BSGS: Bayesian Sparse Group Selection

by Kuo-Jung Lee and Ray-Bing Chen

Abstract An R package **BSGS** is provided for the integration of Bayesian variable and sparse group selection separately proposed by Chen et al. (2011) and Chen et al. (in press) for variable selection problems, even in the cases of large p and small n. This package is designed for variable selection problems including the identification of the important groups of variables and the active variables within the important groups. This article introduces the functions in the **BSGS** package that can be used to perform sparse group selection as well as variable selection through simulation studies and real data.

Introduction

Variable selection is a fundamental problem in regression analysis, and one that has become even more relevant in current applications where the number of variables can be very large, but it is commonly assumed that only a small number of variables are important for explaining the response variable. This sparsity assumption enables us to select the important variables, even in situations where the number of candidate variables is much greater than the number of observations.

BSGS is an R package designed to carry out a variety of Markov chain Monte Carlo (MCMC) sampling approaches for variable selection problems in regression models based on a Bayesian framework. In this package, we consider two structures of variables and create functions for the corresponding MCMC sampling procedures. In the first case where the variables are treated individually without grouping structure, two functions, `CompWiseGibbsSimple` and `CompWiseGibbsSMP`, are provided to generate the samples from the corresponding posterior distribution. In the second case, it is assumed that the variables form certain group structures or patterns, and thus the variables can be partitioned into different disjoint groups. However, only a small number of groups are assumed to be important for explaining the response variable, i.e. the condition of the group sparsity, and we also assume that sparse assumption is held for the variables within the groups. This problem is thus termed a sparse group selection problem Simon et al. (2013); Chen et al. (in press), and the goal is to select the important groups and also identify the active variables within these important groups simultaneously. There are two functions to handle the sparse group selection problems, `BSGS.Simple` and `BSGS.Sample`, which are used to generate the corresponding posterior samples. Once the posterior samples are available, we then can determine the active groups and variables, estimate the parameters of interest and make other statistical inferences.

This paper is organized as follows. We first briefly introduce statistical models that are used to deal with the problems of variable selection in the **BSGS** package. We then describe the tuning parameters in the functions in the **BSGS** package. Two simulations are used to illustrate the details of the implementations of the functions. Finally we present a real economic example to demonstrate the **BSGS** package.

Framework of BSGS

We start with the introduction of individual variable selection problems, and then turn our attention to sparse group selection. For completeness, we describe the model and priors so that one may easily change the inputs of functions in the **BSGS** package for any purpose.

Variable selection

Consider a linear regression model given by

$$Y = \mathbf{X}\beta + \varepsilon, \tag{1}$$

where $Y = (Y_1, \ldots, Y_n)'$ is the response vector of length n, $\mathbf{X} = [X_1, \ldots, X_p]$ is an $n \times p$ design matrix, with X_i as the corresponding i-th variable (regressor) as a potential cause of the variation in the response, β is the corresponding unknown $p \times 1$ coefficient vector, and $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)'$ is the error term, which is assumed to have a normal distribution with a zero mean, and the covariance matrix $\sigma^2 I_n$, I_n is the $n \times n$ identity matrix. To achieve variable selection, we select a subset of $X_1, \ldots, X_p$ to explain the response Y. For this purpose, following the stochastic search variable selection method George and McCulloch (1993), a latent binary variable γ_i taking the value of 0 and 1 is introduced

to indicate whether or not the corresponding variable is selected. That is, if $\gamma_i = 1$, the variable X_i is selected, and otherwise it is not selected.

In this Bayesian framework we basically follow the prior assumption in Chen et al. (2011). We assume the prior distribution of γ_i is a Bernoulli distribution with probability $1 - \rho_i$, and then given γ_i, we assume the prior for β_i is a mixture of point mass at 0 denoted by δ_0 and a normal distribution as follows

$$\beta_i|\gamma_i \sim (1-\gamma_i)\delta_0 + \gamma_i N(0, \tau_i^2), \tag{2}$$

where τ_i is a pre-specified constant. Moreover, (γ_i, β_i) are assumed to be independent for $i = 1, \ldots, p$. Lastly, σ^2 is set to follow an inverse Gamma distribution,

$$\sigma^2 \sim \mathcal{IG}(\nu/2, \nu\lambda/2). \tag{3}$$

Based on the model setting given above, there are two Gibbs samplers for this variable selection. The first procedure is the componentwise Gibbs sampler (CGS) which was introduced by Geweke (1996) and also mentioned in Chen et al. (2011). The other is the stochastic matching pursuit (SMP) algorithm Chen et al. (2011) in which the variables compete to be selected.

Sparse group selection

In traditional variable selection problems, each variable X_i is treated individually; however, in some real applications, a group of variables that behave similarly may be more meaningful in explaining the response. In other words, a group of variables potentially plays a more important role in explaining the response than a single variable can. The variable selection problem thus becomes a group selection one. In group selection problems, the variables within the selected groups are all treated as important. However, this assumption might not be held in practice such as the climate application in Chatterjee et al. (2012). Instead of the group selection problem, we thus consider approaches to sparse group selection, in which the sparse assumption is held for groups and the variables within groups.

