# Variable Selection for Competing Risks in High-Dimensional Covariate Spaces Without and With Missing Data

Dissertation

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of The Ohio State University

By

Guowei Li, MR

Graduate Program in Biostatistics

The Ohio State University

2023

Dissertation Committee:

Kellie J. Archer, Advisor

Jennifer A. Sinnott

Dongjun Chung

Guy Brock

Abstract

Competing risks data sometimes arise in the clinical setting when the primary event of interest competes with one or possibly several other events. The goal is to model the time to the primary event of interest, for example, death due to a specific cause, using available predictors. In gene expression studies, the number of genes often far exceeds the number of subjects, thus it is challenging to select a parsimonious set of features that predicts the outcome. Here, we propose a variable selection method based on the proportional subdistribution hazards model that maximizes the log-partial likelihood function, coupled with a non-convex penalty function. The smoothly clipped absolute deviation (SCAD), minimax concave penalty (MCP) and smooth integration of counting and absolute deviation (SICA) penalty functions are used for variable selection. Optimal tuning parameters are selected using cross-validation. Using simulation studies, we compare this method to the CoxBoost, fastcmprsk and randomForestSRC R packages and show that it works well in high-dimensional settings and generally selects slightly fewer true positives but much fewer false positives.

An issue that further complicates a high-dimensional competing risks model is that some predictors might have missing values. The most common way of dealing with missing data is multiple imputation. However, when applying the variable selection method to the imputed datasets, each imputed dataset may yield a different set of selected predictors. We extend our method to the scenario with missing data, where in the end one set of predictors is selected over multiple imputed datasets. Using simulation studies, we show that this method works well in high-dimensional settings and generally selects the important predictors and few unimportant predictors. We demonstrate these methods when modeling time-to-relapse for acute myeloid leukemia patients who have achieved complete remission using demographic, clinical, and genomic features.

Acknowledgments

I owe my greatest gratitute to my advisor, Dr. Kellie Archer. Dr. Archer is the kindest, most patient and willing-to-help advisor one can get. I could not possibly have completed this dissertation without her help.

I would like to thank Dr. Jennifer Sinnott, Dr. Dongjun Chung and Dr. Guy Brock for taking their time to serve on my committee and making valuable suggestions that improved this dissertation.

I would like to thank Yiran Zhang, Han Fu, Anna Seffernick, Shuai Sun, Yiwen Wang, Justice Ameyi and David Angeles for very helpful discussions.

I would like to thank Dr. Archer and the department of statistics for providing me with Graduate Associateship, without which I could not have completed this degree.

I would like to thank the admission committee that gave me the opportunity to study biostatistics at this great university and all the professors who have taught me, from whom I have learned a lot.

Last but not least, I would like to thank the chairs of graduate studies committee who have provided me with a lot of help and the Office of International Affairs for extending my I-20 so I can complete my degree.

## Vita

2013 . . . . . B.S. Mathematics and Applied Mathematics, Zhejiang University

2015 . . . . . M.R. Statistics, North Carolina State University

## Fields of Study

Major Field: Biostatistics

Table of Contents

List of Tables

List of Figures

# Chapter 1: Introduction and Literature Review

## 1.1 Motivation

This dissertation is motivated by the need to analyze clinical outcome of acute myeloid leukemia (AML) patients to enhance current prognostic risk stratification systems such as the European LeukemiaNet (ELN) [23]. Relevant clinical outcomes and data included are whether the patients achieved a complete remission, date of complete remission, whether the patients who achieved a complete remission relapsed, date of relapse, date of last follow-up, whether the patients were alive at the time of last follow-up, date of death, some demographic, clinical and cytogenetic variables, mutation statuses of known prognostic genes and expression of tens of thousands of transcripts from RNA-sequencing assays. Our research goal then is to identify covariates that have an important association with the time from complete remission to relapse for those who achieved complete remission. Not everyone who achieved complete remission relapsed; some patients were lost to follow-up and others died without relapse. If a patient dies without relapse, they can never relapse. In this case death without relapse is called a competing event because it precludes relapse from happening. Methods for modeling competing risks exist when the number of observations is greater than the number of covariates. However, methods are lacking when the number of observations is less than the number of covariates, such as when including mRNA ex-

pression values from high-throughput genomic assays. Thus, a statistical method that can select variables predictive of a time-to-event outcome with possible censoring and competing events is needed for high-dimensional covariate spaces. Because some values in the dataset are missing, imputation methods are also needed. Relevant literature regarding the proportional hazards model, competing risks models, variable selection methods for competing risks models, methods for imputing missing data and variable selection methods on multiply imputed data will be reviewed in this chapter.

## 1.2 Proportional Hazards Model in Survival Analysis

Cox (1972) proposed the famous proportional hazards model for survival analysis [19]. Let $n$ be the sample size, $T$ and $C$ be the failure and censoring times, respectively, $X = \min(T, C)$, $\Delta = I(T \leq C)$ and $\mathbf{z}$ be a length-$p$ covariate vector. Let $\lambda(t)$ be the hazard function. That is

$$\lambda(t) = \lim_{\Delta t \to 0+} \frac{P(t \leq T < t + \Delta t | t \leq T)}{\Delta t}.$$

The proportional hazards model assumes that

$$\lambda(t|\mathbf{z}) = \lambda_0(t) \exp(\mathbf{z}^T \boldsymbol{\beta}),$$

where $\boldsymbol{\beta}$ is a length-$p$ vector and $\lambda_0(t)$ is the baseline hazard function. The set of individuals at risk before time $t$ is called the risk set at time $t$ and denoted by $R(t)$. The log-partial likelihood is defined as

$$l(\boldsymbol{\beta}) = \sum_{\Delta_i = 1} \left\{ \mathbf{z}_i^T \boldsymbol{\beta} - \log \left[ \sum_{l \in R(t_i)} \exp(\mathbf{z}_l^T \boldsymbol{\beta}) \right] \right\}.$$

$\boldsymbol{\beta}$ represents the log hazard of the covariates and are obtained by maximizing $l(\boldsymbol{\beta})$.

## 1.3 Competing Risk Models

The setting for competing risk models is the same as for general survival analysis, and the only difference is that a subject may fail from any of $K$ causes. Let $\epsilon \in (1, ..., K)$ be the cause of failure.

### 1.3.1 Cause-specific Hazard Models

Holt (1978) proposed two models based on the cause-specific hazard function [45]. The cause-specific hazard function for cause $j$ is defined by

$$\lambda(t, j) = \lim_{\Delta t \to 0+} \frac{P(t \leq T < t + \Delta t, \epsilon = j | t \leq T)}{\Delta t}.$$

The first model is

$$\lambda(t, j|\mathbf{z}) = \lambda_0(t) \exp(\mathbf{z}^T \boldsymbol{\beta}_j)$$

assuming that the cause-specific hazards have the same shape. Inference about $\boldsymbol{\beta}$ is made from maximizing the marginal likelihood

$$L_1(\boldsymbol{\beta}) = \prod_{\Delta_i = 1} \exp(\mathbf{z}_i^T \boldsymbol{\beta}_{j(i)}) / \sum_{h \in R(t_i)} \sum_{j=1}^{K} \exp(\mathbf{z}_h^T \boldsymbol{\beta}_j),$$

where $j(i)$ indicates the cause of failure of the $i$-th person and $R(t)$ is the same as in the proportional hazards model. The second model is

$$\lambda(t, j|\mathbf{z}) = \lambda_{j0}(t) \exp(\mathbf{z}^T \boldsymbol{\beta}_j)$$

allowing the shape to depend on the cause of failure. Inference about $\boldsymbol{\beta}$ is made from maximizing the partial likelihood

$$L_2(\boldsymbol{\beta}) = \prod_{j=1}^{K} \prod_{i=1}^{k_j} \exp(\mathbf{z}_{j(i)}^T \boldsymbol{\beta}_j) / \sum_{h \in R(t_{j(i)})} \exp(\mathbf{z}_h^T \boldsymbol{\beta}_j),$$

where $t_{j(1)}, ..., t_{j(k_j)}$ are the $k_j$ ordered observed survival times by cause $j$.

In addition, Prentice et al. (1978) proposed two different models [79]. The first is an accelerated failure time model given by

$$\lambda(t, j) = \lambda_{j0}[t \exp(\mathbf{z}^T \boldsymbol{\beta}_j)] \exp(\mathbf{z}^T \boldsymbol{\beta}_j).$$

Because of the factorization of the likelihood, inference on a particular $\boldsymbol{\beta}_j$ can proceed by any of the single failure type procedures used to analyze models linear in the logarithm of failure time. Possibilities include parametric approaches based on exponential, Weibull or log-normal distribution or rank procedures based on generalized Wilcoxon or log-rank statistics or other generalized rank tests. A specialization of Holt's (1978) second model is given by

$$\lambda(t, j|\mathbf{z}) = \lambda_0(t) \exp(\gamma_j) \exp(\mathbf{z}^T \boldsymbol{\beta}_j).$$

The hazard functions are restricted to be proportional to each other with proportionality factor $\exp(\gamma_j)$ with, for uniqueness, $\gamma_1 = 0$. A partial likelihood can be written

$$\prod_{j=1}^{K} \prod_{i=1}^{k_j} \exp(\gamma_j + \mathbf{z}_{j(i)}^T \boldsymbol{\beta}_j) / \sum_{k=1}^{K} \sum_{h \in R(t_{j(i)})} \exp(\gamma_k + \mathbf{z}_h^T \boldsymbol{\beta}_k).$$

### 1.3.2 Exponential Model

Lagakos (1978) proposed an exponential model [58]. It assumes that there exist independent "potential" times until failure from causes $1, 2, ..., K$: $T^1, T^2, ..., T^K$ and $T = \min(T^1, T^2, ..., T^K)$. $T^j$ is assumed to be exponentially distributed with rate parameter $\lambda_j$. Given $\mathbf{z}$, assume that $\lambda_j(\mathbf{z}) = \exp(\alpha_j + \mathbf{z}^T\boldsymbol{\beta}_j)$. Let $\lambda_\cdot = \sum_{j=1}^K \lambda_j$. The likelihood contribution of observation $i$ is given by

$$\prod_{j=1}^K \lambda_j(\mathbf{z}_i)^{\Delta_i\epsilon_i=j} \exp[-\lambda_\cdot(\mathbf{z}_i)X_i].$$

Let $f_j$ be the number of failures from cause $j$. The likelihood function can be written as

$$L = \prod_{j=1}^K L_j,$$

where

$$\log(L_j) = \alpha_j f_j + [\sum_{i=1}^n \mathbf{z}_i I(\Delta_i\epsilon_i = j)]^T\boldsymbol{\beta}_j - \sum_{i=1}^n x_i \exp(\alpha_j + \mathbf{z}_i^T\boldsymbol{\beta}_j).$$

$L$ can be maximized by separately maximizing each $L_j$.

### 1.3.3 Parametric Mixture Model

Larson and Dinse (1985) proposed a parametric mixture model [61]. It assumes that

$$P_j(\mathbf{z}) = P(\epsilon = j|\mathbf{z}) = \frac{\exp(\mu_j + \mathbf{z}^T\boldsymbol{\pi}_j)}{\sum_{l=1}^K \exp(\mu_l + \mathbf{z}^T\boldsymbol{\pi}_l)}.$$

For uniqueness, $\mu_K$ is set to 0 and $\boldsymbol{\pi}_K$ is set to $\mathbf{0}$. Assume

$$Q_j(t|\mathbf{z}) = P(T > t|\mathbf{z}, \epsilon = j) = \exp[-\int_0^t h_j(x)\exp(\mathbf{z}^T\boldsymbol{\beta}_j)dx],$$

where $h_j(x)$ is the null hazard function for failure type $j$. $h_j(x)$ is assumed to be a step function:

$$h_j(x) = \exp(\alpha_{jm}) \quad \text{if } x \in I_m,$$

where $I_1, ..., I_M$ are $M$ (prespecified) mutually exclusive intervals that totally exhaust the non-negative real line. An individual who experiences a type $j$ failure at time $t$ contributes $h_j(t)\exp(\mathbf{z}^T\boldsymbol{\beta}_j)P_j(\mathbf{z})Q_j(t|\mathbf{z})$ to the likelihood. An individual who is censored at time $t$ contributes $\sum_{l=1}^{K} P_l(\mathbf{z})Q_l(t|\mathbf{z})$.

An EM algorithm was proposed to obtain the maximum likelihood solution by iteratively solving the simpler problem in which all censored observations are partially complete. The expectation (E) step of the algorithm involves creating a set of "pseudo-data" in which the uncensored observations are left intact and the unit mass associated with each censored observation is fractionated and assigned to $K$ partially complete pseudo-observations of the form $(\epsilon = j, T > t)$. Specifically the fractional mass assigned to this pseudo-observation is

$$W_j(t|\mathbf{z}) = P(\epsilon = j|\mathbf{z}, T > t) = \frac{P_j(\mathbf{z})Q_j(t|\mathbf{z})}{\sum_{l=1}^{K} P_l(\mathbf{z})Q_l(t|\mathbf{z})}.$$

For the $i$-th individual, let $\mathbf{g}_i$ be a vector with length $K$ with the $j$-th element $g_{ij} = I(\Delta_i\epsilon_i = j) + I(\Delta_i = 0)W_j(t_i|\mathbf{z}_i)$. The maximization (M) step of the algorithm involves calculating the parameter values that maximize the log-likelihood of the pseudo-data: $L(P) + \sum_{j=1}^{K} L_j(Q)$, where

$$L(P) = \sum_{i=1}^{n}\sum_{j=1}^{K} g_{ij}\log[P_j(\mathbf{z}_i)]$$

and

$$L_j(Q) = \sum_{i=1}^{n} I(\Delta_i\epsilon_i = j)\{\log[h_j(x_i)] + \mathbf{z}_i^T\boldsymbol{\beta}_j\} + g_{ij}\log[Q_j(x_i|\mathbf{z}_i)].$$

Each $g_{ij}$ is treated as a known constant and separate Newton-Raphson procedures are used to find the values of $\{(\mu_j, \boldsymbol{\pi}_j),\ j = 1, ..., K-1\}$ that maximize $L(P)$ and the values $\boldsymbol{\beta}_j$

and $\alpha_{jm},\ m = 1, ..., M$ that maximize $L_j(Q)$ for $j = 1, ..., K$. The EM algorithm is an iterative procedure that begins by choosing initial estimates of $P_j(\mathbf{z})$ and $Q_j(t|\mathbf{z})$ such as those obtained by ignoring the censored observations. At each subsequent iteration, the algorithm's E-step treats the current estimates of $P_j$ and $Q_j$ as known in order to update each estimate of $W_j(t|\mathbf{z})$ and thus the value of $g_{ij}$, and then the M-step treats the current $g_{ij}$ values as known and updates the estimates of $P_j$ and $Q_j$. Under suitable regularity conditions, these estimates of $P_j$ and $Q_j$ eventually converge to the true ML estimates. The convergence criteria can be based on relative changes in the parameter estimates or the log-likelihood of the pseudo-data.

Ng and McLachlan (2003) gave more details in the M-step of the EM algorithm [75]. Let $g_{ij}^{(s)}$ be the estimate of $g_{ij}$ after the $s$-th iteration,

$$Q_0 = \sum_{i=1}^{n}\sum_{j=1}^{K}[I(\Delta_i\epsilon_i = j) + I(\Delta_i = 0)g_{ij}^{(s)}]\log[P_j(\mathbf{z}_i)],$$
$$Q_j = \sum_{i=1}^{n} I(\Delta_i\epsilon_i = j)\log[h_j(t_i)\exp(\mathbf{z}_i^T\boldsymbol{\beta}_j)Q_j(t_i|\mathbf{z}_i)] + I(\Delta_i = 0)g_{ij}^{(s)}\log[Q_j(t_i|\mathbf{z}_i)]$$

for $j = 1, ..., K$.

$$L(P) + \sum_{j=1}^{K} L_j(Q) = \sum_{j=0}^{K} Q_j.$$

It implies that the estimates of $(\boldsymbol{\mu}_j, \boldsymbol{\pi}_j),\ j = 1, ..., K-1$ and $\boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_K$ can be updated separately by maximizing $Q_0$ and $Q_1, ..., Q_K$, respectively. Let $H_j(t) = \int_0^t h_j(u)du$. Then for $j = 1, ..., K$,

$$Q_j = \sum_{i=1}^{n} I(\Delta_i\epsilon_i = j)\{\log[h_j(t_i)] + \mathbf{z}_i^T\boldsymbol{\beta}_j\} - [I(\Delta_i\epsilon_i = j) + I(\Delta_i = 0)g_{ij}^{(s)}]H_j(x_i)\exp(\mathbf{z}_i^T\boldsymbol{\beta}_j).$$

The maximization of $Q_j$ for $j = 1, ..., K$ is implemented using a conditional approach, and the resulting algorithm can be viewed as an expectation-conditional maximization algorithm.

