Bayesian solutions to high-dimensional data challenges using hybrid search

by

Shiqiang Jin

M.S., Lanzhou University, 2014

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2021

Abstract

In the era of Big Data, variable selection with high-dimensional data has drawn increasing attention. With a large number of predictors, there rises a big challenge for model fitting and prediction. In this dissertation, we propose three different yet interconnected methodologies, which include theory, computation, and real applications for various scenarios of regression analysis. The primary goal in this dissertation is to develop powerful Bayesian solutions to high-dimensional data challenges using a new variable selection strategy, called hybrid search. To effectively reduce computation costs in high-dimensional data analysis, we propose novel computational strategies that can quickly evaluate a large number of marginal likelihoods simultaneously within a single computation.

In Chapter 1, we discuss background and current challenges in high-dimensional variable selection. The motivation of our study is also justified. In Chapter 2, we introduce a new Bayesian method of best subset selection in the context of linear regression. The proposed method rapidly finds the best subset via a hybrid search algorithm that combines deterministic local search and stochastic global search. In Chapter 3, on the basis of the approach in Chapter 2, we extend it to a framework of multivariate linear regression model, which analyzes the relationship between multiple response variables and a common set of predictors. In Chapter 4, we propose a general Bayesian method to perform high-dimensional variable selection for various data types, such as binary, count, continuous and time-to-event (survival) data. Using Bayesian approximation techniques, we develop a general computing strategy that enables us to assess the marginal likelihoods of many candidate models within a single computation. In addition, to accelerate the convergence, we employ a hybrid search algorithm that can quickly explore the model spaces and accurately obtain the global maximum of marginal posterior probabilities.

Bayesian solutions to high-dimensional data challenges using hybrid search

by

Shiqiang Jin

M.S., Lanzhou University, 2014

A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2021

Approved by:

Major Professor
Dr. Gyuhyeong Goh

Copyright

Abstract

In the era of Big Data, variable selection with high-dimensional data has drawn increasing attention. With a large number of predictors, there rises a big challenge for model fitting and prediction. In this dissertation, we propose three different yet interconnected methodologies, which include theory, computation, and real applications for various scenarios of regression analysis. The primary goal in this dissertation is to develop powerful Bayesian solutions to high-dimensional data challenges using a new variable selection strategy, called hybrid search. To effectively reduce computation costs in high-dimensional data analysis, we propose novel computational strategies that can quickly evaluate a large number of marginal likelihoods simultaneously within a single computation.

In Chapter 1, we discuss background and current challenges in high-dimensional variable selection. The motivation of our study is also justified. In Chapter 2, we introduce a new Bayesian method of best subset selection in the context of linear regression. The proposed method rapidly finds the best subset via a hybrid search algorithm that combines deterministic local search and stochastic global search. In Chapter 3, on the basis of the approach in Chapter 2, we extend it to a framework of multivariate linear regression model, which analyzes the relationship between multiple response variables and a common set of predictors. In Chapter 4, we propose a general Bayesian method to perform high-dimensional variable selection for various data types, such as binary, count, continuous and time-to-event (survival) data. Using Bayesian approximation techniques, we develop a general computing strategy that enables us to assess the marginal likelihoods of many candidate models within a single computation. In addition, to accelerate the convergence, we employ a hybrid search algorithm that can quickly explore the model spaces and accurately obtain the global maximum of marginal posterior probabilities.

Table of Contents

List of Figures

List of Tables

Acknowledgments

In retrospect, the journey of my doctoral study is full of rough roads that I cannot walk forward alone. I am very grateful that many great people appeared in my journey and gave me a great deal of support and assistance. To them, I would like to express my sincere appreciation.

First of all, I would like to express my deepest appreciation to my major advisor, Dr. Gyuhyeong Goh, for his invaluable expertise, talent, and knowledge in statistics as well as his patience and encouragement, to help me in completing my research. His rigorous attitude towards academics makes me more in awe of science. His spiritual life and vision of his work exerted a great influence on my attitude towards work.

To committee members, Dr. Wei-Wen Hsu, Dr. Weixing Song, Dr. Jisang Yu, and the outside chairperson, Dr. Yoon-Jin Lee, for their time and effort to serve on my committee, and for their valuable comments when defending my dissertation. Thank Dr. Wei-Wen Hsu for correcting many typos and writing his questions for me on my dissertation.

To the Department of Statistics at K-State for supporting me the funding as a GTA from 2018 spring to 2020 fall.

To professors and staff in the Department of Statistics, both previous and present, including Dr. Abigail Jager, whom I worked with when I serve as a GTA, Department head Dr. Christopher Vahl, who is so nice to me, Bonnie Messmer, Jo Blackburn, and Alisha Dean. To professors whom I learned from in their statistical courses, including Dr. James Neill, Dr. Christopher Vahl, Dr. Juan Du, Dr. Haiyan Wang, Dr. Gyuhyeong Goh, Dr. Trevor Hefley, Dr. Weixing Song, and Dr. Suzanne Dubnicka.

To Dr. Pourab Roy whom I worked with at FDA as an ORISE fellow in the 2020 summer internship.

To my colleagues in Dr. Gyuhyeong Goh's group, Weikang Duan, Min Hua, Dunfung Yang, and Guotao Chu, for helping me with my research.

To my dear father Jianping Jin and mother Youhua Lin for raising me and their unconditional love, my parents-in-law for their love and support. Without them, I cannot go through my journey to the Ph.D. degree.

To my dear brothers and sisters in Manhattan Chinese Christian Fellowship (MCCF), whom we studied Bible and prayed with, especially Dr. Zhenzhen Shi, who introduced me to work as a GRA at ICCM in College of Veterinary from 2016 fall to 2017 fall, and Dr. Jianxiong Li, who gave me a ride to commute to school for a year when I had no car.

To all my classmates and friends at K-State, whom I had wonderful moments with, especially Dr. Jia Liang and Chenchuang Lu, who gave me a lot of support and suggestion when hunting for jobs, Caleb Pfeifer, Jim and Courtney Woods.

I would particularly like to single out my dear wife, Zijun Wu, whom I am united to as one flesh and whom I love most. I want to address my gratitude for her sacrifice, supports and unconditional love. She is the best gift and greatest blessing from God. As Proverb 31:29 says, "Many women do noble things, but you surpass them all". To my bunnies, who accompany me and give me lots of emotional supports and fun time.

Last but not the least, I would like to thank my God through Jesus Christ who guides me in my life, research, and job. "Trust in the Lord with all your heart and lean not on your own understanding; in all your ways submit to him, and he will make your paths straight"–Proverb 3:5-6.

Chapter 1

Introduction

Due to the rapid developments in data storage technologies and data processing software, it now becomes possible to store and accumulate data at an unprecedented speed and a cheap cost. These technology advancements have had a profound influence on data collection and data analysis in many fields of sciences, such as computational genomics, financial engineering, bioinformatics, and cosmology (Donoho, 2000). The difficulty that researchers and data analysts are facing is how to extract useful information from the massive amount of data. In the field of statistics, the advent of big data has led to many challenges in theories, methodologies, and applications. One of the most pressing challenges is the presence of a large number of predictor variables, called the high-dimensional data problem. A very large number of variables that contain many redundant features leads to poor prediction and computational complexity. Therefore, in high-dimensional data analysis, variable selection plays an essential role for building a useful statistical model (Fan and Li, 2006).

1.1 Challenges in high-dimensional variable selection

Recently, many efforts have been put into developing methodologies of variable selection and solving problems of intensive computation for high-dimensional data. To reduce redundant variables and have good model interpretability, finding a parsimonious model (or

sparse estimation) is necessary. Best subset selection is a popular way of selecting a small set of important predictors; for example, see Beale et al. (1967); Hocking and Leslie (1967); Bertsimas et al. (2016). The idea of best subset selection is simply to evaluate all possible combinations of predictors related to the response variable. However, it is well known that best subset selection involves non-convex optimization problems that are computationally intractable when the number of possible predictors is large (see Section 3.2 for more details). Although some penalized likelihood approaches such as Least Absolute Shrinkage and Selection Operator (LASSO), Elastic Net (ENET), and Minimax Concave Penalty (MCP) provide a convex surrogate for nonconvex optimization, their applicability to best subset selection is still limited (Bertsimas et al., 2016).

From a Bayesian perspective, many Markov chain Monte Carlo (MCMC) methods has been proposed an effective way to explore the non-convex model space using the notion of stochastic search (George and McCulloch, 1993, 1997; Brown et al., 1998; Madigan and York, 1995). A Bayesian approach to best subset selection, called Bayesian subset regression (BSR), has been developed by Liang et al. (2013). The key idea of BSR is to employ an adaptive MCMC algorithm, called the stochastic approximation Monte Carlo (Liang et al., 2007). However, the feasibility of MCMC methods deteriorates for the problem of high-dimensional variable selection since it raises computational challenges, including extremely heavy computation and slow convergence.

1.2 Challenges in parallel computing

To cope with the computational intensity, parallel computation techniques have been the focus of much attention. In the statistics literature, one of the representative works is shotgun stochastic search (SSS) (Hans et al., 2007). Although SSS is based on a MCMC approach, it is capable to quickly identify relevant models with the aid of parallel computing. In Zhao et al. (2013), a sequential forward selection approach was developed by using massively parallel processing mode. López et al. (2006) developed a way of paralleling the scatter search for solving a feature subset selection problem using two processors. There is little

doubt that variable selection approaches coupled with parallelism can significantly reduce the running time in the problem of high-dimensional data. However, without exception, parallel-computing algorithms require high-performance computing machines and programming protocols. For example, Hans et al. (2007) utilized 21 processors (1 head node and 20 computer nodes). Wang et al. (2017) implemented parallel sequential backward selection using the performance analysis tools for high-performance computing applications (PATHA), originally proposed by Yoo et al. (2015). In Zhou et al. (2014), 32 treads on a single 8-CPU machine in the national Open Science Grid were employed. However, a practical issue in the application of parallel computing arises from the fact that high-performance machines and programming protocols are not always available to individual users and researchers.

1.3 Challenges in multivariate regression

There are increasing interests in understanding how several dependent variables are regressed on a common set of independent variables. For instance, chemists may intent to predict the concentrations of several sugars (glucose, fructose, and sucrose) in a sample of aqueous solution using several wavelengths recorded by near-infrared analysis. The multivariate linear regression model (MVRM) provides a way of connecting multiple responses to a common set of predictors. It is well known that MVRM provides considerably better prediction performance than the use of multiple univariate regression models due to the additional information from the correlation between dependent variables. Due to this, MVRM has been becoming increasingly popular and widely used in many fields, including quality control (Noorossana et al., 2010a,b), chemistry (Akkartal. et al., 2009), climate science (Jeong. et al., 2012), animal science (Mehmet, 2011), and materials science (Gordon et al., 2018). In statistics and machine learning, many extensions of MVRM has been proposed, such as the Curds and Whey method (Breiman and Friedman, 1997), adaptive multivariate ridge regression (Brown and Zidek, 1980), and Bayesian MVRM (Brown et al., 1998, 2001).

Under the MVRM setting, high-dimensional variable selection is still an important research subject. While abundant studies on high-dimensional variable selection are available

in the literature (e.g., Bedrick, 1994; Fujikoshi and Satoh, 1997; Friedman et al., 2010), there is a lack of research done in a framework of Bayesian MVRM. A representative work has been done by Brown et al. (1998). The key contribution of this work is to extend the stochastic search variable selection (SSVS) of George and McCulloch (1993) to the framework of MVRM. However, Brown et al. (1998) still suffers from computational issues in the presence of high-dimensional data.

1.4 Motivation and dissertation overview

Our motivation comes from the need to address the aforementioned issues related to high-dimensional data analysis. In this dissertation, the main focus is to develop innovative Bayesian methods that can i) be quickly implemented in a single CPU core; ii) identify a best model via a fast global optimization; and iii) apply to various types of data (e.g., Gaussian, multivariate normal, binary, count, and survival time). The remaining parts of the dissertation can be summarized as follows:

- In Chapter 2, we introduce a hybrid best subset search algorithm that combines a deterministic local search and a stochastic global search in a framework of Bayesian linear regression modeling. The proposed hybrid search enables us to find the best subset via a fast global optimization procedure. An attractive feature of the proposed method is that evaluating all possible candidate models for the next update, which is the most expensive part of the computation, is simultaneously accomplished in a single computation. A simulation study and a real data application are discussed to demonstrate the performance of the proposed method.

- In Chapter 3, we develop an extension of the hybrid best subset search algorithm proposed in Chapter 2 to the problem of Bayesian MVRM. A simulation study is performed to demonstrate the superiority of the proposed method comparing to many existing methods including Chapter 2. The proposed method is also applied to real data analysis. To make our proposed method reproducible, we provide several R demo

codes in Appendix.

- In Chapter 4, we develop a general Bayesian variable selection procedure for handling a variety of data types. Using Bayesian approximation techniques, we develop a novel computing strategy that enables us to evaluate all the marginal likelihoods in the neighborhood model space simultaneously within a single computation. Moreover, to accelerate the convergence, we employ a hybrid search algorithm that can quickly explore the model spaces and accurately obtain the global maximum of marginal posterior probabilities. Extensive simulation studies and real data analysis are conducted to investigate the performance of the proposed method.

Chapter 2

Bayesian selection of best subsets via hybrid search

In this chapter, we develop a hybrid best subset search algorithm in a framework of Bayesian linear modeling. In Section 2.1, we summarize the Bayesian model setting and derives the posterior model probability, which serves as a model selection criterion. We present the hybrid best subset search algorithm with a fixed model size in Section 2.2 and with varying sizes in Section 2.3. In Section 2.4, we conduct a series of simulation studies for different scenarios, including models with independent, dependent, and heteroscedastic Gaussian errors. In Section 2.5, an example of real data analysis with gene expression data is provided. Some concluding remarks are given in Section 2.6.

2.1 Basic setup

Consider a multiple linear regression model,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{2.1}$$

where $\mathbf{y} = (y_1, \ldots, y_n)^\top$ is the n-dimensional response vector, $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$ is the $n \times p$ design matrix with $\mathbf{x}_j = (x_{1j}, \ldots, x_{nj})^\top$, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^\top$ is a p-dimensional coefficient

vector, and $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top \sim \text{Normal}(\mathbf{0}_n, \sigma^2\mathbf{I}_n)$. Throughout this chapter, we restrict our attention to high-dimensional regression problems and thus it is always assumed that $p > n$ and $\boldsymbol{\beta}$ contains many zero elements. We further assume that $\mathbf{y}$ and columns of $\mathbf{X}$ are standardized so that the intercept is excluded from our regression analysis. Our goal is to identify the k most important predictors in (2.1), where the best subset size, k, can be considered as being either fixed or varying. To formulate a Bayesian framework for best subset selection, let $\boldsymbol{\gamma} = \{j : \beta_j \neq 0\}$ be an index set of the active predictors. Given $\boldsymbol{\gamma}$, the full model (2.1) reduces to

$$\mathbf{y} = \mathbf{X}_{\boldsymbol{\gamma}}\boldsymbol{\beta}_{\boldsymbol{\gamma}} + \boldsymbol{\epsilon},$$

where $\mathbf{X}_{\boldsymbol{\gamma}}$ and $\boldsymbol{\beta}_{\boldsymbol{\gamma}}$ are a sub-matrix of $\mathbf{X}$ and a sub-vector of $\boldsymbol{\beta}$ that are determined by $\boldsymbol{\gamma}$, respectively. For algebraic and computational convenience, given $\boldsymbol{\gamma}$, we consider conjugate priors for $\boldsymbol{\beta}_{\boldsymbol{\gamma}}$ and σ^2 as follows:

$$\begin{aligned} \boldsymbol{\beta}_{\boldsymbol{\gamma}}|\sigma^2, \boldsymbol{\gamma} &\sim \text{Normal}(0, \tau\sigma^2\mathbf{I}_{|\boldsymbol{\gamma}|}), \\ \sigma^2 &\sim \text{Inverse-Gamma}(a_\sigma/2, b_\sigma/2), \end{aligned}$$

where τ, a_σ, and b_σ are hyperparameters and $|\boldsymbol{\gamma}|$ denotes the number of elements in the set $\boldsymbol{\gamma}$. To impose the constraint $|\boldsymbol{\gamma}| = k$, we define the prior distribution of $\boldsymbol{\gamma}$ by $\pi(\boldsymbol{\gamma}) \propto \mathbb{I}(|\boldsymbol{\gamma}| = k)$, where $\mathbb{I}(\cdot)$ is an indicator function. Let $m(\mathbf{y}|\boldsymbol{\gamma})$ be the marginal likelihood given $\boldsymbol{\gamma}$. Using the kernels of normal density and inverse gamma density, the marginal likelihood can be calculated as

$$\begin{aligned} m(\mathbf{y}|\boldsymbol{\gamma}) &= \int f(\mathbf{y}|\boldsymbol{\beta}_{\boldsymbol{\gamma}}, \sigma^2)\pi(\boldsymbol{\beta}_{\boldsymbol{\gamma}}|\sigma^2, \boldsymbol{\gamma})\pi(\sigma^2)d\boldsymbol{\beta}_{\boldsymbol{\gamma}}d\sigma^2 \\ &\propto \frac{(\tau^{-1})^{\frac{|\boldsymbol{\gamma}|}{2}}}{|\mathbf{X}_{\boldsymbol{\gamma}}^\top\mathbf{X}_{\boldsymbol{\gamma}} + \tau^{-1}\mathbf{I}_{|\boldsymbol{\gamma}|}|^{\frac{1}{2}} \left(\mathbf{y}^\top\mathbf{H}_{\boldsymbol{\gamma}}\mathbf{y} + b_\sigma\right)^{\frac{a_\sigma+n}{2}}}, \end{aligned} \tag{2.2}$$

where $f(\mathbf{y}|\boldsymbol{\beta}_{\boldsymbol{\gamma}}, \sigma^2)$ denotes the reduced model likelihood function given $\boldsymbol{\gamma}$ and $\mathbf{H}_{\boldsymbol{\gamma}} = \mathbf{I}_n - \mathbf{X}_{\boldsymbol{\gamma}}(\mathbf{X}_{\boldsymbol{\gamma}}^\top \mathbf{X}_{\boldsymbol{\gamma}} + \tau^{-1}\mathbf{I}_{|\boldsymbol{\gamma}|})^{-1}\mathbf{X}_{\boldsymbol{\gamma}}^\top$.

From Bayes' theorem, the posterior model probability of $\boldsymbol{\gamma}$ is proportional to $\pi(\boldsymbol{\gamma}|\mathbf{y}) \propto m(\mathbf{y}|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma}) \propto m(\mathbf{y}|\boldsymbol{\gamma})\mathbb{I}(|\boldsymbol{\gamma}| = k)$. Therefore, Bayesian best subset selection can be performed by maximizing $m(\mathbf{y}|\boldsymbol{\gamma})$ over $\boldsymbol{\gamma}$ subject to the constraint $|\boldsymbol{\gamma}| = k$. Keep in mind that high-dimensional and non-convex optimization problems arise in our framework.

Remark. *After selecting the best model $\boldsymbol{\gamma}$, Bayesian inferences about $\boldsymbol{\beta}_{\boldsymbol{\gamma}}$ and σ^2 can be easily made by drawing samples from $\pi(\boldsymbol{\beta}_{\boldsymbol{\gamma}}, \sigma^2|\mathbf{y}, \boldsymbol{\gamma})$. Due to the conjugacy of the priors for $\boldsymbol{\beta}_{\boldsymbol{\gamma}}$ and σ^2, the posterior sampling can be implemented via Gibbs sampling with the closed-form full-conditional distributions.*

2.2 Best subset selection with a fixed-size

In this section, we develop a new Bayesian approach to best subset selection with a fixed subset size k. Let $\hat{\boldsymbol{\gamma}}$ be the current estimate of the best subset of size k. Our strategy is to update $\hat{\boldsymbol{\gamma}}$ by searching over neighbor models iteratively. To this end, for an index set $\boldsymbol{\gamma}$, define $\mathcal{N}_+(\boldsymbol{\gamma}) = \{\boldsymbol{\gamma} \cup \{i\} : i \notin \boldsymbol{\gamma}\}$, which represents the set of larger neighbors of $\boldsymbol{\gamma}$ obtained by adding a new index to $\boldsymbol{\gamma}$. Similarly, define $\mathcal{N}_-(\boldsymbol{\gamma}) = \{\boldsymbol{\gamma} \setminus \{j\} : j \in \boldsymbol{\gamma}\}$ to be the set of smaller neighbors of $\boldsymbol{\gamma}$ obtained by deleting an index from $\boldsymbol{\gamma}$.

We introduce a deterministic search algorithm in Algorithm 1 that converges to a local maximum of $m(\mathbf{y}|\boldsymbol{\gamma})$ subject to $|\boldsymbol{\gamma}| = k$.

Algorithm 1 Deterministic best subset search with a fixed k

1. Define $\hat{\boldsymbol{\gamma}}$ to be an initial subset of size k; and set $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$.
2. **Repeat** for $t = 1, 2, \ldots,$
   a) Compute $\tilde{\boldsymbol{\gamma}}^{(t)} = \arg\max_{\boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})} m(\mathbf{y}|\boldsymbol{\gamma})$;
   `# i.e., select the locally best subset of size k+1`
   b) Compute $\hat{\boldsymbol{\gamma}}^{(t)} = \arg\max_{\boldsymbol{\gamma} \in \mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)})} m(\mathbf{y}|\boldsymbol{\gamma})$;
   `# i.e., select the locally best subset of size k`
   c) Update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t)}$;

   until $\hat{\boldsymbol{\gamma}}^{(t-1)} = \hat{\boldsymbol{\gamma}}^{(t)}$. `# i.e., terminate immediately if no update is made`
3. Return $\hat{\boldsymbol{\gamma}}$.

The following theorem proves the convergence of the proposed deterministic search algorithm.

Theorem 1. *The deterministic search in Algorithm 1 monotonically increases the objective function, $m(\mathbf{y}|\boldsymbol{\gamma})$, subject to $|\boldsymbol{\gamma}| = k$. In addition, the algorithm terminates in a finite number of iterations.*

Proof of Theorem 1. Let $\hat{\boldsymbol{\gamma}}^{(t-1)}$ be the best subset of size k updated by the $t-1$-th iteration. Then, $\hat{\boldsymbol{\gamma}}^{(t)} = \arg\max_{\boldsymbol{\gamma} \in \mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)})} m(\mathbf{y}|\boldsymbol{\gamma})$, where

$$\tilde{\boldsymbol{\gamma}}^{(t)} = \arg \max_{\boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})} m(\mathbf{y}|\boldsymbol{\gamma}).$$

Since $\hat{\boldsymbol{\gamma}}^{(t-1)}$ also belongs to $\mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)})$, we thus have $m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t-1)}) \leq m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t)})$, which proves our first statement. Since the number of all possible states of $\boldsymbol{\gamma}$ satisfying $|\boldsymbol{\gamma}| = k$ is finite, the algorithm terminates in a finite number of iterations. This completes our proof. □

Although the deterministic search algorithm converges rapidly, a possible drawback is that the algorithm can get trapped in a local optimum. As an alternative, we introduce a stochastic search algorithm in Algorithm 2 that converges to a global maximum of $m(\mathbf{y}|\boldsymbol{\gamma})$ with the constraint $|\boldsymbol{\gamma}| = k$. Note that the proposed stochastic search algorithm generates a

Algorithm 2 Stochastic best subset search with a fixed k

1. Define $\hat{\boldsymbol{\gamma}}$ to be an initial subset of size k; and set $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$.
2. **Repeat** for $t = 1, \ldots, T$:
   a) Update $\tilde{\boldsymbol{\gamma}}^{(t)}$ by selecting one from $\mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})$ with probability
   $$\frac{m(\mathbf{y}|\boldsymbol{\gamma})}{\sum_{\boldsymbol{\gamma}' \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})} m(\mathbf{y}|\boldsymbol{\gamma}')}, \ \boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)});$$
   b) Update $\hat{\boldsymbol{\gamma}}^{(t)}$ by selecting one from $\mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)})$ with probability
   $$\frac{m(\mathbf{y}|\boldsymbol{\gamma})}{\sum_{\boldsymbol{\gamma}' \in \mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)})} m(\mathbf{y}|\boldsymbol{\gamma}')}, \ \boldsymbol{\gamma} \in \mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)});$$
   c) **If** $m(\mathbf{y}|\hat{\boldsymbol{\gamma}}) < m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t)})$, **then** update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t)}$.
3. Return $\hat{\boldsymbol{\gamma}}$.

Markov chain, $\{\hat{\boldsymbol{\gamma}}^{(t)}, t = 1, \ldots, T\}$, with the state space $\{\boldsymbol{\gamma} : |\boldsymbol{\gamma}| = k\}$. Hence, if the current estimate $\hat{\boldsymbol{\gamma}}$ is not the global maximum, then we must observe that $m(\mathbf{y}|\hat{\boldsymbol{\gamma}}) < m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t)})$ with probability one as $T \to \infty$. Therefore, as we iterate the algorithm, $\hat{\boldsymbol{\gamma}}$ converges to the global maximum.

Although the stochastic search algorithm eventually reaches the global optimum, it requires the large number of iterations, which is computationally inefficient. To develop a computationally efficient global optimization algorithm, we propose a hybrid best subset selection algorithm that combines the stochastic global search algorithm and the deterministic local search algorithm into one. The proposed hybrid search algorithm is given in Algorithm 3. In the proposed hybrid search algorithm, the deterministic search is first used to find a local optimum in an efficient manner. Then, the stochastic search is employed to check whether or not the deterministic search algorithm has reached the global optimum. Note that if the stochastic search finds a better subset $\hat{\boldsymbol{\gamma}}^{(t)}$ than the current best subset $\hat{\boldsymbol{\gamma}}$ (i.e., when $m(\mathbf{y}|\hat{\boldsymbol{\gamma}}) < m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t)})$ occurs), then we immediately stop the stochastic search procedure and go back to the deterministic search step with the updated state of $\hat{\boldsymbol{\gamma}}$. To improve the performance of stochastic search, we introduce a tuning parameter $\alpha \in (0, 1]$ which acts as

Algorithm 3 Hybrid best subset search with a fixed k

1. Define an initial model of size k as $\hat{\boldsymbol{\gamma}}$.
2. Set $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$.
3. **Repeat** for $t = 1, 2, \ldots$: `# Deterministic Search Step`
   a) Compute $\tilde{\boldsymbol{\gamma}}^{(t)} = \arg\max_{\boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})} m(\mathbf{y}|\boldsymbol{\gamma})$;
   b) Compute $\hat{\boldsymbol{\gamma}}^{(t)} = \arg\max_{\boldsymbol{\gamma} \in \mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)})} m(\mathbf{y}|\boldsymbol{\gamma})$;
   c) Update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t)}$;

   until $\hat{\boldsymbol{\gamma}}^{(t-1)} = \hat{\boldsymbol{\gamma}}^{(t)}$.
4. Set $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$ and choose a tuning parameter $\alpha \in (0, 1]$.
5. **Repeat** for $t = 1, \ldots, T$: `# Stochastic Search Step`
   a) Update $\tilde{\boldsymbol{\gamma}}^{(t)}$ by drawing one from $\mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})$ with probability

$$\frac{m(\mathbf{y}|\boldsymbol{\gamma})^\alpha}{\sum_{\boldsymbol{\gamma}' \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})} m(\mathbf{y}|\boldsymbol{\gamma}')^\alpha}, \ \boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)}); \tag{2.3}$$

   b) Update $\hat{\boldsymbol{\gamma}}^{(t)}$ by drawing one from $\mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)})$ with probability

$$\frac{m(\mathbf{y}|\boldsymbol{\gamma})^\alpha}{\sum_{\boldsymbol{\gamma}' \in \mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)})} m(\mathbf{y}|\boldsymbol{\gamma}')^\alpha}, \ \boldsymbol{\gamma} \in \mathcal{N}_-(\tilde{\boldsymbol{\gamma}}^{(t)}); \tag{2.4}$$

   c) **If** $m(\mathbf{y}|\hat{\boldsymbol{\gamma}}) < m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t)})$, **then** update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t)}$ and immediately go to Step 2.
6. Return $\hat{\boldsymbol{\gamma}}$.

a precision parameter. For example, in Step 5a of Algorithm 3, we select one among $\{\boldsymbol{\gamma} : \boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})\}$ with probabilities $\{m(\mathbf{y}|\boldsymbol{\gamma})^\alpha / \sum_{\boldsymbol{\gamma}' \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})} m(\mathbf{y}|\boldsymbol{\gamma}')^\alpha : \boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})\}$, whereas, in Step 2a of Algorithm 2, we select one among $\{\boldsymbol{\gamma} : \boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})\}$ with probabilities $\{m(\mathbf{y}|\boldsymbol{\gamma}) / \sum_{\boldsymbol{\gamma}' \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})} m(\mathbf{y}|\boldsymbol{\gamma}') : \boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})\}$. As $\alpha \to 0$, the probability distributions in (2.3) and (2.4) will be more spread out so that the chance of getting stuck in the local maximum will be reduced in the stochastic search step. If we set $\alpha = 1$, the stochastic search step is analogous to Algorithm 2. In our simulation study and real data analysis, we set $\alpha = \min\{1, \log(2)/\log(m_{(1)}/m_{(2)})\}$, where $m_{(1)}$ and $m_{(2)}$ are, respectively, the first and second largest values of $\{m(\mathbf{y}|\boldsymbol{\gamma}) : \boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}})\}$.

Note that $m(\mathbf{y}|\hat{\boldsymbol{\gamma}}) < m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t)})$ if and only if $m(\mathbf{y}|\hat{\boldsymbol{\gamma}})^\alpha < m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t)})^\alpha$ for any $\alpha \in (0, 1]$. Hence, as in Algorithm 2, if the current estimate $\hat{\boldsymbol{\gamma}}$ is not the global maximum, then in Step 5 of Algorithm 3 we must observe that $m(\mathbf{y}|\hat{\boldsymbol{\gamma}}) < m(\mathbf{y}|\hat{\boldsymbol{\gamma}}^{(t)})$ with probability one as $T \to \infty$. Therefore, we can reach the global maximum by repeating Algorithm 3 with sufficiently large T.