Here the goal is not only to select the influential groups, but also to identify the important variables within these. To this end, a Bayesian group variable selection approach, the Group-wise Gibbs sampler (GWGS) Chen et al. (in press), is applied. Suppose that, in terms of expert or prior knowledge, the explanatory variables X_i's are partitioned into g non-overlapping groups in a regression model. Each group l contains $j = 1, \ldots, p_l$ variables with $\sum_{l=1}^{g} p_l = p$. Now the model is rewritten as

$$Y = \sum_{l=1}^{g} \mathbf{X}_l \beta_l + \varepsilon, \tag{4}$$

where $\mathbf{X}_l = [X_{l1}, \cdots, X_{lp_l}]$ is the $n \times p_l$ sub-matrix of $\mathbf{X}$, $\beta_l = (\beta_{l1}, \cdots, \beta_{lp_l})'$ is the $p_l \times 1$ coefficient vector of group l. The GWGS works by introducing two nested layers of binary variables to indicate whether a group or a variable is selected or not. At the group-selection level, we define a binary variable $\eta_l = 1$ if group l is selected, and $\eta_l = 0$ otherwise. At the variable-selection level, we define another binary variable $\gamma_{li} = 1$ if the variable i within group l, X_{li}, is selected, and $\gamma_{li} = 0$ otherwise.

We assume the group indicator, η_l, has the Bernoulli distribution with the probability $1 - \theta_l$. Within group l, the prior distribution of γ_{li} conditional on the indicator η_l is defined as

$$\eta_l \sim \text{Ber}(1-\theta_l), \quad \gamma_{li}|\eta_l \sim (1-\eta_l)\delta_0 + \eta_l \text{Ber}(1-\rho_{li}), \tag{5}$$

where δ_0 is a point mass at 0 and $\text{Ber}(1 - \rho_{li})$ is Bernoulli distributed with the probability $1 - \rho_{li}$. Equation (5) implies that if the l-th group is not selected, it turns out that $\gamma_{li} = 0$ for all i. The prior distribution of the coefficient β_{li} given η_l and γ_{li} is given by

$$\beta_{li}|\eta_l, \gamma_{li} \sim (1-\eta_l\gamma_{li})\delta_0 + \eta_l\gamma_{li}\mathcal{N}(0, \tau_{li}^2),$$

where τ_{li} is a pre-specified value Chen et al. (2011). Finally, the variance σ^2 is assumed to have an inverse Gamma distribution, that is, $\sigma^2 \sim \mathcal{IG}(\nu/2, \nu\lambda/2)$. We also assume $(\eta_l, \gamma_{l1}, \beta_{l1}, \ldots, \gamma_{lp_l}, \beta_{lp_l})$, $l = 1, \cdots, g$, are *a priori* independent.

Two sampling procedures are proposed in Chen et al. (in press) for Bayesian sparse group selection. The first is the GWGS. In the GWGS, simulating the indicator variable η_l from the posterior distribution would be computationally intensive, especially when the number of variables within the group is large. To address this issue, Chen et al. (in press) proposed a modified and approximation approach, a sample version of GWGS. In this a Metropolis-Hastings algorithm is adopted to replace the Gibbs sampling method in GWGS.

Implementation

In this section, we describe the default tuning parameters and some details in the implementation of the functions in **BSGS**.

Hyperparameter set-up

- **The tuning parameters, ν and λ:**
  The parameters in the prior distribution of σ^2 are suggested by George and McCulloch (1993) setting the default values of $\nu = 0$ and λ being any positive value. A data-driven choice for this is proposed by Chipman et al. (1997), which sets the value of ν around 2 and the value of λ is set up to be the 99% quantile of the prior of σ^2 that is close to $\sqrt{\text{Var}(Y)}$. In addition, George and McCulloch (1997) simply set $\nu = 10$ and $\lambda = \sqrt{\text{Var}(Y)}$. In our experiences, a larger value of ν tends to result in a larger estimate of σ.
- **The parameter τ:**
  Now we consider the assignment of the value of τ, the prior variance of the regression coefficient for the active variable. It was found that the larger the value of τ, the smaller the conditional probability of $\gamma = 1$ is. As a result, a large value of τ favors a more parsimonious mode. In contrast, a small value of τ would yield more complex models.
- **The parameters ρ and θ:**
  The default of the prior inclusion probabilities of groups and variables is set equal to 0.5. In the case when p is much greater than n and only a small number of variables are considered active, we would assign a larger values to ρ and θ to reflect the prior belief of sparsity.