The M-step is replaced by two conditional maximization (CM) steps. The first involves the calculation of $H_j^{(s+1)}(t)$ by maximization of $Q_j$ with $\boldsymbol{\beta}_j$ fixed at $\boldsymbol{\beta}_j^{(s)}$. The second CM step calculates $\boldsymbol{\beta}_j^{(s+1)}$ by maximization of $Q_j$ with $H_j(t)$ fixed at $H_j^{(s+1)}(t)$. Now rearrange the failure time observations in increasing order and denote the $k_j$ distinct failure times due to the $j$-th cause by $t_{j(1)}, ..., t_{j(k_j)}$. By assuming a step function for $h_j(t)$ with discontinuities at each observed failure time due to the $j$-th cause and considering censored observations as censored at the preceding uncensored failure time, it can be shown that, for fixed $\boldsymbol{\beta}_j$, $Q_j$ is maximized with respect to $H_j(t)$ at

$$H_j^{(s+1)}(t_{j(m)}) = \sum_{i=1}^{m} \frac{d_{ij}}{\sum_{r \in R(t_{j(i)})} [I(\Delta_r \epsilon_r = j) + I(\Delta_r = 0) g_{rj}^{(s)}] \exp(\mathbf{z}_r^T \boldsymbol{\beta}_j)}$$

for $m = 1, ..., k_j$, where $d_{ij}$ is the number of failures due to cause $j$ at time $t_{j(i)}$ and $R(t_{j(i)})$ is the risk set at time $t_{j(i)}$. The solution to the second CM-step, however, does not exist in closed form. As the likelihood function usually has multiple maxima with mixture models, the ECM algorithm should be applied from different initial values to obtain the global maximum, which is usually taken to be the largest of the local maxima obtained.

Chang et al. (2007) also proposed an algorithm based on this model [13]. Assuming there are two competing risks, it considers the likelihood $L_n = \prod_i^n \bar{L}_{(i)}$, where

$$\begin{aligned}
\bar{L}_{(i)} =& \{P_1(\mathbf{z}_i)[H_1(x_i) - H_1(x_i-)] \exp[\mathbf{z}_i^T \boldsymbol{\beta}_1 - H_1(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_1)]\}^{I(\Delta_i \epsilon_i = 1)} \\
& \times \{P_2(\mathbf{z}_i)[H_2(x_i) - H_2(x_i-)] \exp[\mathbf{z}_i^T \boldsymbol{\beta}_2 - H_2(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_2)]\}^{I(\Delta_i \epsilon_i = 2)} \\
& \times \{P_1(\mathbf{z}_i) \exp[-H_1(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_1)] + P_2(\mathbf{z}_i) \exp[-H_2(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_2)]\}^{I(\Delta_i = 0)}.
\end{aligned}$$

$L_n$ is maximized to estimate the parameters. Only step functions are considered for $H_1$ and $H_2$. However, from the definition, $H_1$ and $H_2$ have to be continuous and cannot be step functions and $L_n$ would be equal to 0. This likelihood does not seem very reasonable.

Salesi et al. (2016) specified the parametric forms of $h_j(t)$ [82]. Suppose there are two competing risks. It assumes that given $\epsilon = 1$, $T$ follows the Weibull distribution and given $\epsilon = 2$, $T$ follows the Gompertz distribution. However, no explanation was given as to why $T$ would follow different distribution families for each cause of failure.

### 1.3.4 Semiparametric Mixture Model

Kuk (1992) generalized the parametric mixture model by allowing $h_j(x)$ to be arbitrary hazard functions. The full likelihood

$$L^* = [\prod_{j=1}^{K} \prod_{\Delta_i \epsilon_i = j} h_j(t_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_j) P_j(\mathbf{z}_i) Q_j(t_i|\mathbf{z}_i)] \prod_{\Delta_i = 0} [\sum_{j=1}^{K} P_j(\mathbf{z}_i) Q_j(t_i|\mathbf{z}_i)].$$

Consider a particular realization of causes of failure for censored observations $\boldsymbol{\epsilon}^c = \{\epsilon_i,\ \Delta_i = 0\}$. The likelihood based on this realization and the observed data is given by

$$L^*(\boldsymbol{\epsilon}^c) = [\prod_{j=1}^{K} \prod_{\Delta_i \epsilon_i = j} h_j(t_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_j) P_j(\mathbf{z}_i) Q_j(t_i|\mathbf{z}_i)] \prod_{j=1}^{K} \prod_{\Delta_i = 0} [P_j(\mathbf{z}_i) Q_j(t_i|\mathbf{z}_i)]^{I(\epsilon_i = j)}.$$

Let $n_c$ be the number of censored observations and $\Omega$ the $n_c$-fold Cartesian product of the set $\{1, ..., K\}$. It can be verified that

$$L^* = \sum_{\boldsymbol{\epsilon}^c \in \Omega} L^*(\boldsymbol{\epsilon}^c).$$

Rewrite $L^*(\boldsymbol{\epsilon}^c)$ as

$$L^*(\boldsymbol{\epsilon}^c) = P(\boldsymbol{\epsilon}) \prod_{j=1}^{K} \text{lik}_j(\mathbf{t}),$$

where

$$\text{lik}_j(\mathbf{t}) = \prod_{\Delta_i \epsilon_i = j} h_j(t_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_j) P_j(\mathbf{z}_i) Q_j(t_i|\mathbf{z}_i) \prod_{\Delta_i = 0} [Q_j(t_i|\mathbf{z}_i)]^{I(\epsilon_i = j)}.$$

The paper proposed to replace $\text{lik}_j(\mathbf{t})$ by the marginal likelihood $\text{lik}_j(r)$ based on all possible rank vectors consistent with the uncensored and censored observations from cause $j$. The rank likelihood $\text{lik}_j(r)$ is computed with respect to the conditional distribution of $T$ given $\epsilon = j$. Since the conditional distribution of failure time given failure type follows a proportional hazards model, the result of Kalbfleisch and Prentice (1973) is applicable [52]. Assume there are no ties among the uncensored observations. Let $t_{j(1)}, ..., t_{j(k_j)}$ denote the uncensored type $j$ failure times arranged in increasing order. Set $t_{j(0)} = 0$ and $t_{j(k_j+1)} = \infty$ and define $C_{jl} = \{i : \Delta_i = 0,\ t_{j(l)} \leq x_i < t_{j(l+1)}\},\ l = 0, ..., k_j$. Applying the result of [52], the paper obtains

$$\text{lik}_j(r) = \prod_{m=1}^{k_j} \frac{\exp(\mathbf{z}_{j(m)}^T \boldsymbol{\beta}_j)}{\sum_{l=m}^{k_j} [\exp(\mathbf{z}_{j(l)}^T \boldsymbol{\beta}_j) + \sum_{i \in C_{jl}} I(\epsilon_i = j) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_j)]}.$$

Substituting $\text{lik}_j(r)$ for $\text{lik}_j(\mathbf{t})$ in $L^*(\boldsymbol{\epsilon}^c)$ gives $L(\boldsymbol{\epsilon}^c) = P(\boldsymbol{\epsilon}) \prod_{j=1}^K \text{lik}_j(r)$ and $L = \sum_{\boldsymbol{\epsilon}^c \in \Omega} L(\boldsymbol{\epsilon}^c)$. The computation of $L$ is infeasible in practice so it is approximated by the Monte Carlo method

$$L = K^{n_c} \sum_{\boldsymbol{\epsilon}^c \in \Omega} K^{-n_c} L(\boldsymbol{\epsilon}^c) = K^{n_c} E[L(\boldsymbol{\epsilon}^c)],$$

where the expectation is taken with respect to the distribution that assigns equal probability to each $\boldsymbol{\epsilon}^c \in \Omega$. Let $\boldsymbol{\epsilon}_1^c, ..., \boldsymbol{\epsilon}_r^c$ be $r$ independent realizations of $\boldsymbol{\epsilon}^c$ from the above distribution. Then a Monte Carlo approximation of $L$ is $\tilde{L} = \frac{K^{n_c}}{r} \sum_{i=1}^r L(\boldsymbol{\epsilon}_i^c)$.

Escarela and Bowater (2008) proposed an EM algorithm based on a profile likelihood construction to fit this model [27]. Assuming that cause of failure $\epsilon$ is observed for censored observations, the complete likelihood

$$\begin{aligned} L_c &= \prod_{i=1}^n \{\prod_{j=1}^K [h_j(t_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_j) P_j(\mathbf{z}_i) Q_j(t|\mathbf{z}_i)]^{\Delta_i I(\epsilon_i = j)} \times [P_j(\mathbf{z}_i) Q_j(t_i|\mathbf{z}_i)]^{(1-\Delta_i) I(\epsilon_i = j)}\} \\ &= L_p \times L_S, \end{aligned}$$

where $L_p = \prod_{i=1}^{n} \prod_{j=1}^{K} P_j(\mathbf{z}_i)^{I(\epsilon_i=j)}$ and $L_S = \prod_{i=1}^{n} \prod_{j=1}^{K} [h_j(t_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_j)]^{\Delta_i I(\epsilon_i=j)} Q_j(t_i|\mathbf{z}_i)^{I(\epsilon_i=j)}$.

The E-step in the EM algorithm calculates the expectation of the logarithm of $L_c$ given current estimates of $h_j$, $\boldsymbol{\beta}_j$ and $P_j$ : $l_c = l_p + l_S$, where

$$l_p = \sum_{i=1}^{n} \sum_{j=1}^{K} g_{ij} \log[P_j(\mathbf{z}_i)]$$

and

$$l_S = \sum_{i=1}^{n} \sum_{j=1}^{K} \Delta_i I(\epsilon_i = j)\{\log[h_j(t_i)] + \mathbf{z}_i^T \boldsymbol{\beta}_j\} + g_{ij} \log[Q_j(t_i|\mathbf{z}_i)],$$

where $g_{ij} = \Delta_i I(\epsilon_i = j) + (1 - \Delta_i) \frac{P_j(\mathbf{z}_i) Q_j(t_i|\mathbf{z}_i)}{\sum_{l=1}^{K} P_l(\mathbf{z}_i) Q_l(t_i|\mathbf{z}_i)}$ is the expectation of $\epsilon_i$ given $P_j(\mathbf{z}_i)$ and $Q_j(t_i|\mathbf{z}_i)$. Let $t_{j(1)} < \cdots < t_{j(k_j)}$ denote the distinct uncensored failure times from cause $j$ and $R_{jl}$ denote the set of subjects at risk just prior to $t_{j(l)}$. $l_S$ is approximated by

$$\log \prod_{j=1}^{K} \prod_{l=1}^{k_j} \frac{\exp(\mathbf{z}_{j(l)}^T \boldsymbol{\beta}_j)}{\sum_{m \in R_{jl}} g_{mj} \exp(\mathbf{z}_m^T \boldsymbol{\beta}_j)}.$$

When tied failure times occur from the same cause, a possible approximation is

$$\log \prod_{j=1}^{K} \prod_{l=1}^{k_j} \frac{\exp[(\sum_{i \in D_{jl}} \mathbf{z}_i)^T \boldsymbol{\beta}_j]}{[\sum_{m \in R_{jl}} g_{mj} \exp(\mathbf{z}_m^T \boldsymbol{\beta}_j)]^{|D_{jl}|}},$$

where $D_{jl}$ denotes the set of tied uncensored failures from cause $j$ that occur at time $t_{j(l)}$.

The M-step involves maximizing the log-likelihood with respect to $\boldsymbol{\beta}$ given the current values for $g_{mj}$. Let $t_{j(0)} = 0$. The nonparametric product-limit estimator is adopted which specifies the conditional baseline survival distribution for cause $j$ as

$$Q_{0j}(t) = \prod_{m:\, t_{j(m)} \le t} \alpha_{jm},\ m = 1, ..., k_j,$$

where $\alpha_{jm} \geq 0$ and $\alpha_{j0} = 1$. This is a discontinuous function. But the assumption that $Q_j(t|\mathbf{z}) = \exp[-\int_0^t h_j(x)\exp(\mathbf{z}^T\boldsymbol{\beta}_j)dx]$ implies that $Q_{0j}(t) = \exp[-\int_0^t h_j(x)dx]$, which is a continuous function. This is contradictory to the assumption. The paper states that M-step involves maximizing the approximated $l_S$ but in fact it should be maximizing the approximated $l_p + l_S$. It is unclear what algorithm is used to perform the M-step.

Lu and Peng (2008) used estimating equations to estimate the parameters [69]. Suppose there are two competing risks. Let $Y_i(t) = I(X_i \geq t)$, $N_{ji}(t) = I(X_i \leq t, \Delta_i\epsilon_i = j)$ and $\tilde{\mathbf{z}} = (1, \mathbf{z})^T$. The estimating equations are

$$\begin{aligned}
\sum_{i=1}^{n}[dN_{ji}(t) - Y_i(t)W_j(t|\mathbf{z}_i)\exp(\mathbf{z}_i^T\boldsymbol{\beta}_j)h_j(t)dt] &= 0, \\
\sum_{i=1}^{n}\int_0^{\infty}\mathbf{z}_i[dN_{ji}(t) - Y_i(t)W_j(t|\mathbf{z}_i)\exp(\mathbf{z}_i^T\boldsymbol{\beta}_j)h_j(t)dt] &= \mathbf{0}, \\
\sum_{i=1}^{n}\tilde{\mathbf{z}}_i[I(\Delta_i\epsilon_i = 1) + I(\Delta_i = 0)W_1(X_i|\mathbf{z}_i) - P_1(\mathbf{z}_i)] &= \mathbf{0},
\end{aligned}$$

for $j = 1, 2$. However it is unclear what $dN_{ji}(t)$ means because it is not differentiable at $T_i$ if $\Delta_i\epsilon_i = j$.

### 1.3.5 Proportional Subdistribution Hazards Models

Fine and Gray (1999) proposed a proportional subdistribution hazards model, which has been widely used for modeling competing risks data [32]. The interest is modeling the cumulative incidence function (CIF) for failure from cause $j$ conditional on the covariates,

$$F_j(t|\mathbf{z}) = P(T \leq t, \epsilon = j|\mathbf{z}).$$

The subdistribution hazard, originally described by Gray (1988) [42], is defined as

$$\begin{aligned}\lambda_j(t|\mathbf{z}) &= \lim_{\Delta t \to 0+} \frac{1}{\Delta t} P[t \leq T < t + \Delta t, \epsilon = j | T \geq t \cup (T < t \cap \epsilon \neq j), \mathbf{z}] \\ &= \frac{F_j'(t|\mathbf{z})}{1 - F_j(t|\mathbf{z})} \\ &= -d \log[1 - F_j(t|\mathbf{z})]/dt\end{aligned}$$

The proportional subdistribution hazards model assumes that

$$\lambda_j(t|\mathbf{z}) = \lambda_{j0}(t) \exp(\mathbf{z}^T \boldsymbol{\beta}),$$

where $\lambda_{j0}(t)$ is an unspecified function and $\boldsymbol{\beta}$ is the parameter vector. Then

$$F_j(t|\mathbf{z}) = 1 - \exp[-\int_0^t \lambda_{j0}(s) \exp(\mathbf{z}^T \boldsymbol{\beta}) ds].$$

$\boldsymbol{\beta}$ is estimated by solving

$$\mathbf{U}(\boldsymbol{\beta}) := \sum_{i=1}^{n} \int_0^{\infty} \left[ \mathbf{z}_i - \frac{\sum_{k=1}^{n} w_k(s) Y_k(s) \mathbf{z}_k \exp(\mathbf{z}_k^T \boldsymbol{\beta})}{\sum_{k=1}^{n} w_k(s) Y_k(s) \exp(\mathbf{z}_k^T \boldsymbol{\beta})} \right] w_i(s) dN_i(s) = \mathbf{0}, \tag{1.1}$$

where $N_i(t) = I(T_i \leq t, \epsilon_i = j)$, $Y_i(t) = 1 - N_i(t-)$, $w_i(t) = r_i(t)\hat{G}(t)/\hat{G}(X_i \wedge t)$ is the weight associated with individual $i$, $r_i(t) = I(C_i \geq T_i \wedge t)$ denotes knowledge of vital status on individual $i$ at time $t$ and $\hat{G}$ is the Kaplan-Meier estimate of the survival function of $C$. $\mathbf{U}(\boldsymbol{\beta})$ is the gradient of a well-behaved objective function. The solution is calculated by using a modified Newton algorithm to maximize the objective function.

He et al. (2016) studied this model with adjustments for covariate-dependent censoring [43]. The only difference is that He et al. (2016) estimated the distribution of the censoring

time $C$ using Cox proportional hazards model

$$\lambda_C(t|\mathbf{z}) = \lambda_{C0}(t)\exp(\mathbf{z}^T\boldsymbol{\gamma}).$$

Let $\hat{\boldsymbol{\gamma}}$ be the maximum partial likelihood estimate of $\boldsymbol{\gamma}$. $\Lambda_{C0}(t) = \int_0^t \lambda_{C0}(u)du$. Let $\hat{\Lambda}_{C0}(t)$ be the Breslow estimator of $\Lambda_{C0}(t)$. Then the survival function of $C$ is estimated by

$$\hat{G}(t|\mathbf{z}) = \exp[-\hat{\Lambda}_{C0}(t)\exp(\mathbf{z}^T\hat{\boldsymbol{\gamma}})].$$

Then the weight associated with individual $i$ becomes

$$\hat{w}_i(t) = \frac{r_i(t)\hat{G}(t|\mathbf{z}_i)}{\hat{G}(X_i \wedge t|\mathbf{z}_i)}.$$

This weight is plugged in the estimating equation $\mathbf{U}(\boldsymbol{\beta}) = \mathbf{0}$ to estimate $\boldsymbol{\beta}$.