In general, calculating marginal likelihoods for many candidate models, which is a necessary step in Bayesian model selection, is computationally expensive, even if an explicit form is available as in (2.2). The great merit of the proposed method is that evaluating the marginal likelihoods of all the candidates for the next update can be done simultaneously *within a single computation*. To this end, let $\hat{\boldsymbol{\gamma}}$ be an index set of a model of size k and $\tilde{\boldsymbol{\gamma}}$ be an index set of a model of size $k+1$. For any $i \notin \hat{\boldsymbol{\gamma}}$, it can be shown that

$$\begin{aligned} m(\mathbf{y}|\hat{\boldsymbol{\gamma}} \cup \{i\}) \quad \propto \quad & \left\{ \mathbf{y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{y} - \frac{(\mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{y})^2}{\tau^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i} + b_\sigma \right\}^{-\frac{a_\sigma + n}{2}} \\ & \times (\tau^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i)^{-1/2}, \end{aligned} \tag{2.5}$$

where $\mathbf{x}_i$ is the i-th column of $\mathbf{X}$. The proof of (2.5) is given in Appendix A.1.1. Note that $\mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i$ is the i-th diagonal element of $\mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{X}$ and that $\mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{y}$ is the i-th element of $\mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{y}$. This implies that Eq. (2.5) is the i-th element of the following p-dimensional vector:

$$\begin{aligned} \mathbf{m}_+(\hat{\boldsymbol{\gamma}}) \quad = \quad & \left\{ (\mathbf{y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{y} + b_\sigma) \mathbf{1}_p - \frac{(\mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{y})^2}{\frac{1}{\tau} \mathbf{1}_p + \mathrm{diag}(\mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{X})} \right\}^{-\frac{a_\sigma + n}{2}} \\ & \times \left\{ \frac{1}{\tau} \mathbf{1}_p + \mathrm{diag}(\mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{X}) \right\}^{-1/2}, \end{aligned} \tag{2.6}$$

where $\mathbf{a}^x$ and $\mathbf{a}/\mathbf{b}$ indicate entrywise operations for generic vectors $\mathbf{a}$ and $\mathbf{b}$, accordingly. For example, $\mathbf{a}^x = (a_1^x, \ldots, a_p^x)$ and $\mathbf{a}/\mathbf{b} = (a_1/b_1, \ldots, a_p/b_p)$. Note that $\mathcal{N}_+(\hat{\boldsymbol{\gamma}}) = \{\hat{\boldsymbol{\gamma}} \cup \{i\} : i \notin \hat{\boldsymbol{\gamma}}\}$. Therefore, evaluating $m(\mathbf{y}|\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \mathcal{N}_+(\hat{\boldsymbol{\gamma}})$ can be done simultaneously in a single computation by obtaining the sub-vector of $\mathbf{m}_+(\hat{\boldsymbol{\gamma}})$ for $\{i : i \notin \hat{\boldsymbol{\gamma}}\}$. Similarly, for any $j \in \tilde{\boldsymbol{\gamma}}$,

we can show that

$$m(\mathbf{y}|\tilde{\boldsymbol{\gamma}} \setminus \{j\}) \propto \left\{ \mathbf{y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{y} + \frac{(\mathbf{x}_j^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{y})^2}{\tau^{-1} - \mathbf{x}_j^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_j} + b_\sigma \right\}^{-\frac{a_\sigma + n}{2}} \times (\tau^{-1} - \mathbf{x}_j^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_j)^{-1/2}. \tag{2.7}$$

The proof of (2.7) is given in A.1.2. Define

$$\mathbf{m}_-(\tilde{\boldsymbol{\gamma}}) = \left\{ (\mathbf{y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{y} + b_\sigma) \mathbf{1}_p + \frac{(\mathbf{X}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{y})^2}{\frac{1}{\tau}\mathbf{1}_p - \mathrm{diag}(\mathbf{X}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X})} \right\}^{-\frac{a_\sigma + n}{2}} \times \left\{ \frac{1}{\tau}\mathbf{1}_p - \mathrm{diag}(\mathbf{X}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X}) \right\}^{-1/2}. \tag{2.8}$$

It is easy to check that (2.7) is the j-th element of $\mathbf{m}_-(\tilde{\boldsymbol{\gamma}})$. Since $\mathcal{N}_-(\tilde{\boldsymbol{\gamma}}) = \{\tilde{\boldsymbol{\gamma}} \setminus \{j\} : j \in \tilde{\boldsymbol{\gamma}}\}$, evaluating $m(\mathbf{y}|\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \mathcal{N}_-(\tilde{\boldsymbol{\gamma}})$ can be done simultaneously in a single computation by obtaining the sub-vector of $\mathbf{m}_-(\tilde{\boldsymbol{\gamma}})$ for $\{j : j \in \tilde{\boldsymbol{\gamma}}\}$. Using the aforementioned calculation strategy, we can easily and quickly implement steps 3a), 3b), 5a), and 5b) of Algorithm 3. It is also worth noting that the proposed calculation method enables us to avoid multiple computations of inverse and determinant of matrices in (2.2).

2.3 Best subset selection with varying sizes

In this section, we extend the proposed method to handle the case of varying $k \leq K$, where K is a prespecified upper bound. This is a common setting for high-dimensional best subset selection problems (e.g., Bertsimas et al. 2016; Liang et al. 2013). In a Bayesian framework, this extension can be easily done by assigning an appropriate prior for unknown k. As a non-informative prior, one may consider a discrete uniform prior for k, that is, $k \sim \mathrm{Uniform}\{1, \ldots, K\}$. However, the uniform prior tends to assign larger probability to a larger subset due to the fact that the total number of subsets of size k is $\binom{p}{k}$, which tends to increase exponentially as k increases. To resolve this issue, using a similar idea of Chen

and Chen (2008), we consider

$$\pi(k) \propto \mathbb{I}(k \leq K) / \binom{p}{k}.$$

From Bayes' theorem, Bayesian best subset selection is then performed by maximizing

$$\pi(\boldsymbol{\gamma}|\mathbf{y}) \propto m(\mathbf{y}|\boldsymbol{\gamma}) \mathbb{I}(|\boldsymbol{\gamma}| \leq K) / \binom{p}{|\boldsymbol{\gamma}|}. \tag{2.9}$$

Note that (2.9) is equivalent to the following optimization problem:

$$\max_{\boldsymbol{\gamma}} \left\{ m(\mathbf{y}|\boldsymbol{\gamma}) \mathbb{I}(|\boldsymbol{\gamma}| = k) / \binom{p}{k} \right\} \quad \text{subject to } k \leq K.$$

Therefore, the optimization problem in (2.9) can be solved by proceeding the following steps:

1. For $k = 1, \ldots, K$, compute

$$\hat{\boldsymbol{\gamma}}_k = \arg \max_{\boldsymbol{\gamma}:|\boldsymbol{\gamma}|=k} m(\mathbf{y}|\boldsymbol{\gamma})$$

   using Algorithm 3.

2. Compute

$$\hat{k} = \arg \max_{1 \leq k \leq K} \left\{ \log m(\mathbf{y}|\hat{\boldsymbol{\gamma}}_k) - \log \binom{p}{k} \right\}.$$

3. Return $\hat{\boldsymbol{\gamma}} = \hat{\boldsymbol{\gamma}}_{\hat{k}}$.

Note that if a parallel or cluster computing environment is available, then it can be applied to the first step of the above procedure to run the for-loop over k in parallel. The following theorem shows that the proposed Bayesian approach achieves the model selection consistency in the high-dimensional setting that $p > n$.

Theorem 2. *Define* $\Gamma = \{\boldsymbol{\gamma} : |\boldsymbol{\gamma}| \leq K, \boldsymbol{\gamma} \neq \boldsymbol{\gamma}_*\}$, *where* $\boldsymbol{\gamma}_*$ *is the true model. Assume that* $p = O(n^\xi)$ *for* $\xi \geq 1$. *Under the asymptotic identifiability condition of Chen and Chen (2008), if* $\tau \to \infty$ *as* $n \to \infty$ *but* $\tau = o(n)$, *then the proposed Bayesian subset selection*

possesses the Bayesian model selection consistency, that is,

$$\pi(\boldsymbol{\gamma}_*|\mathbf{y}) > \max_{\boldsymbol{\gamma}\in\Gamma} \pi(\boldsymbol{\gamma}|\mathbf{y}) \tag{2.10}$$

in probability as $n \to \infty$.

Proof of Theorem 2. From the Laplace approximation of Kass and Raftery (1995), we have

$$\log m(\mathbf{y}|\boldsymbol{\gamma}) = \log f(\mathbf{y}|\hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}}, \hat{\sigma}^2) - \frac{|\boldsymbol{\gamma}|}{2}\log n + o_p(\log n),$$

where $\hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}} = (\mathbf{X}_{\boldsymbol{\gamma}}^\top \mathbf{X}_{\boldsymbol{\gamma}})^{-1}\mathbf{X}_{\boldsymbol{\gamma}}^\top \mathbf{y}$ and $\hat{\sigma}^2 = \|\mathbf{H}_{\boldsymbol{\gamma}}\mathbf{y}\|^2/n$. Ignoring a smaller order term than $\log n$, our posterior criterion (2.9) can be approximated as

$$-2\log \pi(\boldsymbol{\gamma}|\mathbf{y}) \;\approx\; -2\log f(\mathbf{y}|\hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}}, \hat{\sigma}^2) + |\boldsymbol{\gamma}|\log n + 2\log\binom{p}{k}, \tag{2.11}$$

which is equivalent to the extended Bayesian information criterion (Chen and Chen, 2008). Therefore, it follows from Theorem 1 of Chen and Chen (2008) that (2.10) holds in probability. □

2.4 Simulation study

2.4.1 Model selection performance comparison

In this section, we investigate the variable selection performance of our best subset selection algorithm on simulated high-dimensional data. We consider three scenarios of data generating processes.

In Scenario I, we generate the data $\{(y_i, x_{i1}, \ldots, x_{ip}) : i = 1, \ldots, n\}$ from the following linear regression model:

$$y_i = \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i, \tag{2.12}$$

where $n = 100$, $(x_{i1}, \ldots, x_{ip})^\top \overset{iid}{\sim} \text{Normal}(\mathbf{0}_p, \mathbf{\Sigma})$ with $\mathbf{\Sigma} = (\Sigma_{ij})_{p\times p}$ and $\Sigma_{ij} = \rho^{|i-j|}$, $(\epsilon_1, \ldots, \epsilon_n) \sim \text{Normal}(0, \mathbf{I}_n)$, four β_j's are randomly selected and then generated from Uniform $\{-1, -2, 1, 2\}$ independently, and the remaining β-coefficients are set equal to 0.

In Scenario II, we consider the same setting as in Scenario I, but we use the following correlated error structure instead of the independent errors: $(\epsilon_1, \ldots, \epsilon_n)^\top \sim \text{Normal}(0, \mathbf{V})$ with $\mathbf{V} = (v_{ij})_{n\times n}$ and $v_{ij} = 0.5^{|i-j|}$.

In Scenario III, we consider the same setting as in Scenario I, but the heteroscedastic error structure and the inverse-wishart random variance for predictors are used as follows: $\epsilon_i \overset{ind}{\sim} \text{Normal}(0, \sigma_i^2)$ with $\sigma_i^2 = 1 + 0.5\sqrt{x_{i1}}$ and $\mathbf{\Sigma} \sim \text{Inverse-Wishart}((1000-1)\mathbf{\Psi}, p+1000)$ with $\mathbf{\Psi} = (\Psi_{ij})_{p\times p}$ and $\Psi_{ij} = \rho^{|i-j|}$.

For each scenario, we consider four different cases: (i) $p = 200$, $\rho = 0.1$, (ii) $p = 200$, $\rho = 0.9$, (iii) $p = 1000$, $\rho = 0.1$, and (iv) $p = 1000$, $\rho = 0.9$. We assume that there is no prior information. Hence, to make our prior distributions non-informative, we set $a_\sigma = b_\sigma = 1$ and $\tau = (\log p)^2$, which satisfies the sufficient condition for model selection consistency discussed in Theorem 2. We further assume that the true number of active predictors is unknown. Hence, we employ the proposed method in Section 2.3 for varying k with $K = \lceil n^{2/3} \rceil = 22$. For the hybrid search algorithm given in Algorithm 3, we set $T = 100$, which is a remarkably small number of iterations compared to existing stochastic search algorithms (e.g., Kirkpatrick et al. 1983; George and McCulloch 1993; Hans et al. 2007), and marginal correlations between the response and each predictor are used to define the initial value of $\hat{\boldsymbol{\gamma}}$. For comparison purposes, we also employ four popular penalized likelihood methods: LASSO (Tibshirani, 1996), ENET (Zou and Hastie, 2005), Smoothly Clipped Absolute Deviation (SCAD) (Fan and Li, 2001), and MCP (Zhang, 2010), where the regularization parameters or tuning parameters are determined by the extended BIC in (2.11) to achieve a fair comparison with the proposed method. In this simulation study, LASSO and ENET are implemented by R package `glmnet` and MCP and SCAD are performed by R package `ncvreg`. To evaluate the variable selection performance, we calculate the false discovery rate, the percentage of selecting the exact true model, the average size of the selected model, and the mean of

Table 2.1: Simulation study results based on 2,000 Monte Carlo replications for Scenario I with Cases i–iv. Notation: FDR—false discovery rate; TRUE—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Case	Method	FDR (s.e.)	TRUE (s.e.)	SIZE (s.e.)	HAM (s.e.)
i	Proposed	0.006 (0.001)	96.900 (0.388)	4.032 (0.004)	0.032 (0.004)
	SCAD	0.034 (0.002)	85.200 (0.794)	4.188 (0.011)	0.188 (0.011)
	MCP	0.035 (0.002)	84.750 (0.804)	4.191 (0.011)	0.191 (0.011)
	ENET	0.016 (0.001)	92.700 (0.582)	4.087 (0.007)	0.087 (0.007)
	LASSO	0.020 (0.002)	91.350 (0.629)	4.109 (0.009)	0.109 (0.009)
ii	Proposed	0.023 (0.002)	88.750 (0.707)	3.985 (0.006)	0.203 (0.014)
	SCAD	0.059 (0.003)	74.150 (0.979)	4.107 (0.015)	0.480 (0.022)
	MCP	0.137 (0.004)	55.400 (1.112)	4.264 (0.020)	1.098 (0.034)
	ENET	0.501 (0.004)	0.300 (0.122)	7.716 (0.072)	5.018 (0.052)
	LASSO	0.276 (0.004)	15.550 (0.811)	5.308 (0.033)	2.038 (0.034)
iii	Proposed	0.004 (0.001)	98.100 (0.305)	4.020 (0.003)	0.020 (0.003)
	SCAD	0.027 (0.002)	87.900 (0.729)	4.145 (0.010)	0.145 (0.010)
	MCP	0.031 (0.002)	86.550 (0.763)	4.172 (0.013)	0.172 (0.013)
	ENET	0.035 (0.002)	84.850 (0.802)	4.181 (0.013)	0.206 (0.012)
	LASSO	0.014 (0.001)	93.850 (0.537)	4.073 (0.007)	0.073 (0.007)
iv	Proposed	0.023 (0.002)	89.850 (0.675)	4.005 (0.005)	0.190 (0.013)
	SCAD	0.068 (0.003)	74.250 (0.978)	4.196 (0.014)	0.493 (0.023)
	MCP	0.152 (0.004)	53.750 (1.115)	4.226 (0.017)	1.202 (0.035)
	ENET	0.417 (0.005)	0.150 (0.087)	6.228 (0.068)	4.089 (0.043)
	LASSO	0.265 (0.004)	19.500 (0.886)	5.139 (0.029)	1.909 (0.035)

Hamming distance based on 2,000 Monte Carlo replications. False discovery rate is defined as $|\hat{\gamma} \setminus \gamma_*|/|\hat{\gamma}|$, and Hamming distance is defined as $|\hat{\gamma} \setminus \gamma_*| + |\gamma_* \setminus \hat{\gamma}|$, where γ_* denotes the index set of the true model.

The results are shown in Tables 2.1, 2.2, and 2.3. In each scenario, the proposed method outperforms all the penalized likelihood methods for all three cases. The proposed method always achieves the smallest false discovery rate and this implies that the proposed method provides the minimum proportion of incorrectly identifying the true active predictors as inactive. According to the selected model size and the Hamming distance, we argue that the proposed method tends to select the closest model to the true model. In addition, the proposed method selects the exact true model with high probability. Hence, this finite-sample performance is supported by our theoretical result in Theorem 2.

Table 2.2: Simulation study results based on 2,000 Monte Carlo replications for Scenario II with Cases i–iv. Notation: FDR—false discovery rate; TRUE—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Case	Method	FDR (s.e.)	TRUE (s.e.)	SIZE (s.e.)	HAM (s.e.)
i	Proposed	0.017 (0.001)	92.150 (0.602)	4.091 (0.008)	0.092 (0.008)
	SCAD	0.032 (0.002)	85.850 (0.780)	4.176 (0.011)	0.175 (0.011)
	MCP	0.033 (0.002)	85.500 (0.780)	4.178 (0.011)	0.177 (0.011)
	ENET	0.020 (0.001)	91.250 (0.632)	4.104 (0.008)	0.104 (0.008)
	LASSO	0.019 (0.001)	91.450 (0.625)	4.101 (0.008)	0.101 (0.008)
ii	Proposed	0.030 (0.002)	85.950 (0.777)	4.030 (0.007)	0.234 (0.014)
	SCAD	0.058 (0.003)	75.100 (0.967)	4.088 (0.013)	0.471 (0.022)
	MCP	0.123 (0.004)	58.800 (1.101)	4.218 (0.019)	0.983 (0.032)
	ENET	0.496 (0.004)	0.300 (0.000)	7.636 (0.072)	4.965 (0.051)
	LASSO	0.269 (0.004)	17.600 (0.852)	5.235 (0.033)	1.987 (0.035)
iii	Proposed	0.016 (0.001)	92.550 (0.587)	4.083 (0.007)	0.084 (0.007)
	SCAD	0.028 (0.002)	87.800 (0.732)	4.152 (0.010)	0.152 (0.010)
	MCP	0.032 (0.002)	86.200 (0.771)	4.191 (0.016)	0.191 (0.016)
	ENET	0.030 (0.002)	85.850 (0.780)	4.149 (0.012)	0.183 (0.011)
	LASSO	0.017 (0.001)	92.450 (0.591)	4.088 (0.008)	0.090 (0.008)
iv	Proposed	0.029 (0.002)	88.100 (0.724)	4.043 (0.006)	0.219 (0.015)
	SCAD	0.058 (0.003)	77.650 (0.932)	4.152 (0.012)	0.419 (0.021)
	MCP	0.148 (0.004)	54.700 (1.113)	4.245 (0.019)	1.169 (0.036)
	ENET	0.427 (0.005)	0.200 (0.100)	6.379 (0.069)	4.204 (0.044)
	LASSO	0.263 (0.004)	19.050 (0.878)	5.157 (0.028)	1.868 (0.033)

2.4.2 Computational speed comparison

In this section, we investigate the computational speed improvement of the proposed method. We are specifically interested in assessing the computational efficiency of the simultaneous marginal likelihood evaluation using the proposed formulas of (2.6) and (2.8). To this end, we generate the data $\{(y_i, x_{i1}, \ldots, x_{ip}) : i = 1, \ldots, n\}$ using the data generating process of Scenario I in Section 2.4.1 for $n = 100$, $\rho = 0.5$, and varying p.

The execution time is measured by implementing Algorithm 3 with $k = 4$ which is the number of the true active predictors. For the purpose of comparison, we also measure the execution time of Algorithm 3 after replacing the proposed simultaneous marginal likelihood evaluation with the for-loop, which is commonly used in Bayesian computation, for evaluating the marginal likelihoods of candidate models for the next move in Steps 3a, 3b, 5a, and 5b.

Table 2.3: Simulation study results based on 2,000 Monte Carlo replications for Scenario III with Cases i–iv. Notation: FDR—false discovery rate; TRUE—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Case	Method	FDR (s.e.)	TRUE (s.e.)	SIZE (s.e.)	HAM (s.e.)
i	Proposed	0.006 (0.001)	96.850 (0.391)	4.031 (0.004)	0.032 (0.004)
	SCAD	0.033 (0.002)	85.450 (0.789)	4.176 (0.011)	0.175 (0.011)
	MCP	0.034 (0.002)	85.150 (0.795)	4.184 (0.011)	0.184 (0.011)
	ENET	0.028 (0.002)	87.250 (0.746)	4.148 (0.010)	0.156 (0.010)
	LASSO	0.020 (0.002)	90.600 (0.653)	4.102 (0.008)	0.104 (0.008)
ii	Proposed	0.042 (0.002)	80.150 (0.892)	3.952 (0.008)	0.385 (0.019)
	SCAD	0.098 (0.004)	62.300 (1.084)	4.132 (0.017)	0.803 (0.029)
	MCP	0.180 (0.005)	44.050 (1.110)	4.282 (0.023)	1.470 (0.038)
	ENET	0.451 (0.005)	0.350 (0.132)	6.287 (0.071)	4.429 (0.046)
	LASSO	0.312 (0.004)	10.800 (0.694)	5.206 (0.036)	2.416 (0.036)
iii	Proposed	0.004 (0.001)	97.650 (0.339)	4.021 (0.003)	0.024 (0.003)
	SCAD	0.029 (0.002)	86.950 (0.753)	4.150 (0.011)	0.164 (0.011)
	MCP	0.032 (0.002)	86.850 (0.780)	4.176 (0.012)	0.177 (0.012)
	ENET	0.062 (0.003)	69.400 (1.031)	4.266 (0.018)	0.428 (0.017)
	LASSO	0.026 (0.002)	86.050 (0.775)	4.114 (0.010)	0.165 (0.010)
iv	Proposed	0.051 (0.003)	80.350 (0.889)	3.995 (0.005)	0.423 (0.021)
	SCAD	0.130 (0.004)	56.550 (1.109)	4.295 (0.018)	1.550 (0.039)
	MCP	0.195 (0.005)	43.500 (1.109)	4.191 (0.018)	1.550 (0.039)
	ENET	0.369 (0.005)	0.400 (0.141)	4.918 (0.062)	3.682 (0.037)
	LASSO	0.290 (0.004)	13.500 (0.7646)	4.957 (0.031)	2.200 (0.034)

The simulation is implemented by R using a Linux-based desktop computer with an Intel Core i5-2400 processor and 16 GB of memory.

Fig. 2.1 displays the trend of the average execution time as p increases. The proposed method has a constant computational speed for increasing values of p, while the average execution time of the for-loop method steadily increases as the number of candidate predictors grows. For example, when $p = 100$, the proposed method takes only 0.122 seconds until convergence occurs, whereas the for-loop requires 4.020 seconds to reach the convergence. When $p = 1,000$, the average execution time of the proposed method is 0.208 seconds while the for-loop requires 20.289 seconds on average to complete the task. The simulation result clearly shows that the proposed method is computationally much faster and more efficient than the use of the for-loop.

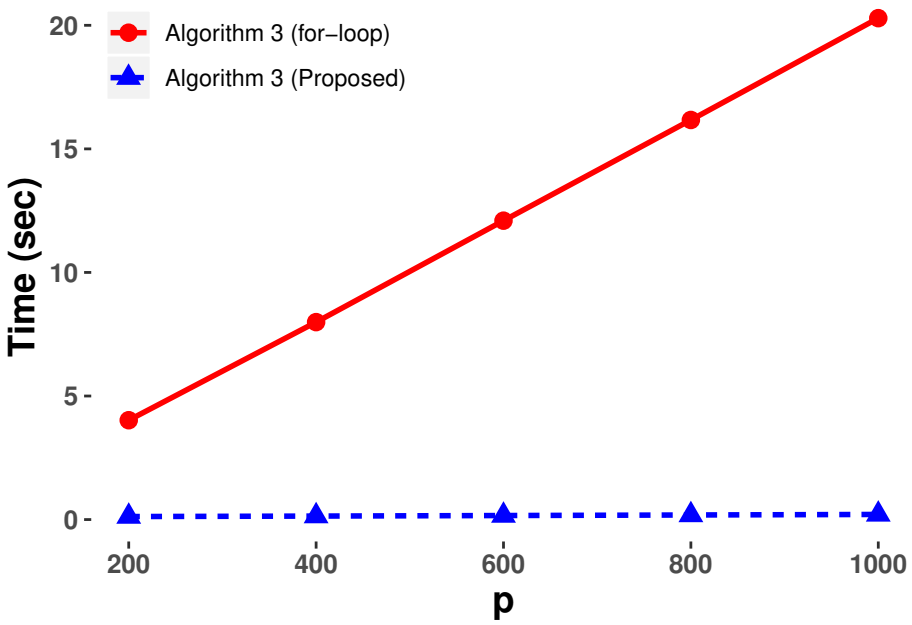

Figure 2.1: Result of computational speed comparison: the proposed simultaneous marginal likelihood evaluation versus the for-loop.

2.5 Real data application

In this section, we apply the proposed method to Breast invasive carcinoma (BRCA) data, which are generated by The Cancer Genome Atlas (TCGA) Research Network: http://cancergenome.nih.gov. We download BRCA data using `R` package `curatedTCGAData`. The data set contains $17,814$ gene expression measurements (recorded on the log scale) of 526 patients with primary solid tumor in TCGA project. BRCA1 is a well-known tumor suppressor gene and its mutations predispose women to breast cancer (Findlay et al., 2018). Our goal here is to identify the best fitting model for estimating an association between BRCA1 (response variable) and the other genes (independent variables). After removing missing values, the data set reduces to $n = 526$ samples with $p = 17,326$ genes.

The proposed method and the penalized likelihood methods as in Section 2.4.1 are applied to the entire data of $n = 526$ and $p = 17,326$. To assess model fitting, for each method, we compute BIC (Schwarz, 1978), extended BIC, and mean squared prediction error (MSPE) for the ordinary least squares (OLS) estimate that is obtained by using the selected predictors, where MSPE is estimated by Monte Carlo cross-validation over 500 replications based on

Table 2.4: Model comparison results for BRCA data

	BIC	extended BIC	MSPE
Proposed	985.93	1120.87	0.60
SCAD	1104.69	1176.42	0.68
MCP	1104.69	1176.42	0.68
ENET	1110.65	1198.68	0.68
LASSO	1104.69	1176.42	0.68

70% of training set and 30% of testing set.

Table 2.4 summarizes model comparison results. As both BIC and extended BIC are minimized at the resulting model from the proposed method, this implies that our Bayesian method is most strongly supported by the data. In addition, the proposed method has the smallest MSPE. Hence, the results demonstrate that the proposed method provides the best fit to the data. The heatmap in Fig. 2.2 displays the OLS coefficient estimates and the p-values of the selected predictors for each method. As a result, the proposed method identifies 8 genes (GINS1, GNL1, VPS25, ARHGAP19, CRBN, NBR2, DTL, and KLHL13) that are statistically related to the human tumor suppressor gene, BRCA1.

According to the human gene database, called GeneCards (`https://www.genecards.org`), GINS1, CRBN, NBR2, DTL, and KLHL13 are known to be associated with diseases including neutropenia, cognitive disability, breast-ovarian cancer, myasthenic syndrome, and Zellweger syndrome. Krishnan et al. (2018) report that GNL1 is related to colorectal and gastric cancer. The study of Herz et al. (2006) shows that VPS25 can serve as a model for human cancer. In Marceaux et al. (2018), it is shown that the premature localization of ARHGAP19 at the cell membrane leads to cytokinesis failure and cell multinucleation.

2.6 Discussion

The proposed hybrid search approach is computationally beneficial and practically effective compared with traditional stochastic search algorithms that are commonly used in Bayesian variable selection. Although the idea of searching over the neighborhood is somewhat similar to that of the MCMC model composition (MC^3) algorithm (Madigan and York, 1995) and

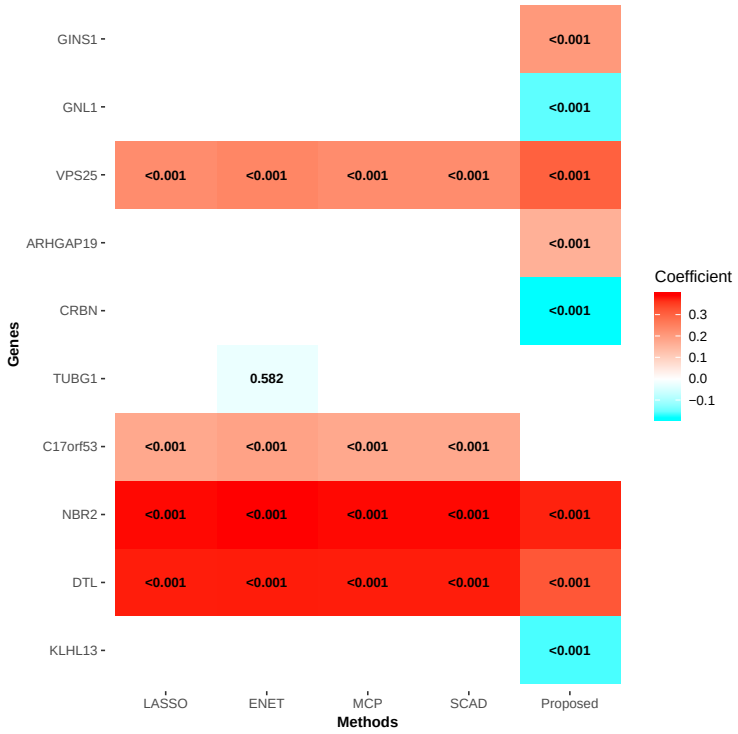

Figure 2.2: A heatmap of OLS coefficient estimates for the selected predictors with p-values without screening.

the shotgun stochastic search (SSS) algorithm (Hans et al., 2007), our search strategy is completely different from the existing methods. While MC^3 and SSS perform a stochastic search over the model space of all possible candidates, we implement a hybrid search over the sub-model space of size k. In addition, evaluating multiple candidates for the next move can be done simultaneously in a single computation in the proposed method.

In addition, as mentioned in Section 2.3, parallel computing, which executes many calculations simultaneously using multiple nodes or multiple processors, can be employed in the proposed framework. The proposed hybrid search algorithm can be applied to frequentist model selection approaches such as Akaike information criterion (AIC), BIC, and Mallows' C_p. Our hybrid search algorithm can be also extended to multivariate regression and generalized linear regression that are topics in Chapters 3 and 4.

Chapter 3

Fast Bayesian best subset selection for high-dimensional multivariate regression

In this chapter, we develop a hybrid search of best subset selection for Bayesian multivariate linear regression models (MVRM). We are interested in (i) choosing k most important predictors among $\binom{p}{k}$ candidate models and (ii) selecting a single best model from among all the 2^p candidate models. The proposed method in this chapter can be considered as an extension of Chapter 2 to a multivariate data case since the classic linear regression model can be viewed as a special case of MVRM with a single response.

The structure of this chapter is as follows. In Section 3.1, we provide a motivating example to show limitations of a representative work of Bayesian high-dimensional variable selection in MVRM. In Section 3.2, we describe a model set-up for Bayesian best subset selection in MVRM. In Section 3.3, we develop a fast best subset selection algorithm in MVRM using the idea of hybrid search. Section 3.4 conducts a simulation study to evaluate the performance of the proposed approach. In Section 3.5, we apply the proposed method to mRNA gene expression data to demonstrate the advantages of our proposed method. Section 3.6 gives our conclusion and some remarks.

3.1 Limitations of existing method

Brown et al. (1998) is a representative work of Bayesian variable selection in MVRM. To identify the variables selected among p regressors, Brown et al. (1998) introduced a latent binary vector $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_p)^\top$ such that $\xi_j = 0$ if the j-th row of the regression coefficient is close to 0 and $\xi_j = 1$ otherwise. With natural conjugate prior distributions, the marginal posterior distribution of $\boldsymbol{\xi}$ can be derived as $\pi(\boldsymbol{\xi}|\mathbf{Y}) = \frac{g(\boldsymbol{\xi}|\mathbf{Y})}{\Sigma_{\boldsymbol{\xi}\in\{0,1\}} g(\boldsymbol{\xi}|\mathbf{Y})}$, where $g(\boldsymbol{\xi}|\mathbf{Y})$ is a proportional form of $\pi(\boldsymbol{\xi}|\mathbf{Y})$. As the exhaustive computation of $\pi(\boldsymbol{\xi}|\mathbf{Y})$ in the model space $\{0,1\}^p$ is not feasible, Brown et al. (1998) employed Gibbs samplers from the full conditional distributions of $\pi(\xi_j|\boldsymbol{\xi}_{-j}, \mathbf{Y}), j = 1, 2, \ldots, p$.