Stopping rule

The posterior distribution is not available in explicit form so we use the MCMC method, and specifically Gibbs sampling to simulate the parameters from this distribution Brooks et al. (2011). To implement the Gibbs sampler, the full conditional distributions of all parameters must be determined. A derivation of the full conditional distributions is provided in Chen et al. (in press). When these have been obtained, the parameters are then updated individually using a Gibbs sampler (where available), or a Metropolis-Hastings sampling algorithm. An MCMC sample will converge to the stationary distribution, i.e. the posterior distribution. We use the batch mean method to estimate the Monte Carlo standard error (MCSE) of the estimate of σ^2 and then decide to stop the simulation once the MCSE is less than a specified value, cf. Flegal et al. (2008). The default minimum number of iterations is 1000. If the MCSE does not achieve the prespecified value, an extra 100 iterations are run until the MCSE is less than the prespecified value. The sample can then be used for statistical inference.

Statistical inference

- **Variable and group selection criteria:**
  In this package, we adopt the median probability criterion proposed by Barbieri and Berger (2004) for group and variable selections. Specifically, for the variable selection problem, we estimate the posterior inclusion probability $P(\gamma_i = 1|Y)$ from the posterior samples and then the i-th variable is selected into the model if the estimated posterior probability is larger than or equal to $1/2$. Here instead of $1/2$, this cut-off value is treated as a tuning parameter, α, which can be specified by users. For the sparse group selection problem, the estimated posterior probability of the l-th group is greater than or equal to α_g, i.e. $P(\eta_l = 1|Y) \geq \alpha_g$, we then include $\mathbf{X}_l$ into the model. Suppose the l-th group is selected, then the i-th variable within this group is selected if $P(\gamma_{li} = 1|\eta_l = 1, Y) \geq \alpha_i$. Here α_g and α_i are two pre-specified values between 0 and 1.
- **Posterior estimates of regression coefficients:**
  We use the Rao-Blackwell method to estimate β by

$$\hat{\beta} = E(\beta|y) \approx \frac{1}{N_M} \sum_{m=1}^{M} \beta_m,$$

  where β_m is the sample in mth iteration, M is the number of iterations, and N_M is the number of nonzero β_m.

Evaluation of model estimation

Regarding the stability of the estimation, we compare the accuracy of selection of the variables by the following measures in the simulation studies: the True Classification Rate (TCR), the True Positive Rate (TPR), and the False Positive Rate (FPR). These are defined as follows

$$\begin{aligned}\text{TCR} &= \frac{\text{number of correctly selected variables}}{\text{number of variables}};\\ \text{TPR} &= \frac{\text{number of correctly selected variables}}{\text{number of active variables}};\\ \text{FPR} &= \frac{\text{number of falsely selected variables}}{\text{number of inactive variables}}.\end{aligned}$$

TCR is an overall evaluation of accuracy in the identification of the active and inactive variables. TPR is the average rate of active variables correctly identified, and is used to measure the power of the method. FPR is the average rate of inactive variables that are included in the regression, and it can be considered as the type I error rate of the selection approach. In these three criteria, it is preferred to have a larger value of TCR or TPR, or a smaller value of FPR.

We also report the mean squared error (MSE),

$$\text{MSE} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2/n,$$

where $\hat{y}_i$ is the prediction of y_i. This is used to evaluate whether the overall model estimation has a good fit with regard to the data set.

Examples

Two simulations and a real example are provided to demonstrate the use of functions in the **BSGS** package.

Simulation I

The traditional variable selection problem is illustrated in this simulation. We use an example to illustrate the functions `CompWiseGibbsSimple` and `CompWiseGibbsSMP` corresponding to the CGS and SMP sampling procedures to simulate the sample from the posterior distribution. Based on the samples, we can decide which variable is important in the regression model. In this simulation, the data Y of length $n = 50$ is generated from a normal distribution with a mean $X\beta$ and $\sigma^2 = 1$. We assume $\beta = (3, -3.5, 4, -2.8, 3.2, \overbrace{0, \ldots, 0}^{p-5})$, $p = 100$, and X is from a multivariate normal distribution with a mean 0 and the covariance matrix Σ as the identity matrix. We then generate the responses based on (1).

```
require(BSGS)
set.seed(1)

## Generate data
num.of.obs <- 50
num.of.covariates <- 100

beta.g <- matrix(c(3, -3.5, 4, -2.8, 3.2, rep(0, num.of.covariates-5)), ncol = 1)
r.true <- (beta.g != 0) * 1

pair.corr <- 0.0 ## pair correlations between covariates
Sigma <- matrix(pair.corr, num.of.covariates, num.of.covariates)
diag(Sigma) <- rep(1, num.of.covariates)
x <- mvrnorm(n = num.of.obs, rep(0, num.of.covariates), Sigma)

sigma2 <- 1
mu <- x %*% beta.g
y <- rnorm(num.of.obs, mu, sigma2)
```

Regarding to the hyperparameters, we simply set $\tau^2 = 10$, but one can use cross-validation to tune the parameter Chen et al. (2011). Following George and McCulloch (1997), we let $\nu = 10$ and