### 1.3.6 Semiparametric Transformation Models

Fine (2001) proposed a semiparametric transformation model for the crude probabilities [31]. Let $F_j(t|\mathbf{z}) = P(T \leq t, \epsilon = j|\mathbf{z})$. Assuming the existence of a known, differentiable function $g(\cdot)$, the model is

$$g[F_j(t|\mathbf{z})] = h(t) - \mathbf{z}^T\boldsymbol{\beta},$$

where $h(t)$ is unspecified, invertible and strictly increasing in $t$. Define the improper variable $Y = T \times I(\epsilon = 1) + \infty \times I(\epsilon \neq 1)$. Suppose

$$h(Y) = \mathbf{z}^T\boldsymbol{\beta} + v,$$

where $v$ is a continuous variable with $h(\infty) < \infty$. But this is impossible because $P(Y = \infty) > 0$ so $P(v = h(\infty) - \mathbf{z}^T\boldsymbol{\beta}) > 0$. But $P(v = h(\infty) - \mathbf{z}^T\boldsymbol{\beta})$ must be 0 since $v$ is a continuous variable.

Similarly, Choi et al. (2021) assumes in [18] that

$$\log(Y) = \mathbf{z}^T\boldsymbol{\beta} + v.$$

But it also suffers from the same issue.

Mao and Lin (2017) presented a class of semiparametric transformation models [71]. Let $g_j$ be a known increasing function and $Q_j$ be an arbitrary increasing function. Assume

$$g_j[F_j(t|\mathbf{z})] = Q_j(t) + \mathbf{z}^T\boldsymbol{\beta}_j.$$

The same likelihood function in Jeong and Fine (2007) [51] is used.

### 1.3.7 Regression Modeling Based on Pseudovalues of the Cumulative Incidence Function

Klein and Anderson (2005) proposed a method based on the pseudovalues from a jackknife statistic constructed from the cumulative incidence curve [55]. Define $N_j(t)$ as number of individuals who have experienced a type $j$ event prior to time $t$ and let $Y(t)$ be the number at risk at time $t$. The cumulative incidence function for cause $j$, $F_j(t)$ is estimated by

$$\hat{F}_j(t) = \int_0^t \prod_{X_i<u} \left[1 - \frac{\sum_{h=1}^K dN_h(X_i)}{Y(X_i)}\right] \frac{dN_j(u)}{Y(u)}.$$

It is unclear if $N_j(t)$ is defined as $\sum_{i=1}^n I(T_i \leq t, \epsilon_i = j)$ or $\sum_{i=1}^n I(X_i \leq t, \Delta_i\epsilon_i = j)$. It would not be calculable if defined the former way. Let $\hat{F}_j^{(i)}(t)$ be the estimated cumulative

incidence function based on the sample obtained by deleting the $i$-th observation. For a grid of points $\tau_1, ..., \tau_M$, define the pseudovalue for the $i$-th subject at time $\tau_h$ as

$$\hat{\theta}_{ih} = n\hat{F}_j(\tau_h) - (n-1)\hat{F}_j^{(i)}(\tau_h),\ i = 1, ..., n,\ h = 1, ..., M.$$

Let $g(\cdot)$ be a link function. Assume a generalized linear model with

$$g(\theta_{ih}) = \alpha_h + \mathbf{z}_i^T\boldsymbol{\gamma} := \mathbf{z}_{ih}^T\boldsymbol{\beta},\ i = 1, ..., n,\ h = 1, ..., M.$$

Define the inverse link by

$$\theta_{ih} = g^{-1}(\mathbf{z}_{ih}^T\boldsymbol{\beta}) := \mu(\mathbf{z}_{ih}^T\boldsymbol{\beta}).$$

Let $\hat{\boldsymbol{\theta}}_i = (\hat{\theta}_{i1}, ..., \hat{\theta}_{iM})$ and $\mathbf{f}_{ij} = (F_j(\tau_1|\mathbf{z}_i), ..., F_j(\tau_M|\mathbf{z}_i))$. Let $d\mu_i(\boldsymbol{\theta})$ be the $(M+p) \times M$ matrix of partial derivatives of $(\mu(\mathbf{z}_{i1}^T\boldsymbol{\beta}), ..., \mu(\mathbf{z}_{iM}^T\boldsymbol{\beta}))^T$ with respect to the parameters. Let $\mathbf{V}_i(\boldsymbol{\beta})$ be a working covariance matrix. The estimating equations to be solved are

$$\sum_{i=1}^{n} d\mu_i(\boldsymbol{\theta})\mathbf{V}_i^{-1}(\boldsymbol{\beta})(\hat{\boldsymbol{\theta}}_i - \mathbf{f}_{ij}) = \mathbf{0}.$$

But it is unclear what $F_j(\tau_h|\mathbf{z}_i),\ h = 1, ..., M$ mean since $F_j$ is what needs to be modeled.

### 1.3.8 Additive Models

Klein (2006) proposed two additive models for the hazard rates or the cumulative incidence functions [54]. The cause-specific hazard function for cause $j$

$$\lambda(t, j) = \lim_{\Delta t \to 0+} \frac{P(t \le T < t + \Delta t, \epsilon = j | t \le T)}{\Delta t}.$$

The first model assumes that

$$\lambda(t, j) = \alpha_0(t) + \mathbf{z}^T\boldsymbol{\beta}.$$

Let $Y_i(t) = I(T_i \geq t),\ N_i(t) = I(X_i \leq t, \epsilon_i = j)$ and $\bar{\mathbf{z}}(t) = \frac{\sum_{i=1}^n Y_i(t)\mathbf{z}_i}{\sum_{i=1}^n Y_i(t)}$. The estimator of $\boldsymbol{\beta}$ is given by

$$\hat{\boldsymbol{\beta}} = \frac{\sum_{i=1}^n \int_0^\infty \mathbf{z}_i - \bar{\mathbf{z}}(t)dN_i(t)}{\sum_{i=1}^n \int_0^\infty Y_i(t)[\mathbf{z}_i - \bar{\mathbf{z}}(t)]^T[\mathbf{z}_i - \bar{\mathbf{z}}(t)]dt}.$$

However this estimator is not calculable if some observations are censored since $Y_i(t)$ cannot be calculated.

The cumulative incidence function for cause $j$ is

$$F_j(t|\mathbf{z}) = P(T \leq t, \epsilon = j|\mathbf{z}).$$

Set a grid of time points, $\tau_1, ..., \tau_M$. The second model assumes that

$$F_j(\tau_h|\mathbf{z}) = F_{j0}(\tau_h|\mathbf{z}) + \mathbf{z}^T\boldsymbol{\alpha}.$$

Let $\gamma_h = F_{j0}(\tau_h|\mathbf{z}),\ \boldsymbol{\beta} = (\gamma_1, ..., \gamma_M, \alpha_1, ..., \alpha_p)^T,\ \theta_{ih} = \gamma_h + \mathbf{z}_i^T\boldsymbol{\alpha}$ and $\boldsymbol{\theta}_i = (\theta_{i1}, ..., \theta_{iM})^T$. Let $\hat{F}_j(t)$ be the estimated cumulative incidence function and $\hat{F}_j^{(i)}(t)$ be the estimated cumulative incidence function based on the sample with the $i$-th observation removed. Let $\hat{\theta}_{ih} = n\hat{F}_j(\tau_h) - (n-1)\hat{F}_j^{(i)}(\tau_h)$. An estimator of $\boldsymbol{\beta}$ is the solution to

$$\sum_{i=1}^n \left(\frac{d\boldsymbol{\theta}_i}{d\boldsymbol{\beta}}\right)^T \mathbf{V}_i^{-1}(\hat{\boldsymbol{\theta}}_i - \boldsymbol{\theta}_i) = \mathbf{0},$$

where $\mathbf{V}_i$ is a working covariance matrix for $\hat{\boldsymbol{\theta}}_i$.

Li, Xue and Long (2017) proposed an additive hazard model for the subdistribution [66].

Let the subdistribution hazard be $\lambda_j(t|\mathbf{z})$. It assumes that

$$\lambda_j(t|\mathbf{z}) = \lambda_{j0}(t|\mathbf{z}) + \mathbf{z}^T\boldsymbol{\beta}.$$

Let $N_i(t) = I(T_i \leq t, \epsilon_i = 1)$, $Y_i(t) = 1 - N_i(t-)$, $r_i(t) = I(C_i \geq T_i \wedge t)$. Let $\hat{G}$ be the Kaplan-Meier estimate of the survival function of the censoring time $C$. Define weight $w_i(t) = \frac{r_i(t)\hat{G}(t)}{\hat{G}(T_i \wedge t)}$. Denote $\tau$ as the maximum follow-up time such that $P(T \geq \tau) > 0$. Let $\bar{\mathbf{z}}(t) = \frac{\sum_{i=1}^n w_i(t)Y_i(t)\mathbf{z}_i}{\sum_{i=1}^n w_i(t)Y_i(t)}$. $\boldsymbol{\beta}$ is estimated by solving

$$\sum_{i=1}^n \int_0^\tau w_i(t)[\mathbf{z}_i - \bar{\mathbf{z}}(t)][dN_i(t) - Y_i(t)\mathbf{z}_i^T\boldsymbol{\beta}] = 0.$$

However, the left-hand side does not seem to be a properly defined integral.

### 1.3.9 Parametric Regression Analysis of Cumulative Incidence Function

Jeong and Fine (2007) proposed parametric regression analysis of cumulative incidence function [51]. For cause-$j$, let $F_j(t|\mathbf{z}) = P(T \leq t, \epsilon = j|\mathbf{z})$. The paper considered

$$g_j[F_j(t|\mathbf{z})] = u_j(t) + \mathbf{z}^T\boldsymbol{\beta}_j,\ j = 1, ..., K,$$

where $g_j(v) = \log\{[(1-v)^{-\alpha_j} - 1]/\alpha_j\}$. The paper said that $-\infty < \alpha_j < \infty$. But $g_j(v)$ cannot be defined when $\alpha_j = 0$. Then the paper claimed that

$$F_j(t|\mathbf{z}) = 1 - [1 + \alpha_j \exp(\mathbf{z}^T\boldsymbol{\beta}_j)u_j(t)]^{-1/\alpha_j}.$$

But actually $F_j(t|\mathbf{z})$ should be equal to $1-\{1+\alpha_j \exp[\mathbf{z}^T\boldsymbol{\beta}_j + u_j(t)]\}^{-1/\alpha_j}$. Then the paper let $u_j(t) = \tau[\exp(\rho t)-1]/\rho$ with $\tau > 0$ and

$$F_j(t|\mathbf{z}) = 1-[1+\alpha_j \exp(\mathbf{z}^T\boldsymbol{\beta}_j)u_j(t)]^{-1/\alpha_j}.$$

However, given this form, when $\alpha_j < 0,\ 1+\alpha_j \exp(\mathbf{z}^T\boldsymbol{\beta}_j)u_j(t)$ could become negative and $[1+\alpha_j \exp(\mathbf{z}^T\boldsymbol{\beta}_j)u_j(t)]^{-1/\alpha_j}$ may not be defined. The authors state that the likelihood function is given by

$$\prod_{i=1}^{n}\{[\prod_{k=1}^{2} F_k'(x_i|\mathbf{z}_i)^{I(\Delta_i\epsilon_i=k)}][1-\sum_{k=1}^{2} F_k(x_i|\mathbf{z}_i)]^{I(\Delta_i=0)}\}$$

though it is not clear why the likelihood function would take this form.

### 1.3.10 Competing Risks Quantile Regression

Peng and Fine (2009) proposed a competing risks quantile regression [78]. For cause-$j$, let $F_j(t|\mathbf{z}) = P(T \le t, \epsilon = j|\mathbf{z})$. Define the conditional quantile $Q_j(\tau|\mathbf{z}) = \inf\{t:\ F_j(t|\mathbf{z}) \ge \tau\}$. Let $\tilde{\mathbf{z}} = (1, \mathbf{z}^T)^T$. Suppose $g(\cdot)$ is a known monotone link function and $0 < \tau_L \le \tau_U < 1$, for $\tau \in [\tau_L, \tau_U]$, assume

$$Q_j(\tau|\mathbf{z}) = g[\tilde{\mathbf{z}}^T\boldsymbol{\beta}_0(\tau)].$$

Let $G(t|\mathbf{z}) = P(C \ge t|\mathbf{z})$ and $\hat{G}(t|\mathbf{z})$ be the Kaplan–Meier estimator. The paper suggested the estimating equation:

$$\sum_{i=1}^{n}\tilde{\mathbf{z}}_i\left(\frac{I\{X_i \le g[\tilde{\mathbf{z}}_i^T\boldsymbol{\beta}_0(\tau)]\}I(\Delta_i\epsilon_i=j)}{\hat{G}(X_i|\mathbf{z}_i)} - \tau\right) = \mathbf{0}.$$

However, this estimating equation may not have a solution. Let $M$ be an extremely large positive number selected to bound $|\mathbf{b}^T\sum_{i=1}^{n}\frac{\tilde{\mathbf{z}}_i I(\Delta_i\epsilon_i=j)}{\hat{G}(X_i)}|$ from above for all $\mathbf{b}$ in the compact

parameter space for $\boldsymbol{\beta}_0(\tau)$. The paper claimed that this equation can be reformulated as locating the minimizer of

$$U(\mathbf{b},\tau)=\sum_{i=1}^{n} I(\Delta_i\epsilon_i=j)\left|\frac{g^{-1}(X_i)}{\hat{G}(X_i)}-\mathbf{b}^T\frac{\tilde{\mathbf{z}}_i}{\hat{G}(X_i)}\right|+\left|M+\mathbf{b}^T\sum_{i=1}^{n}\frac{\tilde{\mathbf{z}}_i I(\Delta_i\epsilon_i=j)}{\hat{G}(X_i)}\right|+|M-\mathbf{b}^T\sum_{i=1}^{n}2\tilde{\mathbf{z}}_i\tau|.$$

However, the paper did not discuss what the compact parameter space for $\boldsymbol{\beta}_0(\tau)$ would be.

### 1.3.11 Absolute Risk Regression

Gerds, Scheike and Andersen (2012) proposed absolute risk regression [37]. It models the cumulative incidence function $F_j(t|\mathbf{z})=F_{j0}(t)\exp(\mathbf{z}^T\boldsymbol{\beta})$. However, $F_j(t|\mathbf{z})$ could exceed 1 with extreme values of $\mathbf{z}^T\boldsymbol{\beta}$ after some value of $t$.

### 1.3.12 Fully Specified Subdistribution Model

Ge and Chen (2012) proposed a fully specified subdistribution model [36]. Two competing risks are considered. Let $T_j$ be the time to failure due to cause $j$ for $j=1,2$ and $T=\min(T_1,T_2)$. It assumes that

$$P(T_1\le t,\epsilon=1)=1-\exp[-H_{10}(t)\exp(\mathbf{z}^T\boldsymbol{\beta}_1)]$$

and

$$P(T_2\le t|\epsilon=2)=1-\exp[-H_{20}(t)\exp(\mathbf{z}^T\boldsymbol{\beta}_2)].$$

Then it claims that the likelihood function is

$$
\begin{aligned}
& L(\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, h_{10}, h_{20} | \mathbf{x}, \mathbf{Z}, \boldsymbol{\Delta\epsilon}) \\
= & \prod_{i=1}^{n} \{h_{10}(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_1) \exp[-H_{10}(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_1)]\}^{I(\Delta_i \epsilon_i = 1)} \\
& \times \{h_{20}(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_2) \exp[-H_{20}(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_2) - H_{10}(\infty) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_1)]\}^{I(\Delta_i \epsilon_i = 2)} \\
& \times (\exp[-H_{10}(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_1)] \\
& - \{1 - \exp[-H_{20}(x_i) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_2)]\} \times \exp[-H_{10}(\infty) \exp(\mathbf{z}_i^T \boldsymbol{\beta}_1)])^{I(\Delta_i = 0)}.
\end{aligned}
$$

This is the same likelihood function as in Jeong and Fine (2007) [51].

### 1.3.13 Constrained Parametric Model for Simultaneous Inference of Two Cumulative Incidence Functions

Shi, Cheng and Jeong (2013) proposed a parametric regression model for the cumulative incidence functions [84]. Assume there are two possible causes of failure and cause 1 is of primary interest. It assumes that the cumulative incidence function for cause 1

$$
F_1(t|\mathbf{z}) = 1 - \left\{ 1 - \frac{p_1 \exp[b_1(t - c_1)] - p_1 \exp(-b_1 c_1)}{1 + \exp[b_1(t - c_1)]} \right\}^{\exp(\mathbf{z}^T \boldsymbol{\beta}_1)}
$$

and

$$
F_2(t|\mathbf{z}) = \frac{(1 - p_1)^{\exp(\mathbf{z}^T \boldsymbol{\beta}_1)} \{\exp[b_2(t - c_2)] - \exp(-b_2 c_2)\}}{1 + \exp[b_2(t - c_2)]}.
$$

The same likelihood function in Jeong and Fine (2007) [51] is used.