However, Brown et al. (1998)'s method has two limitations when it is applied to the problem of high-dimensional best subset selection. First, it involves a heavy computation issue when p is large. The computational challenge arises from the fact that generating a sample from $\pi(\boldsymbol{\xi}|\mathbf{Y})$ is converted into generating p samples sequentially from $\pi(\xi_j|\boldsymbol{\xi}_{-j}, \mathbf{Y})$ for $j = 1, \ldots, p$. To demonstrate this, we conduct a simulation study using a mediocre desktop with Intel® Core™ i5-2400 CPU @ 3.10GHz × 4 and 16 GB memory. Given $n = 100$, we generate the data from the data generating process in Section 3.4 with $\boldsymbol{\Omega} = (0.4^{|i-j|})_{m\times m}$ and $\boldsymbol{\Sigma}_x = (0.8^{|i-j|})_{p\times p}$ for $p = 200$, 400, 600, 800 and 1000. For each simulated data set, we apply Brown et al. (1998)'s method with MCMC size = 1000, 2000, and 5000. Fig. 3.1 summarizes the simulation result based on 50 Monte Carlo replications. The result clearly shows that the average execution time increases as either p increases or the number of MCMC iterations increases. The second limitation is that the method is not applicable to selecting a best subset of size k, where k is a pre-specified value. These limitations motivate us to develop a new Bayesian best subset selection procedure for MVRM.

3.2 Bayesian best subset selection in MVRM

3.2.1 Multivariate regression model

Consider fitting the following MVRM:

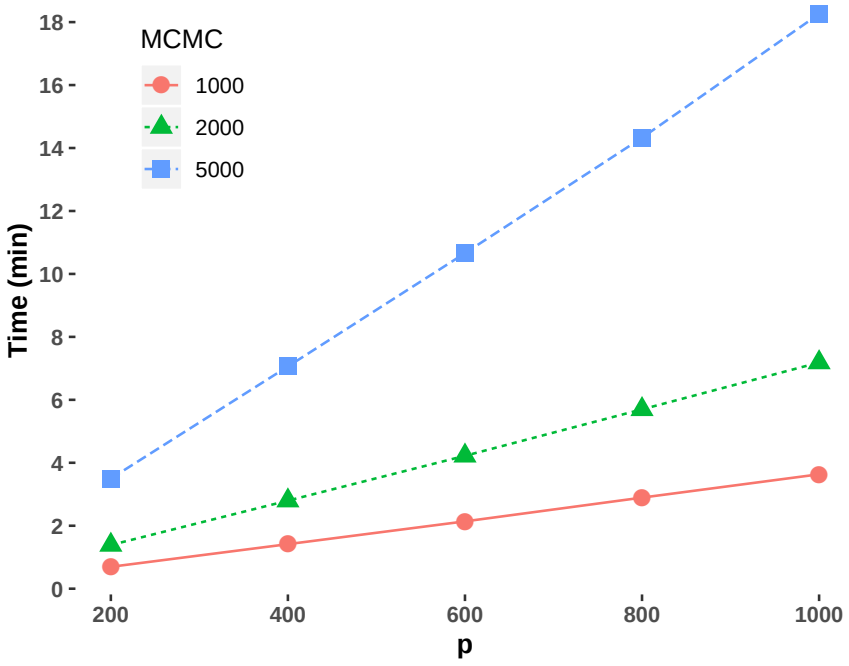

Figure 3.1: Average execution time of Brown et al. (1998) over 50 Monte Carlo replications for different p and MCMC sizes.

$$\mathbf{Y} = \mathbf{X}\mathbf{C} + \mathbf{E}, \tag{3.1}$$

where $\mathbf{Y} \in \mathbb{R}^{n\times m}$ is a response matrix, $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top \in \mathbb{R}^{n\times p}$ is a design matrix of p predictors, and $\mathbf{C} = (\mathbf{c}_1, \ldots, \mathbf{c}_p)^\top \in \mathbb{R}^{p\times m}$ is an unknown coefficient matrix. The rows of error matrix, $\mathbf{E} \in \mathbb{R}^{n\times m}$, are assumed to be i.i.d m-dimensional vectors following multivariate normal distribution $\mathcal{N}_m(\mathbf{0}, \mathbf{\Omega})$, where $\mathbf{\Omega} \in \mathbb{R}^{m\times m}$ is an unknown positive definite covariance matrix. We assume that number of predictors is larger than the sample size, i.e., $p > n$. Note that there are $mp + \binom{m}{2}$ unknown parameters in $\mathbf{C}$ and $\mathbf{\Omega}$, which is much larger than the sample size, imposing a big challenge on model selection. Consequently, finding a parsimonious model is desirable to conduct meaningful inference. Note that in MVRM variable selection is equivalent to imposing row-sparsity on $\mathbf{C}$. To give more details, we define an index set of the active predictors as $\boldsymbol{\gamma} = \{j : \mathbf{c}_j \neq \mathbf{0}\}$. Given $\boldsymbol{\gamma}$, the likelihood

function can be represented as

$$f(\mathbf{Y}|\mathbf{C}_\gamma, \mathbf{\Omega}, \boldsymbol{\gamma}) \sim \mathcal{MN}(\mathbf{X}_\gamma \mathbf{C}_\gamma, \mathbf{I}_n, \mathbf{\Omega}), \tag{3.2}$$

where $\mathcal{MN}$ denotes matrix normal distribution, and $\mathbf{X}_\gamma$ and $\mathbf{C}_\gamma$ are sub-matrices of $\mathbf{X}$ and $\mathbf{C}$ corresponding to $\boldsymbol{\gamma}$, respectively. Note that, given γ, the number of unknown parameters can be reduced to $m|\boldsymbol{\gamma}| + \binom{m}{2}$, where $|\boldsymbol{\gamma}|$ is the number of elements in $\boldsymbol{\gamma}$. We consider the following conjugate prior distributions for parameters $\mathbf{C}_\gamma$ and $\mathbf{\Omega}$:

$$\begin{aligned} \pi(\mathbf{C}_\gamma|\mathbf{\Omega}, \boldsymbol{\gamma}) &\sim \mathcal{MN}(\mathbf{0}, \zeta \mathbf{I}_{|\boldsymbol{\gamma}|}, \mathbf{\Omega}), \\ \pi(\mathbf{\Omega}) &\sim \mathcal{W}^{-1}(\mathbf{\Psi}, \nu), \end{aligned}$$

where $\mathcal{W}^{-1}$ denotes an inverse-wishart distribution and ζ, $\mathbf{\Psi}$, and ν represents deterministic hyperparameters. It can be shown that the marginal likelihood function $f(\mathbf{Y}|\boldsymbol{\gamma})$ is proportional to

$$\begin{aligned} f(\mathbf{Y}|\boldsymbol{\gamma}) &= \int\int f(\mathbf{Y}|\mathbf{C}_\gamma, \mathbf{\Omega}, \boldsymbol{\gamma}) \pi(\mathbf{C}_\gamma|\mathbf{\Omega}, \boldsymbol{\gamma}) \pi(\mathbf{\Omega}) d\mathbf{C}_\gamma d\mathbf{\Omega} \\ &\propto \zeta^{-\frac{m|\boldsymbol{\gamma}|}{2}} |\mathbf{X}_\gamma^\top \mathbf{X}_\gamma + \zeta^{-1}\mathbf{I}_{|\boldsymbol{\gamma}|}|^{-\frac{m}{2}} |\mathbf{Y}^\top \mathbf{H}_\gamma \mathbf{Y} + \mathbf{\Psi}|^{-\frac{n+\nu}{2}} \\ &\equiv s(\mathbf{Y}|\boldsymbol{\gamma}), \end{aligned} \tag{3.3}$$

where $\mathbf{H}_\gamma = \mathbf{I}_n - \mathbf{X}_\gamma(\mathbf{X}_\gamma^\top \mathbf{X}_\gamma + \zeta^{-1}\mathbf{I}_{|\boldsymbol{\gamma}|})^{-1}\mathbf{X}_\gamma^\top$ and $s(\mathbf{Y}|\boldsymbol{\gamma})$ denotes a proportional form of the $f(\mathbf{Y}|\boldsymbol{\gamma})$. Following Chen and Chen (2008), we consider

$$\pi(\boldsymbol{\gamma}) \propto \mathbb{I}(|\boldsymbol{\gamma}| \leq K) / \binom{p}{|\boldsymbol{\gamma}|}, \tag{3.4}$$

where K is the upper bound of the number of active predictors. Then, by the Bayes theorem, the marginal posterior distribution is given as

$$\pi(\boldsymbol{\gamma}|\mathbf{Y}) \propto s(\mathbf{Y}|\boldsymbol{\gamma}) \pi(\boldsymbol{\gamma}), \tag{3.5}$$

which is our model selection criterion.

3.2.2 Best subset selection

Based on the description in James et al. (2014), the best subset selection is defined as finding the k most important variables out of p predictors. The best subset selection algorithm is described in Algorithm 4.

Algorithm 4 Best subset selection

i). Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.

ii). Fixed size: for $k = 1, 2, \ldots, p$,

ii.1). Fit all $\binom{p}{k}$ models that contain exactly k predictors.

ii.2). Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$.

iii). Varying size: Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using a model selection criterion.

In Algorithm 4, James et al. (2014) used a certain model selection criterion, such as R^2 or BIC, to measure the model performance. In this case, a model with the smallest R^2 or BIC is considered to be the *best* model. In our Bayesian MVRM framework, the *best* model is obtained by maximizing $\pi(\boldsymbol{\gamma}|\mathbf{Y})$ in (3.5). As James et al. (2014) commented, while the method of best subset selection is simple and conceptually appealing, the exhaustive evaluation of model selection criterion for all possible models makes it computationally infeasible when p is greater than 40. Note that, for each k, Step ii) requires to consider $\binom{p}{k}$ candidate models to select the best subset model of size k, $\mathcal{M}_k$. If we are interested in selecting the overall best model, we need to evaluate all 2^p candidate models. For example, if $p = 1000$ and $k = 5$, it requires to evaluate $\binom{1000}{5} \approx 8 \times 10^{12}$ candidate models. To select the overall best model, even if p is small, say 40, we still need to evaluate $2^{40} \approx 10^{12}$ candidate models. To address this problem, we propose a fast hybrid search algorithm for Bayesian best subset selection of MVRM in Section 3.3.

3.3 Bayesian best subset selection via hybrid search

In this section, we introduce a hybrid search algorithm for Bayesian best subset selection. In addition, a fast computing strategy that evaluates all neighbor models simultaneously within a single computation is proposed to boost the computation efficiency.

3.3.1 Hybrid best subset search algorithm

Define an addition neighbor of the model $\boldsymbol{\gamma}$ as $\text{nbd}_+(\boldsymbol{\gamma}) = \{\boldsymbol{\gamma} \cup \{j\} : j \notin \boldsymbol{\gamma}\}$, which is a model set including all models obtained by adding one predictor from the $p - k$ remaining predictors. Define a deletion neighbor of $\boldsymbol{\gamma}$ as $\text{nbd}_-(\boldsymbol{\gamma}) = \{\boldsymbol{\gamma} \setminus \{j\} : j \in \boldsymbol{\gamma}\}$, which includes all models obtained by deleting one predictor in $\boldsymbol{\gamma}$. The hybrid best subset search algorithm that we propose in this chapter is summarized in Algorithm 5. Algorithm 5 consists of two

Algorithm 5 Hybrid best subset search with a fixed k

1. Define $\hat{\boldsymbol{\gamma}}$ as the current model with $|\hat{\boldsymbol{\gamma}}| = k$.
2. (**Deterministic search**) Repeat the following steps until get convergence:
   i) Compute $s(\mathbf{Y}|\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}})$. Update $\tilde{\boldsymbol{\gamma}}$ to be the maximizer of $s(\mathbf{Y}|\boldsymbol{\gamma})$;
   ii) Compute $s(\mathbf{Y}|\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \text{nbd}_-(\tilde{\boldsymbol{\gamma}})$. Update $\hat{\boldsymbol{\gamma}}$ to be the maximizer of $s(\mathbf{Y}|\boldsymbol{\gamma})$.
3. Set $\hat{\boldsymbol{\gamma}}^{(0)} \leftarrow \hat{\boldsymbol{\gamma}}$.
4. (**Stochastic search**) Iterate the following steps for $t = 1, \ldots, T$:
   i) Sample $\tilde{\boldsymbol{\gamma}}^{(t)}$ with probabilities proportional to $s_\alpha(\mathbf{Y}|\boldsymbol{\gamma})$ for $\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t-1)})$;
   ii) Sample $\hat{\boldsymbol{\gamma}}^{(t)}$ with probabilities proportional to $s_\alpha(\mathbf{Y}|\boldsymbol{\gamma})$ for $\boldsymbol{\gamma} \in \text{nbd}_-(\tilde{\boldsymbol{\gamma}}^{(t)})$;
   iii) If $\hat{\boldsymbol{\gamma}}^{(t)}$ is better than $\hat{\boldsymbol{\gamma}}$, then update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t)}$ and immediately go to step 2.
5. Return $\hat{\boldsymbol{\gamma}}$.

phases: a deterministic search (to achieve a local optimum) and a stochastic search (to check whether the deterministic search achieves the global optimum). Since the model size is fixed (i.e., $|\boldsymbol{\gamma}| = k$ is given), $\pi(\boldsymbol{\gamma})$ in (3.4) can be reduced to a constant, $\pi(\boldsymbol{\gamma}) \propto 1$. Therefore,

finding the Bayesian best subset of size k can be simplified as

$$\max_{\boldsymbol{\gamma}:|\boldsymbol{\gamma}|=k} s(\mathbf{Y}|\boldsymbol{\gamma}).$$

In the first phase, as discussed in Chapter 2, the deterministic search monotonically increases $s(\mathbf{Y}|\boldsymbol{\gamma})$ and terminates in a finite number of iterations. However, the drawback is that the algorithm is sensitive to the choice of the initial model $\hat{\boldsymbol{\gamma}}$ in Step 1 and the convergence to the global optimum is not guaranteed. Therefore, the second phase (stochastic search) is then introduced to explore a potentially better solution than the local one. If we find a better model than the local maximum (i.e. $s(\mathbf{Y}|\hat{\boldsymbol{\gamma}}^{(t)}) > s(\mathbf{Y}|\hat{\boldsymbol{\gamma}})$ happens), we update $\hat{\boldsymbol{\gamma}}$ by $\hat{\boldsymbol{\gamma}}^{(t)}$ and immediately go back to the deterministic search phase. To further improve the possibility of an escape from a local trap, the escort distribution, defined as $s_\alpha(\mathbf{Y}|\boldsymbol{\gamma}) = \frac{\{s(\mathbf{Y}|\boldsymbol{\gamma})\}^\alpha}{\sum_{\boldsymbol{\gamma}}\{s(\mathbf{Y}|\boldsymbol{\gamma})\}^\alpha}$, is introduced, where $\alpha \in (0, 1]$ is a pre-specified tuning parameter. To explain the role of escort distribution, let us consider escort probabilities of three candidate models with $\alpha = 1$ as follows:

$$s_1(\mathbf{Y}|\boldsymbol{\gamma}) = \left.\frac{\{s(\mathbf{Y}|\boldsymbol{\gamma})\}^\alpha}{\sum_{\boldsymbol{\gamma}}\{s(\mathbf{Y}|\boldsymbol{\gamma})\}^\alpha}\right|_{\alpha=1} = \begin{cases} 0.02 & \text{model 1} \\ 0.90 & \text{model 2} \\ 0.08 & \text{model 3} \end{cases}.$$

Note that model 2 dominates the other two models. However, if we set α close to 0, then the distribution of candidate models would be more dispersed, which flattens the dominant probability and improves small probable models (see Fig. 3.2). We set $\alpha = \min\{1, \frac{\log(2)}{\log(s_{(1)}/s_{(2)})}\}$, where $s_{(1)}$ and $s_{(2)}$ are first and second-largest values of $s(\mathbf{Y}|\boldsymbol{\gamma})$ among $\text{nbd}_+(\boldsymbol{\gamma})$ (or $\text{nbd}_-(\boldsymbol{\gamma})$), so that that first largest value is at most 2 times of the second largest value.

To select the overall best model, we need to extend the hybrid best subset search in Algorithm 5. With varying k, $\pi(\boldsymbol{\gamma})$ cannot be reduced to a constant. Therefore, implementing

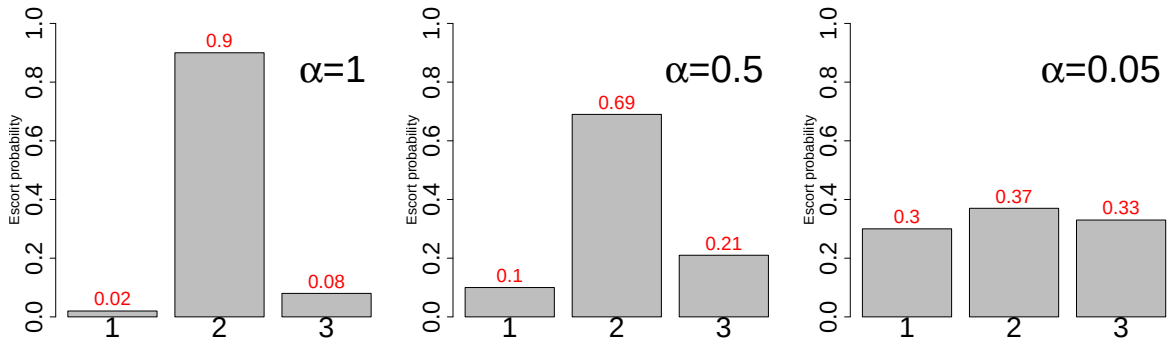

Figure 3.2: Escort distributions of $\pi_\alpha(\boldsymbol{\gamma}|\mathbf{y})$ with $\alpha = 1$, $\alpha = 0.5$, and $\alpha = 0.05$.

Bayesian best subset selection in the MVRM is equivalent to computing

$$\hat{\boldsymbol{\gamma}} = \arg\max_{\boldsymbol{\gamma}} s(\mathbf{Y}|\boldsymbol{\gamma})\mathbb{I}(|\boldsymbol{\gamma}| = k)/\binom{p}{k} \quad \text{subject to } k \leq K. \tag{3.6}$$

The extended algorithm is described in Algorithm 6.

Algorithm 6 Hybrid best subset search with varying k

1. Let $\hat{\boldsymbol{\gamma}}_0$ denote the *null model*, which contains no predictors.
2. Fixed size: for each $k = 1, 2, \ldots, K$,
   Select the best subset model $\hat{\boldsymbol{\gamma}}_k$ via the hybrid best subset search in Algorithm 5.
3. Varying size: pick the single best model among $\hat{\boldsymbol{\gamma}}_0, \ldots, \hat{\boldsymbol{\gamma}}_K$ by finding $\hat{\boldsymbol{\gamma}} = \arg\max_{0 \leq k \leq K}\{s(\mathbf{Y}|\hat{\boldsymbol{\gamma}}_k)/\binom{p}{k}\}$.

3.3.2 Fast computing strategy

As we can see in the hybrid search algorithm, it requires to evaluate (3.3) for p times in each iteration, which involves computations of determinants and inverse matrices that cause heavy computation workloads on the CPU. To release this computation burden, we propose a fast computing algorithm that computes all $s(\mathbf{Y}|\boldsymbol{\gamma})$ simultaneously within a single computation.

Denote $\hat{\boldsymbol{\gamma}}$ an index set of the current model with $|\hat{\boldsymbol{\gamma}}| = k$. For any $i \notin \hat{\boldsymbol{\gamma}}$, it can be shown

that $s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\})$ in (3.3) can be expressed as follows:

$$s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\}) \quad \propto \quad (\zeta^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i)^{-m/2} \left| \mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi} - \frac{\mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{Y}}{\zeta^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i} \right|^{-\frac{n+\nu}{2}} \tag{3.7}$$

The proof of (3.7) is given in Appendix A.2.2. It is apparent that computing $s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\})$ in (3.7) is more efficient than in (3.3). For example, since $s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\})$ in (3.7) shares the same $\mathbf{H}_{\hat{\boldsymbol{\gamma}}}$ for all $i \notin \hat{\boldsymbol{\gamma}}$, we only need to compute $\mathbf{H}_{\hat{\boldsymbol{\gamma}}}$ once for all $i \notin \hat{\boldsymbol{\gamma}}$, whereas (3.3) requires $p - k$ times of computing $\mathbf{H}_{\hat{\boldsymbol{\gamma}} \cup \{i\}}$ for all $i \notin \hat{\boldsymbol{\gamma}}$. By Lemma 2 and 3 in Appendix A.4, (3.7) can be further reduced to

$$s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\}) \quad \propto \quad (\zeta^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i)^{-m/2} \left[1 - \frac{\mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{Y} (\mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi})^{-1} \mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i}{\zeta^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i} \right]^{-\frac{n+\nu}{2}}.$$

It is important to note that $\zeta^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i$ is the i^{th} diagonal element of $\zeta^{-1}\mathbf{I}_p + \mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{X}$ and that $\mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{x}_i$ is the i^{th} column of $\mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{X}$. Therefore, in the addition neighbor of $\hat{\boldsymbol{\gamma}}$ ($\text{nbd}_+(\hat{\boldsymbol{\gamma}}) = \{\hat{\boldsymbol{\gamma}} \cup \{i\} : i \in \hat{\boldsymbol{\gamma}}\}$), $s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\})$ is equal to the i-th element of the following vector:

$$\begin{aligned} \mathbf{s}_+(\hat{\boldsymbol{\gamma}}) \quad \propto \quad & \left(\zeta^{-1}\mathbf{1}_p + \text{diag}(\mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{X})\right)^{-m/2} \cdot \\ & \left[\mathbf{1}_p - \frac{\text{diag}(\mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{Y} (\mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi})^{-1} \mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{X})}{\zeta^{-1}\mathbf{1}_p + \text{diag}(\mathbf{X}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}} \mathbf{X})} \right]^{-\frac{n+\nu}{2}}, \end{aligned} \tag{3.8}$$

where $\mathbf{a}^x = (a_1^x, \ldots, a_p^x)$, $\mathbf{a} \cdot \mathbf{b} = (a_1 b_1, \ldots, a_p b_p)$, $\mathbf{a}/\mathbf{b} = (a_1/b_1, \ldots, a_p/b_p)$ for generic vectors $\mathbf{a}$ and $\mathbf{b}$. Hence, computing all $s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\})$ for all $i \notin \hat{\boldsymbol{\gamma}}$ can be completed simultaneously within one single computation by evaluating the sub-vector of $\mathbf{s}(\mathbf{Y}|\text{nbd}_+(\hat{\boldsymbol{\gamma}}))$ in (3.8).

Let $\tilde{\boldsymbol{\gamma}}$ be an index set with $|\tilde{\boldsymbol{\gamma}}| = k + 1$. For any $\ell \in \tilde{\boldsymbol{\gamma}}$, it can be shown that

$$s(\mathbf{Y}|\tilde{\boldsymbol{\gamma}} \setminus \{\ell\}) \propto (\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell)^{-m/2} \left| \mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi} + \frac{\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y}}{\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell} \right|^{-\frac{n+\nu}{2}}. \tag{3.9}$$

The proof of (3.9) is given in Appendix A.2.3. By Lemma 2 and 3 in Appendix A.4, the

(3.9) can be expressed as:

$$s(\mathbf{Y}|\tilde{\boldsymbol{\gamma}} \setminus \{\ell\}) \propto (\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell)^{-m/2} \left[1 + \frac{\mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} (\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi})^{-1} \mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell}{\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell}\right]^{-\frac{n+\nu}{2}}.$$

Similarly, we can show that $s(\mathbf{Y}|\tilde{\boldsymbol{\gamma}} \setminus \{\ell\})$ is ℓ-th element of the following vector:

$$\begin{aligned} \mathbf{s}_-(\tilde{\boldsymbol{\gamma}}) \propto & \left(\zeta^{-1}\mathbf{1}_p - \mathrm{diag}(\mathbf{X}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X})\right)^{-m/2} \cdot \\ & \left[\mathbf{1}_p + \frac{\mathrm{diag}(\mathbf{X}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} (\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi})^{-1} \mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X})}{\zeta^{-1}\mathbf{1}_p - \mathrm{diag}(\mathbf{X}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X})}\right]^{-\frac{n+\nu}{2}}. \end{aligned} \tag{3.10}$$

Hence, evaluating $s(\mathbf{Y}|\tilde{\boldsymbol{\gamma}} \setminus \{\ell\})$ for all $\ell \in \tilde{\boldsymbol{\gamma}}$ can be completed simultaneously within one single computation by computing the sub-vector of $\mathbf{s}(\mathbf{Y}|\mathrm{nbd}_-(\tilde{\boldsymbol{\gamma}}))$. Applying (3.8) and (3.10) in Algorithm 5, we can further boost the computational speed of Step 2 and 4. Note that since Algorithm 5 is embedded to Step 2 in Algorithm 6, thus the fast computing strategy is also applicable to Algorithm 6.

To help readers understand our idea, two demo `R` codes for the proposed fast computing strategy and the traditional for-loop strategy are given in Lists B.1 and B.2 of Appendix B. More R codes and discussion are also available at `https://calebsjin.github.io/RdemoMVRM/chapter3.html`.

3.4 Simulation study

In this section, several simulation studies are conducted to investigate the performance of our proposed approach. We generate $\{(\mathbf{y}_i, \mathbf{x}_i) : i = 1, \ldots, n\}$ from MVRM as follows:

$$\mathbf{y}_i \overset{ind}{\sim} \mathcal{N}_m(\mathbf{C}^\top \mathbf{x}_i, \boldsymbol{\Omega}),$$

where $n = 100$, $m = 5$, $\mathbf{x}_i \overset{i.i.d}{\sim} \mathcal{N}_p(\mathbf{0}_p, \boldsymbol{\Sigma}_x)$ with $\boldsymbol{\Sigma}_x = (\rho_x^{|i-j|})_{p\times p}$, and $\boldsymbol{\Omega} = (2\rho_e^{|i-j|})_{m\times m}$. We define the true model by $\boldsymbol{\gamma} = \{1, 2, 3, 4, 7, 8, 9, 10\}$. For $\mathbf{C} = (c_{jk})_{p\times m}$, c_{jk} is randomly and independently sampled from Uniform$\{-1.5, -1, 1, 1.5\}$ if $j \in \boldsymbol{\gamma}$ and $c_{jk} = 0$ if $j \notin \boldsymbol{\gamma}$. We

consider 18 scenarios with all possible combinations of the following values of p, ρ_e and ρ_x:

1. $p = 200$, 500 and 1000;
2. ρ_e =0, 0.2 and 0.5;
3. ρ_x =0.2 and 0.8.

In this simulation study, we are interested in finding the overall best model using Algorithm 6. We assess the performance of our proposed approach by comparing it with Brown et al. (1998)'s method (Brown), mLASSO, and mENET.

For the proposed method, we assign non-informative priors on $\mathbf{C}$ and $\mathbf{\Omega}$ with $\mathbf{\Psi} = 0.5 \times \mathbf{I}_5$, $\nu = 0.5$, and $\zeta = \log(n)$. We set $K = \lceil n^{2/3} \rceil = 22$. For the number of MCMC iterations in Step 4 of Algorithm 5, we consider $T = 200$ and $T = 1000$.

For the Brown method, we use the MCMC sample size of 8000 with the burn-in period of 3000. For technical details of the Brown method, see Appendix A.2.1. For mLASSO and mENET, we choose the following four information criteria for tuning parameter selection: AIC (Akaike, 1998), bias-corrected AIC (AICc) (Bedrick, 1994), BIC (Schwarz, 1978) and consistent AIC (CAIC) (Bozdogan, 1987). Note that these information criteria can be viewed as

$$\text{IC}_{\boldsymbol{\gamma}} = nm(\log 2\pi + 1) + n \log |\hat{\mathbf{\Omega}}| + \varphi_{\boldsymbol{\gamma}}, \tag{3.11}$$

where $\hat{\mathbf{\Omega}} = \frac{1}{n}\mathbf{Y}(\mathbf{I}_n - \mathbf{P}_{\boldsymbol{\gamma}})\mathbf{Y}$, $\mathbf{P}_{\boldsymbol{\gamma}} = \mathbf{X}_{\boldsymbol{\gamma}}(\mathbf{X}_{\boldsymbol{\gamma}}^\top \mathbf{X}_{\boldsymbol{\gamma}})^{-1}\mathbf{X}_{\boldsymbol{\gamma}}^\top$, and

$$\varphi_{\boldsymbol{\gamma}} = \begin{cases} 2\left\{m|\boldsymbol{\gamma}| + m(m+1)/2\right\}, & \text{AIC} \\ 2n\left\{m|\boldsymbol{\gamma}| + m(m+1)/2\right\}/(n - |\boldsymbol{\gamma}| - m - 1), & \text{AICc} \\ \left\{m|\boldsymbol{\gamma}| + m(m+1)/2\right\}\log n, & \text{BIC} \\ \left\{m|\boldsymbol{\gamma}| + m(m+1)/2\right\}(1 + \log n), & \text{CAIC} \end{cases}.$$

To make a fair comparison with the proposed method, we apply the constraint of upper

bound $K = 22$ to the selection of tuning parameter of mLASSO and mENET. We use `R` package `glmnet` to implement mLASSO and mENET.

To assess the performance of all approaches from different perspectives, based on 100 Monte Carlo replications, we compute false discover rate (FDR), percentage of selecting the exact true model (TRUE), the average size of the selected model (SIZE), the average Hamming distance (HAM), and the average execution time in second (TIME). We define FDR $= \frac{|\hat{\gamma} \setminus \gamma^*|}{|\hat{\gamma}|}$ and HAM $= |\hat{\gamma} \setminus \gamma^*| + |\gamma^* \setminus \hat{\gamma}|$, where γ^* and $\hat{\gamma}$ are true model and selected model, respectively.

The results of the simulation study are summarized in Tables 3.1, 3.2, and 3.3. Due to the space constraint, we show the results of mLASSO and mENET with CAIC as it provides the best performance. The results for all information criteria are given in Appendix B. Proposed-200 and Proposed-10^3 represent our proposed method with 200 and 1000 MCMC iterations, respectively. The results of Proposed-200 and Proposed-1000 are the same, indicating that 200 is large enough to reach the convergence in Algorithm 6. It is obvious that when $p = 200$ and $\rho_x = 0.2$, the results of all methods are not significantly different for all ρ_e. However, when there exists strong multicollinearity ($\rho_x = 0.8$) in predictors, the performance of our proposed method is good and better than any other methods for all p and ρ_e in terms of FDR, TRUE and HAM. Brown is the second-best, and the next is mLASSO. Compared with the Brown, mLASSO, and mENET in all scenarios, we conclude that our proposed method selects inactive predictors with the smallest possibility (by FDR), the exact true model with the highest probability (by TRUE) and the closest model to the true model (by SIZE and HAM).

From the computation speed, Proposed-200 costs about 7 to 14 seconds, while Brown costs 40 to 120 times more than the proposed method. For mLASSO and mENET, the computational time is within a second due to the use of the well-developed R package (Friedman et al., 2010). However, recall that their performances are very poor especially when the predictors are strongly correlated. Note that the proposed method could be further accelerated by using parallel computing.

Table 3.1: Simulation study of all scenarios of $\rho_e = 0$ based on 100 replications.