$\lambda = \sqrt{\text{Var}(Y)}$. Without any prior information on which variable is important, we let prior inclusion probability $\rho_i = 0.5$ for each variable. We use the rigid regression estimates as the initial values of regression coefficients βs.

```
## Specify the values of hyperparameters and initial values of parameters

tau2 <- 10       ## hyperparameter in Eq. (1)
nu0 <- 10        ## hyperparameter in Eq. (2)
lambda0 <- sd(y) ## hyperparameter in Eq. (2)


## Initial values for parameters
beta.initial <- t(solve(t(x) %*% x + diag(1/5, num.of.covariates)) %*% t(x) %*% y)
sigma2.initial<-1
r.initial <- rbinom(num.of.covariates, 1, 1)
```

Now two functions, `CompWiseGibbsSimple` and `CompWiseGibbsSMP`, are applied to simulate the samples from the posterior distribution. The minimum number of iterations is 1000. The simulation will stop when the MCSE of the estimate of σ^2 is less than 0.1 and otherwise an extra 100 iterations are run until the MCSE is less than this. For stability, we update `num.of.inner.iter.default = 10` times for β and γ and take the last ones as a sample before updating σ^2.

```
num.of.iteration <- 1000
num.of.inner.iter.default <- 10
MCSE.Sigma2.Given <- 0.1

## Apply two sampling functions to generate the samples from the
## posterior distribution.

outputCGS <- CompWiseGibbsSimple(y, x, beta.initial, r.initial, tau2,
  rho, sigma2.initial, nu0, lambda0, num.of.inner.iter.default,
  num.of.iteration, MCSE.Sigma2.Given)

outputSMP <- CompWiseGibbsSMP(y, x, beta.initial, r.initial, tau2,
  rho, sigma2.initial, nu0, lambda0, num.of.inner.iter.default,
  num.of.iteration, MCSE.Sigma2.Given)
```

Once the simulation stops, the posterior samples are used to estimate the posterior quantities of interest. One can then check the number of iterations and computational times for both approaches.

```
## Output from the component-wise Gibbs sampling procedure
outputCGS$Iteration
[1] 1000
outputCGS$TimeElapsed
   user  system elapsed
 61.558   4.815  66.813

## Output from the component-wise Gibbs sampling procedure
outputSMP$Iteration
[1] 1000
outputSMP$TimeElapsed
   user  system elapsed
 45.970   5.907  52.383
```

One can then use the function `CGS.SMP.PE` to identify the important variables and to estimate the parameters. Due to the limitations of space, we do not include the estimates here. A variable i is considered important if the posterior probability of its indicator variable $\gamma_i = 1$ is greater than or equal to $\alpha_i = 1/2$. Once the critical point is decided, two functions `TCR.TPR.FPR.CGS.SMP` and `MSE.CGS.SMP` are carried out to evaluate the performance on the model estimations in terms of TCP, TPR, FTP and MSE.

```
## Output from the component-wise Gibbs sampling procedure
CGS.SMP.PE(outputCGS)

MSE.CGS.SMP(outputCGS, Y=y, X=x)
[1] 2.921087
```

```
TCR.TPR.FPR.CGS.SMP(outputCGS, r.true, 0.5)
$TCR
[1] 1

$TPR
[1] 1

$FPR
[1] 0

## Output from the component-wise Gibbs sampling procedure
CGS.SMP.PE(outputSMP)

MSE.CGS.SMP(outputSMP, Y=y, X=x)
[1] 3.751427

TCR.TPR.FPR.CGS.SMP(outputSMP, r.true, 0.5)
$TCR
[1] 1

$TPR
[1] 1

$FPR
[1] 0
```

Simulation II

We provide another stimulation to illustrate the use of functions for sparse group selection. In the following simulation, the data Y of length $n = 50$ is generated from a normal distribution with a mean $X\beta$ and $\sigma^2 = 1$, where X is from a multivariate distribution with mean 0 and covariance Σ with the pair correlation between variables equal to zero. There are 10 groups of variables. Each group contains 10 variables and some of them are active. More specifically, $\mathbf{X}_l = X_{l1}, \ldots, X_{lp_l}, k = 1, \ldots, 10$ and $p_l = 10$ for all l. We assume the group $l = 1, 2, 5, 8$ are active. Variables $p_1 = 7, 8, 9$ in the group $l = 1$, $p_2 = 1, 2$ in the group $l = 2$, $p_5 = 3$ in the group $l = 5$, and $p_8 = 7$ in the group $l = 8$ are active. The response is generated via (4).

```
require(BSGS)

set.seed(1)

Num.Of.Iteration <- 1000
Num.of.Iter.Inside.CompWise <- 10
num.of.obs <- 50
num.of.covariates <- 100
num.of.group.var <- 10
Group.Index <- rep(1:10, each = 10)

nu <- 0
lambda <- 1
pair.corr <- 0.

Sigma <- matrix(pair.corr, num.of.covariates, num.of.covariates)
diag(Sigma) <- rep(1,num.of.covariates)

X <- mvrnorm(n = num.of.obs, rep(0, num.of.covariates), Sigma)

beta.true <- rep(0, num.of.covariates)
beta.true[c(7, 8, 9, 11, 12, 43, 77)] <- c(3.2, 3.2, 3.2, 1.5, 1.5, -1.5, -2)
beta.true <- cbind(beta.true)
r.true <- (beta.true != 0) * 1

sigma2.true <-1
```

```
Y <- rnorm(num.of.obs, X %*% beta.true, sigma2.true)
```

