size, W:

$$Pr(W = w \mid b) \propto \binom{p}{w} \Gamma(1+w)\Gamma(b+p-w),\ \ w = 0,1,\ldots,p.$$

Notice that for $b = 1$ this reduces to the uniform prior on θ and also on W. As Ley and Steel (2009) highlight, this setting is useful to incorporate prior information about the mean model size, say $w^\star$. This would translate into $b = (p - w^\star)/w^\star$.

Summaries of the posterior distribution and model averaged inference

The simplest summary of the posterior model distribution (3) is its mode

$$\arg\max_{\gamma}\ Pr(M_\gamma \mid \boldsymbol{y}).$$

This model is the model most supported by the information (data and prior) and is normally called the HPM (highest posterior model) or MAP (maximum a posteriori) model. Clearly, a measure of uncertainty regarding this summary is reflected by its posterior probability which should always be reported.

When p is moderate to large, posterior probabilities of individual models can be very small and their interpretation loses appeal. In such situations, posterior inclusion probabilities (normally denoted as PIP) are very useful.

$$Pr(\gamma_i = 1 \mid \boldsymbol{y}) = \sum_{x_i \in M_\gamma} Pr(M_\gamma \mid \boldsymbol{y}). \tag{9}$$

These should be understood as the importance of each variable for explaining the response. Interestingly, these probabilities are used to define another summary, namely the median probability model (MPM) which is the model containing the covariates with inclusion probability larger than 0.5. This model is studied in (Barbieri and Berger, 2004) and they show that, in some situations, it is optimal for prediction.

Extending the idea of inclusion probabilities, it is interesting to obtain measures of joint importance of sets of regressors on the response. For instance, we can compute the posterior probability of two (or more) covariates occurring together in the model or the probability that a covariate enters the model given that another covariate is already present (or not). These quantities can be related to the so-called "jointness" of covariates and are studied, with other related summaries, in Ley and Steel (2007).

A measure of the model complexity is given by

$$Pr(W = w \mid \boldsymbol{y}) = \sum_{M_\gamma : p_\gamma = w} Pr(M_\gamma \mid \boldsymbol{y}), \tag{10}$$

which is the posterior probability mass function of the model size.

The posterior distribution easily allows for obtaining model averaged estimates of any quantity of interest Λ (assuming it has the same meaning across all models). Suppose $\hat{\Lambda}_\gamma$ is the estimate of Λ you would use if M_γ were the true model. Then the model averaged estimate of Λ is

$$\hat{\Lambda} = \sum_{M_\gamma} \hat{\Lambda}_\gamma \, Pr(M_\gamma \mid \boldsymbol{y}). \tag{11}$$

Similarly, the entire posterior distribution of Λ is given by

$$P_{\Lambda|\boldsymbol{y}} = \sum_{M_\gamma} P_{\Lambda|\boldsymbol{y},M_\gamma}\, Pr(M_\gamma \mid \boldsymbol{y}), \tag{12}$$

which has the appeal of formally accounting for model uncertainty.

When Λ refers to regression coefficients (β_i) the model averaged estimates should be used and interpreted with caution as they could be potentially misleading since the 'same' parameter may have a different meaning in different models (Berger and Pericchi, 2001). Also the posterior distribution of β_i is a discrete mixture and hence summaries like the mean are not natural descriptions.

One particular appealing application of this technique is in predicting new values $y^\star$ of the dependent variable associated with certain values of the covariates. In this case Λ could be the moments of $y^\star$ or even the whole predictive distribution. Apart from their intrinsic interest, predictions can be a very useful tool to run predictive checks (often using score functions) *e.g.* to compare various prior specifications.

Numerical methods

There are two main computational challenges in solving a model uncertainty problem. First is the integral in (4) and second is the sum in the denominator of (3) which involves many terms if p is moderate or large.

Fortunately, in normal models, conventional priors combine easily with the likelihood, and conditionally on g lead to closed forms for $m_\gamma(\boldsymbol{y})$. Hence, at most, a univariate integral needs to be computed when g is taken to be random. Interestingly, there have been recent proposals for prior distributions, which despite assuming a hyper prior on g induce closed form marginals using special mathematical functions. This characteristic includes the hyper-g and hyper-g/n of Liang et al. (2008); the Robust prior of Bayarri et al. (2012); the prior of Maruyama and George (2011) and the intrinsic prior in Womack et al. (2014).

The second problem, related with the magnitude of the number of models in $\mathcal{M}$ (i.e. 2^p), could be a much more difficult one. If p is small (say, p in the twenties at most) exhaustive enumeration is possible but if p is larger, heuristic methods need to be implemented. The question of which method should be used has been studied in Garcia-Donato and Martinez-Beneito (2013) which classify strategies as i) MCMC methods to sample from the posterior (3) in combination with estimates based on model visit frequencies and ii) searching methods looking for 'good' models with estimates based on renormalization (i.e with weights defined by the analytic expression of posterior probabilities, cf. (3)). They show that i) is potentially more precise than ii) which could be biased by the searching procedure. Approach i) is the most standard approach but different implementations of ii) have lead to fruitful contributions. The proposals in Raftery et al. (1997) and Fernández et al. (2001) which are based on a Metropolis-Hasting algorithm called MC^3 (originally introduced in Madigan and York, 1995) could be in either class above, while the implementation in Eicher et al. (2011) based on a leaps and bound algorithm proposed by Raftery (1995) is necessarily in ii), since model visit frequencies are not an approximation to model probabilities in this case.

CRAN packages screening

The computation of the posterior distribution in (3) can be implemented in generic Bayesian sofware. This would allow the user a fully customizable (e.g. in terms of priors) environment but can require considerable programming effort. Interesting examples of such approaches include Ntzoufras (2002, 2009) for implementations in WinBUGS (Lunn et al., 2000) or Joseph (2014) in JAGS (Plummer, 2003). Here our focus is different and we are mainly interested in searching for R-packages specifically designed to calculate the posterior distribution (3) and to provide tools for its further analysis. These are expected to be fully automatic for the user.

In what follows we will write the name of the packages using the font package; functions as `function()` and arguments as `argument`.

We seek in CRAN all possible packages that, potentially, could be used to implement the Bayesian approach to variable selection. The key words used to search in CRAN were *Model Selection*, *Variable Selection*, *Bayes Factor* and *Averaging*. Conducting this search on February 17, 2017, we found a total of 13 packages (the version in parentheses) : VarSelectIP(0.2-1); spikeslab(1.1.5) (Ishwaran et al., 2013); spikeslabGAM(1.1-11) (Scheipl, 2011); ensembleBMA(5.1.3) (Fraley et al., 2015); dma(1.2-3) (McCormick et al., 2014); BMA(3.18.6) (Raftery et al., 2015); mglmn(0.0.2) (Katabuchi and Nakamura, 2015); varbvs(2.0-8) (Carbonetto and Stephens, 2012); INLABMA(0.1-8) (Bivand et al., 2015);
tpackBAS(1.4.3) (Clyde, 2017); BayesFactor(0.9.12-2) (Morey et al., 2015); BayesVarSel(1.7.1) (Garcia-Donato and Forte, 2015); BMS(0.3.4) (Zeugner and Feldkircher, 2015) and mombf(1.8.3) (Rossell et al., 2014). As suggested by a referee, we also searched for related packages in `rseek.org` and in the CRAN task view devoted to Bayesian Inference, where the packages monomvn(1.9-7) (Gramacy, 2017) and BoomSpikeSlab(0.7.0) (Scott, 2016) also appeared.

From these, VarSelectIP appeared no longer supported and, from the rest, only the last five of the list found in CRAN seem clearly oriented towards the implementation of conventional priors described in the previous section to perform variable selection in linear models. Hence these will be considered for detailed description and comparison in the following sections. Particularly, BayesVarSel, BAS and BMS seem to be specifically conceived for that task, while the main motivations in BayesFactor and mombf seem different. BayesFactor provides many interesting functionalities to carry out t-tests, ANOVA-type studies and contingency tables using (conventional) Bayes factors with special emphasis on the specification of the hyper parameter g for certain design matrices. The package mombf focuses on a particular type of prior for the model parameters, namely the non-local prior (Johnson and Rossell, 2010, 2012), applied to either the normal scenario considered here or probit models.

Of the other packages we found, spikeslab and spikeslabGAM, implement spike and slab priors in the spirit of the approach by George and McCulloch (1993) and hence are not directly comparable with packages that compute the posterior distribution over the model space. Interestingly, the original spike and slab approach by Mitchell and Beauchamp (1988) (i.e. point-mass mixtures) is implemented in BoomSpikeSlab and in monomvn, with many possibilities to be used as prior inputs such as the Horseshoe (Carvalho et al., 2010) or the Bayesian lasso (Hans, 2010). Additionally, this point-mass spike and slab methodology is used as the base methodology in varbvs but with a specific development by Carbonetto and Stephens (2012) with extreme high dimensional problems ($p >> n$) in mind. Finally, BMA which, as noted by a referee, was a pioneering R-package covering the topic in this paper, provides the posterior distribution over the model space, but based on the BIC criterion. BMA also uses the Monte Carlo model composition MC^3, sampling algorithm in Raftery et al. (1997), but with a somewhat different prior.

Some other packages consider statistical models that are not of the type studied here (linear regression models). This is the case for ensembleBMA, which implements BMA for weather forecasting models and dma which focuses on dynamic models.

INLABMA interacts with the INLA (Rue et al., 2009) methodology for performing model selection within a given list of models. The priors used there are those in the R package INLA which are not model selection priors.

The package mglmn is not Bayesian and it uses the Akaike Information Criterion (AIC).

Selected packages for further comparison

The R packages BAS, BayesFactor, BayesVarSel, BMS and mombf provide functionalities to calculate and study the posterior distribution (3) corresponding to some of the conventional priors described in Table 1. The commands for such calculation are `bas.lm()` in BAS; `regressionBF()` in BayesFactor; `Bvs()`, `PBvs()` and `GibbsBvs()` (for exhaustive enumeration, distributed enumeration and Gibbs sampling) in BayesVarSel; `bms()` in BMS and finally `modelSelection()` in the package mombf.