Method	FDR (s.e)	TRUE (s.e)	SIZE (s.e)	HAM (s.e)	TIME (s.e)
			$p = 200, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	7.054(0.048)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	32.301(0.167)
Brown	0.002(0.002)	98(1.407)	8.02(0.014)	0.02(0.014)	302.692(0.779)
mLASSO-CAIC	0(0)	100(0)	8(0)	0(0)	0.433(0.002)
mENET-CAIC	0(0)	100(0)	8(0)	0(0)	0.427(0.003)
			$p = 200, \rho_x = 0.8$		
Proposed-200	0(0)	98(1.407)	7.98(0.014)	0.02(0.014)	7.635(0.067)
Proposed-10^3	0(0)	98(1.407)	7.98(0.014)	0.02(0.014)	33.845(0.209)
Brown	0.015(0.004)	84(3.685)	8.1(0.046)	0.18(0.044)	284.772(1.222)
mLASSO-CAIC	0.024(0.005)	69(4.648)	8.04(0.063)	0.4(0.067)	0.556(0.005)
mENET-CAIC	0.066(0.009)	49(5.024)	8.49(0.104)	0.79(0.092)	0.536(0.004)
			$p = 500, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	9.83(0.085)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	43.345(0.209)
Brown	0.006(0.002)	95(2.19)	8.05(0.022)	0.05(0.022)	846.797(7.616)
mLASSO-CAIC	0.001(0.001)	99(1)	8.01(0.01)	0.01(0.01)	0.503(0.003)
mENET-CAIC	0(0)	100(0)	8(0)	0(0)	0.492(0.003)
			$p = 500, \rho_x = 0.8$		
Proposed-200	0(0)	98(1.407)	7.98(0.014)	0.02(0.014)	10.393(0.082)
Proposed-10^3	0(0)	98(1.407)	7.98(0.014)	0.02(0.014)	45.609(0.353)
Brown	0.012(0.004)	83(3.775)	8.02(0.047)	0.2(0.047)	837.947(14.617)
mLASSO-CAIC	0.024(0.005)	63(4.852)	7.96(0.075)	0.48(0.07)	0.637(0.005)
mENET-CAIC	0.068(0.009)	43(4.976)	8.43(0.105)	0.87(0.092)	0.62(0.004)
			$p = 1000, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	14.182(0.12)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	64.465(0.421)
Brown	0.008(0.003)	93(2.564)	8.07(0.026)	0.07(0.026)	1752.643(5.724)
mLASSO-CAIC	0.002(0.002)	98(1.407)	8.02(0.014)	0.02(0.014)	0.624(0.004)
mENET-CAIC	0.002(0.002)	98(1.407)	8.02(0.014)	0.02(0.014)	0.606(0.004)
			$p = 1000, \rho_x = 0.8$		
Proposed-200	0(0)	94(2.387)	7.93(0.029)	0.07(0.029)	14.691(0.116)
Proposed-10^3	0(0)	94(2.387)	7.93(0.029)	0.07(0.029)	63.215(0.461)
Brown	0.014(0.004)	82(3.861)	8.07(0.046)	0.19(0.042)	1729.196(8.956)
mLASSO-CAIC	0.025(0.006)	61(4.902)	7.92(0.091)	0.56(0.082)	0.79(0.008)
mENET-CAIC	0.078(0.009)	39(4.902)	8.45(0.115)	1.05(0.119)	0.765(0.006)

Table 3.2: Simulation study of all scenarios of $\rho_e = 0.2$ based on 100 replications.

Method	FDR (s.e)	TRUE (s.e)	SIZE (s.e)	HAM (s.e)	TIME (s.e)
			$p = 200, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	7.067(0.048)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	32.47(0.178)
Brown	0.003(0.002)	97(1.714)	8.03(0.017)	0.03(0.017)	303.72(0.817)
mLASSO-CAIC	0(0)	100(0)	8(0)	0(0)	0.463(0.003)
mENET-CAIC	0(0)	100(0)	8(0)	0(0)	0.457(0.004)
			$p = 200, \rho_x = 0.8$		
Proposed-200	0(0)	99(1)	7.99(0.01)	0.01(0.01)	7.799(0.072)
Proposed-10^3	0(0)	99(1)	7.99(0.01)	0.01(0.01)	34.205(0.274)
Brown	0.015(0.004)	84(3.685)	8.1(0.046)	0.18(0.044)	286.251(1.186)
mLASSO-CAIC	0.031(0.006)	67(4.726)	8.13(0.072)	0.45(0.073)	0.544(0.004)
mENET-CAIC	0.062(0.009)	51(5.024)	8.42(0.102)	0.78(0.096)	0.529(0.004)
			$p = 500, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	9.765(0.075)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	43.654(0.236)
Brown	0.006(0.003)	95(2.19)	8.06(0.028)	0.06(0.028)	825.779(7.549)
mLASSO-CAIC	0(0)	100(0)	8(0)	0(0)	0.537(0.003)
mENET-CAIC	0(0)	100(0)	8(0)	0(0)	0.527(0.004)
			$p = 500, \rho_x = 0.8$		
Proposed-200	0(0)	96(1.969)	7.96(0.02)	0.04(0.02)	10.315(0.085)
Proposed-10^3	0(0)	96(1.969)	7.96(0.02)	0.04(0.02)	45.333(0.35)
Brown	0.013(0.004)	84(3.685)	8.05(0.041)	0.19(0.046)	839.514(14.583)
mLASSO-CAIC	0.024(0.005)	65(4.794)	7.97(0.07)	0.47(0.072)	0.63(0.004)
mENET-CAIC	0.07(0.009)	45(5)	8.47(0.118)	0.91(0.105)	0.62(0.005)
			$p = 1000, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	14.323(0.101)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	61.283(0.346)
Brown	0.011(0.004)	91(2.876)	8.1(0.033)	0.1(0.033)	1751.256(5.606)
mLASSO-CAIC	0.002(0.002)	98(1.407)	8.02(0.014)	0.02(0.014)	0.676(0.006)
mENET-CAIC	0.002(0.002)	98(1.407)	8.02(0.014)	0.02(0.014)	0.645(0.004)
			$p = 1000, \rho_x = 0.8$		
Proposed-200	0(0)	93(2.564)	7.93(0.026)	0.07(0.026)	14.707(0.122)
Proposed-10^3	0(0)	93(2.564)	7.93(0.026)	0.07(0.026)	66.799(0.456)
Brown	0.017(0.004)	80(4.02)	8.08(0.051)	0.22(0.046)	1750.775(8.792)
mLASSO-CAIC	0.028(0.007)	61(4.902)	7.94(0.09)	0.6(0.098)	0.776(0.006)
mENET-CAIC	0.081(0.01)	38(4.878)	8.49(0.123)	1.09(0.121)	0.754(0.006)

Table 3.3: Simulation study of all scenarios of $\rho_e = 0.5$ based on 100 replications.

Method	FDR (s.e)	TRUE (s.e)	SIZE (s.e)	HAM (s.e)	TIME (s.e)
			$p = 200, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	6.994(0.048)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	31.398(0.164)
Brown	0.002(0.002)	98(1.407)	8.02(0.014)	0.02(0.014)	299.103(0.687)
mLASSO-CAIC	0(0)	100(0)	8(0)	0(0)	0.452(0.003)
mENET-CAIC	0(0)	100(0)	8(0)	0(0)	0.444(0.004)
			$p = 200, \rho_x = 0.8$		
Proposed-200	0(0)	96(1.969)	7.96(0.02)	0.04(0.02)	7.606(0.064)
Proposed-10^3	0(0)	96(1.969)	7.96(0.02)	0.04(0.02)	33.918(0.233)
Brown	0.011(0.003)	87(3.38)	8.06(0.04)	0.14(0.038)	284.063(1.011)
mLASSO-CAIC	0.037(0.008)	67(4.726)	8.19(0.098)	0.55(0.101)	0.549(0.005)
mENET-CAIC	0.07(0.009)	51(5.024)	8.51(0.107)	0.85(0.106)	0.529(0.004)
			$p = 500, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	9.692(0.081)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	42.627(0.275)
Brown	0.007(0.003)	94(2.387)	8.06(0.024)	0.06(0.024)	812.635(7.497)
mLASSO-CAIC	0(0)	100(0)	8(0)	0(0)	0.53(0.004)
mENET-CAIC	0(0)	100(0)	8(0)	0(0)	0.52(0.004)
			$p = 500, \rho_x = 0.8$		
Proposed-200	0(0)	95(2.19)	7.95(0.022)	0.05(0.022)	10.321(0.089)
Proposed-10^3	0(0)	95(2.19)	7.95(0.022)	0.05(0.022)	45.724(0.367)
Brown	0.012(0.004)	83(3.775)	8.03(0.044)	0.19(0.044)	796.033(13.979)
mLASSO-CAIC	0.037(0.009)	64(4.824)	8.17(0.125)	0.63(0.123)	0.619(0.005)
mENET-CAIC	0.081(0.01)	44(4.989)	8.6(0.113)	1(0.116)	0.607(0.005)
			$p = 1000, \rho_x = 0.2$		
Proposed-200	0(0)	100(0)	8(0)	0(0)	14.131(0.1)
Proposed-10^3	0(0)	100(0)	8(0)	0(0)	63.703(0.416)
Brown	0.009(0.003)	93(2.564)	8.08(0.031)	0.08(0.031)	1722.443(4.872)
mLASSO-CAIC	0.002(0.002)	98(1.407)	8.02(0.014)	0.02(0.014)	0.663(0.006)
mENET-CAIC	0.001(0.001)	99(1)	8.01(0.01)	0.01(0.01)	0.641(0.004)
			$p = 1000, \rho_x = 0.8$		
Proposed-200	0(0)	93(2.564)	7.93(0.026)	0.07(0.026)	14.804(0.12)
Proposed-10^3	0(0)	93(2.564)	7.93(0.026)	0.07(0.026)	63.869(0.43)
Brown	0.011(0.003)	85(3.589)	8.04(0.042)	0.16(0.039)	1699.55(7.121)
mLASSO-CAIC	0.027(0.007)	62(4.878)	7.89(0.099)	0.63(0.098)	0.763(0.006)
mENET-CAIC	0.084(0.01)	37(4.852)	8.5(0.128)	1.14(0.127)	0.749(0.006)

3.5 Real data analysis

In this section, we apply our proposed approach to mRNA gene expression data called Breast Invasive Carcinoma (BRCA). The data are obtained from the R package `curatedTCGAData` written by Ramos (2019). BRCA data set comprises 526 patients with primary solid tumor and 17814 gene expression measurements in total. BRCA1 and BRCA2 are well-known genes that account for hereditary breast cancer (Yang and Lippman, 1999). Our goal here is to determine the best model for estimating a relationship between BRCA1 & BRCA2 (bivariate response variable) and the other genes (predictors).

After removing some missing data, we have $n = 526$ samples with $17,327$ genes left. In our real data analysis, $m = 2$ represents the number of responses (BRCA1 and BRCA2), and remaining $p = 17,325$ genes are served as independent variables. To compare the performance of the proposed method, we apply the Brown, mLASSO, mENET, LASSO, and ENET to the BRCA data set. To demonstrate the benefit of multivariate approach, we also employ LASSO, ENET, and the method proposed in Chapter 2 (Ch.2 method) by assuming independent errors across the responses and using BRCA1 or BRCA2 as the univariate response variable. Each of LASSO, ENET, and Ch.2 methods will obtain two sets of predictors related to BRCA1 and BRCA2, respectively. The final variable selection result is the combination of two sets of predictors. For the Brown method, the result is based on $12,000$ MCMC samples obtained after an $8,000$ burn-in period with hyperparameters defined in Appendix A.2.1. For mLASSO and mENET, we use AIC, AICc, BIC, and CAIC to select the tuning parameters as in Section 3.4. For LASSO and ENET, the tuning parameters are selected by the extended BIC (EBIC) of Chen and Chen (2008).

We apply Brown, mLASOO, mENET, Ch.2, LASSO, ENET and the proposed methods to the whole data set. To measure statistical significance, we compute the p-value of each selected predictor by fitting the corresponding multivariate linear regression models for each method. Fig. 3.3 displays the number of genes selected by each method and the number of significant coefficients (p-value < 0.05). To evaluate prediction performance, we compute the mean squared prediction error (MSPE), AIC, AICc, BIC, and CAIC by Monte Carlo

cross-validation with the 70% training and 30% testing data sets based on 500 Monte Carlo replications. The result is shown in Table 3.4.

In Table 3.4, the proposed method has smallest average values of MSPE, AIC, AICc, BIC and CAIC. The performance of mLASSO is similar to mENET in terms of MSPE, and they are better than LASSO and ENET. In Fig. 3.3, the proposed approach and the Ch.2 method select 23 and 30 genes, respectively, and all of them are significant, while some genes selected by the penalized likelihood approaches are not significant. Compared with the Ch.2 method, the proposed method selects 7 more significant genes, which shows the merit of our multivariate approach. Hence, we conclude that the proposed method identifies the best fitting model to BRCA data.

Table 3.4: Summary of model comparisons for BRCA data analysis

Method	AIC	AICc	BIC	CAIC	MSPE
Proposed	368.77	381.15	615.15	678.15	0.692
Brown	917.16	918.57	999.29	1020.29	0.989
Ch.2 method	540.66	548.08	732.28	781.28	0.770
mLASSO-AIC	547.52	607.62	1067.66	1200.66	0.808
mLASSO-AICc	547.52	607.62	1067.66	1200.66	0.808
mLASSO-BIC	703.04	717.08	965.06	1032.06	0.864
mLASSO-CAIC	703.04	717.08	965.06	1032.06	0.864
mENET-AIC	562.47	618.67	1066.97	1195.97	0.810
mENET-AICc	562.47	618.67	1066.97	1195.97	0.810
mENET-BIC	705.52	721.34	983.18	1054.18	0.866
mENET-CAIC	705.52	721.34	983.18	1054.18	0.866
LASSO-EBIC	901.06	902.23	975.37	994.37	0.967
ENET-EBIC	902.36	903.77	984.49	1005.49	0.968

3.6 Concluding remarks

In this chapter, we have proposed a novel Bayesian variable selection method under the high-dimensional MVRM. The proposed method combines a deterministic local search with a stochastic global search to perform fast best subset selection. In the deterministic search step, we efficiently evaluate all marginal likelihoods of neighbor models simultaneously by the application of the Sherman-Morrison formula. In the stochastic search step, an MCMC

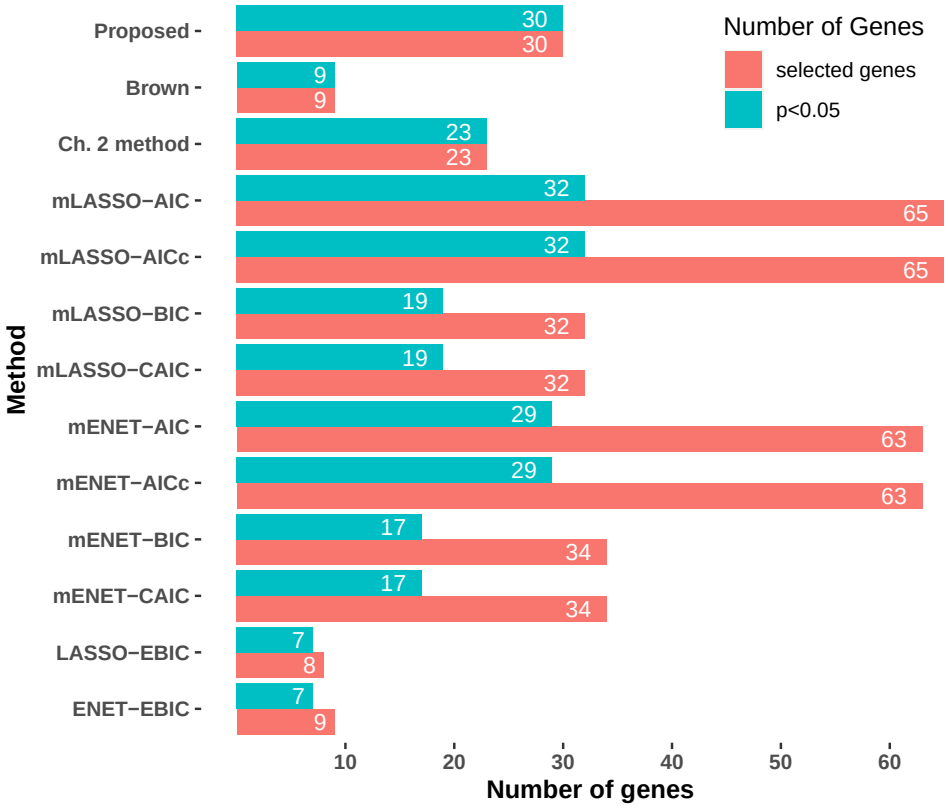

Figure 3.3: The number of selected genes and number of significant coefficients.

procedure is conducted to check whether or not the local optimum is the global solution. The proposed method can be extended to the problem of best subset selection with frequentist model selection criteria, such as AICc and CAIC.

Chapter 4

An approximate Bayesian approach to fast high-dimensional variable selection

In this chapter, we introduce a novel Bayesian high-dimensional variable selection approach that offers the ability to swiftly identify the best model using a single CPU core. The key feature of the proposed method is that it can be applicable to diverse data types, including continuous, binary, count, and survival time.

The remainder of this chapter is organized as follows. In Section 4.1, we describe a general Bayesian model setting for variable selection. Section 4.2 provides a review of some Bayesian model selection algorithms. Section 4.3 introduces the proposed method. In Section 4.4, we exemplify the application of the proposed method for several data types. In Section 4.5, simulation studies are conducted to compare the proposed approach to the hybrid search that computes the exact objective function. In Section 4.6, three real data analyses are conducted to assess the performance of the proposed approach. In Section 4.7, we give some concluding comments and discussion.

4.1 Approximate Bayesian model selection

Suppose n pairs of observations $\{(\mathbf{x}_i, y_i), i = 1, 2, \ldots, n\}$ are identically and independently distributed, where $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^\top$ denotes a p-dimensional vector of predictors. Define $\mathbf{y} = (y_1, y_2, \ldots, y_n)^\top$, where y_i can be binary, categorical, count, or continuous. Let $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top = (\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \ldots, \tilde{\mathbf{x}}_p) \in \mathcal{R}^{n \times p}$, where $\tilde{\mathbf{x}}_j$ is j-th column of $\mathbf{X}$. Suppose $E(y_i \mid \mathbf{x}_i) = h(\mathbf{x}_i^\top \boldsymbol{\theta})$, where $h(\cdot)$ is a known link function. Let $\boldsymbol{\gamma} = \{j : \theta_j \neq 0\}$ represent a reduced model and $|\boldsymbol{\gamma}|$ be the number of active predictor in model $\boldsymbol{\gamma}$. Thus, $\boldsymbol{\theta}_{\boldsymbol{\gamma}}$ is a $|\boldsymbol{\gamma}|$-dimension sub-vector of $\boldsymbol{\theta}$, and $\mathbf{X}_{\boldsymbol{\gamma}}$ is an $n \times |\boldsymbol{\gamma}|$ sub-matrix of $\mathbf{X}$. After assuming the distribution of $\mathbf{y}$, the probability distribution of $\mathbf{y}$ given the model $\boldsymbol{\gamma}$ can be represented by the likelihood function $f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})$.

Let $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma})$ be the prior of $\boldsymbol{\theta}_{\boldsymbol{\gamma}}$ give $\boldsymbol{\gamma}$. By Bayes theorem, we obtain the posterior probability of model γ by $\pi(\boldsymbol{\gamma}|\mathbf{y}) \propto m(\mathbf{y}|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma})$, where $m(\mathbf{y}|\boldsymbol{\gamma}) = \int f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) d\boldsymbol{\theta}_{\boldsymbol{\gamma}}$ is the marginal likelihood function given $\boldsymbol{\gamma}$. Thus, $\pi(\boldsymbol{\gamma}|\mathbf{y})$ or $m(\mathbf{y}|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma})$ can be used as the Bayesian model selection criterion. However, except the Gaussian case, calculation of an explicit form of the marginal likelihood $m(\mathbf{y}|\boldsymbol{\gamma})$ is intractable. Alternatively, an approximate marginal likelihood can be obtained by the Laplace approximation method,

$$\begin{aligned} m(\mathbf{y}|\boldsymbol{\gamma}) &\approx (2\pi)^{|\boldsymbol{\gamma}|/2} |\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\gamma}}|^{1/2} f(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma}) \pi(\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) \\ &= (2\pi n^{-1})^{|\boldsymbol{\gamma}|/2} |\hat{\mathbf{V}}_{\boldsymbol{\gamma}}|^{1/2} f(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma}) \pi(\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}), \end{aligned} \tag{4.1}$$

where

$$\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}} = \arg\max_{\boldsymbol{\theta}_{\boldsymbol{\gamma}}} f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma}) \pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) \tag{4.2}$$

denotes the maximum a posterior (MAP) estimate of $\boldsymbol{\theta}_{\boldsymbol{\gamma}}$, $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\gamma}}$ is the inverse of the negative Hessian matrix of $\log[f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma})]$ evaluated at $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}$, and $\hat{\mathbf{V}}_{\boldsymbol{\gamma}} = n\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\gamma}}$. By treating $|\hat{\mathbf{V}}_{\boldsymbol{\gamma}}|$ as a constant for large n by the law of large numbers, it can be seen that (4.1) is proportional

to $(2\pi n^{-1})^{|\gamma|/2} f(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})\pi(\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma})$. Therefore, we can obtain that

$$\pi(\boldsymbol{\gamma}|\mathbf{y}) \propto m(\mathbf{y}|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma}) \approx (2\pi n^{-1})^{|\gamma|/2} f(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})\pi(\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma}) \equiv S(\boldsymbol{\gamma}). \tag{4.3}$$

Throughout this chapter, $S(\boldsymbol{\gamma})$ is served as the Bayesian model selection criterion.

4.2 Bayesian model selection algorithms

In this section, we review three Bayesian model selection algorithms: deterministic search, stochastic search, and hybrid search. Since all the methods are based on the idea of the neighborhood search (Madigan and York, 1995; Hans et al., 2007), we first introduce the concept of neighborhood space.

4.2.1 Neighborhood space

In a Bayesian model averaging context, Madigan and York (1995) introduced the concept of the model's neighborhood space, which contains the current model and a set of models with either one predictor more or one predictor fewer. Let $\hat{\boldsymbol{\gamma}}$ be the current model such that $|\hat{\boldsymbol{\gamma}}| = k$. Given $\hat{\boldsymbol{\gamma}}$, the neighborhood of $\hat{\boldsymbol{\gamma}}$ is defined by the union of an addition neighbor $\text{nbd}_+(\hat{\boldsymbol{\gamma}})$ and a deletion neighbor $\text{nbd}_-(\hat{\boldsymbol{\gamma}})$, where $\text{nbd}_+(\hat{\boldsymbol{\gamma}}) = \{\hat{\boldsymbol{\gamma}} \cup \{j\}, j \notin \hat{\boldsymbol{\gamma}}\}$ and $\text{nbd}_-(\hat{\boldsymbol{\gamma}}) = \{\hat{\boldsymbol{\gamma}} \setminus \{j\}, j \in \hat{\boldsymbol{\gamma}}\}$. Note that $\text{nbd}_+(\hat{\boldsymbol{\gamma}})$ has $p-k$ neighboring models and $\text{nbd}_-(\hat{\boldsymbol{\gamma}})$ contains k many neighboring models. Hence, the neighborhood of $\hat{\boldsymbol{\gamma}}$ always contains p candidate models for any $\hat{\boldsymbol{\gamma}}$. Denote the neighborhood model space of $\hat{\boldsymbol{\gamma}}$ by

$$\text{nbd}(\hat{\boldsymbol{\gamma}}) = \text{nbd}_+(\hat{\boldsymbol{\gamma}}) \cup \text{nbd}_-(\hat{\boldsymbol{\gamma}}). \tag{4.4}$$

The neighborhood search in this chapter is viewed as a four-step procedure:

- Step 1: Given the current model, define candidate models in the neighborhood of it.
- Step 2: Evaluate all candidate models in the neighborhood.

- Step 3: Identify the best model.
- Step 4: Determine the next update.

Under the above neighborhood search framework, the method to decide the next update (Step 4) depends on algorithms the users adopt.

4.2.2 Neighborhood search algorithms

The key idea of the deterministic search algorithm is to select the local optimum as the next update. A representative deterministic search is the cyclic coordinate descent algorithm (Tseng, 2001), which successively minimizes the objective function by sequentially updating each coordinate axis. From a Bayesian perspective, the cyclic coordinate descent algorithm is equivalent to iterated conditional modes (ICM) algorithm (Besag, 1986), which sequentially updates full conditional posterior distributions. Similarly, the deterministic search based on neighborhood space can be performed by the successive maximization of $S(\boldsymbol{\gamma})$ in (4.3) over $\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}})$. Specifically, when $\hat{\boldsymbol{\gamma}}^{(t)}$ is obtained in the t-th iteration, the $(t+1)$-th iteration proceeds as follow:

$$\hat{\boldsymbol{\gamma}}^{(t+1)} \leftarrow \arg \max_{\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})} S(\boldsymbol{\gamma}).$$

The Algorithm 7 describes this procedure with more details. If we observe $S(\hat{\boldsymbol{\gamma}}^{(t+1)}) > S(\hat{\boldsymbol{\gamma}})$, then we update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t+1)}$, which means the new model is better than the current model. By iteratively updating the best model via the neighborhood search, the deterministic search algorithm quickly converges to the locally best model. This is because the deterministic search monotonically increases the objective function $S(\boldsymbol{\gamma})$ within a finite number of iterations.

To find a global maximum, we can use the stochastic search in Algorithm 8. The main idea of the stochastic search aligns with the Stochastic Shotgun Search (SSS) (Hans et al., 2007), which selects the next update by randomly sampling a model from $\text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$ with probability proportional to $S(\boldsymbol{\gamma})$ normalized within neighborhood space $\text{nbd}(\boldsymbol{\gamma}^{(t)})$. In Step 2-v) of Algorithm 8, a model with larger value of $S(\boldsymbol{\gamma})$ has higher probability to be selected

Algorithm 7 The deterministic search algorithm

1. Set $\hat{\boldsymbol{\gamma}}$ as the current model and $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$.
2. **Repeat** $t = 0, 1, 2, \ldots,$
   i). Define $\text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$;
   ii). Compute $S(\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$;
   iii). Compute $\hat{\boldsymbol{\gamma}}^{(t+1)} = \arg\max_{\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})} S(\boldsymbol{\gamma})$;
   iv). **If** $S(\hat{\boldsymbol{\gamma}}^{(t+1)}) > S(\hat{\boldsymbol{\gamma}})$, **then** update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t+1)}$
   else update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t)}$ and break the loop.
3. Return $\hat{\boldsymbol{\gamma}}$.

as the next update. Note that Algorithm 8 is analogous to an MCMC sampling with the stationary distribution $\pi(\boldsymbol{\gamma}|\mathbf{y})$. Therefore, if the current model $\hat{\boldsymbol{\gamma}}$ is not global optimum, then we always have a chance to observe $S(\boldsymbol{\gamma}^*) > S(\hat{\boldsymbol{\gamma}})$ as $T \to \infty$. In other words, the global maximum can be achieved by repeating this procedure for a sufficiently large number of iterations. Note that Algorithm 8 requires to compute $S(\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$ for every iteration. Hence, in the setting of high dimensional data, the stochastic search is computationally expensive and slow unless parallel computation is available as in Hans et al. (2007). Throughout this study, we assume that any parallel computing environments are not available to make our study more realistic.

To boost the speed of the stochastic search, in Chapter 2, we have proposed a hybrid search which combines deterministic and stochastic searches. Algorithm 9 elaborates on the idea of proposed hybrid search. In Algorithm 9, the deterministic search (in Step 2) is first employed to quickly find a local maximum, then stochastic search (in Step 4) is used to check if the deterministic search achieves the global maximum. In 4-iv) of Algorithm 9, if $S(\boldsymbol{\gamma}^*) > S(\hat{\boldsymbol{\gamma}})$ happens (i.e., a better model is found), then we update $\hat{\boldsymbol{\gamma}}$ by $\boldsymbol{\gamma}^*$ and immediately go back to Step 2. The hybrid search algorithm stops if the stochastic search does not make any update for a long time period. The Table 4.1 summarizes strength and weakness for deterministic, stochastic, and hybrid search algorithms.

As mentioned before, computing $S(\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t-1)})\}$ is very expensive in the

Algorithm 8 The stochastic search algorithm

1. Set $\hat{\boldsymbol{\gamma}}$ as the current model and $\hat{\boldsymbol{\gamma}}^{(0)} \leftarrow \hat{\boldsymbol{\gamma}}$.
2. **Repeat** for $t = 0, 1, 2, \ldots, T$,
   i). Define $\text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$.
   ii). Compute $S(\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$;
   iii). Compute $\boldsymbol{\gamma}^* = \arg\max_{\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})} S(\boldsymbol{\gamma})$.
   iv). **If** $S(\boldsymbol{\gamma}^*) > S(\hat{\boldsymbol{\gamma}})$, **then** update $\hat{\boldsymbol{\gamma}} \leftarrow \boldsymbol{\gamma}^*$.
   v). Update $\hat{\boldsymbol{\gamma}}^{(t+1)}$ by selecting one model from $\text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$ with probability

$$\frac{S(\boldsymbol{\gamma})}{\sum_{\tilde{\boldsymbol{\gamma}} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})} S(\tilde{\boldsymbol{\gamma}})}, \boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)});$$

3. Return $\hat{\boldsymbol{\gamma}}$.

Table 4.1: The strength and weakness of best model search algorithms

Method	Speed	Maximum
Deterministic	Very fast	Local
Stochastic	Slow	Global
Hybrid	Fast	Global

high-dimensional data setting. To address a similar issue, Hans et al. (2007) propose to use a parallel computing technique. However, parallel data processing relies on high performance computing (HPC) environments, which may not be available to individual data analysts. To free from this limitation, in Chapters 2 and 3, we proposed a novel calculation strategy that allows us to simultaneously evaluate the marginal likelihoods of all the candidates within a single computation using the Sherman-Morrison formula and Sylvester's determinant identity. Unfortunately, this idea is not applicable to our general setting since it is limited to Gaussian data. To address this challenge, we propose a new Bayesian approximation approach to a hybrid search framework.

Algorithm 9 The hybrid search algorithm

1. Set $\hat{\boldsymbol{\gamma}}$ as the current model and $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$.

   # Deterministic search begins

2. **Repeat** $t = 0, 1, 2, \ldots,$

   i). Define $\text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$

   ii). Compute $S(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})\}$;

   iii). Compute $\boldsymbol{\gamma}^{(t+1)} = \arg\max_{\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})} S(\boldsymbol{\gamma})$;

   iv). **If** $S(\hat{\boldsymbol{\gamma}}^{(t+1)}) > S(\hat{\boldsymbol{\gamma}})$, **then** update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t+1)}$
   else update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t)}$ and break the loop.