### 1.3.14 Semiparametric Mixture Component Models

Choi and Huang (2014) considered semiparametric analysis of mixture component models on cumulative incidence functions [17]. Assuming there are two competing risks, the paper assumes that

$$P(\epsilon = 1|\mathbf{z}) = \frac{\exp(\mathbf{z}^T\boldsymbol{\gamma})}{1 + \exp(\mathbf{z}^T\boldsymbol{\gamma})}.$$

Let $T_k^*$ be the latent event time for cause $k$. Define $\Lambda_k(t) = -\log[P(X \geq t|\epsilon = k, \mathbf{z})]$. Let $G(t) = \frac{\log(1+rt)}{r}$ for $r > 0$ and $G(t) = t$ for $r = 0$. Let $A_k$ be an increasing but unspecified function with $A_k(0) = 0$. The model posited for the failure times is

$$\Lambda_k(t) = G[\int_0^t I(T_k^* \geq s) \exp(\mathbf{z}^T\boldsymbol{\beta}_k) dA_k(s)].$$

But $\Lambda_k(t)$ is a function of $t$ while the right hand side of the equation is a stochastic process that not only depends on $t$ but also on the random variable $T_k^*$. This equation cannot hold since a deterministic function cannot be equal to a stochastic process.

### 1.3.15 Proportional Odds Cumulative Incidence Model

Eriksson et al. (2015) suggested an estimator for the proportional odds cumulative incidence model [26]. Let $F_j(t) = P(T \leq t, \epsilon = j)$. Consider

$$\text{logit}[F_j(t|\mathbf{z})] = \log[H(t)] + \mathbf{z}^T\boldsymbol{\beta},$$

where $H(t)$ is an increasing positive function with $H(0) = 0$. Define $N_i(t) = I(T_i \leq t, \epsilon_i = j)$, $Y_i(t) = 1 - N_i(t-)$, $r_i(t) = I(C_i \geq T_i \wedge t)$. Let $G$ be the Kaplan–Meier estimator of survival function for the censoring time $C$. Define $w_i(t, G) = \frac{r_i(t)G(t-)}{G[(X_i \wedge t)-]}$. Let $\tau$ denote a finite

maximum follow-up time. Consider the estimating equations

$$\sum_{i=1}^{n} \mathbf{z}_i \int_0^{\tau} w_i(t,G) \left[ dN_i(t) - \frac{Y_i(t)dH(t)}{\exp(-\mathbf{z}_i^T\boldsymbol{\beta}) + H(t-)} \right] = \mathbf{0},$$

and

$$\sum_{i=1}^{n} w_i(t,G) \left[ dN_i(t) - \frac{Y_i(t)dH(t)}{\exp(-\mathbf{z}_i^T\boldsymbol{\beta}) + H(t-)} \right] = 0,\ t \in [0,\tau].$$

$H$ is estimated by a non-decreasing function with jumps only at observed cause-$j$ event times. The estimating equation is solved by a Fisher scoring algorithm.

### 1.3.16 Flexible Parametric Modelling of Cause-specific Hazards

Hinchliffe and Lambert (2013) advocated the use of the flexible parametric survival model to model the cause-specific hazards [44]. A restricted cubic spline function, $s(\log(t)|\boldsymbol{\gamma}, \mathbf{n_0})$ with $N$ knots, a vector of knots $\mathbf{n_0}$ and parameters $\gamma_0, ..., \gamma_{N-1}$ can be written as

$$s(\log(t)|\boldsymbol{\gamma}, \mathbf{n_0}) = \gamma_0 + \gamma_1 y_1 + ... + \gamma_{N-1} y_{N-1}.$$

Let $\phi_j = \frac{n_N - n_j}{n_N - n_1}$. The derived variables $y_1, ..., y_{N-1}$ are calculated as follows

$$y_1 = \log(t)$$
$$y_j = [\log(t) - n_j]_+^3 - \phi_j[\log(t) - n_1]_+^3 - (1-\phi_j)[\log(t) - n_N]_+^3,\ j = 2, ..., N-1.$$

Let $h_j(t) = \lim_{\Delta t \to 0+} \frac{P(t \le T < t+\Delta t, \epsilon = j | T \ge t)}{\Delta t}$ and $H_j(t) = \int_0^t h_j(u)du$. The paper assumes that

$$\log[H_j(t)] = s(\log(t)|\boldsymbol{\gamma}, \mathbf{n_0}) + \mathbf{z}^T\boldsymbol{\beta}.$$

### 1.3.17 Flexible Parametric Modelling of the Cause-specific Cumulative Incidence Function

Lambert et al. (2017) proposed the use of flexible parametric survival models to model the cause-specific CIF [60]. Let the subdistribution hazard function for cause $j$ be

$$h_j(t|\mathbf{z}) = \lim_{\Delta t \to 0+} \frac{P(t \leq T < t + \Delta t, \epsilon = j | T \geq t \cup (T < t \cap \epsilon \neq j), \mathbf{z})}{\Delta t}.$$

Let $H_j(t|\mathbf{z}) = \int_0^t h_j(u|\mathbf{z})du$. Like in Hinchliffe and Lambert (2013) [44], the paper assumes that

$$\log[H_j(t)] = s(\log(t)|\boldsymbol{\gamma}, \mathbf{n_0}) + \mathbf{z}^T\boldsymbol{\beta},$$

where $s(\log(t)|\boldsymbol{\gamma}, \mathbf{n_0})$ is the restricted cubic spline function. In fact the paper wrote the model as

$$\log[H_j(t)] = \log[s(\log(t)|\boldsymbol{\gamma}, \mathbf{n_0})] + \mathbf{z}^T\boldsymbol{\beta},$$

which is perhaps a typo, as $s(\log(t)|\boldsymbol{\gamma}, \mathbf{n_0})$ could be negative. The estimate of the censoring distribution is obtained by fitting a flexible parametric model where being censored is considered as the event. Let $G(t) = P(C \geq t)$. This gives a parametric expression for $G(t)$. The time-dependent weights $w_i(t) = I(t \leq x_i) + \frac{G(t)}{G(x_i)} I(t > x_i)$. Suppose there are two competing risks and the first one is of interest. The contribution of the $i$-th subject to the likelihood is defined as

$$\log(L_i) = I(\Delta_i\epsilon_i = 1)\log[h_1(x_i|\mathbf{z}_i)] - [1 - I(\Delta_i\epsilon_i = 2)]H_1(x_i|\mathbf{z}_i) - I(\Delta_i\epsilon_i = 2)\int_0^\tau w_i(u)h_1(u)du,$$

where $\tau$ the maximum observed follow-up time. But this does not seem like a reasonable likelihood. For example, if the $i$-th subject is censored, $\log(L_i)$ would be $-H_1(x_i|\mathbf{z}_i)$, then

$L_i = \exp[-H_1(x_i|\mathbf{z}_i)] = 1 - P(T_i \leq x_i, \epsilon_i = 1) = P(T_i \leq x_i, \epsilon_i = 2) + P(T_i > x_i)$. But $P(T_i > x_i)$ would be the contribution to likelihood if the $i$-th subject is censored at $x_i$. The paper also mentions that different link functions $g()$ could be used for the model

$$g[F_1(t|\mathbf{z})] = s(\log(t)|\boldsymbol{\gamma}, \mathbf{n_0}) + \mathbf{z}^T\boldsymbol{\beta},$$

where $F_1$ is the cause-1 cumulative incidence function.

### 1.3.18 Weighted NPMLE for the Subdistribution

Bellach et al. (2019) introduced a weighted likelihood function that allows for a direct extension of the Fine–Gray model to a broad class of semiparametric regression models [6]. Let $A(t)$ be the cumulative subdistribution hazard, $A_0$ be an unspecified increasing function and $g$ be a thrice continuously differentiable and strictly increasing function with $g(0) = 0,\ g'(0) > 0$ and $g(\infty) = \infty$. The model proposed is

$$A(t) = g[\exp(\mathbf{z}^T\boldsymbol{\beta})A_0(t)].$$

Let $N_i(t) = I(T_i \leq t, \epsilon_i = j),\ Y_i(t) = 1 - N_i(t-),\ \hat{G}$ be the Kaplan–Meier estimator of $P(C > t),\ w_i(t) = \frac{I(C_i \geq T_i \wedge t)\hat{G}(t)}{\hat{G}(T_i \wedge t)}$ and $\tau$ be the duration of the study. The weighted log-likelihood function is

$$\begin{aligned} l(\boldsymbol{\beta}, A_0) = \sum_{i=1}^{n} ( & \int_0^\tau \log\{\exp(\mathbf{z}_i^T\boldsymbol{\beta})A_0'(t)g'[\exp(\mathbf{z}_i^T\boldsymbol{\beta})A_0(t)]\} I(C_i \geq t) Y_i(t) dN_i(t) \\ & - \int_0^\tau w_i(t) Y_i(t) \exp(\mathbf{z}_i^T\boldsymbol{\beta}) g'[\exp(\mathbf{z}_i^T\boldsymbol{\beta})A_0(t)] dA_0(t)). \end{aligned}$$

It is unclear how this function was obtained. $A_0$ is approximated by a sequence of step functions $A_n^0$, with jumps at the observed events of interest.

### 1.3.19 Copula-based Model

Vasquez and Escarela (2021) proposed Copula-based constructions of the joint distribution of the overall survival time and the cause-specific failure [96]. A copula is a bivariate distribution function with uniform marginals. Suppose there are two competing risks. Let $f_T(t)$ be the probability density function of $T$, $F_T(t)$ be the cumulative distribution function of $T$ and $F_\epsilon(\epsilon)$ be the cumulative distribution function of $\epsilon$. The paper claims that there exists a copula function $CP$ such that the joint density function of $(T, \epsilon)$

$$f_{T,D}(t,d) = f_T(t)\{CP'[F_T(t), F_\epsilon(\epsilon)] - CP'[F_T(t), F_\epsilon(\epsilon-1)]\}I(t>0),\ d=1,2,$$

where $CP'(x,y) = \frac{\partial}{\partial x}CP(x,y)$. However, it is unclear what the joint density function of $(T, \epsilon)$ means since $\epsilon$ cannot have a probability density function as a discrete random variable.

## 1.4 Variable Selection for High-dimensional Competing Risk Data

In this section, existing methods for selecting variables when censoring and competing risks are present are briefly described.

### 1.4.1 Boosting for High-dimensional Time-to-event Data With Competing Risks

Binder et al. (2009) proposed a boosting approach for fitting proportional subdistribution hazards models for high-dimensional data [8]. High-dimensional data are data that have more covariates than observations. The proposed approach is based on two main ideas: first, there are $M$ boosting steps, where in each step some elements of the estimated parameter

vector are updated. The updates are determined by penalized maximum partial likelihood estimation, where the previous boosting steps are incorporated as an offset. Second, there is a distinction between a set of indices of mandatory covariates $S_{mand} \subset \{1, ..., p\}$ and the set of indices of optional covariates $S_{opt} = \{1, ..., p\} \setminus S_{mand}$. In each boosting step, only one element of the estimated parameter vector, corresponding to one optional covariate, is updated. The elements corresponding to the mandatory covariates are updated simultaneously before each boosting step. Suppose the cause of interest is cause 1. The details of the algorithm are as follows:

1. Initialize the offset $\hat{\eta}_{0,i} = 0$, $i = 1, ..., n$ and the estimated parameter vector $\hat{\boldsymbol{\beta}}_0 = \mathbf{0}$.

2. For each boosting step $m = 1, ..., M$,

   (a) Update the elements $s \in S_{mand}$ of $\hat{\boldsymbol{\beta}}_{m-1}$ by one maximum partial likelihood Newton–Raphson step and update the offset via $\hat{\eta}_{m-1,i} = \mathbf{z}_i^T \hat{\boldsymbol{\beta}}_{m-1}$.

   (b) Estimate the parameters $\gamma_{m,s}$ in candidate models

   $$\lambda_1(t|\mathbf{z}_i) = \lambda_{10}(t) \exp(\hat{\eta}_{m-1,i} + \gamma_{m,s} z_{is}),\ s \in S_{opt}.$$

   (c) Determine the best candidate model $s^*$ with parameter estimate $\hat{\gamma}_{m,s^*}$ and perform the update

   $$\hat{\beta}_{m,s} = \begin{cases} \hat{\beta}_{m-1,s} + \hat{\gamma}_{m,s^*} & \text{if } s = s^* \\ \hat{\beta}_{m-1,s} & \text{otherwise} \end{cases}$$

   (d) Update the offset via $\hat{\eta}_{m,i} = \mathbf{z}_i^T \hat{\boldsymbol{\beta}}_m$.

To avoid boosting steps that are too large, the parameters $\hat{\gamma}_{m,s}$ are determined by penalized estimation. The partial log-likelihood provided by Fine and Gray (1999) [32] is augmented

by a penalty term, resulting in

$$l_{pen}(\gamma_{m,s}) = \sum_{i=1}^{n} I(\Delta_i \epsilon_i = 1)\{\hat{\eta}_{m-1,i} + \gamma_{m,s} z_{is} - \log[\sum_{l \in R_i} w_l(X_i) \exp(\hat{\eta}_{m-1,i} + \gamma_{m,s} z_{ls})]\} + \frac{\lambda}{2}\gamma_{m,s}^2,$$

where $R_i = \{l : X_l \geq X_i \text{ or } \Delta_l \epsilon_l > 1\}$ is the risk set that arises when the competing risks process for an individual is stopped just before the time of a competing event and $\lambda$ is a penalty parameter that determines the size of the boosting steps. The formula for $w_l(t)$ is given by $\frac{\hat{G}(t) I(X_l \leq t) \Delta_l}{\hat{G}(X_l)}$ in the paper, different from that in Fine and Gray (1999) [32], which is probably a typo. $\lambda$ is typically chosen such that the number of boosting steps, selected e.g. by cross-validation, is larger than 50, as this number limits the maximal number of non-zero coefficients of the fitted model. The estimates are determined by one Newton–Raphson step, i.e. $\hat{\gamma}_{m,s} = \frac{U_{pen}(0)}{I_{pen}(0)}$, where $U_{pen}(\gamma) = l'_{pen}(\gamma)$ is the score function and $I_{pen}(\gamma) = l''_{pen}(\gamma)$. Correspondingly, the best candidate model is taken to be the one that maximizes the score statistic $\frac{U^2_{pen}(0)}{I_{pen}(0)}$. This method is implemented in the R package CoxBoost, available on GitHub.

### 1.4.2 Penalized Proportional Subdistribution Hazard Model

Fu, Parikh and Zhou (2017) proposed a general penalized variable selection strategy that simultaneously handles variable selection and parameter estimation in the proportional subdistribution hazards model [35]. The log-partial likelihood is defined as

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} \int_0^{\infty} \{\mathbf{z}_i^T \boldsymbol{\beta} - \log[\sum_{q=1}^{n} w_q(s) Y_q(s) \exp(\mathbf{z}_q^T \boldsymbol{\beta})]\} \times w_i(t) dN_i(t)$$

The authors proposed a generalized objective function

$$Q(\boldsymbol{\beta}) = l(\boldsymbol{\beta}) - n \sum_{m=1}^{p} p_\lambda(|\beta_m|),$$

where $p_\lambda(\cdot)$ is the penalty function. The penalized estimator is given by $\tilde{\boldsymbol{\beta}} = \arg\max Q(\boldsymbol{\beta})$. To maximize $Q(\boldsymbol{\beta})$, the authors approximate the log-partial likelihood function by the Newton-Raphson update, and at each iteration, solve an iterative reweighted least square problem subject to penalties. Denote $\boldsymbol{\eta} = \mathbf{Z}\boldsymbol{\beta}$ and define $\mathbf{u} = \partial l/\partial \boldsymbol{\eta}$ and $\mathbf{H} = -\partial^2 l/\partial \boldsymbol{\eta}\partial \boldsymbol{\eta}^T$. The pseudo response vector is $\mathbf{y} = \boldsymbol{\eta} + \mathbf{H}^{-1}\mathbf{u}$. The authors claim that by second order Taylor expansion,

$$-l(\boldsymbol{\beta}) \approx \frac{1}{2}(\mathbf{y} - \boldsymbol{\eta})^T \mathbf{H}(\mathbf{y} - \boldsymbol{\eta}).$$

However, this approximation is flawed as demonstrated in Chapter 2 and a correct approximation is presented as part of this dissertation. They use $BIC = -2l(\tilde{\boldsymbol{\beta}}) + \log(n)\sum_{m=1}^{p} I(\tilde{\beta}_m \neq 0)$ as the criterion to select the tuning parameter $\lambda$. How this criterion works in the competing-risks setting and especially for high-dimensional data is unknown. Therefore, cross-validation type scores are used in this dissertation. Further, the authors only applied this method to low-dimensional data but did not explore how it works for high-dimensional data, which will be studied as part of this dissertation.