Package	BAS	BayesFactor	BayesVarSel	BMS	mombf
Main command	bas.lm()	regressionBF()	Bvs(),PBvs(),GibbsBvs()	bms()	modelSelection()
Prior/Argument	prior=	rscaleCont=	prior.betas=	g=	priorCoef=
C1	"g-prior", alpha=n	-	"gZellner"	"UIP"	zellnerprior (tau=n)
C2	"g-prior", alpha=p^2	-	-	"RIC"	zellnerprior (tau=p^2)
C3	"g-prior", alpha=max(n,p^2)	-	"FLS"	"BRIC"	zellnerprior (tau=max(n,p^2))
C4	"g-prior", alpha=log(n)	-	-	"HQ"	zellnerprior (tau=log(n)))
C5	"EB-local"	-	-	"EBL"	-
R1	"ZS-null"	1	"ZellnerSiow"	-	-
R2	"hyper-g", alpha=a	-	-	"hyper=a"	-
R3	"hyper-g-n", alpha=a	-	"Liangetal"	-	-
R4	-	-	"Robust"	-	-

Table 2: Priors for the parameters within each model. Main commands and corresponding modifying arguments for the different specifications for the hyper parameter g (entries in column 'Prior' refer to Table 1). R3 implemented in BayesVarSel is the case with $a = 3$ as recommended in Liang et al. (2008).

Prior inputs The different conventional priors available in each package and the corresponding argument for their use are described in Table 2.

The implementation of the conventional priors in mombf have certain peculiarities that we now describe. The priors for the common parameters, (α, σ), in mombf do not exactly coincide with (7). In this package, the simplest model M_0 only contains the error term and hence α is not a common parameter. The more popular problem with fixed intercept examined in this paper (cf. (7)) is handled via the modifying argument center=TRUE (given by default) which in turn is equivalent to a prior for α degenerate at its maximum likelihood estimate. This will, especially if n is large enough, often lead to very similar results as with a flat prior on α but small differences could occur because in mombf the variability in this parameter is not taken into account. Also, for σ^2 this package uses an inverse gamma which has the non informative σ^{-2} as a limiting density. Thus,

Package	BAS	BayesFactor	BayesVarSel	BMS	mombf

Prior/Argument	`modelprior=`	`newPriorOdds`
		`(BFobject)=`
			`prior.models=`	`mprior=`	`priorDelta=`

$\theta = 1/2$	`uniform()`	`rep(1,2^ p)`	`"constant"`	`"fixed"`	`modelunifprior()`
$\theta \sim$ Unif(0,1)	`beta.binomial(1,1)`	-	`"ScottBerger"`	`"random"`	`modelbbprior(1,1)`

Table 3: Most popular default priors over the model space (see (8)) within the selected packages. For more flexible options see the text.

differences among the two are expected to be negligible if the parameters in the inverse gamma are small (values of 0.01 are given by default). Another logical argument in `modelSelection()` is `scale`. If it is set to `TRUE` the y's and the x's are scaled to have unitary variance. In this article we are fixing it to `scale=FALSE` so that the data that enter in all the main functions exactly coincide.

All five packages are very rich and flexible regarding the choice of the prior over the model space, $Pr(M_\gamma)$. The access to the standard approaches is described in Table 3. Apart from these standard priors, BMS, following the proposals in Ley and Steel (2009), also allows for the use of a beta distribution for θ in (8) by using `mprior="random"` and modifying the argument `mprior.size` to specify the desired expectation for the model prior distribution (the default option is $p/2$ hence providing the uniform prior on model size). Similarly the mombf package provides a beta prior for θ with parameter (a, b) by setting the corresponding argument to `modelbbprior(a,b)`. In BayesVarSel particular specifications of prior probabilities are available with `mprior="User"` and a $p + 1$ dimensional vector defined in `priorprobs` which describes the prior probability, $Pr(M_\gamma)$, of a single model of each possible size (models of the same size are assumed to have the same prior probability). Finally, BAS handles $Pr(M_\gamma)$ through the argument `modelprior` that allows for a direct implementation of (8) with either a fixed θ (using `modelprior=Bernoulli(probs=theta)`) or following a beta(a, b) distribution (using `modelprior=beta.binomial(a,b)`). Interestingly, BAS allows for other possibilities like truncating the dimension of the models.

For illustration purposes consider the FLS dataset in Ley and Steel (2009) with $p = 41$ potential regressors. These authors study the prior (8) with $\theta \sim Beta(1, b = (41 - \omega^\star)/\omega^\star)$ and $\omega^\star = 7$, reflecting that, a priori, the expected number of regressors is $\omega^\star = 7$. Such a prior can be implemented in BMS with `mprior="random", mprior.size=7`; in mombf with `modelbbprior(1,34/7)` and in BAS with `modelprior=beta.binomial(1,(41-7)/7)`. In BayesVarSel the syntax is quite different and we have to specify `prior.models="User"` and

$$\texttt{priorprobs} = \texttt{dbetabinom.ab}(\texttt{x} = 0 : 41, \texttt{size} = 41, \texttt{shape1} = 1, \texttt{shape2} = 34/7)/\texttt{choose}(41, 0 : 41).$$

Summaries and model averaging The result of executing the main commands for inference under model uncertainty (see Table 2) is an object describing, with a specific structure depending on the package, the posterior distribution (3). For ease of exposition suppose the object created is called `ob`. We compare here the different possibilities to summarize this distribution under each package. This is illustrated in the Supplementary Material which shows the different ways of summarizing the results for each package using one of the studied data sets.

- In BayesFactor, a list of the most probable models and their corresponding Bayes factors (to the null model) can be obtained with the command `head(ob)` or `plot(ob)` over the resulting object.

- In mombf, this list can be obtained with `postprob(ob)` but now best models are displayed with their posterior probabilities. Additionally, inclusion probabilities (9) are contained in `ob$margpp`. In the context of large model spaces, having a list with all the models sampled can be very useful so that the user may program his/her own needs, such as model averaged predictions. Such a list is contained in binary matrix form in mombf in `ob$postSample`. To obtain model averaged estimates we also have the command `rnlp` which produces posterior samples of regression coefficients (from which it is easy to obtain any $\hat{\Lambda}$ in (11) that relates to coefficients).

- In BayesVarSel most probable models and their probabilities are viewed printing the object created, `ob`, while `summary(ob)` displays a table with the inclusion probabilities, the HPM and the MPM (see Subsection). The posterior distribution of the model size (10) is in ob$postprobdim which can be graphed with plotBvs(ob,option="d"). Plots of several measures of the joint importance of two covariates (e.g. joint inclusion probabilities) can be visualized as an image plot with plotBvs(ob, option="j"). With the function GibbsBvs() all models visited are saved in the matrix ob$modelslogBF which, in the last column, have the Bayes factors of each model in log scale. With the function Bvs(), hence performing full enumeration, the most probable models are stored in ob$modelslogBF that contains in the last column the posterior probabilities. BayesVarSel is armed with the function BMAcoeff(ob) (and histBMA() for representation of the object created) that allows to perform model averaged estimates of coefficients through simulations of the posterior distribution. Similarly, predictions based on the posterior distribution can be obtained with the command predictBvs(ob) that again produces simulations from the underlying (model averaged) posterior predictive distribution.

- In BMS the top best models with their probabilities are displayed using topmodels(ob), that can also be plotted with image(ob). A summary(ob) of the resulting object also prints the posterior of the model size (10) that can be plotted with the command plotModelSize(ob). Printing ob displays a table with model averaged estimates of regression coefficients, namely their expected posterior mean and standard deviation (column Post Mean and Post SD respectively). Interestingly, it is possible to compute predictions with the commands predict(ob) (expected predictive mean a posteriori) and pred.density(ob) (mixture predictive density based on a selected number of best models). This package does not save all the models visited but only a (necessarily small) list of the best models sampled in ob$topmod expressed in hexadecimal code.

- In BAS a summary of ob prints the best models, their posterior probabilities and Bayes factors (expressed in relation to the HPM). Here, printing ob we obtain the posterior inclusion probabilities (PIPs). Executing plot(ob) several useful plots are produced, including a graph of PIPs and model size a posteriori. Using image(ob) generates an image plot of the best models providing more insights about which are the most important covariates. We can obtain estimations of regression parameters of various types using the command `coef` and accompanying functions `confint` (for intervals) and `plot`. Finally, BAS allows us to produce predictions from the object created with the function `predict()` whose result can be summarized in credible intervals with the function `confint()`.

Numerical methods Exhaustive enumeration can be performed with BayesFactor, BayesVarSel (command `Bvs()`), BMS (modifying argument `mcmc="enumerate"`) and in BAS (specifying the option `method="deterministic"`)).

When p is larger, exhaustive enumeration is not feasible and this is solved in mombf, BayesVarSel, BMS and BAS by providing specific routines to approximate the posterior distribution in such big model spaces. All four packages implement similar MCMC strategies. BAS, with the option `method="MCMC"` uses a combination of the Metropolis Hastings in Madigan and York (1995) and Raftery et al. (1997) (called MC^3) and a random swap selection of a variable excluded in the current model (see Clyde et al., 2011, for more details). A different technique is offered by BAS, with the option `method="MCMC+BAS"`. In this case the package implements the sampling method (without replacement) called Bayesian Adaptive Sampling (BAS) described in Clyde et al. (2011). With BAS, we have access with `method="MCMC"` to estimates of PIP based on the frequency of visits (`ob$probne0.MCMC`) or on the renormalization of posterior probabilities of sampled models (`ob$probne0.RN`). Of course, both estimates should be close to each other when convergence has been achieved. The rest of the packages essentially implement strategy i) briefly described in Section with the following peculiarities: The packages mombf and BayesVarSel implement the same Gibbs sampling scheme. A minor difference between both is that frequency-based estimates of inclusion probabilities in mombf are refined using Rao-Blackwellization. The methods programmed in BMS are also MCMC strategies to explore the posterior distribution which can be of the type birth and death (modifying argument `mcmc="bd"`) or a reversible jump (`mcmc="rev.jump"`). There is an important difference between the algorithms in mombf, BayesVarSel and in BMS. While in each MCMC step the inclusion/exclusion of *all* p covariates is sampled in mombf and BayesVarSel only one is sampled in BMS. So the updates in BMS will be more "local" and thus typically faster to implement, but convergence may take more updates.

Performance in selected datasets

To compare the selected packages two different scenarios will be considered:

- Exact scenario: data sets with small p and hence all the models can be enumerated.
- Sampling scenario: data sets with moderate to large p where only a small proportion of models can be explored.

As we previously mentioned, mombf does not accommodate the exact scenario nor can BayesFactor be considered in the sampling scenario. Ideally, we should compare all possible packages (in each setup) under the same prior. Table 2 indicates which comparisons are possible. We compared BayesFactor with BayesVarSel using the Zellner-Siow prior (labelled as R1) while we compared mombf and BMS and BayesVarSel using the UIP (C1). In all cases, and strictly for comparative purposes (e.g. BayesFactor does not allow for the Scott and Berger, 2010, prior) the constant prior over the model space was used.

As expected, all four packages produced very similar results in the analyzed datasets. Hence, the question of comparing them reduces basically to comparing computational times and the availability, clarity and organisation of the output.