   # Stochastic search begins

3. Set $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$.

4. **Repeat** for $t = 0, 1, 2, \ldots, T,$

   i). Define $\text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$

   ii). Compute $S(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})\}$;

   iii). Compute $\boldsymbol{\gamma}^{*} = \arg\max_{\boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})} S(\boldsymbol{\gamma})$;

   iv). **If** $S(\boldsymbol{\gamma}^{*}) > S(\hat{\boldsymbol{\gamma}})$, **then** update $\hat{\boldsymbol{\gamma}} \leftarrow \boldsymbol{\gamma}^{*}$ and go back to Step 2 immediately,
   else update $\hat{\boldsymbol{\gamma}}^{(t+1)}$ by selecting one model from $\text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})$ with probability

$$\frac{S(\boldsymbol{\gamma})}{\sum_{\tilde{\boldsymbol{\gamma}} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)})} S(\tilde{\boldsymbol{\gamma}})}, \boldsymbol{\gamma} \in \text{nbd}(\hat{\boldsymbol{\gamma}}^{(t)});$$

5. Return $\hat{\boldsymbol{\gamma}}$.

4.3 Proposed method

4.3.1 Simultaneous evaluation for models in the $\text{nbd}_{+}(\hat{\boldsymbol{\gamma}})$

Given the current model $\hat{\boldsymbol{\gamma}}$, let $\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}$ be the MAP estimate of $\boldsymbol{\theta}_{\hat{\boldsymbol{\gamma}}}$ obtained by (4.2). For $j \notin \hat{\boldsymbol{\gamma}}$, define a model in the addition neighbor, $\hat{\boldsymbol{\gamma}}^{+j} = \hat{\boldsymbol{\gamma}} \cup \{j\} \in \text{nbd}_{+}(\hat{\boldsymbol{\gamma}})$ and rearrange its corresponding coefficient vector $\boldsymbol{\theta}_{\hat{\boldsymbol{\gamma}}^{+j}} = \{\boldsymbol{\theta}_{\hat{\boldsymbol{\gamma}}}, \theta_j\}$, where θ_j is the coefficient corresponding to j-th predictor $\tilde{\mathbf{x}}_j$. Let

$$\tilde{\theta}_j = \arg\max_{\theta_j} f(\mathbf{y}|\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}, \hat{\boldsymbol{\gamma}}^{+j})\pi(\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}|\hat{\boldsymbol{\gamma}}^{+j}). \tag{4.5}$$

Then, we have that

$$\begin{aligned} S(\hat{\boldsymbol{\gamma}}^{+j}) &= (2\pi n^{-1})^{|\hat{\gamma}^{+j}|/2} f(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\hat{\gamma}^{+j}}, \hat{\boldsymbol{\gamma}}^{+j}) \pi(\hat{\boldsymbol{\theta}}_{\hat{\gamma}^{+j}}|\hat{\boldsymbol{\gamma}}^{+j}) \pi(\hat{\boldsymbol{\gamma}}^{+j}) \\ &\geq (2\pi n^{-1})^{\frac{|\hat{\gamma}|+1}{2}} f(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\hat{\gamma}}, \tilde{\theta}_j, \hat{\boldsymbol{\gamma}}^{+j}) \pi(\hat{\boldsymbol{\theta}}_{\hat{\gamma}}, \tilde{\theta}_j|\hat{\boldsymbol{\gamma}}^{+j}) \pi(\hat{\boldsymbol{\gamma}}^{+j}) \qquad (4.6) \\ &\equiv \tilde{S}(\hat{\boldsymbol{\gamma}}^{+j}), \end{aligned}$$

where $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}^{+j}}$ is the MAP estimate of $\boldsymbol{\theta}_{\boldsymbol{\gamma}^{+j}}$. The inequality in (4.6) holds because $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}^{+j}}$ is the MAP estimate, while $(\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}, \tilde{\theta}_j)$ is not. The following theorem shows an important property related to (4.6).

Theorem 3. *$S(\boldsymbol{\gamma})$ is defined by (4.3) where $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}$ is evaluated by (4.2). $\tilde{S}(\boldsymbol{\gamma})$ is an approximate estimate of $S(\boldsymbol{\gamma})$. Define $\boldsymbol{\gamma}_1$ and $\boldsymbol{\gamma}_2$ are two different models. If $\tilde{S}(\boldsymbol{\gamma}_1) > S(\boldsymbol{\gamma}_2)$, then $S(\boldsymbol{\gamma}_1) > S(\boldsymbol{\gamma}_2)$.*

Proof of Theorem 3. Since $S(\boldsymbol{\gamma})$ is evaluated by (4.2), we have $S(\boldsymbol{\gamma}_1) \geq \tilde{S}(\boldsymbol{\gamma}_1)$. Therefore, if $\tilde{S}(\boldsymbol{\gamma}_1) > S(\boldsymbol{\gamma}_2)$, then $S(\boldsymbol{\gamma}_1) \geq \tilde{S}(\boldsymbol{\gamma}_1) > S(\boldsymbol{\gamma}_2)$. This completes our proof. □

By Theorem 3, given two models $\boldsymbol{\gamma}^{+j}$ and $\boldsymbol{\gamma}^{+j_0}$ with $j \neq j_0$, if $\tilde{S}(\hat{\boldsymbol{\gamma}}^{+j}) > S(\hat{\boldsymbol{\gamma}}^{+j_0})$ holds, then we have $S(\hat{\boldsymbol{\gamma}}^{+j}) > S(\hat{\boldsymbol{\gamma}}^{+j_0})$, which implies that we prefer model $\hat{\boldsymbol{\gamma}}^{+j}$ and that $\tilde{S}(\hat{\boldsymbol{\gamma}}^{+j})$ can be a proxy of $S(\hat{\boldsymbol{\gamma}}^{+j})$. In other words, the difference between $S(\cdot)$ and $\tilde{S}(\cdot)$ can be safely ignored for the purpose of variable selection. Compared with $S(\hat{\boldsymbol{\gamma}}^{+j})$, which computes the MAP estimate of a $(k+1)$-dimensional vector $\boldsymbol{\theta}_{\boldsymbol{\gamma}^{+j}}$, the maximization of $\tilde{S}(\hat{\boldsymbol{\gamma}}^{+j})$ is much faster because it requires only the one-dimensional optimization in (4.5). In addition, we propose a novel strategy that allows us to evaluate $\tilde{S}(\hat{\boldsymbol{\gamma}}^{+j})$ for all $j \notin \hat{\boldsymbol{\gamma}}$ simultaneously in a single computation.

The method to estimate $\tilde{S}(\hat{\boldsymbol{\gamma}}^{+j})$ for all $j \notin \hat{\boldsymbol{\gamma}}$ simultaneously: Before simultaneous evaluations of $\tilde{S}(\hat{\boldsymbol{\gamma}}^{+j})$ for all $j \notin \hat{\boldsymbol{\gamma}}$, we need to estimate $\tilde{\theta}_j$ in (4.5). By taking the logarithm of $f(\mathbf{y}|\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\gamma}}, \hat{\boldsymbol{\gamma}}^{+j}) \pi(\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\gamma}}|\hat{\boldsymbol{\gamma}}^{+j})$ and write the resulting equation with respect to

θ_j, we obtain

$$\begin{aligned} l(\theta_j) &= \log \prod_{i=1}^{n} f(y_i|\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}, \hat{\boldsymbol{\gamma}}^{+j}) \pi(\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}} | \hat{\boldsymbol{\gamma}}^{+j}) \\ &= \sum_{i=1}^{n} \log f(y_i|\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}, \hat{\boldsymbol{\gamma}}^{+j}) + \log(\pi(\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}} | \hat{\boldsymbol{\gamma}}^{+j})). \end{aligned} \tag{4.7}$$

Then, taking the first derivative of $l(\theta_j)$ with respective to θ_j leads to

$$u_j(\theta_j) = \frac{\partial l(\theta_j)}{\partial \theta_j} = \sum_{i=1}^{n} \frac{\partial}{\partial \theta_j} \log f(y_i|\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}, \hat{\boldsymbol{\gamma}}^{+j}) + \frac{\partial \log(\pi(\theta_j, \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}} | \hat{\boldsymbol{\gamma}}^{+j}))}{\partial \theta_j}. \tag{4.8}$$

Thus, for all $j \notin \hat{\boldsymbol{\gamma}}$, finding the maximizer of (4.5) can be done by solving $u_j(\theta_j) = 0$. Without loss of generality, we assume the current model $\hat{\boldsymbol{\gamma}} = \{1, 2, \ldots, k\}$. Therefore, for all $j \notin \hat{\boldsymbol{\gamma}}$, $\tilde{\theta}_j$ can be obtained by solving the following system of equations:

$$u_{k+1}(\theta_{k+1}) = 0, \quad u_{k+2}(\theta_{k+2}) = 0, \quad \ldots, \quad u_p(\theta_p) = 0. \tag{4.9}$$

To solve (4.9), we can use the Newton-Raphson method, which repeats the following update until convergence,

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \mathbf{J}_{\mathbf{u}}(\boldsymbol{\theta}^{(t)})^{-1} \mathbf{u}(\boldsymbol{\theta}^{(t)}), \tag{4.10}$$

where $\mathbf{u}(\boldsymbol{\theta}^{(t)}) = (u_{k+1}(\theta_{k+1}^{(t)}), u_{k+2}(\theta_{k+2}^{(t)}), \ldots, u_p(\theta_p^{(t)})$ is a $(p-k)$-dimensional vector and $J_{\mathbf{u}}(\boldsymbol{\theta}^{(t)}) = \{J_j(\theta_l)\}_{j,l \in \{k+1, k+2, \ldots, p\}}$ with $J_j(\theta_l) = \frac{\partial u_j(\theta_j^{(t)})}{\partial \theta_l^{(t)}}$ is a $(p-k) \times (p-k)$ Jacobian matrix. Since θ_j is the only unknown parameter in $u_j(\theta_j^{(t)})$, we have $J_j(\theta_l) = \frac{\partial u_j(\theta_j^{(t)})}{\partial \theta_l^{(t)}} = 0$ for all $j \neq l$. Thus, $J_{\mathbf{u}}(\boldsymbol{\theta}^{(t)})$ is a diagonal matrix. This implies that $\mathbf{J}_{\mathbf{u}}(\boldsymbol{\theta}^{(t)})^{-1}$ can be easily estimated because the inverse of a diagonal matrix is obtained by replacing each diagonal element with its reciprocal. After solving (4.10), we obtain $\tilde{\theta}_j$ for all $j \notin \hat{\boldsymbol{\gamma}}$ simultaneously. Let $\tilde{\boldsymbol{\Theta}}_{\text{nbd}_+(\hat{\boldsymbol{\gamma}})} = \{(\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}, \tilde{\theta}_j) : j \notin \hat{\boldsymbol{\gamma}}\}$ be the set containing $\{\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}, \tilde{\theta}_j\}$ for all $j \notin \hat{\boldsymbol{\gamma}}$. Then applying

each element in $\tilde{\boldsymbol{\Theta}}_{\text{nbd}_+(\hat{\boldsymbol{\gamma}})}$ to (4.6), we can obtain a set of $\tilde{S}(\boldsymbol{\gamma})$ values in $\text{nbd}_+(\hat{\boldsymbol{\gamma}})$,

$$\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}})\}. \tag{4.11}$$

Therefore, the best model in $\text{nbd}_+(\hat{\boldsymbol{\gamma}})$ is $\hat{\boldsymbol{\gamma}}^+ = \arg\max_{\boldsymbol{\gamma}\in\text{nbd}_+(\hat{\boldsymbol{\gamma}})} \tilde{S}(\boldsymbol{\gamma})$. In what follows, we will discuss our strategy to simultaneously evaluate all models in $\text{nbd}_-(\hat{\boldsymbol{\gamma}})$.

4.3.2 Simultaneous evaluation for models in the $\text{nbd}_-(\hat{\boldsymbol{\gamma}})$

For each $j \in \hat{\boldsymbol{\gamma}}$, denote $\hat{\theta}_j$ is an element in the MAP estimate $\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}$. By removing $\hat{\theta}_j$ in $\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}$, we define

$$\tilde{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}} = \hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}} \setminus \hat{\theta}_j, \tag{4.12}$$

where $\hat{\boldsymbol{\gamma}}^{-j} = \hat{\boldsymbol{\gamma}} \setminus \{j\}$ is a model in $\text{nbd}_-(\hat{\boldsymbol{\gamma}})$ and $\tilde{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}$ is a subset of $\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}}$ without $\hat{\theta}_j$. Then we have that

$$\begin{aligned} S(\hat{\boldsymbol{\gamma}}^{-j}) &= (2\pi n^{-1})^{|\hat{\boldsymbol{\gamma}}^{-j}|/2} f(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}, \hat{\boldsymbol{\gamma}}^{-j})\pi(\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}|\hat{\boldsymbol{\gamma}}^{-j})\pi(\hat{\boldsymbol{\gamma}}^{-j}) \\ &\geq (2\pi n^{-1})^{\frac{|\hat{\boldsymbol{\gamma}}|-1}{2}} f(\mathbf{y}|\tilde{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}, \hat{\boldsymbol{\gamma}}^{-j})\pi(\tilde{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}|\hat{\boldsymbol{\gamma}}^{-j})\pi(\hat{\boldsymbol{\gamma}}^{-j}), \\ &\equiv \tilde{S}(\hat{\boldsymbol{\gamma}}^{-j}), \end{aligned} \tag{4.13}$$

where $\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}$ is the MAP estimate of $\boldsymbol{\theta}_{\hat{\boldsymbol{\gamma}}^{-j}}$ obtained by (4.2). Similarly, the inequality in (4.13) holds because $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}^{-j}}$ is the MAP estimate whereas $\tilde{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}$ is not. By Theorem 3, given two models, $\boldsymbol{\gamma}^{-j}$ and $\boldsymbol{\gamma}^{-j_0}$, if $\tilde{S}(\hat{\boldsymbol{\gamma}}^{-j}) > S(\hat{\boldsymbol{\gamma}}^{-j_0})$ holds, then we have $S(\hat{\boldsymbol{\gamma}}^{-j}) > S(\hat{\boldsymbol{\gamma}}^{-j_0})$, which implies we prefer model $\hat{\boldsymbol{\gamma}}^{-j}$ and $\tilde{S}(\hat{\boldsymbol{\gamma}}^{-j})$ can be a proxy of $S(\hat{\boldsymbol{\gamma}}^{-j})$. Hence, the difference between $S(\hat{\boldsymbol{\gamma}}^{-j})$ and $\tilde{S}(\hat{\boldsymbol{\gamma}}^{-j})$ can be safely ignored for the purpose of variable selection. Note that computing $\tilde{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}$ in (4.12) can be done by simply deleting a single element of $\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}$, which is much faster and more efficient than computing the MAP estimate of the $(k-1)$-dimensional vector $\hat{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}$. Define the set $\tilde{\boldsymbol{\Theta}}_{\text{nbd}_-(\hat{\boldsymbol{\gamma}})} = \{\tilde{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}} : j \in \hat{\boldsymbol{\gamma}}\}$ contains $\tilde{\boldsymbol{\theta}}_{\hat{\boldsymbol{\gamma}}^{-j}}$ for all $j \in \hat{\boldsymbol{\gamma}}$. Applying each element in $\tilde{\boldsymbol{\Theta}}_{\text{nbd}_-(\hat{\boldsymbol{\gamma}})}$ to (4.13) for all $j \in \hat{\boldsymbol{\gamma}}$, we can obtain a set of $\tilde{S}(\boldsymbol{\gamma})$

values in $\text{nbd}_-(\hat{\boldsymbol{\gamma}})$,

$$\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_-(\hat{\boldsymbol{\gamma}})\}.$$

Therefore, the best model in $\text{nbd}_-(\hat{\boldsymbol{\gamma}})$ is $\hat{\boldsymbol{\gamma}}^- = \arg\max_{\boldsymbol{\gamma}\in\text{nbd}_-(\hat{\boldsymbol{\gamma}})} \tilde{S}(\boldsymbol{\gamma})$.

4.3.3 The proposed algorithm

Combined with our simultaneous computation strategy, we introduce a new fast algorithm, described in Algorithm 10. The key idea of Algorithm 10 is replace $S(\boldsymbol{\gamma})$ with $\tilde{S}(\boldsymbol{\gamma})$ in Algorithm 9 to implement simultaneous computations. Hence, $\tilde{S}(\boldsymbol{\gamma})$ in the addition and deletion neighbors of $\hat{\boldsymbol{\gamma}}^{(t)}$ are firstly computed in both stages of deterministic and stochastic search. In Step 2-ii), two best models, $\hat{\boldsymbol{\gamma}}^+$ in the addition neighbor and $\hat{\boldsymbol{\gamma}}^-$ in the deletion neighbor, are chosen separately. In Step 2-iii), from $\{\hat{\boldsymbol{\gamma}}^+, \hat{\boldsymbol{\gamma}}^-\}$, we select a model with larger value of the exact objective function $S(\boldsymbol{\gamma})$ as the next update. In Step 5-ii) of stochastic search, we select $\hat{\boldsymbol{\gamma}}^+$ by randomly sampling from $\text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t)})$ with probability proportional to $\tilde{S}(\boldsymbol{\gamma})$ normalized within neighborhood space $\text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t)})$. Using the similar way, we obtain $\hat{\boldsymbol{\gamma}}^-$ in Step 5-iii). In Step 5-v), if we observe $S(\hat{\boldsymbol{\gamma}}^{(t+1)}) > S(\hat{\boldsymbol{\gamma}})$, then we update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t+1)}$ and go back to Step 2 immediately; otherwise, we randomly sample the next update from $\{\hat{\boldsymbol{\gamma}}^+, \hat{\boldsymbol{\gamma}}^-\}$ with probability proportional to $S(\boldsymbol{\gamma})$ normalized within $\{\hat{\boldsymbol{\gamma}}^+, \hat{\boldsymbol{\gamma}}^-\}$.

4.4 Examples

In this section, we show some examples to illustrate how we apply the proposed method to Gaussian, binary, count and time-to-event (survival) data. Since computing $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_-(\hat{\boldsymbol{\gamma}})\}$ is very straightforward, this section focuses on explaining the key steps of evaluation of $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}})\}$ and omits that of $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_-(\hat{\boldsymbol{\gamma}})\}$. We summarize three steps to estimate $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}})\}$.

- Step i). Firstly, derive three key functions for each $j \notin \hat{\boldsymbol{\gamma}}$: $l(\theta_j)$ in (4.7), $u_j(\theta_j)$ in (4.8), and $J_j(\theta_j)$ in (4.9).

Algorithm 10 The proposed approach using hybrid search

1. Set $\hat{\boldsymbol{\gamma}}$ as the current model and $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$.
2. **Repeat** $t = 0, 1, 2, \ldots,$ `# deterministic search`
   i). Compute $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t)})\}$ and $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_-(\hat{\boldsymbol{\gamma}}^{(t)})\}$.
   ii). Compute $\hat{\boldsymbol{\gamma}}^{+} = \arg\max_{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t)})} \tilde{S}(\boldsymbol{\gamma})$ and $\hat{\boldsymbol{\gamma}}^{-} = \arg\max_{\boldsymbol{\gamma} \in \text{nbd}_-(\hat{\boldsymbol{\gamma}}^{(t)})} \tilde{S}(\boldsymbol{\gamma})$.
   iii). Compute $\hat{\boldsymbol{\gamma}}^{(t+1)} = \arg\max_{\boldsymbol{\gamma} \in \{\hat{\boldsymbol{\gamma}}^{+}, \hat{\boldsymbol{\gamma}}^{-}\}} S(\boldsymbol{\gamma})$ by (4.3).
   iv). **If** $S(\hat{\boldsymbol{\gamma}}^{(t+1)}) > S(\hat{\boldsymbol{\gamma}})$, **then** update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t+1)}$, **else** update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t)}$ and break the loop.
3. Return $\hat{\boldsymbol{\gamma}}$.
4. Set $\hat{\boldsymbol{\gamma}}^{(0)} = \hat{\boldsymbol{\gamma}}$.
5. **Repeat** for $t = 0, 1, 2, \ldots, T$ `# stochastic search`
   i). Compute $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t)})\}$ and $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_-(\hat{\boldsymbol{\gamma}}^{(t)})\}$.
   ii). Update $\hat{\boldsymbol{\gamma}}^{+}$ by selecting one model from $\text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t)})$ with probability
   $$\frac{\tilde{S}(\boldsymbol{\gamma})}{\sum_{\boldsymbol{\gamma}' \in \text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t)})} \tilde{S}(\boldsymbol{\gamma}')}, \boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}}^{(t)});$$
   iii). Update $\hat{\boldsymbol{\gamma}}^{-}$ by selecting one model from $\text{nbd}_-(\hat{\boldsymbol{\gamma}}^{(t)})$ with probability
   $$\frac{\tilde{S}(\boldsymbol{\gamma})}{\sum_{\boldsymbol{\gamma}' \in \text{nbd}_-(\hat{\boldsymbol{\gamma}}^{(t)})} \tilde{S}(\boldsymbol{\gamma}')}, \boldsymbol{\gamma} \in \text{nbd}_-(\hat{\boldsymbol{\gamma}}^{(t)});$$
   iv). Compute $\hat{\boldsymbol{\gamma}}^{(t+1)} = \arg\max_{\boldsymbol{\gamma} \in \{\hat{\boldsymbol{\gamma}}^{+}, \hat{\boldsymbol{\gamma}}^{-}\}} S(\boldsymbol{\gamma})$ by (4.3).
   v). **If** $S(\hat{\boldsymbol{\gamma}}^{(t+1)}) > S(\hat{\boldsymbol{\gamma}})$, **then** update $\hat{\boldsymbol{\gamma}} \leftarrow \hat{\boldsymbol{\gamma}}^{(t+1)}$ and go back to Step 2 immediately, **else** update $\hat{\boldsymbol{\gamma}}^{(t+1)}$ by selecting one model from $\{\hat{\boldsymbol{\gamma}}^{+}, \hat{\boldsymbol{\gamma}}^{-}\}$ with probability
   $$\frac{S(\boldsymbol{\gamma})}{\sum_{\boldsymbol{\gamma}' \in \{\hat{\boldsymbol{\gamma}}^{+}, \hat{\boldsymbol{\gamma}}^{-}\}} S(\boldsymbol{\gamma}')}, \boldsymbol{\gamma} \in \{\hat{\boldsymbol{\gamma}}^{+}, \hat{\boldsymbol{\gamma}}^{-}\};$$
6. Return $\hat{\boldsymbol{\gamma}}$.

- Step ii). Secondly, plugging each $u_j(\theta_j)$ and $J_j(\theta_j)$ into (4.10) and repeating it until convergence, we can obtain $\tilde{\boldsymbol{\Theta}}_{\text{nbd}_+(\hat{\boldsymbol{\gamma}})}$ simultaneously.
- Step iii). Finally, applying each element in $\tilde{\boldsymbol{\Theta}}_{\text{nbd}_+(\hat{\boldsymbol{\gamma}})}$ to (4.6), we obtain $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}})\}$.

In addition, an R demo code with detailed comments is available at https://calebsjin.github.io/RdemoMVRM/chapter4.html.

4.4.1 Gaussian data

Suppose y_i are independently generated from normal distribution. The Bayesian hierarchical model given the model $\boldsymbol{\gamma}$ is

$$\begin{aligned} f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma}) &\sim \mathcal{N}(\mathbf{X}_{\boldsymbol{\gamma}}\boldsymbol{\beta}_{\boldsymbol{\gamma}}, \exp(\kappa)\mathbf{I}_n), \\ \pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) &\sim \mathcal{N}(\mathbf{0}, \lambda\mathbf{I}_{|\boldsymbol{\gamma}|+1}), \end{aligned}$$

where $\boldsymbol{\theta}_{\boldsymbol{\gamma}} = (\kappa, \boldsymbol{\beta}_{\boldsymbol{\gamma}})$ and λ is a hyperparameter for the prior $\boldsymbol{\beta}_{\boldsymbol{\gamma}}$. Hence, we have

$$\log f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) = C - \frac{|\boldsymbol{\gamma}|+1}{2}\log(2\pi\lambda) - \frac{n}{2}\kappa - \frac{||\mathbf{y} - \mathbf{X}_{\boldsymbol{\gamma}}\boldsymbol{\beta}_{\boldsymbol{\gamma}}||^2}{2\exp(\kappa)} - \frac{1}{2\lambda}(\kappa^2 + \boldsymbol{\beta}_{\boldsymbol{\gamma}}^\top\boldsymbol{\beta}_{\boldsymbol{\gamma}}),$$

where C is a constant with respect to $\boldsymbol{\theta}_{\boldsymbol{\gamma}}$. It is easy to show that for $j \notin \boldsymbol{\gamma}$,

$$\begin{aligned} l(\beta_j) &= \frac{e^{-\hat{\kappa}}}{2}\left[(\mathbf{x}_j^\top\mathbf{x}_j + \lambda^{-1}e^{\hat{\kappa}})\beta_j^2 - 2(\mathbf{y} - \mathbf{X}_{\boldsymbol{\gamma}}\hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}})^\top\mathbf{x}_j\beta_j + ||\mathbf{y} - \mathbf{X}_{\boldsymbol{\gamma}}\hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}}||^2\right] \\ &\quad -\frac{|\boldsymbol{\gamma}|+1}{2}\log(2\pi\lambda) - \frac{n}{2}\hat{\kappa} - \frac{1}{2\lambda}(\hat{\kappa}^2 + \boldsymbol{\beta}_{\boldsymbol{\gamma}}^\top\boldsymbol{\beta}_{\boldsymbol{\gamma}}) + C, \\ u(\beta_j) &= -e^{-\hat{\kappa}}\left(\beta_j(\mathbf{x}_j^\top\mathbf{x}_j + \lambda^{-1}e^{\hat{\kappa}}) - (\mathbf{y} - \mathbf{X}_{\boldsymbol{\gamma}}\hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}})^\top\mathbf{x}_j\right), \end{aligned}$$

where C is a constant with respect to β_j and $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}} = (\hat{\kappa}, \hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}}) = \arg\max_{\boldsymbol{\theta}_{\boldsymbol{\gamma}}} f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma})$. Since the analytic solution of $u(\beta_j) = 0$ exists, Newton-Raphson method is not necessary in this case. Then we have that

$$\tilde{\beta}_j = \frac{\mathbf{x}_j^\top\mathbf{y} - \mathbf{x}_j^\top\mathbf{X}_{\boldsymbol{\gamma}}\hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}}}{\mathbf{x}_j^\top\mathbf{x}_j + \lambda^{-1}e^{\hat{\kappa}}}.$$

Define $-\boldsymbol{\gamma} = \{j : j \notin \boldsymbol{\gamma}\}$. Hence, $\tilde{\boldsymbol{\Theta}}_{\mathrm{nbd}_+(\boldsymbol{\gamma})}$ can be simultaneously computed by the following vectorized formula:

$$\tilde{\boldsymbol{\Theta}}_{\mathrm{nbd}_+(\boldsymbol{\gamma})} = \frac{\mathbf{X}_{-\boldsymbol{\gamma}}^\top \mathbf{y} - \mathbf{X}_{-\boldsymbol{\gamma}}^\top \mathbf{X}_{\boldsymbol{\gamma}} \hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}}}{\mathrm{diag}(\mathbf{X}_{-\boldsymbol{\gamma}}^\top \mathbf{X}_{-\boldsymbol{\gamma}}) + \lambda^{-1} e^{\hat{\kappa}} \mathbf{1}_{p-|\boldsymbol{\gamma}|}},$$

where $\mathbf{a}/\mathbf{b} = (a_1/b_1, \ldots, a_p/b_p)$ for generic vectors $\mathbf{a}$ and $\mathbf{b}$. By implementing Step ii) and Step iii) in Section 4.4, we can obtain $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \mathrm{nbd}_+(\hat{\boldsymbol{\gamma}})\}$ for linear regression model.

4.4.2 Binary data

Suppose y_i are independent binary outcomes from Bernoulli distribution. To fit the binary regression models, there are many link functions available, such as the logit link, probit link, canonical link, and complementary log-log link. For example, we choose to build a Bayesian logistic regression model with the logit link, then the likelihood function of model $\boldsymbol{\gamma}$ is $f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma}) = \prod_{i=1}^n p_i^{y_i}(1-p_i)^{(1-y_i)}$, where $p_i = \frac{\exp(\mathbf{x}_{i\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})}{1+\exp(\mathbf{x}_{i\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})}$. Suppose that the prior for $\theta_{\boldsymbol{\gamma}}$ is $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) \sim \mathcal{N}(\mathbf{0}, \lambda \mathbf{I}_{|\boldsymbol{\gamma}|})$ with the hyperparameter λ. Then the posterior probability of model $\boldsymbol{\gamma}$, $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\mathbf{y}, \boldsymbol{\gamma})$, can be given as

$$\begin{aligned} f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) &= \prod_{i=1}^n \left[\frac{\exp(\mathbf{x}_{i\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})}{1+\exp(\mathbf{x}_{i\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})}\right]^{y_i} \left[\frac{1}{1+\exp(\mathbf{x}_{i\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})}\right]^{1-y_i} \\ &\quad \times (2\pi\lambda)^{-\frac{k}{2}} \exp\left\{-\frac{1}{2\lambda}\boldsymbol{\theta}_{\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}}\right\}. \end{aligned}$$

To estimate $\tilde{\theta}_j$ for $j \notin \boldsymbol{\gamma}$, we have the explicit forms of $l(\theta_j)$, $u_j(\theta_j)$ and each diagonal element of $\mathbf{J}_{\mathbf{u}}$ as follows:

$$\begin{aligned} l(\theta_j) &= \sum_{i=1}^n \left[y_i\psi_i(\theta_j) - \log(1+\exp(\psi_i(\theta_j)))\right] - \frac{1}{2\lambda}(\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}^\top \hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}} + \theta_j^2) - \frac{k+1}{2}\log(2\pi\lambda) + C, \\ u_j(\theta_j) &= \sum_{i=1}^n x_{ij}\left(y_i - \frac{1}{1+\exp(-\psi_i(\theta_j))}\right) - \frac{\theta_j}{\lambda}, \\ J_j(\theta_j) &= \frac{\partial u_j(\theta_j)}{\partial \theta_j} = -\sum_{i=1}^n \frac{x_{ij}^2 \exp(-\psi_i(\theta_j))}{[1+\exp(-\psi_i(\theta_j))]^2} - \frac{1}{\lambda}, \end{aligned}$$

where C is a constant with respect to θ_j and $\psi_i(\theta_j) = x_{ij}\theta_j + \mathbf{x}_{i\boldsymbol{\gamma}}^\top \hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}$ with the MAP estimate $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}$ obtained by (4.2). By implementing Step ii) and Step iii) in Section 4.4, we can obtain $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}})\}$ for logistic regression model.

4.4.3 Count data

Suppose $\mathbf{y}$ are count data and independently distributed from Poisson distribution. Using the logarithm link, we get the likelihood function $f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma}) = \prod_{i=1}^n e^{-\zeta_i}\zeta_i^{y_i}/y_i!$, where $\zeta_i = \exp(\mathbf{x}_{i\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})$. Let the prior distribution of $\boldsymbol{\theta}_{\boldsymbol{\gamma}}$ be $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) \sim \mathcal{N}(\mathbf{0}, \lambda \mathbf{I}_{|\boldsymbol{\gamma}|})$ with the hyperparameter λ . Hence, the posterior probability of model $\boldsymbol{\gamma}$, $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\mathbf{y}, \boldsymbol{\gamma})$, can be proportional to

$$\begin{aligned} f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) &= \left\{\prod_{i=1}^n \exp(y_i \mathbf{x}_{i\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}}) \exp(-\exp(\mathbf{x}_{i\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}}))/y_i!\right\} \\ &\quad \times (2\pi\lambda)^{-\frac{k}{2}} \exp\left(-\frac{1}{2\lambda}\boldsymbol{\theta}_{\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}}\right). \end{aligned}$$

To estimate $\tilde{\theta}_j$ for $j \notin \boldsymbol{\gamma}$, we have the explicit forms of $l(\theta_j)$, $u_j(\theta_j)$ and each diagonal element of $\mathbf{J}_{\mathbf{u}}$ as follows:

$$\begin{aligned} l(\theta_j) &= \sum_{i=1}^n \Big[y_i\psi_i(\theta_j) - \exp(\psi_i(\theta_j))\Big] - \frac{1}{2\lambda}(\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}^\top \hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}} + \theta_j^2) - \frac{k+1}{2}\log(2\pi\lambda) + C, \\ u_j(\theta_j) &= \sum_{i=1}^n \Big[(y_i - \exp(\psi_i(\theta_j)))x_{ij}\Big] - \frac{\theta_j}{\lambda}, \\ J_j(\theta_j) &= \frac{\partial u_j(\theta_j)}{\partial \theta_j} = -\sum_{i=1}^n x_{ij}^2 \exp(\psi_i(\theta_j)) - \frac{1}{\lambda}, \end{aligned}$$

where C is a constant with respect to θ_j and $\psi_i(\theta_j) = x_{ij}\theta_j + \mathbf{x}_{i\boldsymbol{\gamma}}^\top \hat{\boldsymbol{\beta}}_{\boldsymbol{\gamma}}$ with the MAP estimate $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}$ obtained by (4.2). By implementing Step ii) and Step iii) in Section 4.4, we can obtain $\tilde{S}(\boldsymbol{\gamma})\mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}})\}$ for Poisson regression model.