Sun and Wang (2022) applied the elastic net penalty to the proportional subdistribution hazards model [89]. The elastic net penalty is

$$p_\lambda(\beta) = \lambda[\frac{1}{2}(1-\alpha)\beta^2 + \alpha|\beta|],$$

where $0 < \alpha < 1$. Given $\mathbf{b}$, $l(\boldsymbol{\beta})$ can be approximated by

$$l(\mathbf{b}) + (\boldsymbol{\beta} - \mathbf{b})^T \nabla l(\mathbf{b}) + \frac{1}{2}(\boldsymbol{\beta} - \mathbf{b})^T \boldsymbol{H}_l(\mathbf{b})(\boldsymbol{\beta} - \mathbf{b}),$$

where $\nabla l(\boldsymbol{\beta})$ and $\boldsymbol{H}_l(\boldsymbol{\beta})$ are the gradient and Hessian matrix of $l(\boldsymbol{\beta})$, respectively. $\boldsymbol{H}_l(\mathbf{b})$ is approximated by its expected value $E[\boldsymbol{H}_l(\mathbf{b})] := -\mathbf{J}$. Let $\tilde{\boldsymbol{\beta}}$ be the maximum likelihood estimator that maximizes $l(\boldsymbol{\beta})$. Denote $\boldsymbol{\Sigma}$ as the asymptotic covariance matrix of $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\Sigma}}$

as a consistent estimate of $\mathbf{\Sigma}$. They claimed that $\mathbf{J} = \mathbf{\Sigma}^{-1}$. Then $\mathbf{J}$ is approximated by $\tilde{\mathbf{\Sigma}}^{-1}$. Then they approximated $l(\boldsymbol{\beta})$ by

$$l(\tilde{\boldsymbol{\beta}}) - \frac{1}{2}(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})^T \tilde{\mathbf{\Sigma}}^{-1}(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}}).$$

However, this argument is flawed. First, $\boldsymbol{H}_l(\mathbf{b})$ only depends on $\mathbf{b}$, given the dataset, so there is no need to calculate its expectation. Second, $l(\boldsymbol{\beta})$ cannot be maximized in a high-dimensional data. Third, it is unclear what form $\tilde{\mathbf{\Sigma}}$ takes and why $\mathbf{J} = \mathbf{\Sigma}^{-1}$. Finally, the final approximate failed to include the $(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})^T \nabla l(\tilde{\boldsymbol{\beta}})$ portion. This method is implemented in the R package RAEN, available on GitHub.

### 1.4.3 Scalable Algorithms for Large Competing Risks Data

Kawaguchi et al. (2021) developed a scalable surrogate $l_0$-based method for simultaneous variable selection and parameter estimation for the large $p$ problem [53]. As a scalable approximation to $l_0$-penalized regression, the broken adaptive ridge (BAR) estimator, defined as the limit of an $l_0$-based iteratively reweighted $l_2$-penalization algorithm, has been studied for simultaneous variable selection and parameter estimation. Let the log-partial likelihood be defined as in Fu, Parikh and Zhou (2017) [35]. The BAR estimator of $\boldsymbol{\beta}$ starts with an initial $l_2$-penalized (or ridge) estimator

$$\hat{\boldsymbol{\beta}}^{(0)} = \arg\min\{-2l(\boldsymbol{\beta}) + \xi_n \sum_{m=1}^{p} \beta_m^2\},$$

which is updated iteratively by a reweighted $l_2$-penalized estimator

$$\hat{\boldsymbol{\beta}}^{(s)} = \arg\min \left\{ -2l(\boldsymbol{\beta}) + \lambda_n \sum_{m=1}^{p} \frac{\beta_m^2}{(\hat{\beta}_m^{(s-1)})^2} \right\}, \ s \geq 1,$$

where $\xi_n$ and $\lambda_n$ are nonnegative penalization tuning parameters. The BAR estimator of $\boldsymbol{\beta}$ is defined as the limit of this iterative algorithm:

$$\hat{\boldsymbol{\beta}} = \lim_{s\to\infty} \hat{\boldsymbol{\beta}}^{(s)}.$$

The authors derived a fast cyclic coordinate-wise BAR algorithm that results in the elimination of performing multiple ridge regressions and avoids using a cutoff to introduce sparsity as required by the original BAR algorithm. For a consistent estimate $\tilde{\boldsymbol{\beta}}$ of $\boldsymbol{\beta}$, they considered the Cholesky decomposition $-\mathbf{H}(\tilde{\boldsymbol{\beta}}) = \tilde{\mathbf{X}}'\tilde{\mathbf{X}}$, where $\mathbf{H}(\boldsymbol{\beta})$ is the Hessian matrix of $l(\boldsymbol{\beta})$ and define $\tilde{\mathbf{y}} = (\tilde{\mathbf{X}}')^{-1}[-\mathbf{H}(\tilde{\boldsymbol{\beta}})\tilde{\boldsymbol{\beta}} + \nabla l(\tilde{\boldsymbol{\beta}})]$ as the pseudo-response vector, where $\nabla l(\boldsymbol{\beta})$ is the gradient of $l(\boldsymbol{\beta})$. Approximating the negative log-partial likelihood by $-l(\boldsymbol{\beta}) \approx \frac{1}{2}(\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\boldsymbol{\beta})'(\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\boldsymbol{\beta})$ using a second-order Taylor expansion, they showed that solving the BAR estimator leads to the following solution

$$\hat{\boldsymbol{\beta}}^{(s)} = g(\hat{\boldsymbol{\beta}}^{(s-1)}),$$

where $g(\boldsymbol{\beta}) = [\tilde{\mathbf{X}}'\tilde{\mathbf{X}} + \lambda_n D(\boldsymbol{\beta})]^{-1}\tilde{\mathbf{X}}\tilde{\mathbf{y}}$ and $D(\boldsymbol{\beta}) = \text{diag}(1/\beta_1^2, ..., 1/\beta_p^2)$. Hence, as $s \to \infty$, the limit of the sequence $\{\hat{\boldsymbol{\beta}}^{(s)}\}$ is the fixed point of the function $g(\cdot)$ or the solution to $g(\boldsymbol{\beta}) = \boldsymbol{\beta}$.

They showed that each component of the fixed-point solution of $g$ can be expressed as a function of all the other components in the next theorem.

**Theorem 1.** *Let $\hat{\boldsymbol{\beta}}$ be the fixed-point solution of $g(\cdot)$. Then, for each $m = 1, ..., p$, the $m$-th component of $\hat{\boldsymbol{\beta}}$ can be expressed as follows*

$$\hat{\beta}_m = g_m(\hat{\boldsymbol{\beta}}_{-m}) := \begin{cases} 0, & |b_m| < 2\sqrt{\lambda_n \tilde{\boldsymbol{x}}_m' \tilde{\boldsymbol{x}}_m}, \\ \frac{b_m + sign(b_m)\sqrt{b_m^2 - 4\lambda_n \tilde{\boldsymbol{x}}_m' \tilde{\boldsymbol{x}}_m}}{2\tilde{\boldsymbol{x}}_m' \tilde{\boldsymbol{x}}_m}, & |b_m| \geq 2\sqrt{\lambda_n \tilde{\boldsymbol{x}}_m' \tilde{\boldsymbol{x}}_m}, \end{cases}$$

*where $b_m = \tilde{\boldsymbol{x}}_m'(\tilde{\boldsymbol{y}} - \sum_{i\neq m} \tilde{\boldsymbol{x}}_i \hat{\beta}_i)$.*

In their algorithm, $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{y}}$ are initially estimated using the initial ridge estimate $\hat{\boldsymbol{\beta}}^{(0)}$ and then subsequently updated at step $s$ using the previous estimate $\hat{\boldsymbol{\beta}}^{(s-1)}$. Consequently, at step $s$,

$$\begin{aligned} b_m^{(s)} &= \tilde{\mathbf{x}}_m'(\tilde{\mathbf{y}} - \sum_{i \neq m} \tilde{\mathbf{x}}_i \hat{\beta}_i^{(s-1)}) \\ &= -\frac{\partial^2 l}{\partial \beta_m^2}(\hat{\boldsymbol{\beta}}^{(s-1)})\hat{\beta}_m^{(s-1)} + \frac{\partial l}{\partial \beta_m}(\hat{\boldsymbol{\beta}}^{(s-1)}) \end{aligned}$$

for $m = 1, ..., p$.

Then their algorithm proceeds as follows

1. Set $\hat{\boldsymbol{\beta}}^{(0)} = \hat{\boldsymbol{\beta}}_{ridge}$.

2. For $s = 1, 2, ...$:

   (a) For $m = 1, ..., p$

      i. Calculate $b_m^{(s)} = -\frac{\partial^2 l}{\partial \beta_m^2}(\hat{\boldsymbol{\beta}}^{(s-1)})\hat{\beta}_m^{(s-1)} + \frac{\partial l}{\partial \beta_m}(\hat{\boldsymbol{\beta}}^{(s-1)})$.

      ii. If $|b_m^{(s)}| < 2\sqrt{-\lambda_n \frac{\partial^2 l}{\partial \beta_m^2}(\hat{\boldsymbol{\beta}}^{(s-1)})}$, then $\hat{\beta}_m^{(s)} = 0$;
      otherwise $\hat{\beta}_m^{(s)} = \frac{b_m^{(s)} + \text{sign}(b_m^{(s)})\sqrt{(b_m^{(s)})^2 + 4\lambda_n \frac{\partial^2 l}{\partial \beta_m^2}(\hat{\boldsymbol{\beta}}^{(s-1)})}}{-2\frac{\partial^2 l}{\partial \beta_m^2}(\hat{\boldsymbol{\beta}}^{(s-1)})}$.

   (b) If $||\hat{\boldsymbol{\beta}}^{(s)} - \hat{\boldsymbol{\beta}}^{(s-1)}|| < tol$, then $\hat{\boldsymbol{\beta}}_{BAR} = \hat{\boldsymbol{\beta}}^{(s)}$ and stop.

Suppose the cause of interest is cause 1. The score function is given by

$$\frac{\partial l}{\partial \beta_m}(\boldsymbol{\beta}) = \sum_{i=1}^{n} I(\Delta_i \epsilon_i = 1) z_{im} - \sum_{i=1}^{n} I(\Delta_i \epsilon_i = 1) \frac{\sum_{k \in R_i} z_{km} w_k(X_i) \exp(\mathbf{z}_k^T \boldsymbol{\beta})}{\sum_{k \in R_i} w_k(X_i) \exp(\mathbf{z}_k^T \boldsymbol{\beta})}$$

and the Hessian diagonals are given by

$$\frac{\partial^2 l}{\partial \beta_m^2}(\boldsymbol{\beta}) = \sum_{i=1}^{n} I(\Delta_i \epsilon_i = 1) \left\{ \frac{\sum_{k \in R_i} z_{km}^2 w_k(X_i) \exp(\mathbf{z}_k^T \boldsymbol{\beta})}{\sum_{k \in R_i} w_k(X_i) \exp(\mathbf{z}_k^T \boldsymbol{\beta})} - \left[ \frac{\sum_{k \in R_i} z_{km} w_k(X_i) \exp(\mathbf{z}_k^T \boldsymbol{\beta})}{\sum_{k \in R_i} w_k(X_i) \exp(\mathbf{z}_k^T \boldsymbol{\beta})} \right]^2 \right\},$$

where $R_i = \{y : (X_y \geq X_i) \cup (X_y \leq X_i \cap \Delta_y \epsilon_y > 1)\}$. A lemma is given in the paper:

**Lemma 2.** *Assume that no ties are present. Then, for any* $r, m = 1, ..., p$ *and* $u, v = 0, 1$

$$\sum_{k \in R_i} z_{kr}^u z_{km}^v w_k(X_i) \exp(\boldsymbol{z}_k^T \boldsymbol{\beta}) = \sum_{k \in R_i(1)} z_{kr}^u z_{km}^v \exp(\boldsymbol{z}_k^T \boldsymbol{\beta}) + \hat{G}(X_i) \sum_{k \in R_i(2)} z_{kr}^u z_{km}^v \exp(\boldsymbol{z}_k^T \boldsymbol{\beta}) / \hat{G}(X_k),$$

*where* $R_i(1) = \{y : (X_y \geq X_i)\}$ *and* $R_i(2) = \{y : (X_y < X_i \cap \Delta_y \epsilon_y > 1)\}$.

While $R_i(1)$ grows cumulatively as the event times decrease from largest to smallest, $R_i(2)$ grows cumulatively as the event times increase from smallest to largest. Thus, the ratio of summations for the score and diagonal Hessian values can be calculated in linear time via a forward-backward scan where one scan goes in one direction to calculate the cumulative sums associated with $R_i(1)$ and the other scan goes in the opposite direction to calculate the cumulative sum associated with $R_i(2)$. Therefore, the number of operations can be effectively reduced from $O(n^2)$ to $O(n)$. The BAR algorithm is implemented in the R package pshBAR and the forward–backward scan algorithm is implemented in the R package fastcmprsk, both available on GitHub.

### 1.4.4 Random Survival Forests

Ishwaran et al. (2014) proposed an approach that builds on the framework of random survival forests (RSF) [50]. A single competing risk tree is grown in each bootstrap sample under some splitting rule. Let $\delta = \Delta\epsilon$. Let $x_{(1)} < x_{(2)} < ... < x_{(k)}$ be the distinct event times. Suppose that the proposed split for the root node is of the form $z \leq c$ and $z > c$ for a continuous predictor $z$ (this can be generalized to categorical variables). Such a split forms two daughter nodes containing two new sets of competing risk data. To indicate these data, subscripts of $l$ and $r$ are used for the left and right daughter nodes. The cause specific hazard

function for cause $j$ is given by

$$\alpha_j(t) = \lim_{\Delta t \to 0+} \frac{P(t \leq T \leq t + \Delta t, \epsilon = j | T \geq t)}{\Delta t}.$$

Denote by $\alpha_{jl}(t)$ and $\alpha_{jr}(t)$ the cause $j$ specific hazard rates in the left and right daughter nodes, respectively. The number of individuals at risk at time $t$ in the left and right daughter nodes are, respectively $Y_l(t)$ and $Y_r(t)$, where $Y_l(t) = \sum_{i=1}^n I(X_i \geq t, z_i \leq c)$, $Y_r(t) = \sum_{i=1}^n I(X_i \geq t, z_i > c)$. $Y(t) = \sum_{i=1}^n I(X_i \geq t)$. The number of type $j$ events at time $t$ for the left and right daughters is, respectively,

$$d_{j,l}(t) = \sum_{i=1}^n I(X_i = t, \delta_i = j, z_i \leq c),\ d_{j,r}(t) = \sum_{i=1}^n I(X_i = t, \delta_i = j, z_i > c),$$

and $d_j(t) = \sum_{i=1}^n I(X_i = t, \delta_i = j)$. Define $x_{(m)}, x_{(m)_l}, x_{(m)_r}$ to be the largest times on study in the parent node and the two daughters, respectively.

The first splitting rule is the log-rank test. This is a test of $H_0 : \alpha_{jl}(t) = \alpha_{jr}(t)$ for $t \leq x_{(k)}$. The test is based on the weighted difference of the cause-specific Nelson–Aalen estimates in the two daughter nodes. Specifically, for a split at the value $c$ for variable $z$, the splitting score is

$$L_j^{LR}(z, c) = \frac{1}{\hat{\sigma}_j^{LR}(z, c)} \sum_{i=1}^m W_j(x_{(i)}) \left[ d_{j,l}(x_{(i)}) - \frac{d_j(x_{(i)}) Y_l(x_{(i)})}{Y(x_{(i)})} \right],$$

where the variance estimate is given by

$$(\hat{\sigma}_j^{LR}(z, c))^2 = \sum_{i=1}^m W_j(x_{(i)})^2 d_j(x_{(i)}) \frac{Y_l(x_{(i)})}{Y(x_{(i)})} \left[ 1 - \frac{Y_l(x_{(i)})}{Y(x_{(i)})} \right] \left[ \frac{Y(x_{(i)}) - d_j(x_{(i)})}{Y(x_{(i)}) - 1} \right].$$

Time-dependent weights $W_j(t) > 0$ are used to make the test more sensitive to early or late differences between the cause-specific hazards. The choice $W_j(t) = 1$ corresponds to the

standard log-rank test which has optimal power for detecting alternatives where the cause-specific hazards are proportional. The best split is found by maximizing $|L_j^{LR}(z,c)|$ over $z$ and $c$.

The cause-$j$ specific log-rank splitting rule is useful if the main purpose is to detect variables that affect the cause-$j$ specific hazard. It may not be optimal if the purpose is also prediction of cumulative event probabilities. In this case, better results may be obtained with splitting rules that select variables based on their direct effect on the cumulative incidence. For this reason, they modeled the second splitting rule after Gray's test [42], which tests $H_0 : F_{jl}(t) = F_{jr}(t)$ for $t \leq x_{(k)}$, where $F_{jl}(t)$ and $F_{jr}(t)$ are cumulative incidence functions for the left and the right daughter nodes, respectively. For notational simplicity, consider analysis of event $j = 1$ and assume the number of causes of failure $K = 2$. Gray's statistic for testing the null hypothesis is

$$\int_0^{x_{(k)}} W_j(s) R_l(s) \left[ \frac{d\hat{F}_{jl}(s)}{1 - \hat{F}_{jl}(s)} - \frac{d\hat{F}_j(s)}{1 - \hat{F}_j(s)} \right],$$

where

$$R_l(t) = I(x_{(m)_l} \geq t) Y_l(t) [1 - \hat{F}_{jl}(t-)] / \hat{S}_l(t-),$$

and $\hat{F}_j(t)$ is the Aalen–Johansen estimator [1]

$$\hat{F}_j(t) = \sum_{i=1}^{m(t)} \hat{S}(x_{(i-1)}) d_j(x_{(i)}) / Y(x_{(i)}),$$

where $\hat{S}(t)$ is the Kaplan–Meier estimator for the survival function of $T$ and $m(t) = \max\{i : x_{(i)} \leq t\}$.