For the computational comparisons to be fair all the calculations have been done on an iMac computer with Intel Core i5, 2.7 GHz processor. The code used to compute results provided here is publicly available as supplementary material to this paper accesible through the journal website.

Exact Scenario

We considered two data sets that we briefly describe.

US crime data. The US Crime data set was first examined by Ehrlich (1973) and is available from R-package `MASS` (Venables and Ripley, 2002). This data set has a total of $n = 47$ observations (corresponding to states in the US) of $p = 15$ potential covariates aimed at explaining the crime rate in a particular category per head of the population.

Returns to schooling. This data set, used by Tobias and Li (2004) and Ley and Steel (2012), concerns returns to education. As these are microeconomic data, the number of observations is much larger. In particular, we have a response variable (log of hourly wages) recorded for $n = 1190$ white males in the US in 1990, and a total of $p = 26$ possible regressors.

For both scenarios we directly compare the time needed to exactly calculate the posterior distribution with BAS, BayesVarSel and BMS using the C1 prior for the model parameters and the uniform prior (with fixed $\theta = 1/2$) on the model space. A referee pointed out that BAS is very demanding in terms of memory allocation and indeed we experienced that, for large enough p, this may cause an unexpected halt of execution. For the returns to schooling data set this was even the case when we changed from a 8GB RAM computer to a machine with more memory resource (48GB). These memory problems are likely due to a computational strategy where all the models sampled are retained in RAM memory. This ‘limitation’ is acknowledged in the documentation of the package which states that “For p less than 20-25, BAS can enumerate all models depending on memory availability”.

The computational times for this experiment are presented in Table 4. Results clearly indicate that BayesVarSel is more affected by the sample size since it performs better than BMS for the Crime data set ($n = 47$) but not for the returns to schooling application ($n = 1190$). The Bayes factors depend on the data only through the sum of squared errors and we know that BayesVarSel computes this statistic from scratch for each model and, thus, the n matters in that calculation. Hence a likely reason for the differences in computational time between BayesVarSel and BMS packages when n increases would be that the algorithms in BMS have implemented specific numerical recipes (see e.g. Eklund and Karlsson, 2007; Hammarling and Lucas, 2008) to update the sum of squared errors once a variable is added/dropped without the need to compute the sum of squared errors from scratch.

The comparison between BayesFactor and BayesVarSel, now using the R1 prior, is summarized in the same table for the Crime data set where we can clearly see that BayesFactor is outperformed by BayesVarSel. For the Crime data, running time for BAS is approximately constant for the two priors considered while the time in BayesVarSel increases with R1 compared to C1. This behaviour is likely due to the numerical integration algorithm implemented for the Zellner-Siow prior (also used in R3 but not in R4).

Data set	Prior	BMS	BayesVarSel	BayesFactor	BAS	mombf
Crime $p = 15$	C1 unif	3.07 secs	0.35 secs	-	0.10 secs	0.92 secs
Returns to schooling $p = 26$	C1 unif	1.69 hrs	11.32 hrs	-	-	-
Crime $p = 15$	R1 unif	-	1.26 secs	12.59 mins	0.12 secs	-

Table 4: Computational times in exact scenario (observed variations over repetitions of the experiment were negligible)

Table 4 also illustrates the large difference in computational cost between an exhaustive analysis with $p = 15$ covariates (where $\mathcal{M}$ has $2^{15} = 32,768$ models) and $p = 26$, leading to a model space with 67 million models, which is about 2000 times larger. Computational cost goes up by a factor of about 2000 for BMS, which is therefore roughly linear in the size of model space, and thus seems virtually unaffected by the number of observations n. Our intuition is that this is a consequence of how the statistics are computed within each package, as commented above. Notice that the returns to schooling data have not been analysed by BAS due to RAM memory limitations; 48GB of RAM was not enough for this example. A referee pointed out that it does work with more resources but we do not know how much RAM memory is needed for problems of this size.

Sampling Scenario

We considered here he following three data sets.

Ozone. These data were used by Casella and Moreno (2006), Berger and Molina (2005) and Garcia-Donato and Martinez-Beneito (2013) and contain $n = 178$ measures of ozone concentration in the atmosphere with a total of $p = 35$ covariates. Details on the data can be found in Casella and Moreno (2006).

GDP growth. This dataset is larger than Ozone with a total of $p = 67$ potential drivers for the annual GDP growth per capita between 1960 and 1996 for $n = 88$ countries. This data set was used in Sala-I-Martin et al. (2004) and revisited by Ley and Steel (2007).

Boston housing. This dataset was used recently in Schäfer and Chopin (2013) and contains $n = 506$ observations of $p = 103$ covariates formed by the 13 columns of the original data set, all first order interactions and a squared version of each covariate (except for the binary variable `CHAS`).

For the Ozone dataset, exact inclusion probabilities, (9), are reported in Garcia-Donato and Martinez-Beneito (2013) for the C1 prior. These are the result of an intensive computational experiment aimed at comparing different searching methods. These numbers provide a "gold standard" and thus allow us to define a simple measure to compare the computational efficiency of the different packages. For a given computational time, t, we calculate

$$\Delta_t = \max_{i=1,\dots,p} |\widehat{Pr}_t(\gamma_i = 1 \mid \boldsymbol{y}) - Pr(\gamma_i = 1 \mid \boldsymbol{y})|,$$

where $\widehat{Pr}_t(\gamma_i = 1 \mid \boldsymbol{y})$ is the estimate of the corresponding PIP at time t provided by the package. Clearly, the faster Δ_t approaches zero, the more efficient is the package. In Figure 1 we have plotted Δ_t for mombf; BayesVarSel (label Bvs); the two algorithms in BMS (the label BMSbd indicates

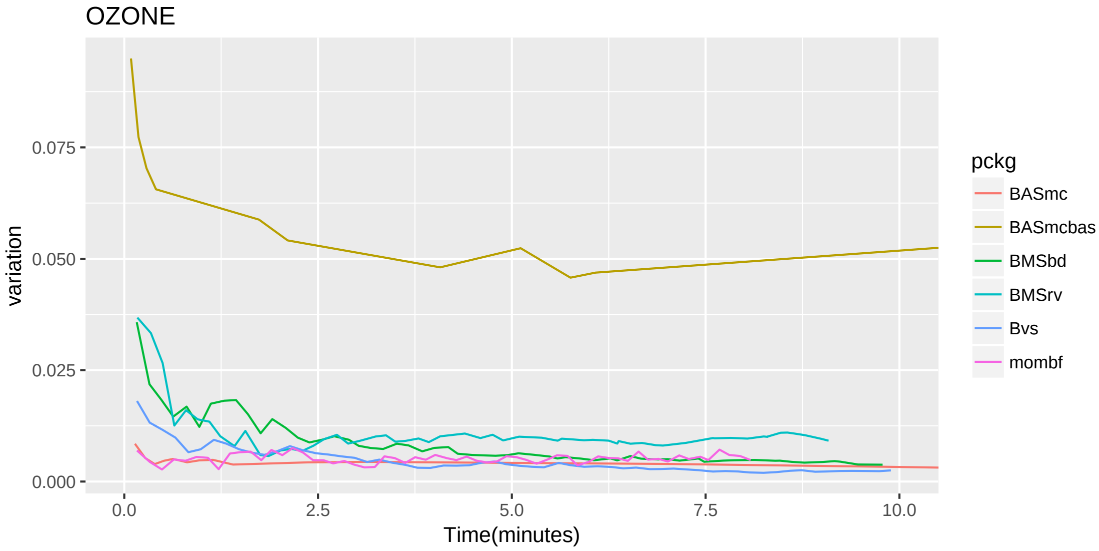

Figure 1: Ozone dataset: maximum difference with the real inclusion probabilities (Δ_t) as a function of computational time. The labels used are defined in the text.

birth/death and BMSrv corresponds to reversible jump) and two in BAS (BASmc is obtained with `method="MCMC"` with PIP obtained via frequency of visits and BASmcbas corresponds to `method="MCMC+BAS"` with PIP calculated by renormalization of Bayes factor). The same labels will be used for the rest of the paper. Notice that all estimates are based on frequencies except for BASmcbas.

All approaches –except for BASmcbas– behave quite satisfactorily, providing very reliable estimates with a small computational time (a maximum discrepancy with the exact values of 0.01 in less than 2.5 minutes). It seems that BASmc and mombf are slightly more efficient than the rest while BMSrv in BMS is less efficient. The observed less optimal behaviour of BASmcbas could be explained by the known bias introduced by the combination of heuristic methods to search good models plus estimation based on renormalization (an effect fully reported in Garcia-Donato and Martinez-Beneito, 2013).

In the GDP growth and the Boston housing examples, we cannot compute Δ_t simply because the PIPs are unknown. Nevertheless, we observe that for a sufficiently large computational time, all

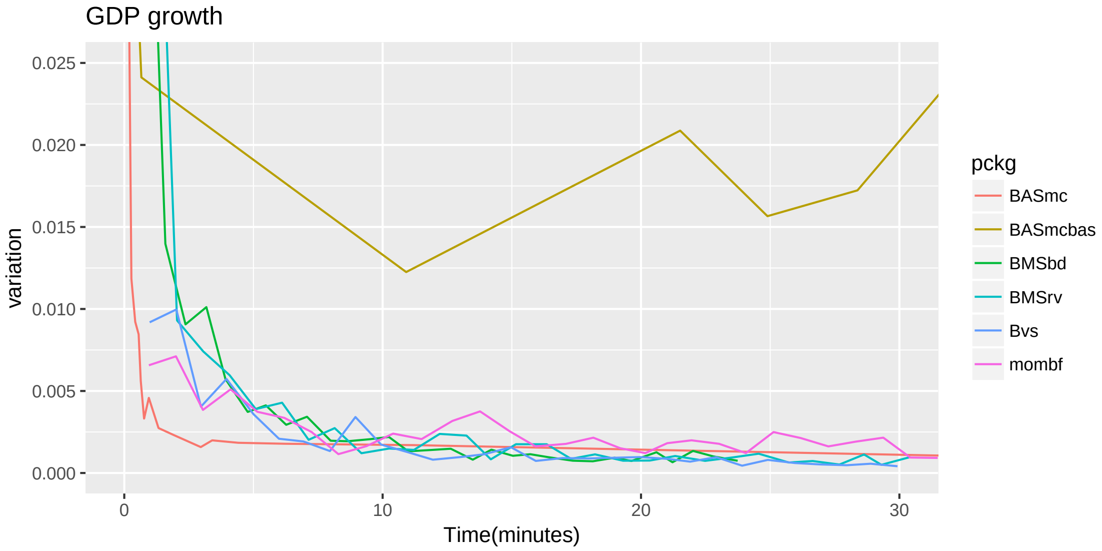

Figure 2: GDP growth data: variations in PIP ($\Delta_{t,t-dt}$) as a function of computational time with $dt = 60$ seconds (starting after the burning period).

procedures except for BASmcbas, converged to almost identical PIPs. Hence, and even in the unlikely case that none of them were capturing the 'truth' it seems that the fairest way to compare the packages is computing time until 'convergence'. This is what we have represented in Figures 2 and 3 where the y-axes display the difference between estimates at consecutive computational times, *i.e.*

$$\Delta_{t,t-dt} = \max_{i=1,\ldots,p} |\widehat{Pr}_t(\gamma_i = 1 \mid \boldsymbol{y}) - \widehat{Pr}_{t-dt}(\gamma_i = 1 \mid \boldsymbol{y})|,$$

where $dt = 60$ seconds was used and we have verified that PIPs converge.