4.4.4 Time-to-event data

Let the random variable T_i denote time to the event of our interest, the random variable C_i be censored time, and Δ_i be the censored indicator. Suppose we have observed a censored survival data set from a sample of triplets $(y_i, \Delta_i, \mathbf{x}_i)$, $i = 1, 2, \ldots, n$, where

$$
\begin{aligned}
&y_i = \min\{T_i, C_i\},\\
&\Delta_i = \mathbb{I}(T_i \leq C_i),\\
&\mathbf{x}_i = (x_1, x_2, \ldots, x_p)^\top \text{ is a vector of covariates.}
\end{aligned}
$$

Then the hazard rate of the event at time t given $\mathbf{x}_i$ under the Cox proportional hazards model is

$$\lambda(t|\mathbf{x}_i) = \lambda_0(t)\exp(\mathbf{x}_i^\top \boldsymbol{\theta}),$$

where $\lambda_0(t)$ is the unknown baseline hazard rate and $\boldsymbol{\theta}$ is a p-dimensional vector of regression coefficients (Cox, 1972). Let $Z_i(u) = \mathbb{I}(y_i \geq u)$ be the indicator for i-th individual at risk at time u. The partial likelihood (Cox, 1975) is

$$\mathrm{PL}(\boldsymbol{\theta}) = \prod_{i=1}^{n} \left[\frac{\exp(\mathbf{x}_i^\top \boldsymbol{\theta})}{\sum_{l=1}^{n} \exp(\mathbf{x}_l^\top \boldsymbol{\theta}) Z_l(y_i)} \right]^{\delta_i},$$

where δ_i, the observed value of Δ_i, is 0 if y_i is a censoring time, and 1 otherwise. The log partial likelihood function of $\boldsymbol{\theta}$ is

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \delta_i \left[\mathbf{x}_i^\top \boldsymbol{\theta} - \log\left(\sum_{l=1}^{n} \exp(\mathbf{x}_l^\top \boldsymbol{\theta}) Z_l(y_i) \right) \right].$$

The score function is given as

$$U(\boldsymbol{\theta}) = \sum_{i=1}^{n} \delta_i \left[\mathbf{x}_i - \frac{\sum_{l=1}^{n} \mathbf{x}_l \exp(\mathbf{x}_l^\top \boldsymbol{\theta}) Z_l(y_i)}{\sum_{l=1}^{n} \exp(\mathbf{x}_l^\top \boldsymbol{\theta}) Z_l(y_i)} \right].$$

Assume that the prior distribution for $\boldsymbol{\theta}_{\boldsymbol{\gamma}}$ is $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) \sim \mathcal{N}(0, \lambda \mathbf{I}_{|\boldsymbol{\gamma}|})$. Thus, the posterior probability of model $\boldsymbol{\gamma}$, $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\mathbf{y}, \boldsymbol{\gamma})$, can be given as

$$PL(\boldsymbol{\theta}_{\boldsymbol{\gamma}})\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) = \prod_{i=1}^{n} \left[\frac{\exp(\mathbf{x}_i^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})}{\sum_{l=1}^{n} \exp(\mathbf{x}_l^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}}) Z_l(y_i)} \right]^{\delta_i} \times (2\pi\lambda)^{-\frac{k}{2}} \exp\left(-\frac{1}{2\lambda} \boldsymbol{\theta}_{\boldsymbol{\gamma}}^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}} \right).$$

To estimate $\tilde{\theta}_j$ for $j \notin \boldsymbol{\gamma}$, we derive the explicit forms of $l(\theta_j)$, $u_j(\theta_j)$ and each diagonal element of $\mathbf{J}_{\mathbf{u}}$ as follows:

$$\begin{aligned}
\ell(\theta_j) &= \sum_{i=1}^{n} \delta_i \left[\psi_i(\theta_j) - \log\left(\sum_{l=1}^{n} \exp(\psi_l(\theta_j)) Z_l(y_i) \right) \right] - \frac{1}{2\lambda}(\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}^\top \hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}} + \theta_j^2) \\
&\quad -\frac{k+1}{2}\log(2\pi\lambda) + C, \\
u_j(\theta_j) &= \sum_{i=1}^{n} \delta_i \left[z_{ij} - \frac{\sum_{l=1}^{n} z_{lj} \exp(\psi_l(\theta_j)) Z_l(y_i)}{\sum_{l=1}^{n} \exp(\psi_l(\theta_j)) Z_l(y_i)} \right] - \frac{\theta_j}{\lambda} \\
&= \sum_{i=1}^{n} \delta_i \left[z_{ij} - \sum_{l=1}^{n} z_{lj} w_l(y_i, \theta_j) \right] - \frac{\theta_j}{\lambda}, \\
J_j(\theta_j) &= \frac{\partial u_j(\theta_j)}{\partial \theta_j} = -\sum_{i=1}^{n} \delta_i \left[\sum_{l=1}^{n} z_{lj}^2 w_l(y_i, \theta_j) - \left(\sum_{l=1}^{n} z_{lj} w_l(y_i, \theta_j) \right)^2 \right] - \frac{1}{\lambda},
\end{aligned}$$

where C is a constant with respect to θ_j, $\psi_i(\theta_j) = x_{ij}\theta_j + \mathbf{x}_{i\boldsymbol{\gamma}}^\top \hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}$ with the MAP estimate $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}$ obtained by (4.2), and

$$w_l(y_i, \theta_j) = \frac{\exp(\psi_l(\theta_j)) Z_l(y_i)}{\sum_{l=1}^{n} \exp(\psi_l(\theta_j)) Z_l(y_i)}.$$

Using Step ii) and Step iii) in Section 4.4, we can obtain $\tilde{S}(\boldsymbol{\gamma}) \mathbb{I}\{\boldsymbol{\gamma} \in \text{nbd}_+(\hat{\boldsymbol{\gamma}})\}$ for Cox proportional hazards model.

4.5 Simulation study

In this section, we conduct multiple simulation studies to demonstrate the performance of the proposed approach for binary, count, and Gaussian data. We consider to use linear

regression model for fitting Gaussian data, Poisson regression model for count data, and logistic regression model for binary data.

Suppose we have n pairs of independent observations $\mathcal{D} = \{(\mathbf{x}_i, y_i), i = 1, 2, \ldots, n\}$ with predictors $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^\top$ and response y_i for $i = 1, 2, \ldots, n$. Let $\mathbf{y} = (y_1, y_2, \ldots, y_n)^\top$ be an $n \times 1$ response vector and $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$ be an $n \times p$ covariates matrix with $n < p$. We set $\pi(\boldsymbol{\gamma}) \propto 1/\binom{p}{|\boldsymbol{\gamma}|}$ as the prior distribution for the model $\boldsymbol{\gamma}$ according to Chen and Chen (2008) for all simulation studies. When generating data for each data type, we consider two cases with Case I, $n = 300$ and $p = 1000$, and Case II, $n = 600$ and $p = 2000$. The details of data generation process is shown below. Note that S1 represents Gaussian data scenario, S2 binary data, and S3 count data. For Case I:

S1. $y_i \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{x}_i^\top \boldsymbol{\beta}_{\boldsymbol{\gamma}}, \exp(\kappa))$, where $\boldsymbol{\gamma} = \{1, 3, 5, 7, 9\}$, $\boldsymbol{\beta}_{\boldsymbol{\gamma}} = (1, -1, 1, -1, 1)^\top$, $\kappa = \log 3$, and $\mathbf{x}_i \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{0}_p, \boldsymbol{\Sigma}_x)$ with $\boldsymbol{\Sigma}_x = (\Sigma_{ij})_{p\times p}$ and $\Sigma_{ij} = 0.6^{|i-j|}$.

S2. $y_i \overset{i.i.d}{\sim} \textit{Bernoulli}(p_i)$ with $p_i = g_2(\mathbf{x}_i^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})$, where $g_2(x) = \frac{1}{1+\exp(-x)}$, $\boldsymbol{\gamma} = \{1, 3, 5\}$, $\boldsymbol{\theta}_{\boldsymbol{\gamma}} = (0.8, 1 - 1)^\top$ and $\mathbf{x}_i \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{0}_p, \boldsymbol{\Sigma}_x)$ with $\boldsymbol{\Sigma}_x = (\Sigma_{ij})_{p\times p}$ and $\Sigma_{ij} = 0.6^{|i-j|}$.

S3. $y_i \overset{i.i.d}{\sim} \textit{Poisson}(\mu_i)$ with $\mu_i = g_3(\mathbf{x}_i^\top \boldsymbol{\theta}_{\boldsymbol{\gamma}})$, where $g_3(x) = \exp(x)$, $\boldsymbol{\gamma} = \{1, 3, 5\}$, $\boldsymbol{\theta}_{\boldsymbol{\gamma}} = (0.8, 1, -1)^\top, \mathbf{x}_i = \boldsymbol{\Phi}(\mathbf{z}_i) - 0.5\mathbf{1}_p$ with $\boldsymbol{\Phi}(\cdot)$ the CDF of standard normal distribution, and $\mathbf{z}_i \overset{i.i.d}{\sim} \mathcal{N}(\mathbf{0}_p, \boldsymbol{\Sigma}_z)$ with $\boldsymbol{\Sigma}_z = (\Sigma_{ij})_{p\times p}$ and $\Sigma_{ij} = 0.6^{|i-j|}$.

For S1 in Case II, we consider the same model setting as in S1 of Case I, but $\kappa = \log 6$. For S2 in Case II, we consider the same model setting as in S2 of Case I, but the true model is $\boldsymbol{\gamma} = \{1, 3, 5, 7, 9\}$ with its corresponding true coefficients $\boldsymbol{\theta}_{\boldsymbol{\gamma}} = (1, 0.9, -0.9, 0.9, -0.9)^\top$. For S3 in Case II, we consider the same model setting as in S3 of Case I, but the true model is $\boldsymbol{\gamma} = \{1, 3, 5, 7, 9\}$ with its corresponding true coefficients $\boldsymbol{\theta}_{\boldsymbol{\gamma}} = (0.8, 0.8, -0.8, 0.8, -0.8)^\top$.

Note that in S1, $\boldsymbol{\theta}_{\boldsymbol{\gamma}} = (\kappa, \boldsymbol{\beta}_{\boldsymbol{\gamma}})$ with the first element, κ, always selected in the process of variable selection. The initial model is set as the predictor that has the largest correlation with the response variable. In S2 and S3, we set their first predictors as $\mathbf{x}_1 = \mathbf{1}_n$, thus θ_1 is the intercept. The initial model is set as $\hat{\boldsymbol{\gamma}} = \{1\}$ and is always selected afterward. The prior distributions for $\boldsymbol{\theta}_{\boldsymbol{\gamma}}$ are set as $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) \sim \mathcal{N}(\mathbf{0}, 10^5\mathbf{I}_{|\boldsymbol{\gamma}|+1})$ for S1 and $\pi(\boldsymbol{\theta}_{\boldsymbol{\gamma}}|\boldsymbol{\gamma}) \sim \mathcal{N}(\mathbf{0}, 10^5\mathbf{I}_{|\boldsymbol{\gamma}|})$

for S2 and S3. For the number of MCMC iterations, we consider $T = 200$ for Algorithm 8 and $T = 100$ for Algorithm 9 and 10. Based on the model settings, we conduct two simulation studies to investigate the performance of computational speed and variable selection for the proposed method.

4.5.1 Simulation study I

To compare the computational speed of the proposed method (Algorithm 10) to Algorithm 7-9, we run a simulation study based on 100 Monte Carlo replications using a mediocre laptop with Intel® Core™ i5-7300HQ CPU @ 2.50GHz × 4 and 7.6 GB memory. Note that all the simulation studies is implemented by a single CPU core, although the laptop we use has 4 cores. The average time costs in minute for S1-S3 are recorded and illustrated in Figure 4.1. The common pattern is that the square sign (Algorithm 7) and solid dot (Algorithm 10) are partially or almost completely overlapped; the empty dot (Algorithm 8) is about twice as long as the triangle sign (Algorithm 9). For example, in S3 of Case II, the run time of proposed and deterministic search algorithms is very close and less than one minute, while the hybrid search spends about 22 minutes and stochastic search is about 43 minutes.

4.5.2 Simulation study II

Table 4.2: Measurements to evaluation model selection performance. Note that $\boldsymbol{\gamma}^*$ and $\hat{\boldsymbol{\gamma}}$ denotes the index set of the true model and selected model, respectively.

Abbreviation	Explanation	Formula
FPR	False positive rate	$\frac{\lvert\hat{\boldsymbol{\gamma}}\setminus\boldsymbol{\gamma}^*\rvert}{p-\lvert\boldsymbol{\gamma}^*\rvert}$
FNR	False negative rate	$\frac{\lvert\boldsymbol{\gamma}^*\setminus\hat{\boldsymbol{\gamma}}\rvert}{\lvert\boldsymbol{\gamma}^*\rvert}$
ACC	Accuracy	$\mathbb{I}(\hat{\boldsymbol{\gamma}} = \boldsymbol{\gamma}^*)$
SIZE	Size of $\hat{\boldsymbol{\gamma}}$	$\lvert\hat{\boldsymbol{\gamma}}\rvert$
HAM	Hamming distance	$\lvert\boldsymbol{\gamma}^* \setminus \hat{\boldsymbol{\gamma}}\rvert + \lvert\hat{\boldsymbol{\gamma}} \setminus \boldsymbol{\gamma}^*\rvert$

To evaluate the performance of variable selection, we select five measurements displayed

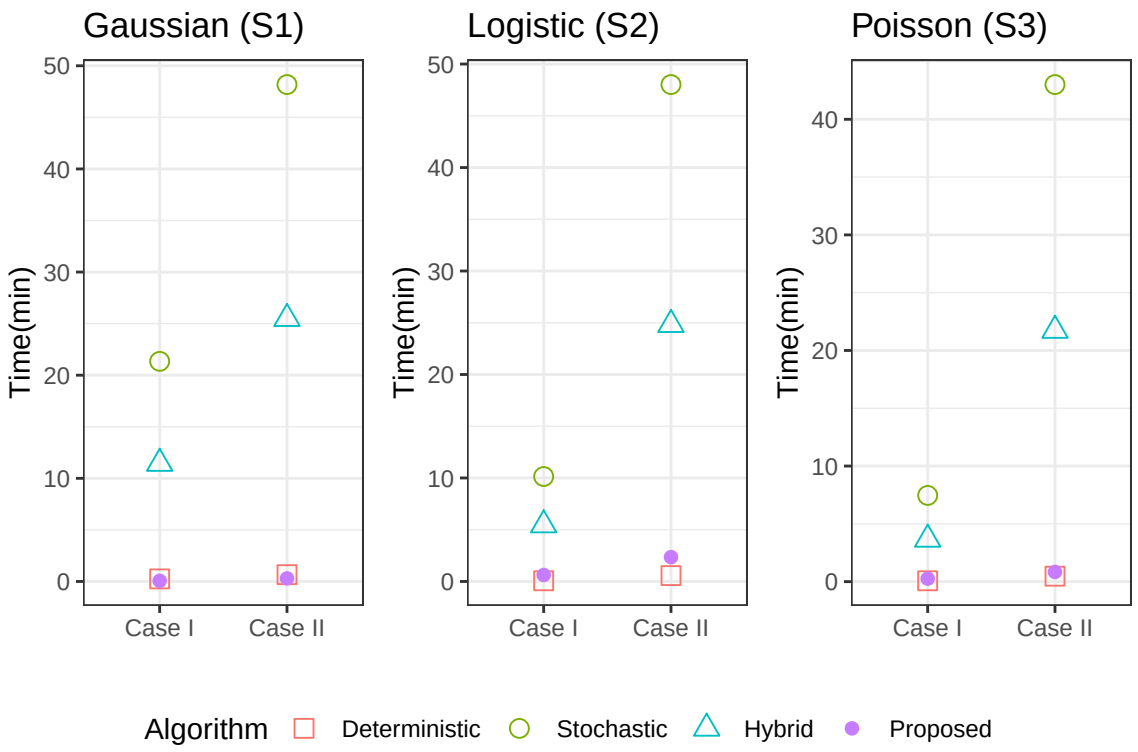

Figure 4.1: Comparison of computational speed in Algorithm 7-10.

in Table 4.2. Based on 500 Monte Carlo replications, the average of five measurements are computed and shown in Table 4.3 and 4.4. For example, in S2 of Case I, deterministic search has FPR% 0.001%, which means on average there are 0.001% covariates incorrectly detected as true ones among 500 replications. FNR% is about 31%, indicating averagely 31% true predictors are incorrectly diagnosed as false ones among 500 replications. For its ACC%, it means among 500 replications, 52% of them find the exactly true model. For its SIZE, the average size of selected model is about 2.1. For hamming distance, on average there is about one predictor incorrectly detected as a true or false predictor. In Table 4.3 and 4.4, in terms of FPR%, FNR%, ACC%, SIZE and HAM, it is apparent that the proposed method has almost the same performance with stochastic and hybrid search algorithms, which achieve the global optima, and is much better than deterministic search, which achieves the local optimum. For instance, in S2 of Case I, ACC% in stochastic, hybrid and proposed methods are about 91%, much higher than that of deterministic search, 52%. However, as is shown in second sub-figure of Figure 4.1, the average time cost of the proposed is much less than the stochastic

and the hybrid. The simulation studies obviously demonstrate that the proposed method achieves the same accuracy with Algorithm 8 and 9 but has much better computational efficiency.

Table 4.3: Simulation study for Case I $(n, p) = (300, 1000)$ based on 500 repliations. Notation: s.e.— standard error;

Method	FPR% (s.e)	FNR% (s.e)	ACC% (s.e)	SIZE (s.e)	HAM (s.e)
			Gaussian (S1)		
Deterministic	0.001(0.001)	10.72(1.128)	82.6(1.697)	4.478(0.057)	0.550(0.057)
Stochastic	0.001(0)	0(0)	98.8(0.487)	5.012(0.005)	0.012(0.005)
Hybrid	0.001(0)	0(0)	98.8(0.487)	5.012(0.005)	0.012(0.005)
Proposed	0.001(0)	0.24(0.138)	98.6(0.526)	4.998(0.007)	0.022(0.009)
			Logistic (S2)		
Deterministic	0.001(0)	30.933(1.470)	52.0(2.237)	2.080(0.044)	0.936(0.044)
Stochastic	0(0)	4.733(0.734)	91.4(1.255)	2.862(0.022)	0.146(0.022)
Hybrid	0(0)	4.733(0.734)	91.4(1.255)	2.862(0.022)	0.146(0.022)
Proposed	0.001(0)	4.800(0.736)	91.2(1.268)	2.862(0.022)	0.150(0.022)
			Poisson (S3)		
Deterministic	0(0)	20.333(1.346)	67.8(2.092)	2.390(0.040)	0.610(0.040)
Stochastic	0(0)	4.867(0.726)	91.0(1.281)	2.854(0.022)	0.146(0.022)
Hybrid	0(0)	4.867(0.726)	91.0(1.281)	2.854(0.022)	0.146(0.022)
Proposed	0(0)	4.867(0.726)	91.0(1.281)	2.854(0.022)	0.146(0.022)

Table 4.4: Simulation study for Case II $(n, p) = (600, 2000)$ based on 500 replications. Notation: s.e.— standard error;

Method	FPR% (s.e)	FNR% (s.e)	ACC% (s.e)	SIZE (s.e)	HAM (s.e)
			Gaussian (S1)		
Deterministic	0.001(0)	2.2(0.485)	94.8(0.994)	4.900(0.024)	0.120(0.025)
Stochastic	0.001(0)	0(0)	99.0(0.445)	5.010(0.004)	0.010(0.004)
Hybrid	0.001(0)	0(0)	98.6(0.526)	5.016(0.006)	0.016(0.006)
Proposed	0.001(0)	0(0)	99.0(0.445)	5.010(0.004)	0.010(0.004)
			Logistic (S2)		
Deterministic	0(0)	11.360(1.031)	78.2(1.848)	4.438(0.051)	0.574(0.052)
Stochastic	0(0)	0.800(0.287)	97.8(0.657)	4.964(0.015)	0.044(0.015)
Hybrid	0(0)	0.640(0.270)	98.4(0.562)	4.972(0.014)	0.036(0.014)
Proposed	0(0)	0.480(0.211)	98.4(0.562)	4.980(0.011)	0.028(0.011)
			Poisson (S3)		
Deterministic	0(0)	13.520(1.028)	72.0(2.010)	4.326(0.051)	0.678(0.051)
Stochastic	0(0)	0.120(0.069)	99.4(0.346)	4.994(0.003)	0.006(0.003)
Hybrid	0(0)	0.120(0.069)	99.4(0.346)	4.994(0.003)	0.006(0.003)
Proposed	0(0)	0.120(0.069)	99.4(0.346)	4.994(0.003)	0.006(0.003)

4.6 Real data analysis

In this section, we conduct three real examples for Gaussian, binary, and count data sets and fit them with linear, logistic and Poisson regression models, respectively. The model settings for each data sets are the same as we used in the simulation studies. To evaluate the performance of the proposed method, we compare it with four penalized regression approaches, LASSO, ENET, LASSO and SCAD. The goal in this real data analysis is to find the best fitting model among them. In order to make a fair comparison, the tuning parameters in penalized regression approaches are carefully selected to get the best performance. We consider two methods to select the best tuning parameters. In the first method, we use "cv.glmnet" and "cv.ncvreg" functions in the R packages of "glmnet" and "ncvreg" to select the tuning parameters with the minimal cross-validation errors. In the second method, the best tuning parameters are determined by minimal EBIC (Chen and Chen, 2008). To measure the performance of variable selection, we select a family of model selection criteria (FMSC), including the BIC, EBIC, modified BIC (MBIC) of Wang et al. (2009), corrected RIC (RICc) of Zhang (2010) and modified RIC (MRIC) of Foster and George (1994). The FMSC of model $\boldsymbol{\gamma}$ can be summarized by $\text{FMSC}_{\boldsymbol{\gamma}} = -2f(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma}) + |\hat{\boldsymbol{\gamma}}|\eta_{\boldsymbol{\gamma}}$, where $\hat{\boldsymbol{\theta}}_{\boldsymbol{\gamma}} = \arg\max_{\boldsymbol{\theta}_{\boldsymbol{\gamma}}} f(\mathbf{y}|\boldsymbol{\theta}_{\boldsymbol{\gamma}}, \boldsymbol{\gamma})$ and

$$\eta_{\boldsymbol{\gamma}} = \begin{cases} 2\log p & \text{mRIC} \\ 2(\log p + \log\log p) & \text{RICc} \\ \log n & \text{BIC} \\ \log n \log(\log p) & \text{MBIC} \\ \log n + 2\log\binom{p}{|\hat{\boldsymbol{\gamma}}|}/|\hat{\boldsymbol{\gamma}}| & \text{EBIC} \end{cases}.$$

All results of model selection performance evaluated by FMSC for three data sets are displayed in Table 4.5. Table 4.6 presents predictors selected by each method for three examples. In addition, based on variable selected by the proposed method in each example, we build Bayesian regression model and compute the median values of coefficients and 95%

credible intervals, together with their corresponding MLEs, shown in Table 4.7. When fitting Bayesian regression models, we use R package "rstanarm" (Goodrich et al., 2020) to obtain the MCMC samplings, and use "HPDinterval" function in "coda" package to compute median values and lower bound (LB) and upper bound (UB) of 95% credible interval based on the MCMC samplings.

4.6.1 Gaussian data: Ovarian carcinoma

For the example of Gaussian data, we use ovarian serous cystadenocarcinoma (OV) data set generated by The Cancer Genome Atlas (TCGA) research network: `http://cancergenome.nih.gov`. From the R package called "curatedTCGAData" (Ramos, 2019), we can download the OV data set, including 18631 gene expression measurements of 563 patients with primary solid tumors. Since BRCA1 germline mutations account for many genetic alterations associated with genetic ovarian cancer (Antoniou et al., 2003), we select BRCA1 as the response variable and remaining genes as covariates.

All the results are shown in Table 4.5, 4.6 and 4.7. In Table 4.5, the EBIC-method and CV-method represent two methods aforementioned to select the best tuning parameters with minimal cross-validation errors and EIBC values, respectively. NUM means the number of covariates are selected by each model. The proposed method selects five genes (TBL1Y, LOC146909, CCNB1, NBR2 and ATAD5 in Table 4.6), and its BIC (1074.7), EBIC (1141.2), MBIC (1107.3) RICc (1139.4), and MRIC (1119.1) are smallest compared with other methods. This demonstrates that the proposed model fits this Gaussian data best. In Table 4.6, we show the genes selected by EBIC-methods and neglect genes selected by CV-methods because they select too many genes, although they have minimal cross-validation errors. All EBIC-methods consistently select CIT, NBR2, ADAD5 and TOP2A genes. In Table 4.7, all Bayesian estimates (Median) are very close to corresponding MLEs and included in 95% confidence intervals.

4.6.2 Binary data: RNA-Seq

For the example of binary data set, what we use is "RNA-Seq" data set downloaded from the UCI machine learning repository: https://archive.ics.uci.edu/ml/datasets/gene+expression+cancer+RNA-Seq and maintained by the cancer genome atlas pan-cancer analysis project. The data set includes 20531 gene expressions levels in 801 patients having one of five different types of tumor: 300 patients with Breast Cancer (BRCA), 146 with Kidney Renal Clear Cell Carcinoma (KIRC), 78 with Colon Adenocarcinoma (COAD), 141 with Lung Adenocarcinoma (LUAD), and 136 with Prostate Adenocarcinoma (PRAD). To make the data set fittable to binary scenario, we generate a binary response variable $y_i = 1$ if ith observation has a BRCA solid tumor, and $y_i = 0$ if ith observation has a solid tumor other than BRCA. To reduce dimensionality and computational cost, we do the variable screening by performing FDR screening method (Benjamini and Hochberg, 1995; Craiu and Sun, 2008), which computes adjusted p-values for adaptive FDR control, to detect a group of covariates that has close relationships with the response variable. Given an FDR threshold, covariates with adjusted p-values smaller than it are selected as candidate variables. The details to compute adjusted p-values is shown in Appendix A.3.1. FDR screening can be used based on various models, including logistic regression model for the binary outcome (Chang et al., 2020). After performing the FDR screening method, there are 2059 genes left for the purpose of variable selection. We apply the proposed method and logistic regression models penalized by LASSO, ENET, MCP, and SCAD penalties to classify the binary response.

All results are shown in Table 4.5, 4.6 and 4.7. Table 4.5 shows that the proposed method select three genes (Gene 7964, 8349, and 15589 in Table 4.6) and has smallest FMSC values compared with other penalties approaches. This definitely shows that the proposed method is the best model to fit "RNA-seq" data set among them. In Table 4.6, we can observe that Gene 7964 and Gene 15589 are common genes selected by all methods. In Table 4.7, the Bayesian estimates (Median) is not as close to MLEs as other two data scenarios, this may be due to high correlations among three genes. But 95% credible intervals of coefficients still include their corresponding MLEs.

4.6.3 Count data: Crime and Communities

The example of count data is titled as "Crime and Communities" and downloaded from UCI machine learning repository website: https://archive.ics.uci.edu/ml/datasets/Communities+and+Crime+Unnormalized. The data contains some information of communities in the US, which combines three data sets: socio-economic data from 1990 Census, law enforcement data from the 1990 US Law Enforcement Management and Administration Statistics survey, and crime data from the 1995 FBI UCR over 50 states. This whole data set has 2216 observations with 147 variables. There are 18 potential response variables involving crime, such as murders (number of murders in 1995) and robberies (number of robberies in 1995). The independent variables in the data set involves the information of (i): community, such as population for community and mean people per household; (ii): law enforcement, such as total requests for police per police officer and the number of different kinds of drugs seized. We consider the number of murders in 1995 as the response variable and all variables regarding community and law enforcement as the predictors. To generate the high-dimensional issue, we only consider the observations within the Connecticut. After removing variables with missing values, we obtain the data set with $n = 71$ and $p = 103$. We apply our proposed method to the data set and compare it with Poisson regression models with penalties of LASSO, ENET, MCP and SCAD.

The results are displayed in Table 4.5, 4.6 and 4.7. In Table 4.5, it shows that three variables are selected by the proposed method and it has best performance in terms of FMSC. Table 4.6 gives the abbreviated variable names selected by each method and the information of them is displayed in the Table B.7 of Appendix B. In Table 4.7, Bayesian Poisson regression model has the estimates close to corresponding MLEs, covered by 95% credible intervals.

Table 4.5: Model selection performance evaluated by BIC, EBIC, MBIC, RICc and MRIC for Gaussian, binary, and count data.

	BIC	EBIC	MBIC	RICc	MRIC	NUM
Gaussian data						
Proposed	1074.7382	1141.1623	1107.2998	1139.3635	1119.0809	5
EBIC-LASSO	1095.1960	1167.4999	1127.7668	1166.8092	1148.5235	4
EBIC-ENET	1095.1960	1167.4999	1127.7668	1166.8092	1148.5235	4
EBIC-MCP	1095.1960	1167.4999	1127.7668	1166.8092	1148.5235	4
EBIC-SCAD	1095.1960	1167.4999	1127.7668	1166.8092	1148.5235	4
CV-LASSO	1216.9863	2907.2430	2397.6789	3812.9631	3150.1096	145
CV-ENET	1253.3920	2962.9995	2450.3700	3885.1753	3213.1790	147
CV-MCP	1084.6155	1943.9425	1613.8915	2248.3292	1951.1880	65
CV-SCAD	1146.9390	2084.5993	1733.2139	2435.9757	2106.8347	72
Binary data						
Proposed	38.8263	93.5041	66.4278	89.3793	73.1226	3
EBIC-LASSO	62.4658	104.6621	83.1682	100.3838	88.1909	2
EBIC-ENET	55.5049	110.1866	83.1081	106.0623	89.8051	3
EBIC-MCP	47.4836	102.1654	75.0868	98.0410	81.7839	3
EBIC-SCAD	62.4658	104.6621	83.1682	100.3838	88.1909	2
CV-LASSO	213.9476	538.6971	434.7732	618.4070	488.3495	31
CV-ENET	501.4396	1139.5007	1018.9996	1449.3914	1144.5692	74
CV-MCP	73.5445	206.3565	149.4533	212.5774	167.8701	10
CV-SCAD	86.9162	240.1280	176.6266	251.2278	198.3920	12
Count data						
Proposed	185.4760	216.1584	194.6094	217.8657	205.5805	3
EBIC-LASSO	206.4090	243.0828	217.8257	246.8961	231.5395	4
EBIC-ENET	206.0983	242.7720	217.5150	246.5854	231.2288	4
EBIC-MCP	199.0646	235.7384	210.4814	239.5518	224.1951	4
EBIC-SCAD	197.6563	228.3387	206.7897	230.0460	217.7608	3
CV-LASSO	204.2596	251.8182	220.2431	260.9416	239.4423	6
CV-ENET	204.3902	270.4669	229.5070	293.4618	259.6773	10
CV-MCP	199.0646	235.7384	210.4814	239.5518	224.1951	4
CV-SCAD	197.6563	228.3387	206.7897	230.0460	217.7608	3

Table 4.6: Predictors selected by proposed, EBIC-LASSO, EBIC-ENET, EBIC-MCP and EBIC-SCAD.