The steps required to construct a competing risks forest can be summarized as follows.

1. Draw $B$ bootstrap samples from the learning data.

2. Grow a competing risk tree for each bootstrap sample. At each node of the tree, randomly select $M \leq p$ candidate variables. The node is split using the candidate variable that maximizes a competing risk splitting rule.

3. Grow the tree to full size under the constraint that a terminal node should have no less than $n_0 > 0$ unique cases.

The concordance index and the prediction error defined by the integrated Brier score (BS) are used to assess prediction performance. The concordance index (C-index) estimates the probability that, in a randomly selected pair of cases, the case that fails first had a worse predicted outcome. The BS is the squared difference between actual and predicted outcome. The cause-$j$ mortality $M_j(x_{(k)}|\mathbf{z}) = \int_0^{x_{(k)}} F_j(t|\mathbf{z})$. Let $c_{i,b}$ be the number of times case $i$ occurs in the bootstrap sample used to grow the $b$-th tree. To define the CIF for the $b$-th tree, take a subject's covariate vector $\mathbf{z}$ and drop it down the tree. Let $h_b(\mathbf{z})$ denote the indices for cases from the learning data whose covariates share the terminal node with $\mathbf{z}$. Denoting node-specific event counts by $N_{j,b}(t|\mathbf{z}) = \sum_{i \in h_b(\mathbf{z})} c_{i,b} I(X_i \leq t, \delta_i = j)$ and the number at risk by $Y_b(t|\mathbf{z}) = \sum_{i \in h_b(\mathbf{z})} c_{i,b} I(X_i \geq t)$, this subject's CIF is defined as

$$\hat{F}_{j,b}(t|\mathbf{z}) = \int_0^t \frac{\hat{S}_b(u-|\mathbf{z})}{Y_b(u|\mathbf{z})} dN_{j,b}(u|\mathbf{z}),$$

where $\hat{S}_b(t|\mathbf{z}) = \prod_{u \leq t}[1 - \sum_j dN_{j,b}(u|\mathbf{z})/Y_b(u|\mathbf{z})]$ is this subject's Kaplan–Meier estimate of event-free survival. The ensemble estimate of the CIF and the cause-$j$ mortality, respectively, equal

$$\bar{F}_j(t|\mathbf{z}) = \frac{1}{B}\sum_{b=1}^{B} \hat{F}_{j,b}(t|\mathbf{z}), \; \bar{M}_j(x_{(k)}|\mathbf{z}) = \int_0^{x_{(k)}} \bar{F}_j(t|\mathbf{z}) dt.$$

Subjects are ranked by ensemble cause-$j$ mortality. Subject $i$ is said to have a higher risk of event $j$ than case $i'$ if $\bar{M}_j(x_{(k)}|\mathbf{z}_i) > \bar{M}_j(x_{(k)}|\mathbf{z}_{i'})$. [100] described a time-truncated concor-

dance index for competing risks, which in the current setting is

$$C_j(x_{(k)}) = P[\bar{M}_j(x_{(k)}|\mathbf{z}_i) > \bar{M}_j(x_{(k)}|\mathbf{z}_{i'})|T_i \leq x_{(k)},\ \epsilon_i = j \text{ and } (T_i < T_{i'} \text{ or } \epsilon_{i'} \neq j)].$$

The time-dependent BS ([41] and [38]) and its integral (IBS) are also considered to assess the performance of the ensemble CIF:

$$IBS_j(x_{(m)}) = \int_0^{x_{(m)}} BS_j(t)dt = \int_0^{x_{(m)}} E[I(T_i \leq t, \delta_i = j) - \bar{F}_j(t|\mathbf{z})]^2 dt.$$

For reporting an internal error rate, out-of-bag (OOB) ensembles are used. The OOB data are used to construct the OOB ensemble. Let $O_i \subset \{1, ..., B\}$ be the index set of trees where $c_{i,b} = 0$; i.e., $O_i$ records trees where subject $i$ is OOB. The OOB ensemble estimate of the CIF is

$$\bar{F}_j^{oob}(t|\mathbf{z}_i) = \frac{1}{|O_i|} \sum_{b \in O_i} \hat{F}_{j,b}(t|\mathbf{z}_i).$$

Denote $(X_i, \delta_i, \mathbf{z}_i),\ 1 \leq i \leq n'$ for a validation dataset of size $n'$. Based on these data, the prediction error can be estimated using inverse probability of censoring weights (IPCWs) ([38] and [100]). Let $\hat{G}(t)$ be the Kaplan–Meier estimate of the censoring distribution of censoring time. The OOB-IPCW estimate of $C_j$ at $x_{(k)}$ is

$$\hat{C}_j(x_{(k)}) = \frac{\sum_i \sum_{i'} (A_{ii'}/\hat{w}_{ii',1} + B_{ii'}/\hat{w}_{ii',2}) Q_{ii'}^{oob} I(X_i \leq x_{(k)}, \delta_i = j)}{\sum_i \sum_{i'} (A_{ii'}/\hat{w}_{ii',1} + B_{ii'}/\hat{w}_{ii',2}) I(X_i \leq x_{(k)}, \delta_i = j)},$$

where $\hat{w}_{ij,1} = \hat{G}(X_i-)\hat{G}(X_i),\ \hat{w}_{ij,2} = \hat{G}(X_i-)\hat{G}(X_j-), A_{ij} = I(X_i < X_j),\ B_{ij} = I(X_i \geq X_j \text{ and } \Delta_j \epsilon_j \neq j),\ Q_{ij}^{oob} = I[\bar{L}^{oob}(x_{(k)}|\mathbf{z}_i) < \bar{L}^{oob}(x_{(k)}|\mathbf{z}_j)]$ and $L(x_{(k)}|\mathbf{z}) = \sum_{j=1}^K \int_0^{x_{(k)}} F_j(t|\mathbf{z})dt$. The definition of $\hat{w}_{ij,1}$ probably contains a typo and its correct expression is most likely $\hat{G}(X_i-)\hat{G}(X_j)$ or $\hat{G}(X_j-)\hat{G}(X_i)$. Using weights $\hat{w}_i(t) = I(X_i \leq t, \Delta_i = 1)/\hat{G}(X_i) + I(X_i >$

$t)/\hat{G}(t)$ ([8]), the OOB estimate of the integrated BS for event $j$ is given by

$$\widehat{IBS}_j^{oob}(x_{(k)}) = \int_0^{x_{(k)}} \frac{1}{n} \sum_{i=1}^{n} \hat{w}_i(t)[I(X_i \leq t, \delta_i = j) - \bar{F}_j^{oob}(t|\mathbf{z}_i)]^2 dt.$$

Note that extremely large weights may occur, but can be avoided by evaluating the IPCW statistics at an earlier time point $t < x_{(k)}$.

RSF variable selection typically involves filtering variables on the basis of variable importance (VIMP). VIMP measures the increase (or decrease) in prediction error for the forest ensemble when a variable is randomly "noised-up" [12]. A large positive VIMP shows that the prediction accuracy of the forest is substantially degraded when a variable is noised-up; thus a large VIMP indicates a potentially predictive variable. The authors calculated VIMP by random node assignment [49]. In random node assignment, cases are dropped down a tree and randomly assigned to a daughter node whenever the parent node splits on the target variable. To compute event-specific VIMP, first estimate the prediction error. Then the data are noised up by random node assignment, and the prediction error is recomputed. The difference in these two values gives the VIMP for each variable for each event $j$. Minimal depth assesses the predictiveness of a variable by the depth of the first split of a variable relative to the root node of a tree. Variables are selected using minimal depth variable selection [48]. Those variables whose event-specific VIMP are positive, and that meet a minimal depth threshold (estimated from the forest), represent the final selected set of variables. This method is implemented in the R package randomForestSRC [47].

### 1.4.5 Penalized Binomial Regression Model

Ambrogi and Scheike (2016) attempted to directly model cause-1 cumulative incidence function $F_1(t|\mathbf{z})$ assuming [3]

$$\text{logit}[F_1(t|\mathbf{z})] = \alpha(t) + \mathbf{z}^T\boldsymbol{\beta}.$$

Let $N_i(t) = I(X_i \leq t, \Delta_i \epsilon_i = 1)$, $\mathbf{N}(t) = (N_1(t), ..., N_n(t))$ and $\mathbf{F}_1(t|\mathbf{z}) = (F_1(t|\mathbf{z}_1), ..., F_1(t|\mathbf{z}_n))$. Let $D_{\alpha,i}(t) = \frac{\partial F_1(t|\mathbf{z}_i)}{\partial \alpha(t)}$ and $D_{\boldsymbol{\beta},i}(t) = \frac{\partial F_1(t|\mathbf{z}_i)}{\partial \boldsymbol{\beta}}$. Let $\mathbf{D}_\alpha(t) = (D_{\alpha,1}(t), ..., D_{\alpha,n}(t))^T$ and $\mathbf{D}_{\boldsymbol{\beta}}(t)$ be a $n \times p$ matrix with the $i$-th row being $D_{\boldsymbol{\beta},i}(t)$. Define inverse probability of censoring weighting weight matrix $\mathbf{W}(t) = \text{diag}(W_i(t))$ with $W_i(t) = \Delta_i(t)/S_C(\min(t, X_i)|\mathbf{z}_{mand,i})$, where $\Delta_i(t) = I(\min(T_i, t) \leq C_i)$, $S_C(t|\mathbf{z}_{mand}) = P(C > t|\mathbf{z}_{mand})$ and $\mathbf{z}_{mand}$ is the set of covariates that have to be included in the model. The estimated quantity is $\hat{\mathbf{W}}(t)$ with $\hat{W}_i(t) = \Delta_i(t)/\hat{S}_C(\min(t, X_i)|\mathbf{z}_{mand,i})$. The regression function $\alpha(t)$ and regression parameter $\boldsymbol{\beta}$ can be estimated based on the following estimating equations:

$$U_\alpha(t) := \mathbf{D}_\alpha^T(t)\hat{\mathbf{W}}(t)\{\mathbf{N}(t) - \mathbf{F}_1(t|\mathbf{z})\} = 0 \tag{1.2}$$

$$U_{\boldsymbol{\beta}}(t) := \int_0^\tau \mathbf{D}_{\boldsymbol{\beta}}^T(t)\hat{\mathbf{W}}(t)\{\mathbf{N}(t) - \mathbf{F}_1(t|\mathbf{z})\}dt = \mathbf{0} \tag{1.3}$$

where $\tau$ is the last time point considered. Note that the estimates of $\alpha(t)$ will be piecewise constant functions that change their value only after events of type 1 so only the score equations for $\alpha(t)$ in the jump times need to be considered. In the case of high-dimensional covariates, a specific set of time-points is considered only to reduce the computations, and the baseline $\alpha(t)$ is thus reduced to a finite-dimensional parameter. When profiling out both the baseline and $\boldsymbol{\beta}_{mand}$, $U_\alpha(t)$ becomes $U_{p,mand}(\boldsymbol{\beta}_{option})$, where $\boldsymbol{\beta}_{option}$ is the parameter vector corresponding to the covariates that do not have to be included in the model. However, how this "profiling out" exactly works seems unclear. The penalized estimating functions for $\boldsymbol{\beta}$ can be written as

$$U_{p,mand}(\boldsymbol{\beta}) - n\mathbf{q}_\lambda(\boldsymbol{\beta}_{option})\text{sign}(\boldsymbol{\beta}_{option})$$

where $\mathbf{q}_\lambda(\boldsymbol{\beta}_{option}) = (q_{\lambda,1}(\beta_1), ..., q_{\lambda,p_{option}}(\beta_{p_{option}}))$ and $q_{\lambda,m}()$ for $m = 1, .., p_{option}$ are functions depending on the coefficients. The interest here is in cases where $q_{\lambda,m}()$ is the derivative of some penalty function.

### 1.4.6 Penalized Cause-specific Hazards Models

Saadati et al. (2018) [81] considers modeling competing risks data in high dimensions using a penalized cause-specific hazards (CSHs) approach. The CSH for cause $j$ is given by

$$\lambda(t, j|\mathbf{z}) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t,\ \epsilon = j | T \geq t)}{\Delta t}.$$

The proportional CSH model for cause $j$ is given by

$$\lambda(t, j|\mathbf{z}) = \lambda_{0j}(t) \exp(\mathbf{z}^T \boldsymbol{\beta}_j),$$

where $\lambda_{0j}(t)$ is the baseline hazard for cause $j$ at time $t$ and $\boldsymbol{\beta}_j$ is the vector of cause-specific regression coefficients. The partial likelihood for cause $j$ is defined in [79] as

$$L_j(\boldsymbol{\beta}_j) = \prod_{i=1}^{k_j} \frac{\exp(\mathbf{z}_{j(i)} \boldsymbol{\beta}_j)}{\sum_{l \in R(t_{j(i)})} \exp(\mathbf{z}_l \boldsymbol{\beta}_j)},$$

where $t_{j(i)}, i = 1, ..., k_j$ denote the $k_j$ times of failure of cause $j$, $\mathbf{z}_{j(i)}$ denotes the corresponding covariates, $R(t_{j(i)})$ is the set of study subjects known to be at risk just prior to $t_{j(i)}$. $l_j(\boldsymbol{\beta}_j) = \log[L_j(\boldsymbol{\beta}_j)]$. The LASSO-penalized log-partial likelihood for cause $j$ is maximized:

$$\max\{l_j(\boldsymbol{\beta}_j) - \lambda_j ||\boldsymbol{\beta}_j||_1\}.$$

The tuning parameters $\lambda_j$ are obtained via cross-validation with respect to minimal deviance (i.e., the penalized log-partial likelihood) as defined by [85].

Suppose there are two causes of failure. As an extension, the authors attempt to link the two independently penalized CSH models by choosing the optimal tuning parameters $\lambda_1$ and $\lambda_2$ with respect to minimal prediction error of the event of interest at a fixed time $s$. Schoop

et al. (2011) [83] defined prediction error (a.k.a. Brier score) for the event of interest $j$ as

$$PE_j(t) = E[I(T \leq t, \epsilon = j) - \pi_j(t|\mathbf{z})]^2,$$

where $\pi_j(t|\mathbf{z})$ denotes the predicted cumulative incidence function $F_j(t|\mathbf{z})$. The suggested algorithm works as follows:

1. Set up a grid of lambda values that ranges from the smallest to the largest: $\lambda_{ji}$, $j = 1, 2$, $i = 1, ..., I$.

2. Partition the data into, for example, 10 folds. For each fold

   (a) use 9 of 10 folds to fit cause-specific penalized regression model for the cause of interest for all $i = 1, ..., I$.

   (b) predict for each patient in the 10th fold the probability of event 1.

3. Calculate the prediction error $PE_1(s)$ at some time point $s$.

4. Select the pair $(\lambda_{1r_1^*}, \lambda_{2r_2^*})$ with the smallest average prediction error and fit the final CSH model using the optimal tuning parameters.

The authors advocate to choose $s$ as a clinically/biologically relevant time point. In the absence of such a time point related to the clinical context, it is also possible to use the integrated Brier score rather than choosing an arbitrary time point $s$.

### 1.4.7 Penalized Quantile Regression

Li, Tian and Tang (2019) developed a variable selection procedure based on penalized estimating equations for competing risks quantile regression [64]. Let $F_1(t|\mathbf{z})$ be the cumulative incidence function for cause 1. The conditional quantile is defined as $Q_1(\tau|\mathbf{z}) = \inf\{t :$

$F_1(t|\mathbf{z}) \geq \tau\}$. Let $g(\cdot)$ be a known monotone increasing and continuously differentiable link function. Let $\tilde{\mathbf{z}} = (1, \mathbf{z})^T$, for $\tau \in [\tau_L, \tau_U]$, $Q_1(\tau|\mathbf{z})$ is modeled as

$$Q_1(\tau|\mathbf{z}) = g[\tilde{\mathbf{z}}^T \boldsymbol{\beta}_0(\tau)].$$

Define $T_1^* = I(\epsilon = 1) \times T + I(\epsilon \neq 1) \times \infty$. The authors present the following modified model:

$$\begin{aligned} g^{-1}(T_1^*) &= Q_1(\tau|\mathbf{z}) + \tilde{e} \\ &= \tilde{\mathbf{z}}^T \boldsymbol{\beta}_0(\tau) + \tilde{e}, \end{aligned}$$

where $\tilde{e}$ is an error term with $\tau$-quantile assumed to be zero. However this model is confusing as it seems to imply that $Q_1(\tau|\mathbf{z}) = \tilde{\mathbf{z}}^T \boldsymbol{\beta}_0(\tau)$ and it is not clear that this is correct. Additionally, $P(T_1^* = \infty) > 0$ so $P(g^{-1}(T_1^*) = \infty) > 0$ and then $P(\tilde{e} = \infty) > 0$. This is impossible for a proper random variable.