Here we suppose that we have no prior information on the parameters. We let $\nu = 0$ and $\lambda = 1$ which corresponds to the non-informative prior for σ^2. Also, we let $\rho_i = 0.5$ and $\theta = 0.5$ for prior inclusion probabilities of groups and variables. Finally, we let $\tau = 1$ for the variance of each regression coefficient.

```
## hyperparameters
tau2.value <- rep(1, num.of.covariates)
rho.value <- rep(0.5, num.of.covariates)
theta.value <- rep(0.5, num.of.group.var)
```

With the reasonable assignment of the initial values of parameters, we apply two functions, `BSGS.Simple` and `BSGS.Sample`, to estimate the posterior quantities of interest and in turn identify the important groups and variables. For illustration, we stop the simulation when the MCSE of estimate of σ^2 is less than 0.5.

```
## Initial values and stopping point
r.value <- rbinom(num.of.covariates, 1, 0.5)
eta.value <- rbinom(num.of.group.var, 1, 0.5)
beta.value <- cbind(c(t(solve(t(X) %*% X +
  diag(1/5, num.of.covariates)) %*% t(X) %*% Y) )) # beta.true
sigma2.value <- 1
MCSE.Sigma2.Given <- 0.5

## Apply two sampling approaches to generate samples
outputSimple <- BSGS.Simple(Y, X, Group.Index, r.value, eta.value, beta.value,
  tau2.value, rho.value, theta.value, sigma2.value, nu, lambda,
  Num.of.Iter.Inside.CompWise, Num.Of.Iteration, MCSE.Sigma2.Given)

outputSample <- BSGS.Sample(Y, X, Group.Index, r.value, eta.value, beta.value,
  tau2.value, rho.value, theta.value, sigma2.value, nu, lambda,
  Num.of.Iter.Inside.CompWise, Num.Of.Iteration, MCSE.Sigma2.Given)
```

One can easily use the function `BSGS.PE` to estimate the posterior probabilities of $\eta_l = 1$ and $\gamma_{li} = 1|\eta_l = 1$ based on the samples generated from the posterior distribution. To investigate which sampling approach provides a better model estimation, one can calculate MSE by the function `MSE.BSGS`. Furthermore, the function `TCR.TPR.FPR.BSGS` is used to evaluate the performance on the accuracy of selection on variables. All functions are illustrated as follows. We take $\alpha_i = \alpha_g = 0.5$ in this example.

```
## The posterior quantities estimated by two sampling approaches respectively.

## Output from the simple version of BSGS
outputSimple$Iteration
[1] 1000
outputSimple$TimeElapsed
   user  system elapsed
238.755   0.007 239.037
BSGS.PE(outputSimple)$eta.est
    1     2     3     4     5     6     7     8     9    10
1.000 1.000 0.115 0.200 0.926 0.099 0.108 0.991 0.053 0.171
MSE.BSGS(outputSimple, Y=Y, X=X)
[1] 0.574

TCR.TPR.FPR.BSGS(outputSimple, r.true, 0.5)
$TCR
[1] 0.97

$TPR
[1] 1

$FPR
[1] 0.0323

## Output from the sample version of BSGS
```

```
outputSample$Iteration
[1] 3700
outputSample$TimeElapsed
   user  system elapsed
105.535   0.163 105.822
BSGS.PE(outputSample)$eta.est
      1       2       3       4       5       6       7       8       9      10
1.00000 0.90514 0.04189 0.02459 0.88000 0.01000 0.00486 0.91135 0.00486 0.00730
MSE.BSGS(outputSample, Y=Y, X=X)
[1] 2.16
TCR.TPR.FPR.BSGS(outputSample, r.true, 0.5)
$TCR
[1] 0.97

$TPR
[1] 1

$FPR
[1] 0.0323
```

One may be interested in how the different stopping points would affect the computational effort and model estimation. We thus compare the computational times and accuracy of parameter estimation in terms of TCR, TPR, FPR, and MSE for two different sampling approaches by using different MCSEs to stop the simulation and the results are shown in Tables 1 and 2. It has been found that for the case of $n = 50$ and $p = 100$ the simple version would converge faster than the sample version. However, the sample version would require less effort to produce the sample than the simple version.

MCSE	# of iterations	runtime (in sec)	MSE	TCR	TPR	FPR	$\hat{\sigma}^2$
0.5	1000	239	0.58	0.97	1	0.03	1.143
0.25	1000	239	0.58	0.97	1	0.03	1.143
0.1	1100	263	0.55	0.98	1	0.02	1.112

Table 1: Results based on simple version procedure for sparse group selection for different stopping points.