Predictor names	Proposed	LASSO	ENET	MCP	SCAD
			Gaussian data		
TBL1Y	X				
LOC146909	X				
CCNB1	X				
CIT		X	X	X	X
NBR2	X	X	X	X	X
ATAD5	X	X	X	X	X
TOP2A		X	X	X	X
			Binary data		
Gene89				X	
Gene7964	X	X	X	X	X
Gene8349	X				
Gene15589	X	X	X	X	X
Gene17905			X		
			Count data		
Population	X				
PersPerFam	X				
MedOwnCostPctInc	X				
racePctHisp		X	X		
NumUnderPov		X	X	X	
HousVacant		X	X		X
PctVacantBoarded		X			
NumKidsBornNeverMar			X		
pctWRetire				X	
NumInShelters				X	X
nonViolPerPop				X	
agePct16t24					X

Table 4.7: Bayesian and MLE estimates for regression model selected by the proposed method for Gaussian, binary, and count data; Notation: LB and UB are lower bound and upper bound of 95% confidence or credible interval.

	Bayesian			MLE		
Variables	Median	LB	UB	Estimate	LB	UB
			Gaussian data			
TBL1Y	-0.154	-0.213	-0.091	-0.155	-0.215	-0.095
LOC146909	0.262	0.204	0.320	0.263	0.423	0.530
CCNB1	0.150	0.090	0.209	0.149	0.237	0.358
NBR2	0.477	0.423	0.532	0.477	0.205	0.321
ATAD5	0.299	0.239	0.358	0.298	0.091	0.207
			Binary data			
Intercept	5.707	2.487	10.138	4.962	0.780	9.143
Gene7964	-18.618	-33.004	-8.372	-17.160	-32.773	-1.548
Gene8349	-11.906	-20.652	-5.105	-9.822	-17.713	-1.930
Gene15589	24.618	6.411	46.704	26.738	-0.658	54.133
			Count data			
Intercept	0.914	-5.270	6.668	0.967	-5.192	7.126
population	1.842	1.584	2.099	1.835	1.566	2.104
PersPerFam	-11.307	-18.389	-4.301	-11.377	-18.559	-4.196
MedOwnCostPctInc	8.529	5.862	11.287	8.587	5.803	11.371

4.7 Concluding remarks

While the proposed approach provides the same model selection performance as the traditional stochastic and hybrid search algorithms, it greatly alleviates the computational burden. The proposed method enables us to evaluate all candidate models for the next update simultaneously within a single computation without using parallel computing.

Regardless of data types, any parametric or semi-parametric regression model can be applied to the proposed approach as long as it satisfies the following three conditions: (1) a proportional form of the (pseudo) likelihood function is known, (2) the likelihood function has unimodal, and (3) the second derivative of the likelihood function exists.

Chapter 5

Conclusion

In this dissertation, we have developed three Bayesian approaches that can

- identify a best model via a fast global optimization;
- be quickly implemented in a single CPU core;
- apply to various types of data (e.g., Gaussian, multivariate, binary, count, and survival data).

The proposed methods have been validated by simulation studies and real data application. In addition, some important theoretical properties have been established.

In Chapter 2, we have developed a Bayesian variable selection of best subsets for linear model. The best subsets selection involves intensive computation and non-convex optimization problems. To solve them, we have proposed the hybrid search that combines the deterministic search and stochastic search to obtain the global maximum. To improve the computational speed, we have proposed a novel strategy that can simultaneously evaluate the exact marginal likelihoods of all neighbor models within a single computation. We have shown that the Bayesian variable selection criterion achieves the model selection consistency in the high-dimensional setting. The proposed hybrid search algorithm can be applied to frequentist model selection approaches such as AIC, BIC, and Mallows C_p. A possible limitation of the proposed method is that the applicability is limited to linear regression models

with Gaussian errors.

In Chapter 3, we have extended the proposed method in Chapter 2 to multivariate linear regression models. The benefit of multivariate regression is to take the advantage of the information in the covariance matrix of response variables which cannot be not taken into account under the classic linear regression framework. However, the extended method still requires an explicit proportional form of the marginal likelihood function.

In Chapter 4, we have proposed a more general method applicable to various data types, such as continuous, binary, count, and time-to-event data. Based on Laplace approximation, we have proposed a fast computing strategy that can simultaneously estimate all the marginal likelihoods of the neighborhood in a single computation. To accelerate the convergence, we employed a hybrid search algorithm that can quickly and accurately obtain the global maximum of marginal posterior probabilities in the model space. More general regression models can be also applied to the proposed approach, including binary regression models with various links, multinomial regression, and negative binomial models. It is worth noting that the proposed approximate Bayesian method requires a sufficiently large sample size to obtain an accurate result. In addition, the posterior mode must be unique. For future research, the proposed method can be extended to the multivariate generalized linear model, which describes how multiple responses with different types of data are regressed by a common set of predictors.

Bibliography

H. Akaike. Information theory and an extension of the maximum likelihood principle. In E. Parzen, K. Tanabe, and G. Kitagawa, editors, *Selected Papers of Hirotugu Akaike*, pages 199–213. Springer New York, New York, NY, 1998.

E. Akkartal., M. Mendes., and I. Kursun. Multivariate analysis of the relations between some blood biochemistry parameters, morphological characters and tonic immobility in broiler. *Asian Journal of Chemistry*, 21(4):2869–2874, 2009.

A. Antoniou, P. D. P. Pharoah, S. Narod, and H. A. Risch. Average risks of breast and ovarian cancer associated with BRCA1 or BRCA2 mutations detected in case series unselected for family history: a combined analysis of 22 studies. *American journal of human genetics*, 72(5):1117–1130, 2003.

E. M. L. Beale, M. G. Kendall, and D. W. Mann. The discarding of variables in multivariate analysis. *Biometrika*, 54(3/4):357–366, 1967.

E. J. Bedrick. Model selection for multivariate regression in small samples. *Journal of Biometrics*, pages 226–231, 1994.

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300, 1995.

D. Bertsimas, A. King, and R. Mazumder. Best subset selection via a modern optimization lens. *The Annals of Statistics*, 44(2):813–852, 2016.

J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society. Series B (Methodological)*, 48(3):259–302, 1986.

H. Bozdogan. Model selection and akaike's information criterion (AIC): The general theory and its analytical extensions. *Psychometrika*, 52(3):345–370, 1987.

L. Breiman and J. H. Friedman. Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 59(1): 3–54, 1997.

P. J. Brown and J. V. Zidek. Adaptive multivariate ridge regression. *Annals of Statistics*, 8 (1):64–74, 01 1980.

P. J. Brown, M. Vannucci, and T. Fearn. Multivariate bayesian variable selection and prediction. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 60(3): 627–641, 1998.

P. J. Brown, T. Fearn, and M. Vannucci. Bayesian wavelet regression on curves with application to a spectroscopic calibration problem. *Journal of the American Statistical Association*, 96(454):398–408, 2001.

C. Chang, C. Sung, and H. Hsiao. HDMAC: A web-based interactive program for high-dimensional analysis of molecular alterations in cancer. *Scientific Reports*, 10(3953), 2020.

J. Chen and Z. Chen. Extended Bayesian information criteria for model selection with large model spaces. *Biometrika*, 95(3):759–771, 2008.

D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(2):187–220, 1972.

D. R. Cox. Partial likelihood. *Biometrika*, 62(2):269–276, 08 1975.

R. V. Craiu and L. Sun. Choosing the lesser evil: Trade-off between false discovery rate and non-discovery rate. *Statistica Sinica*, 18(3):861–879, 2008.

D. L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. *Aide-Memoire of the lecture in AMS conference on Math challenges of 21st Centrury*, 2000.

J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360, 2001.

J. Fan and R. Li. Statistical challenges with high dimensionality: Feature selection in knowledge discovery. *Proceedings of the Madrid International Congress of Mathematicians*, 2006.

G. M. Findlay, R. M. Daza, B. Martin, M. D. Zhang, A. P. Leith, M. Gasperini, J. D. Janizek, X. Huang, L. M. Starita, and J. Shendure. Accurate classification of BRCA1 variants with saturation genome editing. *Nature*, 562(7726):217–222, 2018.

D. P. Foster and E. I. George. The risk inflation criterion for multiple regression. *The Annals of Statistics*, 22(4):1947–1975, 1994.

J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

Y. Fujikoshi and K. Satoh. Modified AIC and Cp in multivariate linear regression. *Biometrika*, 84(3):707–716, 1997.

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

E. I. George and R. E. McCulloch. Approaches for bayesian variable selection. *Statistica Sinica*, 7(2):339–373, 1997.

B. Goodrich, J. Gabry, I. Ali, and S. Brilleman. rstanarm: Bayesian applied regression modeling via Stan., 2020. R package version 2.21.1.

D. A. Gordon, W.-H. Huang, D. M. Burns, R. H. French, and L. S. Bruckman. Multivariate multiple regression models of poly(ethylene-terephthalate) film degradation under outdoor and multi-stressor accelerated weathering exposures. *PLOS ONE*, 13(12):1–30, 12 2018.

C. Hans, A. Dobra, and M. West. Shotgun stochastic search for "large p" regression. *Journal of the American Statistical Association*, 102(478):507–516, 2007.

H.-M. Herz, Z. Chen, H. Scherr, M. Lackey, C. Bolduc, and A. Bergmann. vps25 mosaics display non-autonomous cell survival and overgrowth, and autonomous apoptosis. *Development*, 133(10):1871–1880, 2006.

R. R. Hocking and R. N. Leslie. Selection of the best subset in regression analysis. *Technometrics*, 9(4):531–540, 1967.

G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning: With Applications in R*. Springer Publishing Company, Incorporated, 2014.

D. I. Jeong., A. St-Hilaire., T. B. M. J. Ouarda., and P. Gachon. Multivariate multiple regression analysis based on principal component scores to study relationships between some pre- and post-slaughter traits of broilers. *Climatic Change*, 114:567–591, 2012.

R. E. Kass and A. E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995.

S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

R. Krishnan, N. Boddapati, and S. Mahalingam. Interplay between human nucleolar gnl1 and rps20 is critical to modulate cell proliferation. *Scientific Reports*, 8, 2018.

F. Liang, C. Liu, and R. J. Carroll. Stochastic approximation in Monte Carlo computation. *Journal of the American Statistical Association*, 102(477):305–320, 2007.

F. Liang, Q. Song, and K. Yu. Bayesian subset modeling for high-dimensional generalized linear models. *Journal of the American Statistical Association*, 108(502):589–606, 2013.

F. G. López, M. G. Torres, B. M. Batista, J. A. M. Pérez, and J. M. Moreno-Vega. Solving feature subset selection problem by a parallel scatter search. *European Journal of Operational Research*, 169(2):477–489, 2006.

D. Madigan and J. York. Bayesian graphical models for discrete data. *International Statistical Review*, 63(2):215–232, 1995.

C. Marceaux, D. Petit, J. Bertoglio, and M. D. David. Phosphorylation of arhgap19 by cdk1 and rock regulates its subcellular localization and function during mitosis. *Journal of Cell Science*, 131(5), 2018.

M. Mehmet. Multivariate multiple regression analysis based on principal component scores to study relationships between some pre- and post-slaughter traits of broilers. *Tarim Bilimleri Dergisi-journal of Agricultural Sciences*, 17:77–83, 2011.

R. Noorossana, M. Eyvazian, A. Amiri, and M. A. Mahmoud. Statistical monitoring of multivariate multiple linear regression profiles in phase i with calibration application. *Quality and Reliability Engineering International*, 26(3):291–303, 2010a.

R. Noorossana, M. Eyvazian, and A. Vaghefi. Phase ii monitoring of multivariate simple linear profiles. *Computers & Industrial Engineering*, 58(4):563 – 570, 2010b.

M. Ramos. *curatedTCGAData: Curated Data From The Cancer Genome Atlas (TCGA) as MultiAssayExperiment Objects*, 2019. R package version 1.4.3.

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

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 58(1):267–288, 1996.

P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*, 109:475–494, 2001.

H. Wang, B. Li, and C. Leng. Shrinkage tuning parameter selection with a diverging number of parameters. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(3):671–683, 2009.

J. Wang, W. Yoo, A. Sim, P. Nugent, and K. Wu. Parallel variable selection for effective performance prediction. *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, pages 208–217, 2017.

X. Yang and M. E. Lippman. BRCA1 and BRCA2 in breast cancer. *Breast Cancer Research and Treatment*, 54(1):1–10, Mar 1999.

W. Yoo, M. Koo, Y. Cao, A. Sim, P. Nugent, and K. Wu. Patha: Performance analysis tool for hpc applications. *2015 IEEE 34th International Performance Computing and Communications Conference (IPCCC)*, pages 1–8, 2015.

C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.

Z. Zhao, R. Zhang, J. Cox, D. Duling, and W. Sarle. Massively parallel feature selection: an approach based on variance preservation. *Machine Learning*, 92(1):195–220, 2013.

Y. Zhou, U. Porwal, C. Zhang, H. Q. Ngo, L. Nguyen, C. Ré, and V. Govindaraju. Parallel feature selection inspired by group testing. *Advances in Neural Information Processing Systems*, 2014.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

Appendix A

Calculations

A.1 Calculation in Chapter 2

A.1.1 Calculation in (2.5)

For any $i \notin \hat{\boldsymbol{\gamma}}$, from (2.2), we have

$$\begin{aligned} m(\mathbf{y}|\hat{\boldsymbol{\gamma}} \cup \{i\}) \quad &\propto \quad |\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}^{\top}\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}} + \tau^{-1}\mathbf{I}_{k+1}|^{-1/2} \\ &\quad \times \left(\mathbf{y}^{\top}\mathbf{H}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}\mathbf{y} + b_{\sigma}\right)^{-\frac{a_{\sigma}+n}{2}}. \end{aligned} \tag{A.1}$$

It follows from the Sherman-Morrison formula that

$$\mathbf{H}_{\hat{\boldsymbol{\gamma}}\cup\{i\}} = \mathbf{H}_{\hat{\boldsymbol{\gamma}}} - \frac{\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i\mathbf{x}_i^{\top}\mathbf{H}_{\hat{\boldsymbol{\gamma}}}}{\tau^{-1} + \mathbf{x}_i^{\top}\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i}. \tag{A.2}$$

Using the Sylvester's determinant identity and the Sherman-Morrison formula, we obtain

$$
\begin{aligned}
&|\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}^\top \mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}} + \tau^{-1}\mathbf{I}_{k+1}| \\
&= \tau^{-(k+1)}|\tau \mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}^\top + \mathbf{I}_n| \\
&= \tau^{-(k+1)}|\tau \mathbf{X}_{\hat{\boldsymbol{\gamma}}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n + \tau \mathbf{x}_i\mathbf{x}_i^\top| \\
&= \tau^{-(k+1)}|\tau \mathbf{X}_{\hat{\boldsymbol{\gamma}}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n|\{1 + \tau \mathbf{x}_i^\top(\tau \mathbf{X}_{\hat{\boldsymbol{\gamma}}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n)^{-1}\mathbf{x}_i\} \\
&= |\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top \mathbf{X}_{\hat{\boldsymbol{\gamma}}} + \tau^{-1}\mathbf{I}_k|(\tau^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i). \qquad \text{(A.3)}
\end{aligned}
$$

Applying (A.2) and (A.3) to (A.1), we thus have

$$
\begin{aligned}
m(\mathbf{y}|\hat{\boldsymbol{\gamma}}\cup\{i\}) \quad \propto \quad & \left\{\mathbf{y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{y} - \frac{(\mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{y})^2}{\tau^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i} + b_\sigma\right\}^{-\frac{a_\sigma+n}{2}} \\
& \times(\tau^{-1} + \mathbf{x}_i^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i)^{-1/2}
\end{aligned}
$$

for any $i \notin \hat{\boldsymbol{\gamma}}$.

A.1.2 Calculation in (2.7)

For any $j \in \tilde{\boldsymbol{\gamma}}$, (2.2) leads to

$$
\begin{aligned}
m(\mathbf{y}|\tilde{\boldsymbol{\gamma}} \setminus \{j\}) \quad \propto \quad & |\mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{j\}}^\top \mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{j\}} + \tau^{-1}\mathbf{I}_k|^{-1/2} \\
& \times \left(\mathbf{y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}\setminus\{j\}}\mathbf{y} + b_\sigma\right)^{-\frac{a_\sigma+n}{2}}. \qquad \text{(A.4)}
\end{aligned}
$$

From the Sherman-Morrison formula, we have

$$
\mathbf{H}_{\tilde{\boldsymbol{\gamma}}\setminus\{j\}} = \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} + \frac{\mathbf{H}_{\tilde{\boldsymbol{\gamma}}}\mathbf{x}_j\mathbf{x}_j^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}}}{\tau^{-1} - \mathbf{x}_j^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}}\mathbf{x}_j}. \qquad \text{(A.5)}
$$

From the Sylvester's determinant identity and the Sherman-Morrison formula, we obtain

$$
\begin{aligned}
&|\mathbf{X}_{\tilde{\gamma}\setminus\{j\}}^\top \mathbf{X}_{\tilde{\gamma}\setminus\{j\}} + \tau^{-1}\mathbf{I}_k| \\
&= \tau^{-k}|\tau \mathbf{X}_{\tilde{\gamma}\setminus\{j\}} \mathbf{X}_{\tilde{\gamma}\setminus\{j\}}^\top + \mathbf{I}_n| \\
&= \tau^{-k}|\tau \mathbf{X}_{\tilde{\gamma}} \mathbf{X}_{\tilde{\gamma}}^\top + \mathbf{I}_n - \tau \mathbf{x}_j \mathbf{x}_j^\top| \\
&= \tau^{-k}|\tau \mathbf{X}_{\tilde{\gamma}} \mathbf{X}_{\tilde{\gamma}}^\top + \mathbf{I}_n|\{1 - \tau \mathbf{x}_j^\top (\tau \mathbf{X}_{\tilde{\gamma}} \mathbf{X}_{\tilde{\gamma}}^\top + \mathbf{I}_n)^{-1}\mathbf{x}_j\} \\
&= \tau^2 |\mathbf{X}_{\tilde{\gamma}}^\top \mathbf{X}_{\tilde{\gamma}} + \tau^{-1}\mathbf{I}_{k+1}|(\tau^{-1} - \mathbf{x}_j^\top \mathbf{H}_{\tilde{\gamma}} \mathbf{x}_j). \qquad \text{(A.6)}
\end{aligned}
$$

Hence, applying (A.5) and (A.6) to (A.4), we have

$$
\begin{aligned}
m(\mathbf{y}|\tilde{\gamma}\setminus\{j\}) \propto &\left\{ \mathbf{y}^\top \mathbf{H}_{\tilde{\gamma}} \mathbf{y} + \frac{(\mathbf{x}_j^\top \mathbf{H}_{\tilde{\gamma}} \mathbf{y})^2}{\tau^{-1} - \mathbf{x}_j^\top \mathbf{H}_{\tilde{\gamma}} \mathbf{x}_j} + b_\sigma \right\}^{-\frac{a_\sigma + n}{2}} \\
&\times (\tau^{-1} - \mathbf{x}_j^\top \mathbf{H}_{\tilde{\gamma}} \mathbf{x}_j)^{-1/2}
\end{aligned}
$$

for any $j \in \tilde{\gamma}$.

A.2 Calculations in Chapter 3

A.2.1 Brown method

Consider a MVRM,

$$\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{E},$$

where $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n)^\top$ is the $n \times q$ response matrix, $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$ is the $n \times p$ design matrix, $\mathbf{B} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_p)^\top$ is a $p \times q$ unknown coefficient matrix, and $\mathbf{E} = (\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n)^\top$ is an $n \times q$ error matrix with $\mathbf{e}_i \overset{i.i.d}{\sim} \mathcal{N}_q(\mathbf{0}, \boldsymbol{\Sigma})$. To identify the variables selected among p regressors, Brown et al. (1998) introduced a latent binary vector $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_p)^\top$ with

$$
\xi_j = \begin{cases} 1 & \text{if } \boldsymbol{\beta}_j \text{ is active} \\ 0 & \text{if } \boldsymbol{\beta}_j \text{ is not active} \end{cases}.
$$

The conjugate priors are considered as follows:

$$\begin{aligned}
&\pi(\mathbf{B}_{\boldsymbol{\xi}}|\boldsymbol{\Sigma},\boldsymbol{\xi}) \sim \mathcal{MN}(\mathbf{0},\mathbf{H}_{\boldsymbol{\xi}},\boldsymbol{\Sigma}) \\
&\pi(\boldsymbol{\Sigma}) \sim \mathcal{W}^{-1}(\mathbf{Q},\delta) \\
&\pi(\boldsymbol{\xi}) \overset{ind}{\sim} \prod_{j=1}^{p} \mathrm{Ber}(\omega_j),
\end{aligned}$$

where $\mathbf{Q}$, $\mathbf{H}_{\boldsymbol{\xi}}$ and δ are hyperparameters. Then, by Bayesian theorem, the marginal posterior distribution can be obtained by

$$\begin{aligned}
& m(\boldsymbol{\xi}|\mathbf{Y}) \propto m(\mathbf{Y}|\boldsymbol{\xi})\pi(\boldsymbol{\xi}) \\
= & \iint f(\mathbf{Y}|\mathbf{B}_{\boldsymbol{\xi}},\boldsymbol{\Sigma})\pi(\mathbf{B}_{\boldsymbol{\xi}}|\boldsymbol{\Sigma},\boldsymbol{\xi})\pi(\boldsymbol{\Sigma})d\mathbf{B}d\boldsymbol{\Sigma}\pi(\boldsymbol{\xi}) \\
\propto & |\mathbf{H}_{\boldsymbol{\xi}}\mathbf{K}_{\boldsymbol{\xi}}|^{-\frac{q}{2}}|\mathbf{Y}^\top(\mathbf{I}_n - \mathbf{X}_{\boldsymbol{\xi}}\mathbf{K}_{\boldsymbol{\xi}}^{-1}\mathbf{X}_{\boldsymbol{\xi}}^\top)\mathbf{Y} + \mathbf{Q}|^{-\frac{n+\delta}{2}}\pi(\boldsymbol{\xi}) \\
\equiv & g(\boldsymbol{\xi}|\mathbf{Y}), \qquad \text{(A.7)}
\end{aligned}$$

where $\mathbf{K}_{\boldsymbol{\xi}} = \mathbf{X}_{\boldsymbol{\xi}}^\top\mathbf{X}_{\boldsymbol{\xi}} + \mathbf{H}_{\boldsymbol{\xi}}^{-1}$ and $g(\boldsymbol{\xi}|\mathbf{Y})$ is the proportional form of $\pi(\boldsymbol{\xi}|\mathbf{Y})$. Brown et al. (1998) set $\mathbf{Q} = k\mathbf{I}_q$ and $\mathbf{H}_{\boldsymbol{\xi}} = c(\mathbf{X}_{\boldsymbol{\xi}}^\top\mathbf{X}_{\boldsymbol{\xi}})^{-1}$ and $\omega_j = \omega$, then $\pi(\boldsymbol{\xi}) = \prod_{j=1}^{p}\omega_j^{\xi_j}(1-\omega_j)^{1-\xi_j} = \omega^{|\boldsymbol{\xi}|}(1-\omega)^{p-|\boldsymbol{\xi}|}$. Apply $\mathbf{Q} = k\mathbf{I}_q$ and $\mathbf{H}_{\boldsymbol{\xi}} = c(\mathbf{X}_{\boldsymbol{\xi}}^\top\mathbf{X}_{\boldsymbol{\xi}})^{-1}$ into (A.7), hence we have

$$\begin{aligned}
\log g(\boldsymbol{\xi}|\mathbf{Y}) \quad = \quad & -\frac{n+\delta}{2}\log|k\mathbf{I}_q + \mathbf{Y}^\top\mathbf{Y} - \frac{c}{c+1}\mathbf{Y}^\top\mathbf{X}_{\boldsymbol{\xi}}(\mathbf{X}_{\boldsymbol{\xi}}^\top\mathbf{X}_{\boldsymbol{\xi}}^{-1})\mathbf{X}_{\boldsymbol{\xi}}^\top\mathbf{Y}| - \\
& \frac{q|\boldsymbol{\xi}|}{2}\log(c+1) + |\boldsymbol{\xi}|\log\omega + (p-|\boldsymbol{\xi}|)\log(1-\omega).
\end{aligned}$$

Exhaustive computation of $g(\boldsymbol{\xi}|\mathbf{Y})$ for 2^p values of $\boldsymbol{\xi}$ becomes prohibitive even in a super computer when p is larger than 40. In this circumstances, using MCMC sampling to explore marginal posterior is possible. Brown et al. (1998) use Gibbs sampler to generate each $\boldsymbol{\xi}$ value component-wise from full conditional distributions $\pi(\xi_j|\boldsymbol{\xi}_{-j},\mathbf{Y})$, where $\boldsymbol{\xi}_{-j} = \boldsymbol{\xi} \setminus \xi_j$. It is easy to show that

$$\pi(\xi_j = 1|\boldsymbol{\xi}_{-j},\mathbf{Y}) = \frac{\theta_j}{\theta_j+1},$$

where

$$\theta_j = \frac{g(\xi_j = 1, \boldsymbol{\xi}_{-j}|\mathbf{Y})}{g(\xi_j = 0, \boldsymbol{\xi}_{-j}|\mathbf{Y})}.$$

The complete algorithm of Brown et al. (1998) is described in Algorithm 11:

Algorithm 11 Brown: MCMC sampling for $\boldsymbol{\xi}$

1. Initialize $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots, \xi_p)$.
2. **Repeat** for $t = 1, 2, \ldots$ until convergence
   1). $a \leftarrow g(\xi_1^{(t)} = 1, \boldsymbol{\xi}_{-1}^{(t)}|\mathbf{Y})$ and $b \leftarrow g(\xi_1^{(t)} = 0, \boldsymbol{\xi}_{-1}^{(t)}|\mathbf{Y})$.
   2). $\theta_1 \leftarrow \frac{a}{b}$; $p_1 \leftarrow \frac{\theta_1}{\theta_1+1}$.
   3). $\xi^* \sim \text{Ber}(p_1)$; $\xi_1^{(t)} \leftarrow \xi^*$.
   4). **Repeat** for $j = 2, \ldots, p$:
      i) $\kappa = a\xi^* + b(1-\xi^*)$; $\tilde{\boldsymbol{\xi}} \leftarrow \boldsymbol{\xi}^{(t)}$.
      ii) **If** $\xi_j^{(t)} = 1$,
         $\tilde{\xi}_j \leftarrow 0$; $a \leftarrow \kappa$; $b \leftarrow g(\tilde{\boldsymbol{\xi}}|\mathbf{Y})$.
      else
         $\tilde{\xi}_j \leftarrow 1$; $a \leftarrow g(\tilde{\boldsymbol{\xi}}|\mathbf{Y})$; $b \leftarrow \kappa$.
      iii) $\theta_j \leftarrow \frac{a}{b}$; $p_j \leftarrow \frac{\theta_j}{\theta_j+1}$.
      iv) $\xi^* \sim Ber(p_j)$; $\xi_j^{(t)} \leftarrow \xi^*$.
   5). Get $\boldsymbol{\xi}^{(t)}$

For the hyperparameter, we follow the same setting with Brown et al. (1998), which are $\delta = 2 + q, \omega = 20/p, k = 2$ and $c = 10$. Note that since our definition of inverse-wishart distribution is different from that of Brown et al. (1998), the values of q and k we set here are after adjustment to make them consistent with Brown et al. (1998).

A.2.2 Calculation in (3.7)

For any $i \notin \hat{\boldsymbol{\gamma}}$, $|\hat{\boldsymbol{\gamma}} \cup \{i\}| = k + 1$, hence $s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\})$ in Eq. (3.3) can be expressed as

$$s(\mathbf{Y}|\hat{\boldsymbol{\gamma}} \cup \{i\}) = \zeta^{-\frac{m(k+1)}{2}} |\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}^\top \mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}} + \zeta^{-1}\mathbf{I}_{k+1}|^{-\frac{m}{2}} |\mathbf{Y}^\top \mathbf{H}_{\hat{\boldsymbol{\gamma}}\cup\{i\}} \mathbf{Y} + \boldsymbol{\Psi}|^{-\frac{n+\nu}{2}}. \tag{A.8}$$

From the Lemma 1 and Sherman-Morrison formula, we have

$$\begin{aligned}
\mathbf{H}_{\hat{\boldsymbol{\gamma}}\cup\{i\}} &= (\zeta \mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}^\top + \mathbf{I}_n)^{-1} \\
&= (\mathbf{I}_n + \zeta \mathbf{X}_{\hat{\boldsymbol{\gamma}}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top + \zeta \mathbf{x}_i\mathbf{x}_i^\top)^{-1} \\
&= (\mathbf{H}_{\hat{\boldsymbol{\gamma}}}^{-1} + \zeta \mathbf{x}_i\mathbf{x}_i^\top)^{-1} \\
&= \mathbf{H}_{\hat{\boldsymbol{\gamma}}} - \frac{\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i\mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}}{\zeta^{-1} + \mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i}. \qquad \text{(A.9)}
\end{aligned}$$

By using Lemma 1 and Lemma 2, we have

$$\begin{aligned}
& |\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}^\top\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}} + \zeta^{-1}\mathbf{I}_{k+1}| \\
= \; & \zeta^{-(k+1)}|\zeta \mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}\cup\{i\}}^\top + \mathbf{I}_n| \\
= \; & \zeta^{-(k+1)}|\zeta \mathbf{X}_{\hat{\boldsymbol{\gamma}}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n + \zeta \mathbf{x}_i\mathbf{x}_i^\top| \\
= \; & \zeta^{-(k+1)}|\zeta \mathbf{X}_{\hat{\boldsymbol{\gamma}}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n|\{1 + \zeta \mathbf{x}_i^\top(\zeta \mathbf{X}_{\hat{\boldsymbol{\gamma}}}\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n)^{-1}\mathbf{x}_i\} \\
= \; & |\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top\mathbf{X}_{\hat{\boldsymbol{\gamma}}} + \zeta^{-1}\mathbf{I}_k|(\zeta^{-1} + \mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i). \qquad \text{(A.10)}
\end{aligned}$$

Applying (A.9) and (A.10) to (A.8), we have

$$\begin{aligned}
s(\mathbf{Y}|\hat{\boldsymbol{\gamma}}\cup\{i\}) &= \zeta^{-\frac{m(k+1)}{2}}|\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top\mathbf{X}_{\hat{\boldsymbol{\gamma}}} + \zeta^{-1}\mathbf{I}_k|^{-\frac{m}{2}}(\zeta^{-1} + \mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i)^{-\frac{m}{2}} \times \\
&\quad \left|\mathbf{Y}^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{Y} + \boldsymbol{\Psi} - \frac{\mathbf{Y}^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i\mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{Y}}{\zeta^{-1} + \mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i}\right|^{-\frac{n+\nu}{2}} \\
&= \zeta^{-\frac{m(k+1)}{2}}|\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top\mathbf{X}_{\hat{\boldsymbol{\gamma}}} + \zeta^{-1}\mathbf{I}_k|^{-\frac{m}{2}}(\zeta^{-1} + \mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i)^{-\frac{m}{2}}\left|\mathbf{Y}^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{Y} + \boldsymbol{\Psi}\right|^{-\frac{n+\nu}{2}} \times \\
&\quad \left[1 - \frac{\mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{Y}(\mathbf{Y}^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{Y} + \boldsymbol{\Psi})^{-1}\mathbf{Y}^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i}{\zeta^{-1} + \mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i}\right]^{-\frac{n+\nu}{2}} \qquad \text{(A.11)} \\
&= c_{\hat{\boldsymbol{\gamma}}}^+ \times (\zeta^{-1} + \mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i)^{-\frac{m}{2}}\left[1 - \frac{\mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{Y}(\mathbf{Y}^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{Y} + \boldsymbol{\Psi})^{-1}\mathbf{Y}^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i}{\zeta^{-1} + \mathbf{x}_i^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{x}_i}\right]^{-\frac{n+\nu}{2}},
\end{aligned}$$

where $c_{\hat{\boldsymbol{\gamma}}}^+ = \zeta^{-\frac{m(k+1)}{2}}|\mathbf{X}_{\hat{\boldsymbol{\gamma}}}^\top\mathbf{X}_{\hat{\boldsymbol{\gamma}}} + \zeta^{-1}\mathbf{I}_k|^{-\frac{m}{2}}\left|\mathbf{Y}^\top\mathbf{H}_{\hat{\boldsymbol{\gamma}}}\mathbf{Y} + \boldsymbol{\Psi}\right|^{-\frac{n+\nu}{2}}$ is a constant with respect to $i \notin \hat{\boldsymbol{\gamma}}$ and (A.11) holds by Lemma 2 and 3.