Li et al. (2023) studied variable selection based on penalized weighted quantile regression [65]. The paper assumes that $Q_1(\tau|\mathbf{z}) = g[\tilde{\mathbf{z}}^T \boldsymbol{\beta}_0(\tau)]$. The penalized weighted objective function is minimized to estimate $\boldsymbol{\beta}_0(\tau)$:

$$\begin{aligned} Q_p[\boldsymbol{\beta}(\tau), w_i(\hat{F}_1)] = &\sum_{i=1}^{n} \{w_i(\hat{F}_1)\rho_\tau[g^{-1}(x_i) - \mathbf{z}_i^T \boldsymbol{\beta}(\tau)] + [1 - w_i(\hat{F}_1)]\rho_\tau[g^{-1}(x^\infty) - \mathbf{z}_i^T \boldsymbol{\beta}(\tau)]\} \\ &+ \sum_{m=1}^{p} p_\lambda(|\beta_m(\tau)|), \end{aligned}$$

where $x^\infty$ is any value sufficiently large to exceed all $\mathbf{z}_i^T \boldsymbol{\beta}(\tau)$, $\rho_\tau(u) = u[\tau - I(u \leq 0)]$ is called the "check" function, $\hat{F}_1(t) = \frac{1}{n} \sum_{i=1}^{n} \frac{I(x_i \leq t, \Delta_i \epsilon_i = 1)}{\hat{G}(x_i)}$ is the IPCW estimator for $F_1(t)$, $\hat{G}$ is the Kaplan–Meier estimator for the survival function of $C$ and $p_\lambda$ is a penalty function.

The estimated weights

$$w_i(\hat{F}_1) = \begin{cases} 1, & \Delta_i \epsilon_i = 1 \\ 0, & \Delta_i \epsilon_i \neq 1,\ \hat{F}_1(c_i) > \tau \\ \frac{\tau - \hat{F}_1(c_i)}{1 - \hat{F}_1(c_i)}, & \Delta_i \epsilon_i \neq 1,\ \hat{F}_1(c_i) \leq \tau. \end{cases}$$

However, it is unclear how to determine $x^{\infty}$. And, when $\Delta_i \epsilon_i > 1,\ c_i$ is unobserved so $\hat{F}_1(c_i)$ cannot be calculated thus $w_i(\hat{F}_1)$ cannot be calculated.

### 1.4.8 Regularized Weighted Nonparametric Likelihood Approach

Tapak et al. (2021) proposed a penalized weighted nonparametric likelihood approach [90] based on the model of Bellach et al. (2019) [6]. Let $l(\boldsymbol{\beta}, A_0)$ be the weighted log-likelihood function defined in Bellach et al. (2019) [6]. The regularized estimator is defined as

$$\hat{\boldsymbol{\beta}} = \arg\max[l(\boldsymbol{\beta}, A_0) - \sum_{m=1}^{p} p_\lambda(|\beta_m|)],$$

where $p_\lambda(\cdot)$ is a penalty function. $A_0$ is approximated by a sequence of step functions $(A_n^0)$ with jumps at the observed events of interest. In the original paper, only $g(x) = x$ and $g(x) = \log(1 + x)$ were considered. The maximization in this paper was utilized through the algorithm proposed by Goeman (2010) [40], which is a combination of gradient ascent optimization with the Newton-Raphson algorithm. This algorithm follows the gradient of the likelihood from a given starting value of $\boldsymbol{\beta}$. The algorithm automatically switches to a Newton-Raphson algorithm when it gets close to the optimum to avoid slow convergence.

## 1.5 Penalty Functions

### 1.5.1 Least Absolute Shrinkage and Selection Operator

Tibshirani (1996) proposed a method for penalized estimation in linear models and thus permits fitting over-parameterized models [93]. Let $y_1, ..., y_n$ be the responses. For a tuning parameter $s$, the least absolute shrinkage and selection operator (LASSO) is defined by

$$(\hat{\alpha}, \hat{\boldsymbol{\beta}}) = \text{argmin}\{\sum_{i=1}^{n}(y_i - \alpha - \sum_{m=1}^{p} z_{im}\beta_m)^2\}, \text{ subject to } \sum_{m=1}^{p} |\beta_m| \leq s.$$

Tibshirani (1997) then extended this method to the Cox model [94]. Let $l(\boldsymbol{\beta})$ be the log partial likelihood and assume that the $z_{im}$ are standardized so that $\sum_{i=1}^{n} z_{im} = 0$, $\sum_{i=1}^{n} z_{im}^2 = n$. The author proposed to estimate $\boldsymbol{\beta}$ via the criterion

$$\hat{\boldsymbol{\beta}} = \text{argmin } l(\boldsymbol{\beta}), \text{ subject to } \sum_{m=1}^{p} |\beta_j| \leq s.$$

Implementations of this model have included the R packages glmnet [33, 85] and glmpath [77], among others. The LASSO penalty was used in competing risks models in Ambrogi and Scheike (2016) [3], Fu, Parikh and Zhou (2017) [35] and Saadati et al. (2018) [81].

### 1.5.2 Smoothly Clipped Absolute Deviation Penalty

Fan and Li (2001) proposed the smoothly clipped absolute deviation (SCAD) penalty [29]. The continuous differentiable penalty function $p_\lambda(\beta)$ is defined by

$$p'_\lambda(\beta) = \lambda[I(\beta \leq \lambda) + \frac{(a\lambda - \beta)_+}{(a-1)\lambda} I(\beta > \lambda)]$$

for some $a > 2$ and $\beta > 0$. Fan and Li (2002) then extended this method to the Cox model [28] and obtained the penalized log partial likelihood

$$l(\boldsymbol{\beta}) - n \sum_{m=1}^{p} p_\lambda(|\beta_m|).$$

The SCAD penalty was used in competing risks models in Ambrogi and Scheike (2016) [3], Fu, Parikh and Zhou (2017) [35], Li, Tian and Tang (2019) [64], Kawaguchi et al. (2021) [53], implemented in the R package fastcmprsk, and Tapak et al. [90].

### 1.5.3 Smooth Integration of Counting and Absolute Deviation Penalties

Lv and Fan (2009) proposed the smooth integration of counting and absolute deviation (SICA) penalties [70]. For $a > 0$, the family of penalties are given by

$$p_\lambda(\beta) = \frac{\lambda(a+1)\beta}{a+\beta}, \ \beta \geq 0.$$

The SICA penalty has not been used in competing risks models.

### 1.5.4 Minimax Concave Penalty

Zhang (2010) proposed the minimax concave penalty (MCP) [104]. The MCP is defined as

$$p_\lambda(\beta) = \lambda \int_0^\beta (1 - \frac{x}{\gamma\lambda})_+ dx$$

with a regularization parameter $\gamma > 0$ for $\beta \geq 0$. The MCP penalty was used in competing risks models in Fu, Parikh and Zhou (2017) [35], Kawaguchi et al. (2021) [53], implemented in the R package fastcmprsk, and Tapak et al. [90].

### 1.5.5 Simulation Study Comparing Penalty Functions Applied to Cox Model

Bradic et al. (2011) conducted a simulation study comparing variable selection performance of LASSO, SCAD, SICA and MCP when applied to the Cox model [10]. LASSO always performed worse than SCAD, selecting fewer true positives and more false positives than SCAD, in the high-dimensional scenario, with 100 observations and 5000 predictors. Thus, only SCAD, SICA and MCP will be used in this dissertation.

## 1.6 Coordinate Descent Algorithm

Friedman et al. (2007) considered "one-at-a-time" coordinate-wise descent algorithms for a class of convex optimization problems including $L_1$ (LASSO)-penalized regression [34]. Simon et al. (2011) extended the method to Cox model, regularized by the elastic net penalty [85], which has been implemented in the R package glmnet. Breheny and Huang (2011) applied the algorithm for fitting linear and logistic regression with MCP and SCAD [11], implemented in the R package ncvreg. The coordinate descent algorithm will be used in this dissertation to fit penalized competing risks models using the SCAD, SICA, and MCP penalties.

## 1.7 Missing Data Imputation

In this section, methods for imputing missing data with both continuous and categorical variables, assuming the data are missing at random, that might be able to handle high-dimensioal data and have been implemented in R packages are reviewed. Let $D_s,\ s = 1, ..., q$, be $q$ possibly incomplete variables and $\mathbf{D} = (D_1, ..., D_q)$. Let $\mathbf{D}_{-s} = (D_1, ..., D_{s-1}, D_{s+1}, ..., D_q)$ denote

the collection of the $q-1$ variables in $\mathbf{D}$ except $D_s$. Denote $m_s$ the indices of the observations that are missing in variable $D_s$ and $o_s = \{1, ..., n\} \setminus m_s$.

### 1.7.1 Iterative Stepwise Regression Imputation

Templ, Kowarik and Filzmoser (2011) proposed an algorithm called IRMI for iterative model-based imputation using robust methods [92]. The algorithm can be summarized as follows:

1. Initialize the missing values using a simple imputation technique (e.g. k-nearest neighbor or mean imputation; the median is used by default).

2. Sort the variables according to the amount of missing values in decreasing order.

3. If the response is continuous, a robust regression method is applied; if the response is categorical, generalized linear regression is applied using all the other variables as covariates (optionally, a robust method can be selected). Optionally, it is possible to use a stepwise model selected by AIC, to include only the $k$ most important variables in the regression.

4. Estimate the regression coefficients with the corresponding model and use the estimated regression coefficients to replace the missing values.

5. Repeat steps 3-4 for each variable with missing values.

6. Repeat steps 3–5 until the imputed values stabilize.

There is an option to add a random error term to the imputed values, creating the possibility for multiple imputation. The error term has mean 0 and a variance corresponding to the (robust) variance of the regression residuals from the observations of the observed response. To provide adequate variances of the imputed data, the error term has to be multiplied by a factor $\sqrt{1 + \frac{|m_s|_0}{n}}$. Additionally, the level of noise can be controlled by a scale parameter. This method is implemented in the R package VIM [57].

### 1.7.2 MissForest

Stekhoven and Bühlmann (2012) proposed an iterative imputation method based on a random forest called missForest [86]. To begin, an initial guess for the missing values in $\mathbf{D}$ is made using mean imputation or another imputation method. Then, the variables $D_s,\ s = 1, ..., q$ are sorted according to the amount of missing values starting with the lowest amount. For each variable $D_s$, the missing values are imputed by first fitting a random forest with response $\mathbf{d}_s^{o_s}$ and predictors $\mathbf{d}_{-s}^{o_s}$; then, the missing values $\mathbf{d}_s^{m_s}$ are predicted by applying the trained random forest to $\mathbf{d}_{-s}^{m_s}$. The stopping criterion is met as soon as the difference between the newly imputed data matrix and the previous one increases for the first time with respect to both continuous and categorical variables. The difference for the set of continuous variables $\mathbf{N}$ is defined as

$$\Delta_{\mathbf{N}} = \frac{\sum_{s \in \mathbf{N}} (\mathbf{D}_{s,new}^{imp} - \mathbf{D}_{s,old}^{imp})^2}{\sum_{s \in \mathbf{N}} (\mathbf{D}_{s,new}^{imp})^2}$$

and for the set of categorical variables $\mathbf{F}$ as

$$\Delta_{\mathbf{F}} = \frac{\sum_{s \in \mathbf{F}} \sum_{i=1}^{n} I(\mathbf{D}_{i,s,new}^{imp} \neq \mathbf{D}_{i,s,old}^{imp})}{\#NA},$$

where $\#NA$ denotes the number of missing values in the categorical variables. It is a little unclear how exactly $\Delta_{\mathbf{N}}$ is calculated. This method is implemented in the R package missForest [87].

### 1.7.3 Fully Conditional Specification

van Buuren and Groothuis-Oudshoorn (2011) implemented the fully conditional specification method in the R package *mice* [95]. The MICE algorithm samples iteratively from condi-

tional distributions of the form $P(D_1|\mathbf{D}_{-1}, \theta_1), ..., P(D_q|\mathbf{D}_{-q}, \theta_q)$. The parameters $\theta_1, ..., \theta_q$ are specific to the respective conditional densities. Starting from a simple draw from observed marginal distributions, the $t$-th iteration of chained equations is a Gibbs sampler that successively draws

$$\begin{aligned}
\theta_1^{*(t)} &\sim P(\theta_1|D_1^{o_1}, D_2^{(t-1)}, ..., D_q^{(t-1)}) \\
D_1^{*(t)} &\sim P(D_1|D_1^{o_1}, D_2^{(t-1)}, ..., D_q^{(t-1)}, \theta_1^{*(t)}) \\
&\vdots \\
\theta_q^{*(t)} &\sim P(\theta_q|D_q^{o_q}, D_1^{(t)}, ..., D_{q-1}^{(t)}) \\
D_q^{*(t)} &\sim P(D_q|D_q^{o_q}, D_1^{(t)}, ..., D_{q-1}^{(t)}, \theta_q^{*(t)}),
\end{aligned}$$

where $D_s^{(t)} = (D_s^{o_s}, D_s^{*(t)})$ is the $s$-th imputed variable at iteration $t$. Note that in the paper $D_q^{*(t)} \sim P(D_q|D_q^{o_q}, D_1^{(t)}, ..., D_q^{(t)}, \theta_q^{*(t)})$, which may be a typo as $D_q^{(t)} = (D_q^{o_q}, D_q^{*(t)})$ and $D_q^{*(t)}$ is what is to be drawn. This method is also implemented in the mi package [88].

### 1.7.4 Multiple Imputation With Denoising Autoencoders

Lall and Robinson (2022) proposed an approach called Multiple Imputation with Denoising Autoencoders (MIDAS) [59]. MIDAS implements multiple imputation with the aid of artificial neural networks. A neural network consists of a series of nested nonlinear functions usually depicted as interconnected nodes organized in layers. Input data are fed into the network through an input layer, processed by nodes in one or more hidden layers, and returned via nodes in an output layer. The model for a "forward pass"—or computation of output values given input data—through layer $h$ of a neural network is:

$$\mathbf{y}^{(h)} = \sigma(\mathbf{W}^{(h)}\mathbf{y}^{(h-1)} + \mathbf{b}^{(h)}),$$

where $\mathbf{y}^{(h)}$ is a vector of outputs from layer $h$, $\mathbf{y}^{(0)}$ is the input, $\mathbf{W}^{(h)}$ is a matrix of weights connecting the nodes in layer $h-1$ with the nodes in layer $h$, $\mathbf{b}$ is a vector of biases for layer $h$, and $\sigma$ is a nonlinear activation function. It is unclear if $\mathbf{y}^{(0)}$ is the data from one subject or the data from one variable. The final-layer activation function is $\Phi$, which might be different from $\sigma$. The parameters are trained to minimize a loss function that measures the distance between actual and predicted outputs. Training involves four steps, collectively known as an epoch, which are repeated until some convergence criterion is met: (1) performing a forward pass through the network using current parameters; (2) calculating the loss function; (3) using the chain rule to calculate error gradients with respect to weights in each layer, a technique called backpropagation; and (4) adjusting weights in the direction of the negative gradient for the next forward pass.

One class of neural networks is the denoising autoencoders (DA). Classical autoencoders consist of two parts. First, an encoder deterministically maps an input vector $\mathbf{y}^{(0)}$ to a lower-dimensional representation $\mathbf{y}$ by compressing it through a series of shrinking hidden layers that culminate in a "bottleneck" layer. Second, a decoder maps $\mathbf{y}$ back to a reconstructed vector $\mathbf{v}$ with the same probability distribution and dimensions as $\mathbf{y}^{(0)}$ by passing it through a parallel series of expanding hidden layers culminating in the output layer. To map $\mathbf{v}$ as closely as possible to $\mathbf{y}^{(0)}$, weights are adjusted by backpropagation to minimize a loss function.

DA were developed to prevent autoencoders from learning an identical representation of the input while enabling them to extract more robust features from the data. They achieve these benefits by partially corrupting inputs through the injection of stochastic noise.

MIDAS modifies the standard DA model in two key ways. First, as part of the initial corruption process, it forces all missing values—in addition to a random subset of inputs—to 0. The task of the DA is thus to predict corrupted values that were both originally missing and originally observed using a loss function that only includes the latter. Second, to further

reduce the risk of overfitting, MIDAS regularizes the DA with the complementary technique of dropout. Dropout involves randomly removing nodes in the hidden layers of a network during training, typically by multiplying outputs from each of these layers by a Bernoulli vector. Dropout training proceeds by sampling an arbitrary number of "thinned" networks, with a different set of nodes dropped in each iteration. To produce multiple imputations, MIDAS samples multiple thinned networks. The default activation function is exponential linear unit. The final-layer activation function is chosen according to the distribution of the input data, with identity and softmax functions assigned to continuous and categorical variables, respectively. MIDAS employs root mean squared error (RMSE) and cross-entropy loss functions for continuous and categorical variables, respectively.