In the GDP growth data set, we can not find big differences in the performance of all but BASmcbas and all of these behave, again, very satisfactorily. It seems that the procedure Bvs (implemented by BayesVarSel) tends to 0 faster than the rest of algorithms while the performance of the package mombf manifests slightly more variability.

In the Boston housing problem the package BayesVarSel is clearly penalized (with respect to the GDP growth data) by the large number of observations, hence displaying slower convergence. Nevertheless, even in this challenging example with more than 100 possible covariates (and over 500 observations),

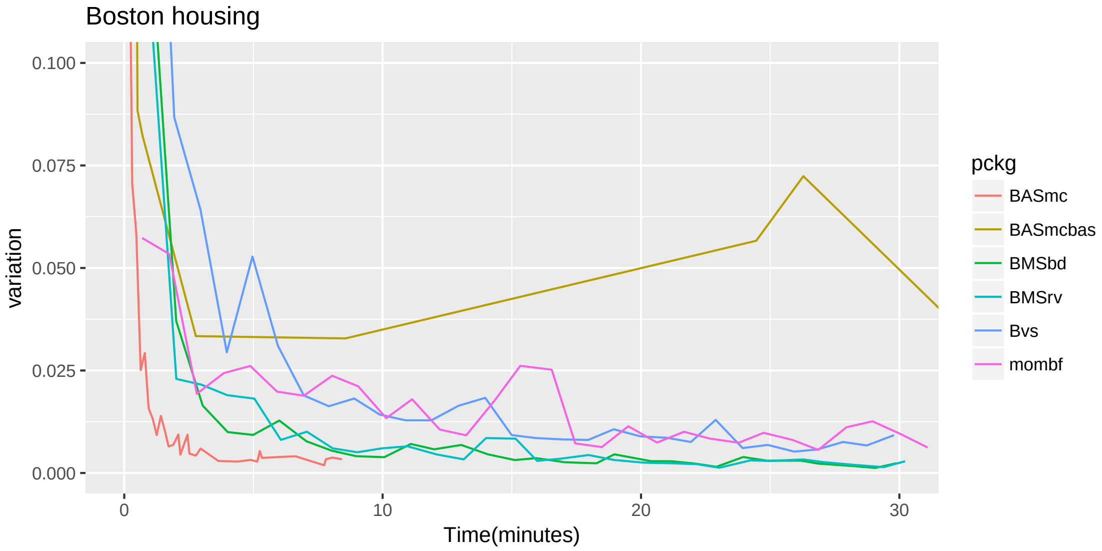

Figure 3: Boston Housing data: variations in PIP ($\Delta_{t,t-dt}$) as a function of computational time with $dt = 60$ seconds (starting after the burning period). The BASmc line is shorter than the rest because the programme stops automatically due to some internal convergence criteria.

all methods barring BASmcbas lead to practical convergence within 10 minutes of running time.

Other Features

Besides the characteristics analysed so far (prior inputs, numerical methods and summaries), there are several other features of the packages that are potentially relevant for the applied user. We list some here under three categories: the interface, extra functionalities and documentation.

The interface In general, all five packages have simple interfaces with quite intuitive syntaxes. One minor difference is that in BAS, BayesVarSel and BayesFactor the dependent and explanatory variables are defined with the use of `formula` (hence inspired by well-known `R` commands like `lm`) while in mombf these are defined through the arguments `y` and `x`. In BMS the dependent variable should be in the first column of the data provided and the rest play the role of explanatory variables.

Extra functionalities

- *Fixed covariates.* By default only the intercept is included in all the competing models (cf. (1)) in all packages (but recall this is handled in mombf via centering). There could be situations where we wish to assume that certain covariates affect the response and these should always be included in the analysis (see, for instance Camarero et al., 2015). Packages BAS, BMS and BayesVarSel include this possibility in their main commands.

- *Main terms and interactions.* On occasion, it is convenient to conserve the hierarchy between the explanatory variables in the way that interactions (or higher polynomial terms) are only included if the main terms are included (Peixoto, 1987). In Chipman et al. (1997) this is called the "heredity principle". This would translate into a reduction of the model space. The package BMS accommodates this possibility through a modification of the sampling algorithm.

- *Model comparison.* A complementary tool to the BMA exercise would be comparing separately some of the competing models (e.g. comparing the HPM and the MPM). These type of comparisons can be performed in BMS, BayesVarSel and BayesFactor.

- *Convergence.* BAS and BMS include several interesting tools to analyse the convergence of the sampling methods implemented. For instance, in BAS the command `diagnostics()` produce a comparative plot of the estimations of PIP based on frequencies and on re-normalization. Similarity of both estimates suggests convergence of the MCMC (see e.g. Fernández et al., 2001).

- *Parallel computation.* BMS, BayesVarSel and mombf have facilities to perform computations in parallel.

- *Jointness measures.* BayesVarSel provides the functions `Jointness()` and `plotBvs()` (with certain values of the argument `option`) to compute and visualize these type of measures. It is also worth mentioning the post on the blog https://modelaveraging.wordpress.com describing how different jointness measures can be computed using `BMS`

Documentation The five packages come with a detailed help with useful examples. Further, mombf, BMS and BAS have a comprehensive *vignette* with additional illustrations and written more pedagogically than the help documentations. The main functionalities and usage of the package BayesVarSel are explained in the paper Garcia-Donato and Forte (2016).

The packages BMS and BayesFactor are documented in the websites associated with Feldkircher and Zeugner (2014) (http://bms.zeugner.eu) and Morey (2015) (http://bayesfactor.blogspot.com.es), respectively. These sites contain manuals as well as valuable additional information, especially to users less familiar with model uncertainty techniques.

Conclusions and recommendations

In this paper, we have examined the performance and the built-in possibilities of various R-packages available in CRAN for the purpose of Bayesian variable selection in linear regression. In particular, we compare the packages BAS, BMS, BayesVarSel, mombf and BayesFactor. It is clear that all packages concerned lead to very similar results (with the only exception of the method BAS in BAS), which is reassuring for the user. However, they do differ in the prior choices they allow, the way they present the output and the numerical strategies used. The latter affects both CPU times and memory requirements. In this regard, BAS is overall faster than the rest but with a very high cost in terms of memory allocation (particularly problematic for large p and/or long runs of MCMC). BayesVarSel and BMS do not have such stringent requirements about memory and we believe that BayesVarSel is a good choice for small or moderate values of n, while BMS is preferable when n is large. The package BayesFactor can not deal with larger values of p and seems relatively slow, thus is not recommended for general use. mombf uses a slightly different prior from the one we focus on here (and which is the most commonly used), but is relatively competitive and closely approximates the PIPs after a short run time, albeit with slightly more variability than BMS or BayesVarSel.

In practice, users may be interested in specific features, such as always including certain covariates, that will dictate the choice of package. On the basis of its performance, the flexibility of prior choices and the extra features allowed, we would generally recommend the use of BAS (using BASmc) in an ideal situation with extremely large memory resources, and whenever memory allocation is an issue (as it often is in practice) we would recommend BayesVarSel for small or moderate values of n, and BMS when n is large.

Throughout the paper, we have assumed that $n > p$ but there is an obvious and increasing interest in the problem with $n < p$ and particularly with $n << p$. In principle conventional priors cannot handle this situation since the prior covariance matrix is singular for many of the competing models. This is of course a limitation that does not affect other choice of priors like the spike-and-slab family since they assume independence, but then the attractive properties of the conventional priors are lost. Yet there is a simple way to overcome this difficulty within the conventional methodology through assigning zero prior probability to singular models (this has been assumed, for example, in the context of non-local priors in Johnson and Rossell (2012)). This approach is assumed by default in the packages mombf and BMS and can be easily incorporated with the arguments `priorprobs` and modelprior in BayesVarSel and BAS respectively. Another possibility is the approach described in Berger et al. (2016) which can be viewed as a formalization of the suggestion in Liang et al. (2008) about the use of generalized inverses. As this is a very recent proposal, it is still not incorporated in any of the packages (but we expect it be in the near future).

Acknowledgments

The authors would like to thank David Rossell for valuable comments on a preliminary version of this paper. We are grateful to two anonymous referees and the editors for many suggestions that have helped us to substantially improve the paper.

References

Barbieri, M. M. and J. O. Berger (2004). Optimal predictive model selection. *The Annals of Statistics 32*(3), 870–897.

Bayarri, M. J., J. O. Berger, A. Forte, and G. García-Donato (2012). Criteria for Bayesian model choice with application to variable selection. *The Annals of Statistics 40*, 1550–1577.

Berger, J. O., G. García-Donato, M. A. Martinez-Beneito, and V. Peña (2016, July). Bayesian variable selection in high dimensional problems without assumptions on prior model probabilities. arXiv:1607.02993.

Berger, J. O. and G. Molina (2005). Posterior model probabilities via path-based pairwise priors. *Statistica Neerlandica 59*(1), 3–15.

Berger, J. O. and L. R. Pericchi (2001). *Objective Bayesian Methods for Model Selection: Introduction and Comparison*, Volume 38 of *Lecture Notes–Monograph Series*, pp. 135–207. Beachwood, OH: Institute of Mathematical Statistics.

Bivand, R. S., V. Gómez-Rubio, and H. Rue (2015). Spatial data analysis with R-INLA with some extensions. *Journal of Statistical Software 63*(20), 1–31.

Camarero, M., A. Forte, G. García-Donato, Y. Mendoza, and J. Ordoñez (2015). Variable selection in the analysis of energy consumption-growth nexus. *Energy Economics 52, Part A*, 207–216.

Carbonetto, P. and M. Stephens (2012). Scalable variational inference for Bayesian variable selection in regression, and its accuracy in genetic association studies. *Bayesian Analysis 7*(1), 73–108.

Carvalho, C., N. Polson, and J. Scott (2010). The horseshoe estimator for sparse signals. *Biometrika 97*(2), 465–480.