MCSE	# of iterations	runtime (in sec)	MSE	TCR	TPR	FPR	$\hat{\sigma}^2$
0.5	3700	105	2.16	0.97	1	0.03	1.418
0.25	9500	226	0.58	0.97	1	0.03	1.418
0.1	31200	4148	0.48	0.97	1	0.03	1.047

Table 2: Results based on sample version procedure for sparse group selection for different stopping points.

We further investigate the accuracy of parameter estimations and the selection of the variables in terms of MSE, TCR, TPR, and FPR for the two sampling approaches when the number of covariates increases but each group has 10 variables. We consider $p = 300, 500$, and 1000 and the simulation is terminated when the MCSE of the estimate of σ^2 is less than 0.5 for illustration. One may use different values of MCSE, but more computational time may be needed. Table 3 shows that the simulation stops with the same number of iterations and the parameter estimates and accuracy of variable selection show little difference. On the other hand, the results in Table 4 for the sample version GWGS show that more iterations are needed, under the same stopping rule. The two approaches thus perform equally well on the selection of variables, but by comparing MSEs it is evident that the simple version of GWGS outperforms with regard to the predictions. Although the sample version has less computational intensity, it needs more iterations to achieve the stopping point. If one is interested in selecting important variables, both approaches are effective. But if one is interested in choosing a model which fits the data well, it is thus suggested one uses the simple version approach.

Next, we compare the computational time when the number of variables in the group increases. We perform an experiment in which there are seven groups, each containing 15 variables. The assignments of hyperparameters are the same as those in Simulation II. Table 5 shows the computational time plus

# of covariates	# of iterations	runtime (in sec)	MSE	TCR	TPR	FPR	$\hat{\sigma}^2$
300	1000	212	0.56	0.95	0.86	0.05	1.94
500	1000	273	0.99	0.98	1	0.02	1.803
1000	1000	263	0.55	0.98	1	0.02	1.112

Table 3: Results based on the simple version procedure for sparse group selection for different numbers of covariates, with pair-correlation equal to 0.

# of covariates	# of iterations	runtime (in sec)	MSE	TCR	TPR	FPR	$\hat{\sigma}^2$
300	2100	127	1.63	0.99	1	0.01	1.819
500	14600	2258	4.73	0.97	1	0.03	1.601
1000	39900	39567	8.90	0.99	1	<0.01	2.158

Table 4: Results based on the sample version procedure for sparse group selection for different numbers of covariates, with pair-correlation equal to 0.

the model estimations. It can be seen that the sample version is strongly recommended when the number of variables within a group is greater than 15.

Sampling version	# of iterations	runtime (in sec)	MSE	TCR	TPR	FPR	$\hat{\sigma}^2$
Simple	1000	8330	0.11	0.93	1	0.08	0.90
Sample	2400	71	0.75	0.99	1	0.01	1.06

Table 5: Comparison between simple and sample versions for sparse group selection when the number of covariates within a group is 15.

A real economic example

This subsection further illustrates the functions in the **BSGS** based on an economic dataset from Rose and Spiegel Rose and Spiegel (2010, 2011, 2012) which is available at `http://faculty.haas.berkeley.edu/arose` The response variable is the 2008-2009 growth rate for the crisis measure. Rose and Spiegel originally consider 119 explanatory factors for the crisis for as many as 107 countries, but there are missing data for a number of these.

```
require(BSGS)
## the whole data set
data(Crisis2008)
```

To maintain a balanced data set, we use 51 variables for a sample of 72 countries. For more information about the balanced data, please see the description of '`Crisis2008`' in the **BSGS**. The balanced data is then analyzed to illustrate the main sampling function `BSGS.Simple` to simulate the sample from the posterior distribution. In the analysis, we demean the response so that it is not necessary to include the intercept into the design matrix. All variables are standardized except the dummy variables.

```
set.seed(1)
data(Crisis2008BalancedData)

var.names <- colnames(Crisis2008BalancedData)[-1]
country.all <- rownames(Crisis2008BalancedData)
cov.of.interest <- colnames(Crisis2008BalancedData)[-1]

Y <- Crisis2008BalancedData[, 1]
Y <- Y - mean(Y)
X <- Crisis2008BalancedData[, -1]

if (NORMALIZATION) {
  dummy.variable <- cov.of.interest[lapply(apply(X, 2, unique), length) == 2]
  non.dummy.X <- X[, !(colnames(X) %in% dummy.variable)]
```

```
  X.normalized <- apply(non.dummy.X, 2, function(XX) (XX - mean(XX))/sd(XX))
  X[, !(colnames(X) %in% dummy.variable)] <- X.normalized
}
```

As discussed in Ho (in press), these variables can be classified into the nine theoretical groups of the crisis' origin (the number in parentheses indicates the number of variables considered in the group): principal factors (10), financial policies (three), financial conditions (four), asset price appreciation (two), macroeconomic policies (four), institutions (11), geography (four), financial linkages (one), and trade linkages (12). Based on this information, we assign a group index to each variable.

```
Group.Index <- rep(1:9, c(10, 3, 4, 2, 4, 11, 4, 1, 12))
```