A.2.3 Calculation in (3.9)

For any $\ell \in \tilde{\boldsymbol{\gamma}}$, $|\tilde{\boldsymbol{\gamma}} \setminus \{\ell\}| = k$, hence $s(\mathbf{Y}|\tilde{\boldsymbol{\gamma}} \setminus \{\ell\})$ in (3.3) can be expressed as

$$s(\mathbf{Y}|\tilde{\boldsymbol{\gamma}} \setminus \{\ell\}) = \zeta^{-\frac{mk}{2}} |\mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}}^\top \mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}} + \zeta^{-1}\mathbf{I}_k|^{-\frac{m}{2}} |\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}} \mathbf{Y} + \boldsymbol{\Psi}|^{-\frac{n+\nu}{2}}. \tag{A.12}$$

From the Lemma 1 and Sherman-Morrison formula, we have

$$\begin{aligned}
\mathbf{H}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}} &= (\zeta \mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}} \mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}}^\top + \mathbf{I}_n)^{-1} \\
&= (\mathbf{I}_n + \zeta \mathbf{X}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X}_{\tilde{\boldsymbol{\gamma}}}^\top - \zeta \mathbf{x}_\ell \mathbf{x}_\ell^\top)^{-1} \\
&= (\mathbf{H}_{\tilde{\boldsymbol{\gamma}}}^{-1} - \zeta \mathbf{x}_\ell \mathbf{x}_\ell^\top)^{-1} \\
&= \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} + \frac{\mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}}}{\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell}.
\end{aligned} \tag{A.13}$$

By using Lemma 1 and Lemma 2, we have

$$\begin{aligned}
& |\mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}}^\top \mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}} + \zeta^{-1}\mathbf{I}_k| \\
= \; & \zeta^{-k} |\zeta \mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}} \mathbf{X}_{\tilde{\boldsymbol{\gamma}}\setminus\{\ell\}}^\top + \mathbf{I}_n| \\
= \; & \zeta^{-k} |\zeta \mathbf{X}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X}_{\tilde{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n - \zeta \mathbf{x}_\ell \mathbf{x}_\ell^\top| \\
= \; & \zeta^{-k} |\zeta \mathbf{X}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X}_{\tilde{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n| \{1 - \zeta \mathbf{x}_\ell^\top (\zeta \mathbf{X}_{\tilde{\boldsymbol{\gamma}}} \mathbf{X}_{\tilde{\boldsymbol{\gamma}}}^\top + \mathbf{I}_n)^{-1} \mathbf{x}_\ell\} \\
= \; & \zeta^2 |\mathbf{X}_{\tilde{\boldsymbol{\gamma}}}^\top \mathbf{X}_{\tilde{\boldsymbol{\gamma}}} + \zeta^{-1}\mathbf{I}_{k+1}| (\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell).
\end{aligned} \tag{A.14}$$

Applying (A.12) and (A.13) to (A.14), we have

$$
\begin{aligned}
s(\mathbf{Y}|\tilde{\boldsymbol{\gamma}} \setminus \ell) &= \zeta^{-\frac{mk}{2}} |\mathbf{X}_{\tilde{\boldsymbol{\gamma}}}^\top \mathbf{X}_{\tilde{\boldsymbol{\gamma}}} + \zeta^{-1}\mathbf{I}_{k+1}|^{-\frac{m}{2}} (\zeta^{-1} + \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell)^{-\frac{m}{2}} \times \\
&\quad \left| \mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi} + \frac{\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y}}{\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell} \right|^{-\frac{n+\nu}{2}} \\
&= \zeta^{-\frac{mk}{2}} |\mathbf{X}_{\tilde{\boldsymbol{\gamma}}}^\top \mathbf{X}_{\tilde{\boldsymbol{\gamma}}} + \zeta^{-1}\mathbf{I}_{k+1}|^{-\frac{m}{2}} (\zeta^{-1} + \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell)^{-\frac{m}{2}} \left| \mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi} \right|^{-\frac{n+\nu}{2}} \times \\
&\quad \left[\mathbf{1}_p + \frac{\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell (\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi})^{-1} \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y}}{\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell} \right]^{-\frac{n+\nu}{2}} \qquad \text{(A.15)} \\
&= c_{\tilde{\boldsymbol{\gamma}}}^- \times (\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell)^{-\frac{m}{2}} \left[\mathbf{1}_p + \frac{\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell (\mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi})^{-1} \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y}}{\zeta^{-1} - \mathbf{x}_\ell^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{x}_\ell} \right]^{-\frac{n+\nu}{2}},
\end{aligned}
$$

where $c_{\tilde{\boldsymbol{\gamma}}}^- = \zeta^{-\frac{mk}{2}} |\mathbf{X}_{\tilde{\boldsymbol{\gamma}}}^\top \mathbf{X}_{\tilde{\boldsymbol{\gamma}}} + \zeta^{-1}\mathbf{I}_{k+1}|^{-\frac{m}{2}} \left| \mathbf{Y}^\top \mathbf{H}_{\tilde{\boldsymbol{\gamma}}} \mathbf{Y} + \boldsymbol{\Psi} \right|^{-\frac{n+\nu}{2}}$ is a constant with respect to $\ell \notin \tilde{\boldsymbol{\gamma}}$ and (A.15) holds by Lemma 2 and 3.

A.3 Calculations in Chapter 4

A.3.1 FDR screening

Section three of (Craiu and Sun, 2008) gives us the details of how to obtain adjusted p-values. Consider a sequence of hypothesis testings: $H_1, H_2, \ldots, H_m$ with corresponding p-values $p_1, p_2, \ldots, p_m$. Let $p_{(1)}, p_{(2)}, \ldots, p_{(m)}$ be the ordered p-value sequence. The FDR control procedure proposed by Benjamini and Hochberg (1995) can be performed by means of FDR adjusted p-value. The FDR adjusted p-value corresponding to $p_{(i)}$ is

$$
p_{(i)}^{\text{FDR}} = \min \left\{ \frac{m p_{(k)}}{k} : k \geq i \right\}.
$$

Covariates $\{\mathbf{x}_{(i)} : p_{(i)}^{\text{FDR}} \leq \alpha^*\}$ will be selected as candidate covariates, where α^* is the FDR threshold. This leads to FDR $\leq \alpha^*$.

A.4 Important Lemmas

Lemma 1. *Consider an $n \times p$ matrix $\mathbf{X}$ and a scalar ζ, then*

$$\begin{aligned}
\mathbf{H} &= \mathbf{I}_n - \mathbf{X}(\mathbf{X}^\top\mathbf{X} + \zeta^{-1}\mathbf{I}_p)^{-1}\mathbf{X}^\top \\
&= \mathbf{I}_n - \mathbf{X}\mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + \zeta^{-1}\mathbf{I}_n)^{-1} \\
&= \{(\mathbf{X}\mathbf{X}^\top + \zeta^{-1}\mathbf{I}_n) - \mathbf{X}\mathbf{X}^\top\}(\mathbf{X}\mathbf{X}^\top + \zeta^{-1}\mathbf{I}_n)^{-1} \\
&= \zeta^{-1}(\mathbf{X}\mathbf{X}^\top + \zeta^{-1}\mathbf{I}_n)^{-1} \\
&= (\zeta\mathbf{X}\mathbf{X}^\top + \mathbf{I}_n)^{-1}.
\end{aligned}$$

Lemma 2. *Suppose $\mathbf{A}$ is an invertible square matrix and $\mathbf{U}$ and $\mathbf{V}$ are $n \times m$ matrices, then*

$$\det(\mathbf{A} + \mathbf{U}\mathbf{V}^\top) = \det(\mathbf{I}_m + \mathbf{V}^\top\mathbf{A}^{-1}\mathbf{U})\det(\mathbf{A})$$

Lemma 3. *Suppose $\mathbf{u}$ and $\mathbf{v}$ are m-dimension vectors, then*

$$|\mathbf{I}_m - \mathbf{u}\mathbf{v}^\top| = 1 - \mathbf{u}^\top\mathbf{v}$$

Appendix B

Additional tables

Table B.1: Simulation study for AIC, AICc, BIC and CAIC of $\rho_e = 0$ and $\rho_x = 0.2$ based on 100 replications. Notation: FDR—false discovery rate; TRUE%—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Method	FDR (s.e)	FOR (s.e)	TRUE% (s.e)	SIZE (s.e)	HAM (s.e)
			$p = 200, \rho_x = 0.2$		
mLASSO-AIC	0.573(0.007)	0(0)	0(0)	19.11(0.238)	11.11(0.238)
mENET-AIC	0.533(0.015)	0(0)	5(2.19)	18.33(0.389)	10.33(0.389)
mLASSO-AICc	0.205(0.015)	0(0)	21(4.094)	10.48(0.225)	2.48(0.225)
mENET-AICc	0.132(0.014)	0(0)	39(4.902)	9.49(0.179)	1.49(0.179)
mLASSO-BIC	0.011(0.003)	0(0)	90(3.015)	8.1(0.03)	0.1(0.03)
mENET-BIC	0.009(0.003)	0(0)	93(2.564)	8.08(0.031)	0.08(0.031)
mLASSO-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)
mENET-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)
			$p = 500, \rho_x = 0.2$		
mLASSO-AIC	0.596(0.004)	0(0)	0(0)	19.99(0.184)	11.99(0.184)
mENET-AIC	0.578(0.008)	0(0)	0(0)	19.42(0.257)	11.42(0.257)
mLASSO-AICc	0.298(0.017)	0(0)	11(3.145)	12.12(0.312)	4.12(0.312)
mENET-AICc	0.185(0.018)	0(0)	37(4.852)	10.38(0.273)	2.38(0.273)
mLASSO-BIC	0.006(0.003)	0(0)	95(2.19)	8.06(0.028)	0.06(0.028)
mENET-BIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)
mLASSO-CAIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)
mENET-CAIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)
			$p = 1000, \rho_x = 0.2$		
mLASSO-AIC	0.6(0.004)	0(0)	0(0)	20.22(0.188)	12.22(0.188)
mENET-AIC	0.579(0.008)	0(0)	1(1)	19.42(0.234)	11.42(0.234)
mLASSO-AICc	0.312(0.018)	0(0)	15(3.589)	12.45(0.329)	4.45(0.329)
mENET-AICc	0.187(0.019)	0(0)	38(4.878)	10.53(0.306)	2.53(0.306)
mLASSO-BIC	0.008(0.003)	0(0)	94(2.387)	8.07(0.029)	0.07(0.029)
mENET-BIC	0.005(0.003)	0(0)	96(1.969)	8.05(0.026)	0.05(0.026)
mLASSO-CAIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)
mENET-CAIC	0.002(0.002)	0(0)	98(1.407)	8.02(0.014)	0.02(0.014)

Table B.2: Simulation study for AIC, AICc, BIC and CAIC of $\rho_e = 0.2$ and $\rho_x = 0.2$ based on 100 replications. Notation: FDR—false discovery rate; TRUE%—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Method	FDR (s.e)	FOR (s.e)	TRUE% (s.e)	SIZE (s.e)	HAM (s.e)
			$p = 200, \rho_x = 0.2$		
mLASSO-AIC	0.565(0.01)	0(0)	1(1)	19.04(0.3)	11.04(0.3)
mENET-AIC	0.539(0.015)	0(0)	5(2.19)	18.57(0.382)	10.57(0.382)
mLASSO-AICc	0.18(0.014)	0(0)	24(4.292)	10.1(0.201)	2.1(0.201)
mENET-AICc	0.124(0.014)	0(0)	40(4.924)	9.42(0.19)	1.42(0.19)
mLASSO-BIC	0.01(0.003)	0(0)	92(2.727)	8.09(0.032)	0.09(0.032)
mENET-BIC	0.009(0.003)	0(0)	93(2.564)	8.08(0.031)	0.08(0.031)
mLASSO-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)
mENET-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)
			$p = 500, \rho_x = 0.2$		
mLASSO-AIC	0.594(0.004)	0(0)	0(0)	19.85(0.174)	11.85(0.174)
mENET-AIC	0.558(0.01)	0(0)	0(0)	18.78(0.306)	10.78(0.306)
mLASSO-AICc	0.252(0.018)	0(0)	20(4.02)	11.38(0.298)	3.38(0.298)
mENET-AICc	0.163(0.017)	0(0)	42(4.96)	10.06(0.253)	2.06(0.253)
mLASSO-BIC	0.007(0.003)	0(0)	94(2.387)	8.06(0.024)	0.06(0.024)
mENET-BIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)
mLASSO-CAIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)
mENET-CAIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)
			$p = 1000, \rho_x = 0.2$		
mLASSO-AIC	0.597(0.005)	0(0)	0(0)	20.06(0.188)	12.06(0.188)
mENET-AIC	0.573(0.011)	0(0)	1(1)	19.46(0.305)	11.46(0.305)
mLASSO-AICc	0.279(0.02)	0(0)	20(4.02)	11.99(0.347)	3.99(0.347)
mENET-AICc	0.19(0.02)	0(0)	37(4.852)	10.66(0.331)	2.66(0.331)
mLASSO-BIC	0.007(0.003)	0(0)	94(2.387)	8.06(0.024)	0.06(0.024)
mENET-BIC	0.006(0.002)	0(0)	95(2.19)	8.05(0.022)	0.05(0.022)
mLASSO-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)
mENET-CAIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)

Table B.3: Simulation study for AIC, AICc, BIC and CAIC of $\rho_e = 0.5$ and $\rho_x = 0.2$ based on 100 replications. Notation: FDR—false discovery rate; TRUE%—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Method	FDR (s.e)	FOR (s.e)	TRUE% (s.e)	SIZE (s.e)	HAM (s.e)
			$p = 200, \rho_x = 0.2$		
mLASSO-AIC	0.531(0.013)	0(0)	1(1)	18.07(0.374)	10.07(0.374)
mENET-AIC	0.454(0.02)	0(0)	10(3.015)	16.31(0.468)	8.31(0.468)
mLASSO-AICc	0.131(0.013)	0(0)	39(4.902)	9.47(0.177)	1.47(0.177)
mENET-AICc	0.108(0.013)	0(0)	49(5.024)	9.19(0.157)	1.19(0.157)
mLASSO-BIC	0.004(0.002)	0(0)	96(1.969)	8.04(0.02)	0.04(0.02)
mENET-BIC	0.004(0.002)	0(0)	96(1.969)	8.04(0.02)	0.04(0.02)
mLASSO-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)
mENET-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)
			$p = 500, \rho_x = 0.2$		
mLASSO-AIC	0.569(0.009)	0(0)	0(0)	19.14(0.284)	11.14(0.284)
mENET-AIC	0.542(0.014)	0(0)	2(1.407)	18.54(0.369)	10.54(0.369)
mLASSO-AICc	0.187(0.016)	0(0)	30(4.606)	10.28(0.233)	2.28(0.233)
mENET-AICc	0.121(0.014)	0(0)	47(5.016)	9.38(0.182)	1.38(0.182)
mLASSO-BIC	0.007(0.003)	0(0)	95(2.19)	8.07(0.033)	0.07(0.033)
mENET-BIC	0.003(0.002)	0(0)	97(1.714)	8.03(0.017)	0.03(0.017)
mLASSO-CAIC	0.001(0.001)	0(0)	99(1)	8.01(0.01)	0.01(0.01)
mENET-CAIC	0.002(0.002)	0(0)	98(1.407)	8.02(0.014)	0.02(0.014)
			$p = 1000, \rho_x = 0.2$		
mLASSO-AIC	0.576(0.009)	0(0)	0(0)	19.41(0.272)	11.41(0.272)
mENET-AIC	0.548(0.014)	0(0)	1(1)	18.85(0.381)	10.85(0.381)
mLASSO-AICc	0.23(0.016)	0(0)	21(4.094)	10.9(0.25)	2.9(0.25)
mENET-AICc	0.153(0.017)	0(0)	46(5.009)	9.91(0.24)	1.91(0.24)
mLASSO-BIC	0.006(0.002)	0(0)	95(2.19)	8.05(0.022)	0.05(0.022)
mENET-BIC	0.003(0.002)	0(0)	97(1.714)	8.03(0.017)	0.03(0.017)
mLASSO-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)
mENET-CAIC	0(0)	0(0)	100(0)	8(0)	0(0)

Table B.4: Simulation study for AIC, AICc, BIC and CAIC of $\rho_e = 0.2$ and $\rho_x = 0.8$ based on 100 replications. Notation: FDR—false discovery rate; TRUE%—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Method	FDR (s.e)	FOR (s.e)	TRUE% (s.e)	SIZE (s.e)	HAM (s.e)
			$p = 200, \rho_x = 0.8$		
mLASSO-AIC	0.408(0.02)	0(0)	6(2.387)	14.89(0.464)	7.03(0.456)
mENET-AIC	0.388(0.02)	0(0)	7(2.564)	14.4(0.462)	6.56(0.461)
mLASSO-AICc	0.149(0.016)	0(0)	37(4.852)	9.7(0.235)	1.88(0.236)
mENET-AICc	0.14(0.013)	0(0)	28(4.513)	9.45(0.185)	1.63(0.182)
mLASSO-BIC	0.038(0.008)	0.001(0)	63(4.852)	8.17(0.101)	0.59(0.096)
mENET-BIC	0.078(0.009)	0.001(0)	35(4.794)	8.54(0.107)	0.96(0.089)
mLASSO-CAIC	0.025(0.006)	0.001(0)	64(4.824)	7.96(0.084)	0.5(0.076)
mENET-CAIC	0.074(0.008)	0.001(0)	35(4.794)	8.46(0.109)	0.94(0.084)
			$p = 500, \rho_x = 0.8$		
mLASSO-AIC	0.515(0.015)	0(0)	1(1)	17.36(0.426)	9.7(0.412)
mENET-AIC	0.459(0.02)	0(0)	7(2.564)	16.29(0.492)	8.57(0.487)
mLASSO-AICc	0.151(0.016)	0(0)	31(4.648)	9.58(0.225)	1.98(0.22)
mENET-AICc	0.141(0.014)	0(0)	30(4.606)	9.42(0.187)	1.72(0.188)
mLASSO-BIC	0.04(0.008)	0.001(0)	58(4.96)	8.13(0.096)	0.65(0.091)
mENET-BIC	0.081(0.009)	0(0)	39(4.902)	8.55(0.106)	0.99(0.098)
mLASSO-CAIC	0.031(0.007)	0.001(0)	58(4.96)	7.95(0.093)	0.63(0.086)
mENET-CAIC	0.071(0.008)	0.001(0)	39(4.902)	8.41(0.105)	0.93(0.091)
			$p = 1000, \rho_x = 0.8$		
mLASSO-AIC	0.539(0.016)	0(0)	3(1.714)	17.97(0.385)	10.57(0.41)
mENET-AIC	0.498(0.019)	0(0)	4(1.969)	17.09(0.448)	9.51(0.464)
mLASSO-AICc	0.208(0.017)	0(0)	23(4.23)	10.24(0.273)	2.88(0.292)
mENET-AICc	0.185(0.016)	0(0)	25(4.352)	9.98(0.248)	2.44(0.238)
mLASSO-BIC	0.041(0.008)	0(0)	55(5)	8(0.096)	0.76(0.107)
mENET-BIC	0.086(0.009)	0(0)	35(4.794)	8.52(0.123)	1.12(0.111)
mLASSO-CAIC	0.033(0.007)	0(0)	58(4.96)	7.9(0.102)	0.72(0.108)
mENET-CAIC	0.075(0.008)	0(0)	35(4.794)	8.3(0.115)	1.08(0.102)

Table B.5: Simulation study for AIC, AICc, BIC and CAIC of $\rho_e = 0$ and $\rho_x = 0.8$ based on 100 replications. Notation: FDR—false discovery rate; TRUE%—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Method	FDR (s.e)	FOR (s.e)	TRUE% (s.e)	SIZE (s.e)	HAM (s.e)
			$p = 200, \rho_x = 0.8$		
mLASSO-AIC	0.43(0.018)	0(0)	4(1.969)	15.21(0.429)	7.35(0.418)
mENET-AIC	0.37(0.021)	0(0)	11(3.145)	14.11(0.472)	6.25(0.467)
mLASSO-AICc	0.138(0.015)	0.001(0)	37(4.852)	9.54(0.233)	1.74(0.227)
mENET-AICc	0.143(0.012)	0.001(0)	26(4.408)	9.45(0.168)	1.65(0.164)
mLASSO-BIC	0.042(0.008)	0.001(0)	61(4.902)	8.25(0.099)	0.59(0.096)
mENET-BIC	0.093(0.009)	0.001(0)	34(4.761)	8.76(0.111)	1.06(0.1)
mLASSO-CAIC	0.028(0.006)	0.001(0)	62(4.878)	8.02(0.082)	0.5(0.073)
mENET-CAIC	0.074(0.008)	0.001(0)	34(4.761)	8.46(0.103)	0.94(0.081)
			$p = 500, \rho_x = 0.8$		
mLASSO-AIC	0.492(0.018)	0(0)	2(1.407)	16.92(0.444)	9.24(0.44)
mENET-AIC	0.44(0.02)	0(0)	7(2.564)	15.63(0.47)	7.91(0.46)
mLASSO-AICc	0.16(0.015)	0(0)	28(4.513)	9.65(0.204)	2.03(0.202)
mENET-AICc	0.145(0.014)	0(0)	29(4.56)	9.45(0.187)	1.77(0.188)
mLASSO-BIC	0.055(0.01)	0(0)	56(4.989)	8.33(0.128)	0.81(0.117)
mENET-BIC	0.085(0.009)	0(0)	37(4.852)	8.61(0.109)	1.03(0.099)
mLASSO-CAIC	0.037(0.008)	0.001(0)	56(4.989)	8.05(0.111)	0.69(0.094)
mENET-CAIC	0.073(0.009)	0.001(0)	37(4.852)	8.41(0.108)	0.97(0.092)
			$p = 1000, \rho_x = 0.8$		
mLASSO-AIC	0.561(0.013)	0(0)	1(1)	18.45(0.345)	11.05(0.367)
mENET-AIC	0.51(0.017)	0(0)	2(1.407)	17.15(0.417)	9.61(0.429)
mLASSO-AICc	0.185(0.017)	0(0)	30(4.606)	9.92(0.265)	2.58(0.274)
mENET-AICc	0.176(0.016)	0(0)	26(4.408)	9.86(0.249)	2.34(0.249)
mLASSO-BIC	0.041(0.008)	0(0)	57(4.976)	8.01(0.099)	0.75(0.11)
mENET-BIC	0.087(0.009)	0(0)	36(4.824)	8.53(0.111)	1.11(0.104)
mLASSO-CAIC	0.031(0.007)	0(0)	59(4.943)	7.89(0.099)	0.69(0.106)
mENET-CAIC	0.079(0.008)	0(0)	36(4.824)	8.41(0.107)	1.05(0.097)

Table B.6: Simulation study for AIC, AICc, BIC and CAIC of $\rho_e = 0.5$ and $\rho_x = 0.8$ based on 100 replications. Notation: FDR—false discovery rate; TRUE%—percentage of the true model detected; SIZE—selected model size; HAM—Hamming distance; s.e.— standard error.

Method	FDR (s.e)	FOR (s.e)	TRUE% (s.e)	SIZE (s.e)	HAM (s.e)
			$p = 200, \rho_x = 0.8$		
mLASSO-AIC	0.407(0.02)	0(0)	7(2.564)	14.85(0.457)	6.99(0.448)
mENET-AIC	0.369(0.02)	0(0)	7(2.564)	13.95(0.454)	6.11(0.447)
mLASSO-AICc	0.13(0.015)	0.001(0)	38(4.878)	9.47(0.247)	1.69(0.241)
mENET-AICc	0.133(0.014)	0.001(0)	30(4.606)	9.46(0.227)	1.66(0.222)
mLASSO-BIC	0.038(0.009)	0.001(0)	64(4.824)	8.15(0.114)	0.63(0.106)
mENET-BIC	0.089(0.01)	0.001(0)	36(4.824)	8.73(0.132)	1.07(0.119)
mLASSO-CAIC	0.027(0.007)	0.001(0)	65(4.794)	7.98(0.09)	0.54(0.087)
mENET-CAIC	0.078(0.009)	0.001(0)	36(4.824)	8.5(0.112)	0.98(0.097)
			$p = 500, \rho_x = 0.8$		
mLASSO-AIC	0.454(0.018)	0(0)	1(1)	15.83(0.472)	8.19(0.459)
mENET-AIC	0.417(0.021)	0(0)	6(2.387)	15.07(0.484)	7.37(0.478)
mLASSO-AICc	0.148(0.016)	0(0)	29(4.56)	9.56(0.235)	1.96(0.229)
mENET-AICc	0.151(0.015)	0(0)	27(4.462)	9.61(0.225)	1.93(0.224)
mLASSO-BIC	0.047(0.009)	0.001(0)	56(4.989)	8.2(0.119)	0.76(0.113)
mENET-BIC	0.082(0.009)	0(0)	37(4.852)	8.57(0.107)	1.01(0.102)
mLASSO-CAIC	0.036(0.007)	0.001(0)	56(4.989)	8.01(0.1)	0.67(0.089)
mENET-CAIC	0.073(0.008)	0.001(0)	37(4.852)	8.43(0.105)	0.95(0.093)
			$p = 1000, \rho_x = 0.8$		
mLASSO-AIC	0.527(0.016)	0(0)	2(1.407)	17.58(0.405)	10.2(0.436)
mENET-AIC	0.495(0.018)	0(0)	3(1.714)	16.96(0.449)	9.38(0.466)
mLASSO-AICc	0.183(0.016)	0(0)	27(4.462)	9.78(0.228)	2.42(0.25)
mENET-AICc	0.178(0.015)	0(0)	21(4.094)	9.81(0.227)	2.29(0.224)
mLASSO-BIC	0.052(0.009)	0(0)	53(5.016)	8.13(0.12)	0.89(0.128)
mENET-BIC	0.092(0.01)	0(0)	33(4.726)	8.64(0.152)	1.22(0.138)
mLASSO-CAIC	0.044(0.009)	0(0)	57(4.976)	8(0.117)	0.84(0.132)
mENET-CAIC	0.082(0.009)	0(0)	33(4.726)	8.39(0.109)	1.11(0.105)

```r
# Y is the response matrix, X is the design matrix
# r is the index set of current model
# p represents dimension, i.e. number of predictors
# p_r is the index set that indexes are not in current model

X.r <- X[, r] # n by k submatrix of X
p_r <- setdiff(seq(1, p), r) # p-k vector
I_k <-  diag(1, k) # identity matrix

## Loop
log.s.plus1 <- rep(NA, length(p_r))
j <- 1
for (i in p_r) {
  rUi <- sort(c(r, i))  # a model in addition neighbor
  X.rUi <- X[, rUi]  # n by k+1 submatrix of X
  XtX <- crossprod(X.rUi) + 1/zeta * I_k1
  H.rUi <- I_n - X.rUi %*% solve(XtX) %*% t(X.rUi)
  # logarithm of Eq (3) for a model in additional neighbor
  log.s.Y.rUi <- -m * (k + 1)/2 * log(zeta) - m/2 * log(det(
     XtX)) - (n + v)/2 * log(det(t(Y) %*% H.rUi %*% Y + Psi))
  log.s.plus[j] <- log.s.Y.rUi
  j <- j + 1
}
```

Listing B.1: R code demo for computing $\mathbf{s}(\mathbf{Y}|\mathrm{nbd}_+(\hat{\boldsymbol{\gamma}}))$ with (3.3) using the for-loop

```r
# Y is the response matrix, X is the design matrix
# r is the index set of current model
# p represents dimension, i.e. number of predictors
# p_r is the index set that indexes are not in current model

## Proposed Method
X_r <- X[, p_r]  # n by p-k m sub-matrix of X
H.r <- I_n - X.r %*% solve(crossprod(X.r) + 1/zeta * I_k) %*%
    t(X.r)  # n by n matrix
d <- 1/zeta + colSums(H.r %*% X_r * X_r)  # p-k dimension
   vector
YHX_r <- t(Y) %*% H.r %*% X_r  # p-k by m matrix
YHY_1 <- solve(t(Y) %*% H.r %*% Y + Psi)  # m by m matrix
u <- colSums(YHY_1 %*% YHX_r * YHX_r)  # p-k dimension vector
# logarithm of (8)
log.s.plus.approx <- -m/2 * log(d) - (n + v)/2 * log(1 - u/d)
```

Listing B.2: R code demo for computing $\mathbf{s}(\mathbf{Y}|\mathrm{nbd}_+(\hat{\boldsymbol{\gamma}}))$ with (3.8)

Table B.7: Information of variables in Table 4.6

Abbrevation	Information
Population	population for community
PersPerFam	mean number of people per family
MedOwnCostPctInc	median owners cost as a percentage of household income
racePctHisp	percentage of population that is of hispanic heritage
NumUnderPov	number of people under the poverty level
HousVacant	number of vacant households
PctVacantBoarded	percent of vacant housing that is boarded up
NumKidsBornNeverMar	number of kids born to never married
pctWRetire	percentage of households with retirement income in 1989
NumInShelters	number of people in homeless shelters
nonViolPerPop	total number of non-violent crimes per 100K popuation
agePct16t24	percentage of population that is 16-24 in age