Casella, G. and E. Moreno (2006). Objective Bayesian variable selection. *Journal of the American Statistical Association 101*(473), 157–167.

Chipman, H., M. Hamada, and C. F. J. Wu (1997). A Bayesian variable-selection approach for analyzing designed experiments with complex aliasing. *Technometrics 39*(4), 372–381.

Clyde, M. (2017). *BAS: Bayesian Adaptive Sampling for Bayesian Model Averaging.* R package version 1.4.3.

Clyde, M. A., J. Ghosh, and M. L. Littman (2011). Bayesian adaptive sampling for variable selection and model averaging. *Journal of Computational and Graphical Statistics 20*(1), 80–101.

Ehrlich, I. (1973). Participation in illegitimate activities: a theoretical and empirical investigation. *Journal of Political Economy 81*(3), 521–567.

Eicher, T., C. Papageorgiou, and A. E. Raftery (2011). Default priors and predictive performance in Bayesian model averaging, with application to growth determinants. *Journal of Applied Econometrics 26*, 30–55.

Eklund, J. and S. Karlsson (2007). Computational efficiency in Bayesian model and variable selection. Technical report, Central Bank of Iceland, Economics Department.

Feldkircher, M. and S. Zeugner (2009). Benchmark priors revisited: On adaptive shrinkage and the supermodel effect in bayesian model averaging. *IMF Working Papers*.

Feldkircher, M. and S. Zeugner (2014). R-package BMS Bayesian Model Averaging in R. http://bms.zeugner.eu.

Fernández, C., E. Ley, and M. F. Steel (2001). Benchmark priors for Bayesian model averaging. *Journal of Econometrics 100*, 381–427.

Foster, D. and E. I. George (1994). The Risk Inflation Criterion for Multiple Regression. *The Annals of Statistics 22*, 381–427.

Fraley, C., A. E. Raftery, J. M. Sloughter, T. Gneiting, and U. of Washington. (2015). *ensembleBMA: Probabilistic Forecasting using Ensembles and Bayesian Model Averaging.* R package version 5.1.2.

Garcia-Donato, G. and A. Forte (2015). *BayesVarSel: Bayes Factors, Model Choice And Variable Selection In Linear Models*. R package version 1.6.1.

Garcia-Donato, G. and A. Forte (2016, November). BayesVarSel: Bayesian Testing, Variable Selection and Model Averaging in Linear Models using R. arXiv:1611.08118.

Garcia-Donato, G. and M. A. Martinez-Beneito (2013). On Sampling strategies in Bayesian variable selection problems with large model spaces. *Journal of the American Statistical Association 108*(501), 340–352.

George, E. I. and R. E. McCulloch (1993). Variable selection via Gibbs sampling. *Journal of the American Statistical Association 88*(423), 881–889.

Gramacy, R. B. (2017). *monomvn: Estimation for Multivariate Normal and Student-t Data with Monotone Missingness*. R package version 1.9-7.

Hammarling, S. and C. Lucas (2008). Updating the QR factorization and the least squares problem. Technical report, Manchester Institute for Mathematical Sciences School of Mathematics.

Hans, C. (2010). Model uncertainty and variable selection in bayesian lasso regression. *Statistics and Computing 20*, 221–229.

Ishwaran, H., J. Rao, and U. Kogalur (2013). *spikeslab: Prediction and Variable Selection Using Spike and Slab Regression*. R package version 1.1.5.

Jeffreys, H. (1961). *Theory of Probability* (3rd ed.). Oxford University Press.

Johnson, V. E. and D. Rossell (2010). On the use of non-local prior densities in Bayesian hypothesis tests. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 72*(2), 143–170.

Johnson, V. E. and D. Rossell (2012). Bayesian model selection in high-dimensional settings. *Journal of the American Statistical Association 107*(498), 649–660.

Joseph, M. (2014, March). Stochastic search variable selection in JAGS. http://https://www.r-bloggers.com/stochastic-search-variable-selection-in-jags/.

Kass, R. E. and L. Wasserman (1995). A reference Bayesian test for nested hypotheses and its relationship to the schwarz criterion. *Journal of the American Statistical Association 90*(431), 928–934.

Katabuchi, M. and A. Nakamura (2015). *mglmn: Model Averaging for Multivariate GLM with Null Models.* R package version 0.0.2.

Ley, E. and M. F. Steel (2007). Jointness in Bayesian variable selection with applications to growth regression. *Journal of Macroeconomics 29*(3), 476 – 493. Special Issue on the Empirics of Growth Nonlinearities.

Ley, E. and M. F. Steel (2009). On the effect of prior assumptions in Bayesian model averaging with applications to growth regression. *Journal of Applied Econometrics 24*(4), 651–674.

Ley, E. and M. F. Steel (2012). Mixtures of g-priors for Bayesian model averaging with economic applications. *Journal of Econometrics 171*(2), 251 – 266. Bayesian Models, Methods and Applications.

Liang, F., R. Paulo, G. Molina, M. A. Clyde, and J. O. Berger (2008). Mixtures of g-priors for Bayesian variable selection. *Journal of the American Statistical Association 103*(481), 410–423.

Lunn, D. J., A. Thomas, N. Best, and D. Spiegelhalter (2000). WinBUGS-a Bayesian modelling framework: concepts, structure and extensibility. *Statistics and Computing 10*, 325–337.

Madigan, D. and J. York (1995). Bayesian graphical models for discrete data. *International Statistical Review 63*, 215–232.

Maruyama, Y. and E. I. George (2011). Fully Bayes factors with a generalized g-prior. *The Annals of Statistics 39*(5), 2740–2765.

McCormick, T. H., A. E. Raftery, and D. Madigan (2014). *dma: Dynamic model averaging.* R package version 1.2-2.

Mitchell, T. and J. Beauchamp (1988). Bayesian variable selection in linear regression. *Journal of the Americal Statistical Association 83*, 1023–1032.

Morey, R. (2015). BayesFactor an R package for Bayesian data analysis. http://bayesfactorpcl.r-forge.r-project.org.

Morey, R. D., J. N. Rouder, and T. Jamil (2015). *BayesFactor: Computation of Bayes Factors for Common Designs.* R package version 0.9.11-1.

Ntzoufras, I. (2002). Gibbs variable selection using BUGS. *Journal of statistical software 7*, 1–19.

Ntzoufras, I. (2009). *Bayesian modeling using WinBUGS.* Wiley series in computational statistics. John Wiley & Sons.

Peixoto, J. (1987). Hierarchical variable selection in polynomial regression models. *American Statistician 44*(1), 26–30.

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. In *Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003).*

R Core Team (2015). *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing.

Raftery, A., J. Hoeting, C. Volinsky, I. Painter, and K. Y. Yeung (2015). *BMA: Bayesian Model Averaging.* R package version 3.18.4.

Raftery, A. E. (1995). Bayesian model selection in social research. *Sociological Methodology 25*, 111–163.

Raftery, A. E., D. Madigan, and J. Hoeting (1997). Bayesian model averaging for linear regression models. *Journal of the American Statistical Association 92*, 179–191.

Rossell, D., J. D. Cook, D. Telesca, and P. Roebuck (2014). *mombf: Moment and Inverse Moment Bayes factors.* R package version 1.5.9.

Rouder, J. N., R. D. Morey, P. L. Speckman, and J. M. Province (2012). Default bayes factors for anova designs. *Journal of Mathematical Psychology 56*, 356–374.

Rouder, J. N., P. L. Speckman, D. Sun, R. D. Morey, and G. Iverson (2009). Bayesian t-tests for accepting and rejecting the null hypothesis. *Psychonomic Bulletin and Review 16*, 225–237.

Rue, H., S. Martino, and N. Chopin (2009). Approximate Bayesian inference for latent gaussian models using integrated nested Laplace approximations (with discussion). *Journal of the Royal Statistical Society: Series B 71*(2), 319–392.

Sala-I-Martin, X., G. Doppelhofer, and R. I. Miller (2004). Determinants of long-term growth: A Bayesian averaging of classical estimates (BACE) approach. *American Economic Review 94*(4), 813–835.

Schäfer, C. and N. Chopin (2013). Sequential monte carlo on large binary sampling spaces. *Statistics and Computing 23*(2), 163–184.

Scheipl, F. (2011). spikeslabGAM: Bayesian variable selection, model choice and regularization for generalized additive mixed models in R. *Journal of statistical software 43*(14), 1–24.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics 6*(2), 461–464.

Scott, J. G. and J. O. Berger (2010). Bayes and empirical-Bayes multiplicity adjustment in the variable-selection problem. *The Annals of Statistics 38*(5), 2587–2619.

Scott, S. L. (2016). *BoomSpikeSlab: MCMC for Spike and Slab Regression.* R package version 0.7.0.

Tobias, J. L. and M. Li (2004). Returns to schooling and Bayesian model averaging: A union of two literatures. *Journal of Economic Surveys 18*(2), 153–180.

Venables, W. N. and B. D. Ripley (2002). *Modern Applied Statistics with S* (Fourth ed.). New York: Springer-Verlag. ISBN 0-387-95457-0.

Womack, A., L. León-Novelo, and G. Casella (2014). Inference from Intrinsic Bayes' procedures under model selection and uncertainty. *Journal of the Americal Statistical Association 109*, 1040–1053.

Zellner, A. (1986). On assessing prior distributions and Bayesian regression analysis with g-prior distributions. In A. Zellner (Ed.), *Bayesian Inference and Decision techniques: Essays in Honor of Bruno de Finetti*, pp. 389–399. Edward Elgar Publishing Limited.

Zellner, A. and A. Siow (1980). Posterior odds ratio for selected regression hypotheses. In J. M. Bernardo, M. DeGroot, D. Lindley, and A. F. M. Smith (Eds.), *Bayesian Statistics 1*, pp. 585–603. Valencia: University Press.

Zellner, A. and A. Siow (1984). *Basic Issues in Econometrics.* Chicago: University of Chicago Press.

Zeugner, S. and M. Feldkircher (2015). Bayesian model averaging employing fixed and flexible priors: The BMS package for R. *Journal of Statistical Software 68*(4), 1–37.