Since the number of covariates within the group is moderate, it is recommended that the simple version GWGS be applied to generate the samples. In this example, we have tested different values of τ^2 and finally we set $\tau^2 = 10$ due to the minimal MSE's. Here the stopping rule is when the MCSE of the estimate of σ^2 is less than or equal to 0.1 for the simple version. No prior information is provided to indicate which group or variable is more important, so we let $\theta = 0.5$ and $\eta = 0.5$. We let $\nu = 0$ and $\lambda = 1$ resulting in a non-informative prior for σ^2. In each group, we will update parameters "`Num.of.Iter.Inside.CompWise = 100`" times within a group for stability.

```
Num.Of.Iteration <- 1000
Num.of.Iter.Inside.CompWise <- 100
num.of.obs <- nrow(X)
num.of.covariates <- ncol(X)
num.of.groups <- length(unique(Group.Index))
nu <- 0
lambda <- 1
beta.est <- lm(Y ~ X - 1)$coef
beta.est[is.na(beta.est)] <- 0
beta.value <- beta.est
tau2.value <- rep(1, num.of.covariates)

sigma2.value <- 1

r.value <- rep(0, num.of.covariates)
eta.value <- rep(0, num.of.groups)

tau2.value <- rep(10, num.of.covariates)

rho.value <- rep(0.5, num.of.covariates)
theta.value <- rep(0.5, num.of.groups)

MCSE.Sigma2.Given <- 0.5

outputCrisis2008 <- BSGS.Simple(Y, X, Group.Index, r.value, eta.value, beta.value,
  tau2.value, rho.value, theta.value, sigma2.value, nu, lambda,
  Num.of.Iter.Inside.CompWise, Num.Of.Iteration, MCSE.Sigma2.Given)
```

The posterior probabilities of η_l and γ_{li} are shown in Figure 1. Based on the median probability criterion, the group (or variable) whose posterior probability is larger than or equal to 0.5 is selected as an important group (or variable). It is found that only the groups, "Financial Policies" and "Trade Linkages" are considered to have an influence on the economic crisis. Moreover, within each important group, we also find that only some of the variables may make a contribution to explain the response.

Summary

This paper illustrated the usage of a new R package, **BSGS**, for identifying the important groups of variables and important variables in linear regression models. Furthermore, **BSGS** can be easily implemented with problems of a large p and small n. The grouping idea is also applicable to other regression and classification settings, for example, the multi-response regression and multi-class classification problems. We envision future additions to the package that will allow for extensions to these models.

We are confident that this package can be applied to many important real-world problems by keeping flexibility with regard to selecting variables within a group based on the hierarchical assignment of two layers of indicator variables. For instance, in the gene-set selection problem, a biological