Appendix A: Supplementary Material: Summarizing the output for the Crime data set

BMS

- Call

```
> library(BMS)#Version: 0.3.4
> ob <- bms(X.data=lUScrime, g="UIP", mprior="uniform", nmodel=100,g.stats = F)
```

- `print()`

```
> ob
```

```
##             PIP    Post Mean    Post SD Cond.Pos.Sign Idx
## Ineq 0.9886667  1.43717383 0.38685782    1.00000000  12
## Ed   0.9836667  1.88879241 0.60288400    1.00000000   2
## Prob 0.8800000 -0.20653933 0.11614165    0.00000000  13
## M    0.8406667  1.15074387 0.69006929    1.00000000   1
## NW   0.6100000  0.05948973 0.05845147    1.00000000   8
## U2   0.6016667  0.20400737 0.21596352    0.99889197  10
## Po1  0.5996667  0.56267682 0.54463150    1.00000000   3
## Po2  0.4906667  0.40936002 0.55299266    0.96535326   4
## Pop  0.3606667 -0.02271586 0.03980480    0.00000000   7
## GDP  0.3080000  0.18101443 0.35033505    1.00000000  11
## So   0.2760000  0.03782410 0.09277546    0.97463768  15
## Time 0.2670000 -0.06113723 0.14041361    0.05617978  14
## U1   0.1640000 -0.01984539 0.14783355    0.35772358   9
## LF   0.1453333  0.05646821 0.27907629    0.86009174   5
## M.F  0.1373333 -0.01963118 0.66576789    0.46116505   6
##
## Mean no. regressors               Draws             Burnins
##            "7.6533"              "3000"              "1000"
##                Time  No. models visited      Modelspace 2^K
##     "0.522115 secs"              "1426"             "32768"
##           % visited         % Topmodels            Corr PMP
##               "4.4"                "48"            "0.6996"
##            No. Obs.         Model Prior             g-Prior
##                "47"     "uniform / 7.5"               "UIP"
```

- `plot()` and `image()`

```
> plot(ob)
```

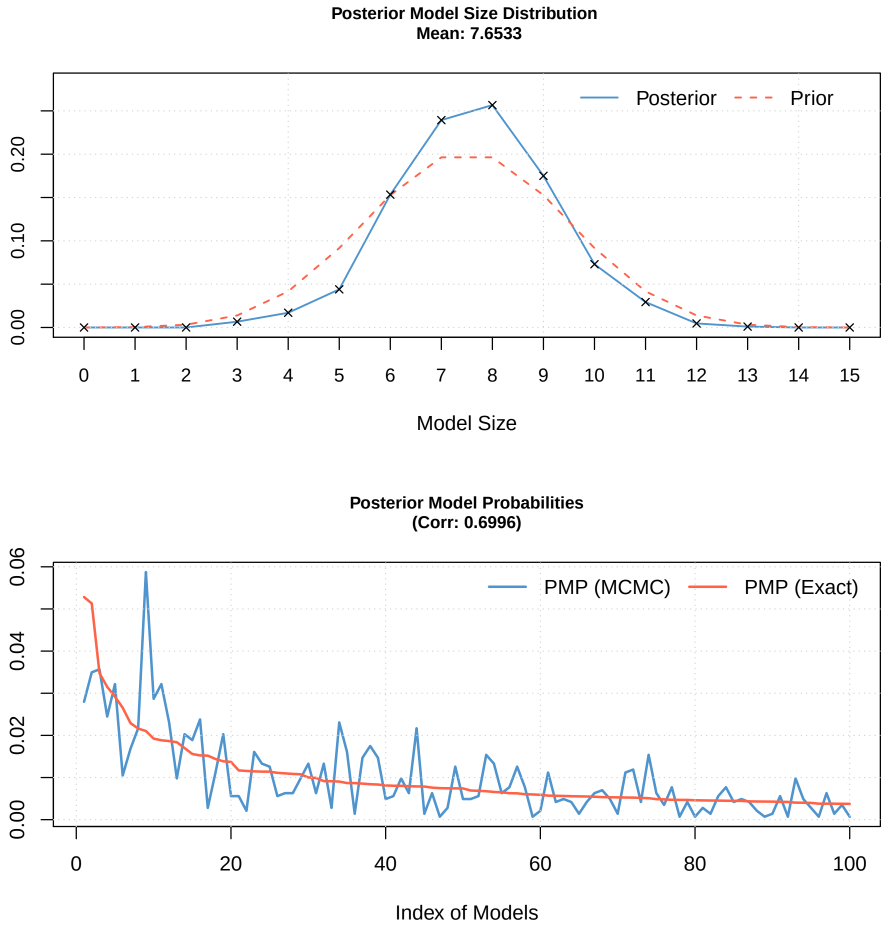

Figure 4: Prior/Posterior probabilities of model size and convergence performance plotted using `plot(ob)`

```
> image(ob)
```

- Predictive density

```
> plot(pred.density(ob, newdata = lUScrime[1,-1]))
```

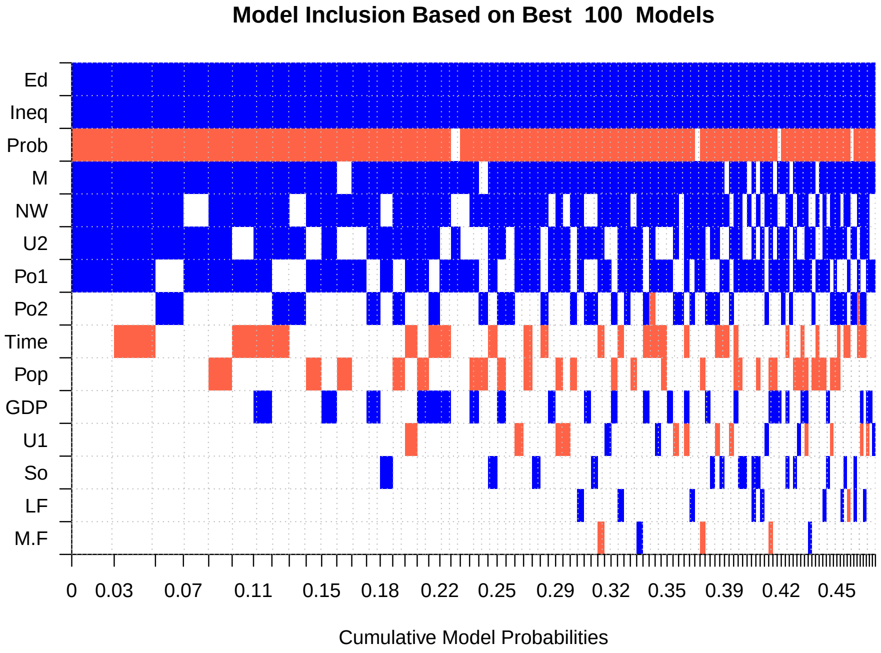

Figure 5: Model inclusion probabilities based on the best 100 models using `image(ob)`

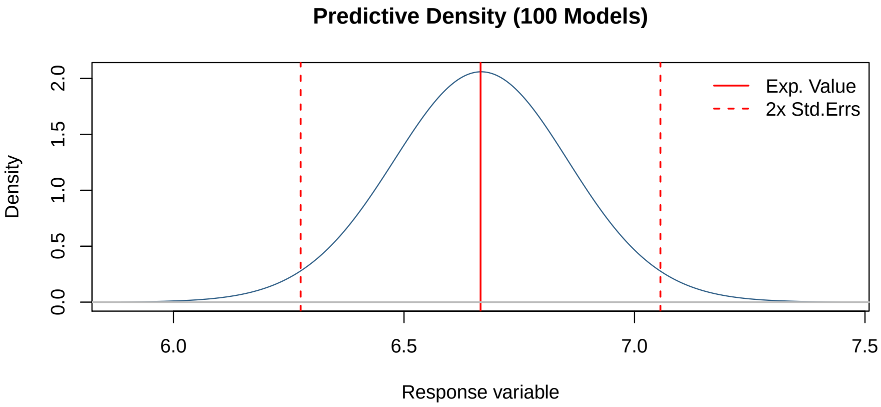

Figure 6: Predictive density using the 100 most probable models

BayesFactor

- Call:

```
library(BayesFactor)#Version: 0.9.12-2
ob <- regressionBF(formula=y~. ,data=lUScrime, rscaleCont = 1, noSample=TRUE)
```

```
> head(ob)
```

```
## Bayes factor analysis
## --------------
## [1] M + Ed + Po1 + NW + U2 + Ineq + Prob + Time       : 23217774828 ±0%
## [2] M + Ed + Po1 + NW + U2 + Ineq + Prob              : 22390162522 ±0%
## [3] M + Ed + Po2 + NW + U2 + Ineq + Prob              : 15146223575 ±0%
## [4] M + Ed + Po1 + Pop + NW + U2 + Ineq + Prob        : 13801956643 ±0%
## [5] M + Ed + Po1 + U2 + Ineq + Prob                   : 12988208736 ±0%
## [6] M + Ed + Po1 + NW + U2 + GDP + Ineq + Prob + Time : 11922310558 ±0%
##
## Against denominator:
##   Intercept only
## ---
## Bayes factor type: BFlinearModel, JZS
```

- plot(head(ob))

```
> plot(head(ob))
```

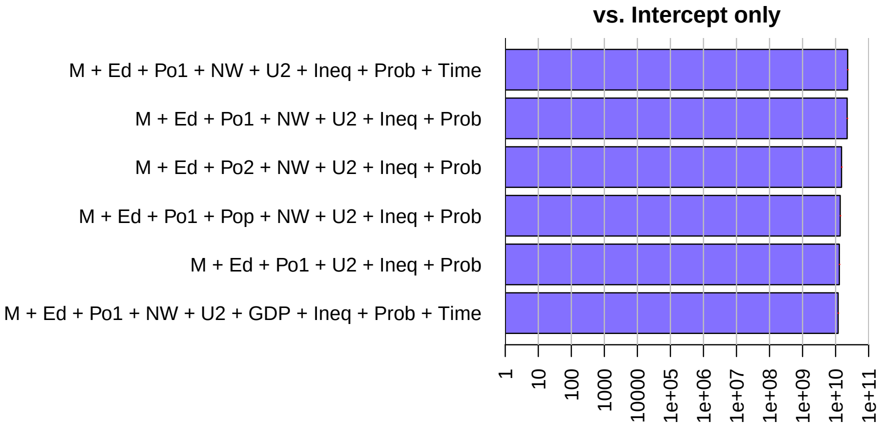

Figure 7: Bayes Factors using package BayesFactor

BayesVarSel

- Call:

```
> library(BayesVarSel)#Version: 1.7.1
> ob <- Bvs(formula="y~.", data=lUScrime, prior.betas="gZellner",
+           prior.models = "Constant", n.keep=10)
```

- `print()` and `summary()`

```
> ob
```

```
##
## Call:
## Bvs(formula = "y~.", data = lUScrime, prior.betas = "gZellner",
##     prior.models = "Constant", n.keep = 10)
##
## The 10 most probable models and their probabilities are:
##    M Ed Po1 Po2 LF M.F Pop NW U1 U2 GDP Ineq Prob Time So        prob
## 1  *  *   *                 *     *        *    *         0.024695841
## 2  *  *   *                 *     *        *    *    *    0.023987471
## 3  *  *       *             *     *        *    *         0.016258755
## 4  *  *   *                       *        *    *         0.014728156
## 5  *  *   *              *  *     *        *    *         0.013640810
## 6  *  *   *                 *              *    *    *    0.012415831
## 7  *  *   *                 *     *   *    *    *    *    0.010720692
## 8  *  *       *             *     *        *    *    *    0.010106903
## 9  *  *       *                   *        *    *         0.009834356
## 10 *  *   *              *  *              *    *         0.008994454
```

```
> summary(ob)
```

```
##
## Call:
## Bvs(formula = "y~.", data = lUScrime, prior.betas = "gZellner",
##     prior.models = "Constant", n.keep = 10)
##
## Inclusion Probabilities:
##      Incl.prob. HPM MPM
## M        0.8504   *   *
## Ed       0.9776   *   *
## Po1      0.6655   *   *
## Po2      0.4216
## LF       0.1567
## M.F      0.1603
## Pop      0.3302
## NW       0.6793   *   *
## U1       0.2083
## U2       0.5996   *   *
## GDP      0.3125
## Ineq     0.9975   *   *
## Prob     0.8963   *   *
## Time     0.3333
## So       0.2307
## ---
## Code: HPM stands for Highest posterior Probability Model and
##  MPM for Median Probability Model.
##
```

- `plotBvs()` The BayesVarSel package has its own plot function named as `plotBvs()` which shows different information depending of the value of the argument `option`. If option is `"d"` a Barplot of the posterior probabilities of each model dimension is printed. If we set `option="j"` we obtain the posterior probability of every two covariates being together in the model. `option="c"` present the posterior conditional probability of a variable (in rows) given an other variable (in columns) is already in the model. Finally `option="n"` plots the posterior conditional probability of a variable (in rows) being in the model given than other variable (in columns) is not.

```
> plotBvs(ob, option="d")
```

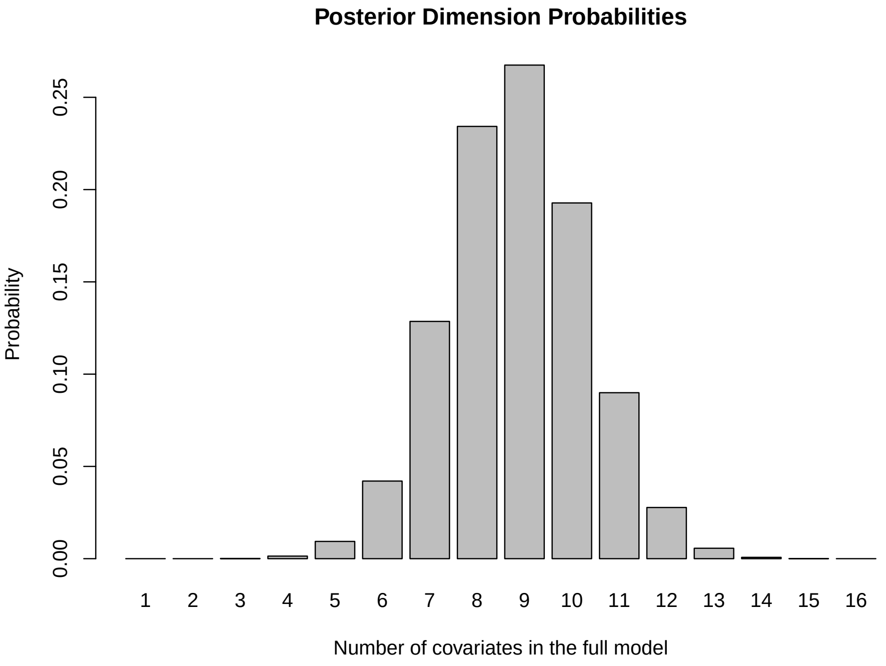

Figure 8: Posterior probabilities of each dimension

```
> ob.cond <- plotBvs(ob,option="c")
```

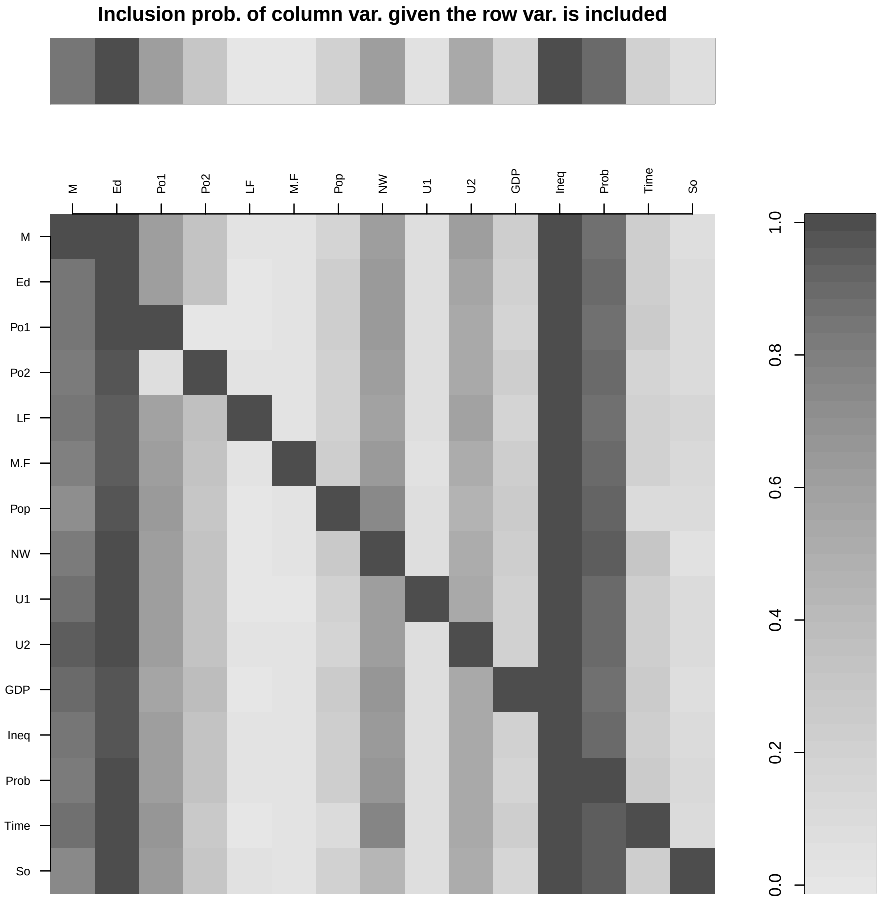

Figure 9: Posterior conditional probabilities

mombf

- Call:

```
> library(mombf)#Version: 1.8.3
> ob <- modelSelection(y=lUScrime$y, x=lUScrime[,-1],center=TRUE,scale=FALSE,
+                 priorCoef=zellnerprior(tau=length(lUScrime$y)),
+                 priorDelta=modelunifprior(),priorVar=igprior(alpha=.001,lambda=.001))
```

- `print()`

```
> ob
```

```
## msfit object with 15 variables and normal residual distribution
## Use postProb() to get posterior model probabilities
## Elements $margpp, $postMode, $postSample and $coef contain further information (see
help('msfit') and help('modelSelection') for details)
```

```
> ob$margpp #PIP
```

```
##         M        Ed       Po1       Po2        LF       M.F       Pop
## 0.8620971 0.9809170 0.6519821 0.4323931 0.1576645 0.1600437 0.3314804
##        NW        U1        U2       GDP      Ineq      Prob      Time
## 0.6939326 0.2110625 0.6176200 0.3199392 0.9978756 0.9058535 0.3410301
##        So
## 0.2317813
```

```
> ob$postMode #Posterior Mode
```

```
##    M   Ed  Po1  Po2   LF  M.F  Pop   NW   U1   U2  GDP Ineq Prob Time   So
##    1    1    1    0    0    0    0    1    0    1    0    1    1    1    0
```

BAS

- Call:

```
> library(BAS)#Version: 1.4.3
> ob <- bas.lm(formula="y~.", data = lUScrime, prior = "g-prior",
+               alpha = dim(lUScrime)[1], modelprior = uniform(),
+               method = "deterministic",n.models = 2^(dim(lUScrime)[2]))
```

- print()

```
> ob
```

```
##
## Call:
## bas.lm(formula = "y~.", data = lUScrime, n.models = 2^(dim(lUScrime)[2]), prior =
"g-prior", alpha = dim(lUScrime)[1], modelprior = uniform(), method = "deterministic")
##
##
## Marginal Posterior Inclusion Probabilities:
## Intercept M Ed Po1 Po2 LF
## 1.0000 0.8504 0.9776 0.6655 0.4216 0.1567
## M.F Pop NW U1 U2 GDP
## 0.1603 0.3302 0.6793 0.2083 0.5996 0.3125
## Ineq Prob Time So
## 0.9975 0.8963 0.3333 0.2307
```

- summary()

```
> summary(ob)
```

```
##           P(B != 0 | Y)  model 1    model 2    model 3    model 4
## Intercept     1.0000000  1.00000  1.0000000  1.0000000  1.0000000
## M             0.8503615  1.00000  1.0000000  1.0000000  1.0000000
## Ed            0.9775864  1.00000  1.0000000  1.0000000  1.0000000
## Po1           0.6654873  1.00000  1.0000000  0.0000000  1.0000000
## Po2           0.4215797  0.00000  0.0000000  1.0000000  0.0000000
## LF            0.1567424  0.00000  0.0000000  0.0000000  0.0000000
## M.F           0.1603299  0.00000  0.0000000  0.0000000  0.0000000
## Pop           0.3301836  0.00000  0.0000000  0.0000000  0.0000000
## NW            0.6792925  1.00000  1.0000000  1.0000000  0.0000000
## U1            0.2082608  0.00000  0.0000000  0.0000000  0.0000000
## U2            0.5996084  1.00000  1.0000000  1.0000000  1.0000000
## GDP           0.3124840  0.00000  0.0000000  0.0000000  0.0000000
## Ineq          0.9974810  1.00000  1.0000000  1.0000000  1.0000000
## Prob          0.8963338  1.00000  1.0000000  1.0000000  1.0000000
## Time          0.3333490  0.00000  1.0000000  0.0000000  0.0000000
## So            0.2306890  0.00000  0.0000000  0.0000000  0.0000000
## BF                   NA  1.00000  0.9713161  0.6583609  0.5963832
## PostProbs            NA  0.02470  0.0240000  0.0163000  0.0147000
## R2                   NA  0.82650  0.8420000  0.8229000  0.8046000
## dim                  NA  8.00000  9.0000000  8.0000000  7.0000000
## logmarg              NA 24.55728 24.5281755 24.1392769 24.0404071
##              model 5
## Intercept  1.0000000
## M          1.0000000
## Ed         1.0000000
## Po1        1.0000000
## Po2        0.0000000
## LF         0.0000000
## M.F         0.0000000
## Pop         1.0000000
## NW          1.0000000
## U1          0.0000000
## U2          1.0000000
## GDP         0.0000000
## Ineq        1.0000000
## Prob        1.0000000
## Time        0.0000000
## So          0.0000000
## BF          0.5523522
## PostProbs   0.0136000
## R2          0.8375000
## dim         9.0000000
## logmarg    23.9637095
```