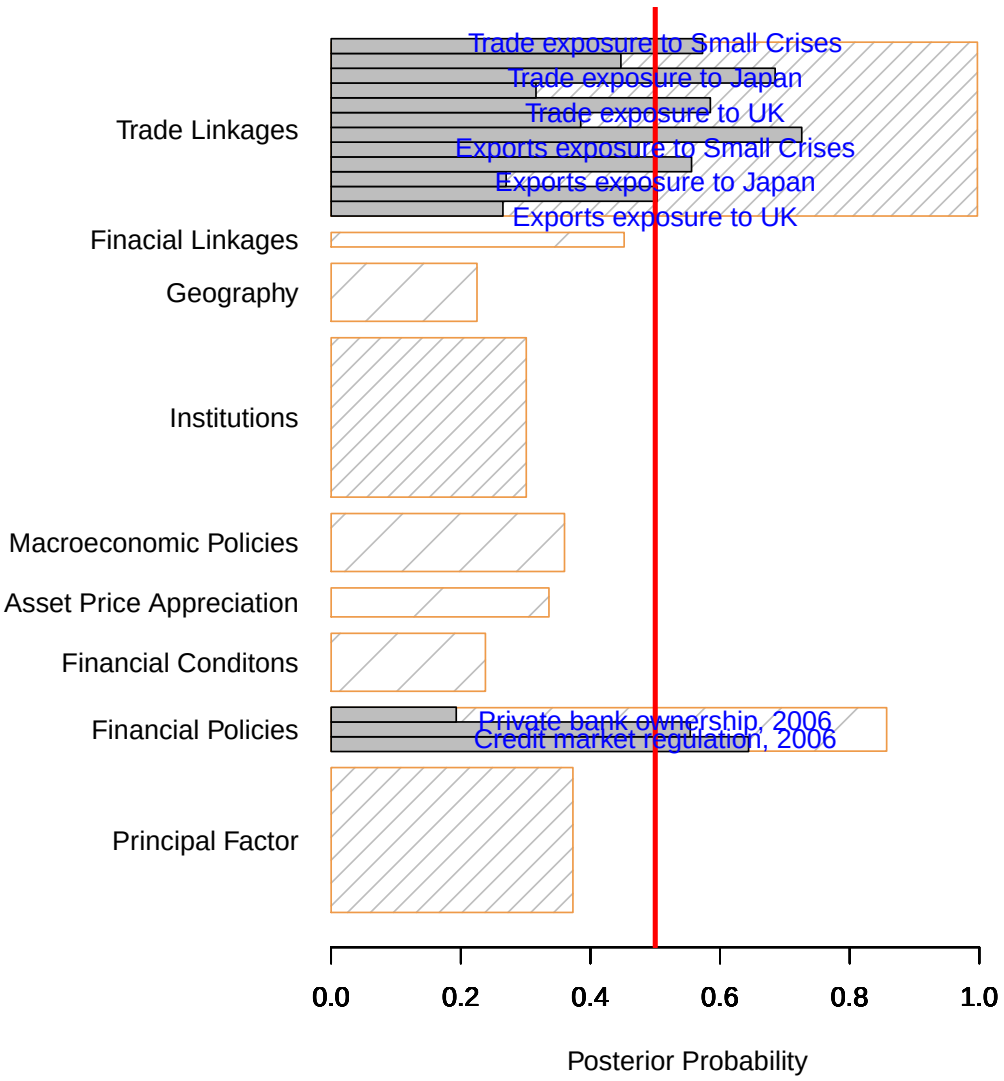

Figure 1: The group selection results for the cross-country severity of the crisis. The orange border indicates the posterior probability for the group selection, and the gray bar indicates the posterior probability for variable selection in the selected groups.

pathway may be related to a certain biological process, but it may not necessarily mean all the genes in the pathway are all related to the biological process. We may want not only to remove unimportant pathways effectively, but also identify important genes within important pathways.

Acknowledgment

This work is partially supported by the Ministry of Science and Technology under grant MOST 103-2633-M-006 -002 (Lee); the Ministry of Science and Technology under grant MOST 103-2118-M-006-002-MY2 (Chen), the Mathematics Division of the National Center for Theoretical Sciences in Taiwan.

Bibliography

M. Barbieri and J. O. Berger. Optimal predictive model selection. *Annals of Statistics*, 32:870–897, 2004. [p124]

S. Brooks, A. Gelman, G. L. Jones, and X. Meng. *Handbook of Markov Chain Monte Carlo*. Chapman & Hall/CRC, Boca Raton, 2011. [p124]

S. Chatterjee, K. Steinhaeuser, A. Banerjee, S. Chatterjee, and A. Ganguly. Sparse group lasso: Consistency and climate applications. In *Proceedings of the 2012 SIAM International Conference on Data Mining*, chapter 4, pages 47–58. 2012. [p123]

R.-B. Chen, C.-H. Chu, T.-Y. Lai, and Y.-N. Wu. Stochastic matching pursuit for Bayesian variable selection. *Statistics and Computing*, 21:247–259, 2011. [p122, 123, 125]

R.-B. Chen, C.-H. Chu, S. Yuan, and Y. N. Wu. Bayesian sparse group selection. *Journal of Computational and Graphical Statistics*, in press. doi: 10.1080/10618600.2015.1041636. [p122, 123, 124]

H. Chipman, M. Hamada, and C. Wu. A Bayesian variable-selection approach for analyzing designed experiments with complex aliasing. *Technometrics*, 39:372–381, 1997. [p124]

J. M. Flegal, M. Haran, and G. L. Jones. Markov chain Monte Carlo: Can we trust the third significant figure? *Statistical Science*, 23:250–260, 2008. [p124]

E. I. George and R. E. McCulloch. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88:881–889, 1993. [p122, 124]

E. I. George and R. E. McCulloch. Approaches for Bayesian variable selection. *Statistica Sinica*, 7: 339–373, 1997. [p124, 125]

J. Geweke. Variable selection and model comparison in regression. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, editors, *Bayesian Statistics*, volume 5, pages 609–620. Oxford University Press, Oxford, 1996. [p123]

T.-K. Ho. Looking for a needle in a haystack: Revisiting the cross-country causes of the 2008–09 crisis by Bayesian model averaging. *Economica*, in press. [p131]

A. K. Rose and M. M. Spiegel. Cross-country causes and consequences of the 2008 crisis: International linkages and American exposure. *Pacific Economic Review*, 15:340–363, 2010. [p130]

A. K. Rose and M. M. Spiegel. Cross-country causes and consequences of the crisis: An update. *European Economic Review*, 55:309–324, 2011. [p130]

A. K. Rose and M. M. Spiegel. Cross-country causes and consequences of the 2008 crisis: Early warning. *Japan and the World Economy*, 24:1–16, 2012. [p130]

N. Simon, J. Friedman, T. Hastie, and R. Tibshirani. A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22:231–245, 2013. [p122]

Kuo-Jung Lee
Assistant Professor
Department of Statistics, National Cheng-Kung University
Tainan, Taiwan 70101
Taiwan
kuojunglee@mail.ncku.edu.tw

Ray-Bing Chen
Professor
Department of Statistics, National Cheng-Kung University
Tainan, Taiwan 70101
Taiwan
rbchen@mail.ncku.edu.tw