- plot()

```
> plot(ob)
```

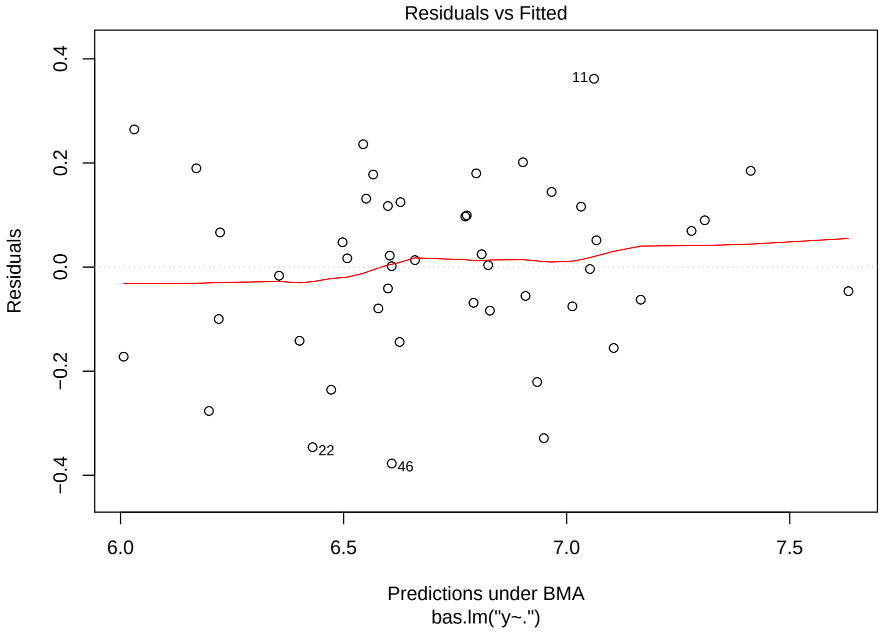

Figure 10: Residuals

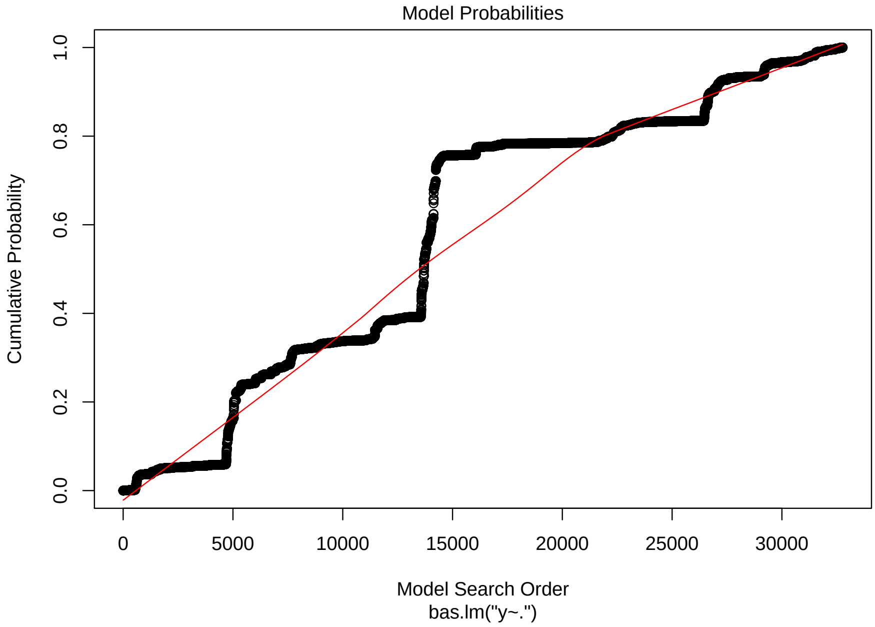

Figure 11: Cummulative Probabilities

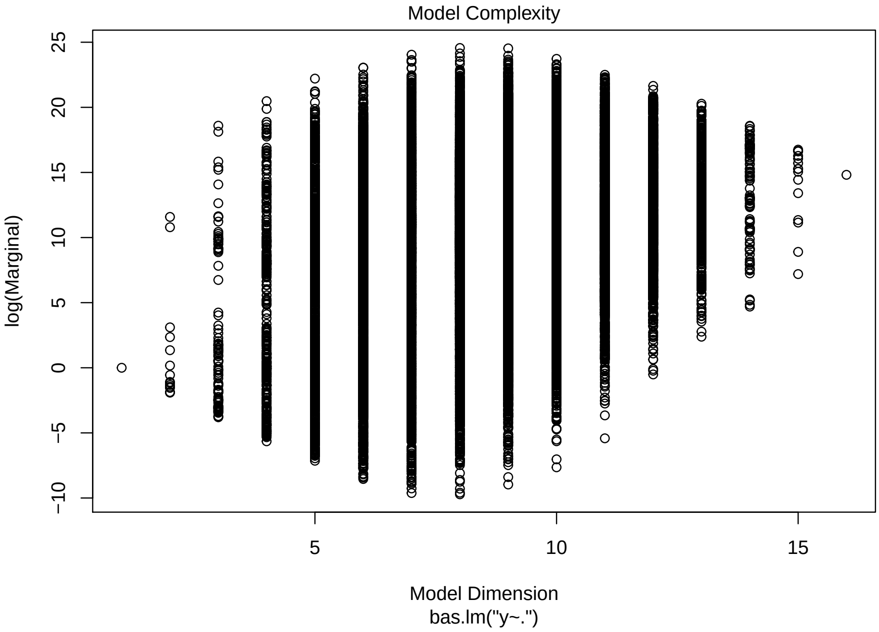

Figure 12: Probabilities by dimension

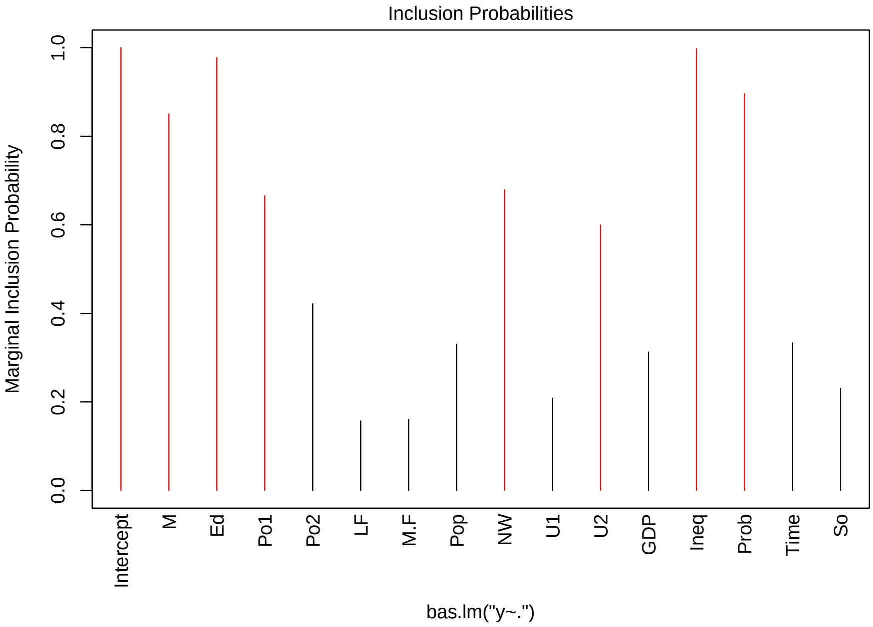

Figure 13: Posterior inclusion probabilities

Appendix B. Selected Packages. CRAN description

For the selected packages the actual descriptions on `CRAN` are:

BayesFactor (Morey et al., 2015)

- Version: 0.9.11-1
- Date: 2015-03-20
- Title: Functions to compute Bayes factor hypothesis tests for common research designs and hypotheses.
- Description: This package contains function to compute Bayes factors for a number of research designs and hypotheses, including t tests, ANOVA, and linear regression, and contingency tables.
- Basic references: Rouder et al. (2012) and Rouder et al. (2009)

BayesVarSel, (Garcia-Donato and Forte, 2015)

- Version: 1.6.1
- Date: 2015-01-26
- Title: Bayes Factors, Model Choice And Variable Selection In Linear Models
- Description: Conceived to calculate Bayes factors in linear models and then to provide a formal Bayesian answer to testing and variable selection problems. From a theoretical side, the emphasis in the package is placed on the prior distributions and BayesVarSel allows using a wide range of them (Jeffreys, 1961; Zellner and Siow, 1980, 1984; Zellner, 1986; Fernández et al., 2001; Liang et al., 2008; Bayarri et al., 2012). The interaction with the package is through a

friendly interface that syntactically mimics the wellknown `lm` command of R. The resulting objects can be easily explored providing the user very valuable information (like marginal, joint and conditional inclusion probabilities of potential variables; the highest posterior probability model, HPM; the median probability model, MPM) about the structure of the true -data generating- model. Additionally, BayesVarSel incorporates abilities to handle problems with a large number of potential explanatory variables through parallel and heuristic versions (Garcia-Donato and Martinez-Beneito, 2013) of the main commands.

- Basic references: Bayarri et al. (2012) and Garcia-Donato and Martinez-Beneito (2013)

BMS (Zeugner and Feldkircher, 2015)

- Version: 0.3.3
- Date: 2013-11-21
- Title: Bayesian Model Averaging Library.
- Description: Bayesian Model Averaging for linear models with a wide choice of (customizable) priors. Built-in priors include coefficient priors (fixed, flexible and hyper-g priors), 5 kinds of model priors, moreover model sampling by enumeration or various MCMC approaches. Post-processing functions allow for inferring posterior inclusion and model probabilities, various moments, coefficient and predictive densities. Plotting functions available for posterior model size, MCMC convergence, predictive and coefficient densities, best models representation, BMA comparison.
- Basic reference is Feldkircher and Zeugner (2009)

mombf(Rossell et al., 2014)

- Version: 1.6.0

- Date: 2015-07-02
- Title: Moment and Inverse Moment Bayes Factors
- Description: Model selection and parameter estimation based on non-local priors. Routines are provided to compute Bayes factors, marginal densities and to perform variable selection in regression setups. Routines to evaluate prior densities, distribution functions, quantiles and modes are included.