The algorithm proceeds in three stages. In the first stage, the input data are prepared for training. Categorical variables are converted into separate dummy variables for each unique class and continuous variables are rescaled between 0 and 1. A missingness indicator matrix $\mathbf{M}$ is constructed for the input data. All missing values are set to 0. A DA is then initialized according to the dimensions of the data; the default architecture is a three-layer network with 256 nodes per layer. In the training stage, the following five steps are repeated: (1) the input data and $\mathbf{M}$ are shuffled and sliced rowwise into paired mini-batches to accelerate convergence; (2) mini-batch inputs are partially corrupted through multiplication by a Bernoulli vector with default probability of taking the value 1 set to 0.8; (3) outputs from half of the nodes in hidden layers are corrupted using the same procedure; (4) a forward pass through the DA is conducted and the reconstruction error on predictions of the corrupted values that were originally observed is calculated using the loss functions; and (5) loss values are aggregated into a single term and backpropagated through the DA, with the resulting error gradients used to adjust weights for the next epoch. Finally, once training is complete, the whole of the input data is passed into the DA, which attempts to reconstruct all corrupted values. A completed dataset is then constructed by replacing the missing values with

predictions from the network's output. This stage is repeated multiple times. This method is implemented in the R package rMIDAS [80].

## 1.8 Variable Selection on Multiply Imputed Data

In this section, methods of selecting variables on multiple imputed datasets when there are missing values are reviewed.

### 1.8.1 Multiple Imputation-Least Absolute Shrinkage and Selection Operator

Chen and Wang (2013) proposed a multiple imputation-least absolute shrinkage and selection operator (MI-LASSO) variable selection method as an extension of the LASSO method to multiply-imputed data [14]. The linear regression model is assumed:

$$Y_i = \beta_0 + \sum_{j=1}^{p} z_{ij}\beta_j + \epsilon_i,\ i = 1, ..., n$$

Let $m$ be the number of imputed datasets. Let $\hat{\beta}_{1,j}, ..., \hat{\beta}_{m,j}$ denote the estimated coefficients for $z_j$ on the $m$ imputed datasets. The following function is minimized to estimate the coefficients

$$\sum_{d=1}^{m}\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p} z_{d,ij}\beta_{d,j})^2 + \lambda\sum_{j=1}^{p}\sqrt{\sum_{d=1}^{m}\beta_{d,j}^2}.$$

The penalty function used is called the group LASSO penalty. The estimated coefficients $(\hat{\beta}_{1,j}, ..., \hat{\beta}_{m,j})$ for each covariate $z_j$ will either be all exactly zero or be all nonzero. The local quadratic-approximation method is used to solve the optimization problem. Suppose we already have the estimates $\hat{\beta}_{d,j}^{(t)},\ d = 1, ..., m$ at the $t$-th iteration. As long as $\sum_{d=1}^{m}(\hat{\beta}_{d,j}^{(t)})^2 > 0$,

we have the following approximation

$$\sqrt{\sum_{d=1}^{m} \beta_{d,j}^2} \approx \frac{\sum_{d=1}^{m} \beta_{d,j}^2}{\sqrt{\sum_{d=1}^{m} (\hat{\beta}_{d,j}^{(t)})^2}}.$$

Then the optimization problem can be approximated by

$$\sum_{d=1}^{m} \left[ \sum_{i=1}^{n} (y_{d,i} - \beta_0 - \sum_{j=1}^{p} z_{d,ij} \beta_{d,j})^2 + \lambda \sum_{j=1}^{p} \frac{\beta_{d,j}^2}{\sqrt{\sum_{d=1}^{m} (\hat{\beta}_{d,j}^{(t)})^2}} \right].$$

It can be seen that the estimated coefficients can be obtained by solving $m$ separate ridge regressions. The iterations continue until convergence. One possible limitation for this approximation is that once a group of coefficients are shrunken to zero, they will stay at zero. To avoid this inflexibility, $\hat{\beta}_{1,j}^{(t)}, ..., \hat{\beta}_{m,j}^{(t)}$ are set to $10^{-10}$ when $\sum_{d=1}^{m} (\hat{\beta}_{d,j}^{(t)})^2 \leq m10^{-20}$. This method has a couple of drawbacks. First, it is unclear what the initial estimates are obtained. Second, ridge regression does not force coefficients to 0 so it is unclear how variables are selected. Third, setting $\hat{\beta}_{1,j}^{(t)}, ..., \hat{\beta}_{m,j}^{(t)}$ to $10^{-10}$ when $\sum_{d=1}^{m} (\hat{\beta}_{d,j}^{(t)})^2 \leq m10^{-20}$ forces coefficients to always be nonzero, preventing variables from being excluded from the model. Du et al. (2022) also considered another penalty, called the group adaptive LASSO penalty:

$$\lambda \sum_{j=1}^{p} \hat{a}_j \sqrt{\sum_{d=1}^{m} \beta_{d,j}^2}$$

where $\hat{a}_j = (\sqrt{\sum_{d=1}^{m} \hat{\beta}_{d,j}^2} + \frac{1}{nm})^{-\gamma}$, $\gamma = \lceil \frac{2\nu}{1-\nu} + 1 \rceil$ and $\nu = \frac{\log(pm)}{\log(nm)}$ [24]. $\hat{\beta}_{d,j}$ is estimated using the group LASSO penalty.

### 1.8.2 Stability Selection Combined With Bootstrap Imputation

Long and Johnson (2015) proposed a resampling approach that combines bootstrap imputation and stability selection in the linear regression setting [68]. First, a series of bootstrap datasets $\{(\mathbf{y}^{(b)}, \mathbf{Z}^{(b)}), b = 1, ..., B\}$ are generated of the original data. Then, for each bootstrapped dataset, impute the missing values in $\mathbf{Z}^{(b)}$ and the resultant imputed datasets are denoted by $\{(\mathbf{y}^{(b)}, \mathbf{Z}_I^{(b)}), b = 1, ..., B\}$. For the $b$-th imputed dataset, find $\hat{\boldsymbol{\beta}}_\lambda^{(b)}$ that minimizes

$$||\mathbf{y}^{(b)} - \mathbf{Z}_I^{(b)}\boldsymbol{\beta}||_2^2 + \lambda \sum_{j=1}^{p} \frac{|\beta_j|}{w_j^{(b)}},$$

where $w_j^{(b)}$'s are independently identically distributed random variables in $[\alpha, 1]$ with $\alpha \in (0, 1)$. Denote the set of non-zero parameter estimates in $\hat{\boldsymbol{\beta}}^{(b)}$ by $\hat{S}_\lambda^{(b)}$. Let $\Lambda$ denote a set of feasible values for $\lambda$. The final estimated active set is defined as

$$\hat{S}_\pi = \{j : \max_{\lambda \in \Lambda}(\Pi_j^\lambda) \geq \pi\},$$

where $\Pi_j^\lambda = \frac{\sum_{b=1}^{B} I(j \in \hat{S}_\lambda^{(b)})}{B}$ and $\pi \in (0, 1)$ is a threshold for selecting a predictor and is often set to between 0.6 and 0.9 in practice.

### 1.8.3 Multiple Imputation-based Weighted Elastic Net

Wan et al. (2015) proposed a multiple imputation-based weighted elastic net method based on stacked MI data and a weighting scheme for each observation in the stacked data set in the linear regression setting [98]. Let $m$ be the number of imputed datasets. It minimizes the following function:

$$\frac{1}{2n} \sum_{i=1}^{n} \sum_{d=1}^{m} w_i (y_i - \beta_0 - \sum_{j=1}^{p} z_{d,ij}\beta_j)^2 + \lambda P_\alpha(\boldsymbol{\beta}),$$

where $P_\alpha(\boldsymbol{\beta}) = \alpha \sum_{j=1}^{p} |\beta_j| + \frac{1-\alpha}{2} \sum_{j=1}^{p} \beta_j^2$ and $w_i = \frac{f_i}{m}$, where $f_i$ is the fraction of observed values for subject $i$.

Du et al. (2022) also considered setting $w_i = \frac{1}{m}$ and the stacked elastic net penalty

$$\alpha \sum_{j=1}^{p} |\beta_j| + (1-\alpha) \sum_{j=1}^{p} \beta_j^2$$

and the stacked adaptive elastic net penalty

$$\alpha \sum_{j=1}^{p} \hat{a}_j |\beta_j| + (1-\alpha) \sum_{j=1}^{p} \beta_j^2,$$

where $\hat{a}_j = (\sqrt{\sum_{d=1}^{m} |\hat{\beta}_j|} + \frac{1}{nm})^{-\gamma}$, $\gamma = \lceil \frac{2\nu}{1-\nu} + 1 \rceil$ and $\nu = \frac{\log(p)}{\log(nm)}$ [24]. $\hat{\beta}_j$ is estimated using the stacked elastic net penalty.

### 1.8.4 Multiple Imputation Random LASSO

Liu et al. (2016) propose a multiple imputation random LASSO method in the linear regression setting [67]. Impute the dataset $m$ times. For each imputed data set, generate $B$ bootstrap samples. For the $b$-th bootstrap sample in the $i$-th imputation, apply lasso-OLS to obtain estimates $\hat{\beta}_{ij}^{(b)}$ for $\beta_j$, $i = 1, .., m$, $j = 1, ..., p$. Compute the importance measure of variable $z_j$ by

$$I_j = \frac{|\sum_{i=1}^{m} \sum_{b=1}^{B} \hat{\beta}_{ij}^{(b)}|}{mB}.$$

For the $b$-th bootstrap sample, randomly select $\lceil p/2 \rceil$ candidate variables with selection probability of $z_j$ proportional to its importance measure $I_j$. Let $\Lambda$ be a grid of $K$ exponential decaying sequence of tuning parameters $\lambda$'s, apply lasso-OLS to obtain estimates $\hat{\beta}_{ij\lambda}^{(b)}$ for

$\beta_j,\ j = 1, ..., p,\ \lambda \in \Lambda$. Then calculate the empirical probability

$$\hat{\Pi}_j^{\lambda} = \frac{\sum_{i=1}^{m} \sum_{b=1}^{B} I(\hat{\beta}_{ij\lambda}^{(b)} \neq 0)}{mB},$$

where $\hat{\beta}_{ij\lambda}^{(b)} = 0$ if varaible $j$ is not sampled. The important variables are those in the stable variable set: $\{j : \max_{\lambda \in \Lambda} \hat{\Pi}_j^{\lambda} \geq \pi_{thr}\}$, where a threshold $\pi_{thr}$ is chosen by cross-validation with the one-standard error rule. The lasso-OLS estimator is a two-step procedure. First, compute the lasso estimator, where the tuning parameter $\lambda$ is chosen from cross-validation. Next, the lasso-OLS estimator is the ordinary least squares estimator obtained by regressing the outcome on the subset of variables chosen by lasso.

### 1.8.5 Use the Magnitude of the Parameter Estimates for Selection

Zahid et al. (2020) proposed to use the magnitude of the parameter estimates of each candidate predictor across all the imputed datasets for its selection [103]. LASSO regression (or some other variable selection technique) is fit to each imputed dataset. All those predictors which appear in all models of the $m$ imputed datasets are selected. For the rest of the predictors, a continuous predictor $z_j$ is selected if

$$\frac{\sum_{d=1}^{m} |\hat{\beta}_{d,j}|}{\sum_{j=1}^{p} \sum_{d=1}^{m} |\hat{\beta}_{d,j}|} \geq \frac{1}{p}.$$

For a categorical predictor $z_j$ with $Ca_j + 1$ categories, $Ca_j$ dummy variables are created. Let $\beta_{d,jk}$ be the parameter associated with the $k$-th dummy variable. $z_j$ is selected if

$$\frac{\sum_{k=1}^{Ca_j} \sum_{d=1}^{m} |\hat{\beta}_{d,jk}|}{\sum_{j=1}^{p} \frac{\sum_{k=1}^{Ca_j} \sum_{d=1}^{m} |\hat{\beta}_{d,jk}|}{Ca_j}} \geq \frac{1}{p}.$$

It is unclear if continuous predictors are included in the calculation of the denominator.

## 1.9 Summary

In this chapter the proportional hazards model, competing risks models, methods to select variables for competing risks data, relevant penalty functions, coordinate descent algorithm, methods for imputing missing data in high-dimensional data and methods for selecting variables when the data contain missing values were reviewed. In Chapter 2, a penalized proportional hazards model with cross-validation is proposed for variable selection. In Chapter 3, the method proposed in Chapter 2 is evaluated by simulations and applied to the AML dataset. In Chapter 4, the penalized proportional hazards model with cross-validation method is extended to the situation when there are missing values in the data. In Chapter 5, the method proposed in Chapter 4 is evaluated by simulations and applied to the AML dataset. In Chapter 6, we comment on the proposed methods and possible direction of future research.

# Chapter 2: Penalized Proportional Subdistribution Hazards Model Selecting Tuning Parameter With Cross-validation

## 2.1 Proposed Method

In this chapter, a method is proposed to select predictors based on the proportional subdistribution hazards model in Fine and Gray (1999) [32] for high-dimensional data. We choose to focus on the proportional subdistribution hazards model because it directly models the cumulative incidence function thus the predictor effects have relatively simple interpretation. As before, let $n$ be the sample size, $T$ and $C$ be the failure and censoring times, respectively, $X = \min(T, C),\ \Delta = I(T \leq C),\ \mathbf{z}$ be a length-$p$ covariate vector and $\epsilon \in (1, ..., K)$ be the cause of failure. Assuming cause 1 is the cause of interest, the model is solved by the following equation

$$\mathbf{U}(\boldsymbol{\beta}) := \sum_{i=1}^{n} \int_0^{\infty} \left[ \mathbf{z}_i - \frac{\sum_{m=1}^{n} w_m(s) Y_m(s) \mathbf{z}_m \exp(\mathbf{z}_m^T \boldsymbol{\beta})}{\sum_{m=1}^{n} w_m(s) Y_m(s) \exp(\mathbf{z}_m^T \boldsymbol{\beta})} \right] w_i(s) dN_i(s) = \mathbf{0}, \tag{2.1}$$

where $N_i(t) = I(T_i \leq t, \epsilon_i = 1)$, $Y_i(t) = 1 - N_i(t-)$, $w_i(t) = r_i(t)\hat{G}(t)/\hat{G}(X_i \wedge t)$ is the weight associated with individual $i$, $r_i(t) = I(C_i \geq T_i \wedge t)$ denotes knowledge of vital status on

individual $i$ at time $t$ and $\hat{G}$ is the Kaplan-Meier estimate of the survival function of $C$.

Let's simplify $\mathbf{U}(\boldsymbol{\beta})$ first. If $\Delta_i = 0$, then $N_i(t) = 0$ when $t \leq C_i$ and $r_i(t) = 0$ when $t > C_i$ thus $w_i(t) = 0$. If $\Delta_i = 1$ but $\epsilon_i \neq 1$, then $N_i(t) = 0,\ t \in [0, \infty)$. Otherwise, $N_i(t) = I_{[T_i,\infty)}(t)$. As long as no observation is censored at any $T_i,\ s.t.\ \Delta_i\epsilon_i = 1$, which is usually the case, the integrand in each of the integrals in the expression of $\mathbf{U}(\boldsymbol{\beta})$ is left-continuous at any $T_i,\ s.t.\ \Delta_i\epsilon_i = 1$. Then

$$\mathbf{U}(\boldsymbol{\beta}) = \sum_{\Delta_i\epsilon_i=1} \left[ \mathbf{z}_i - \frac{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \mathbf{z}_m \exp(\mathbf{z}_m^T \boldsymbol{\beta})}{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})} \right]. \tag{2.2}$$

Now define $l(\boldsymbol{\beta}) = \sum_{\Delta_i\epsilon_i=1} \{\mathbf{z}_i^T \boldsymbol{\beta} - \log[\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})]\}$. Let $\nabla l(\boldsymbol{\beta})$ be the gradient of $l(\boldsymbol{\beta})$. We can see that $\nabla l(\boldsymbol{\beta}) = \mathbf{U}(\boldsymbol{\beta})$. We call $l(\boldsymbol{\beta})$ the log-partial likelihood function. Now we prove that $l(\boldsymbol{\beta})$ is concave.

**Theorem 3.** *$l(\boldsymbol{\beta})$ is concave.*

*Proof.* For any $h = 1, ..., p$

$$\begin{aligned}
\frac{\partial l}{\partial \beta_h} &= \sum_{\Delta_i\epsilon_i=1} \left[ z_{ih} - \frac{\sum_{s=1}^{n} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) z_{sh}}{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})} \right] \\
&= \sum_{\Delta_i\epsilon_i=1} \left[ z_{ih} - \sum_{s=1}^{n} z_{sh} \frac{w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta})}{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})} \right] \\
&= \sum_{\Delta_i\epsilon_i=1} [z_{ih} - \sum_{s=1}^{n} z_{sh} v_{is}(\boldsymbol{\beta})] \\
&= \sum_{\Delta_i\epsilon_i=1} [z_{ih} - \bar{z}_{ih}(\boldsymbol{\beta})],
\end{aligned}$$

where $v_{is}(\boldsymbol{\beta}) = \frac{w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta})}{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})}$ and $\bar{z}_{ih}(\boldsymbol{\beta}) = \sum_{s=1}^{n} z_{sh} v_{is}(\boldsymbol{\beta})$.

For any $q = 1, ..., p$,

$$
\begin{aligned}
\frac{\partial^2 l}{\partial \beta_h \partial \beta_q} &= -\sum_{\Delta_i \epsilon_i = 1} \sum_{s=1}^{n} z_{sh} \frac{\partial v_{is}(\boldsymbol{\beta})}{\partial \beta_q} \\
&= -\sum_{\Delta_i \epsilon_i = 1} \sum_{s=1}^{n} z_{sh} \{ \frac{w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) z_{sq} \sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})}{[\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})]^2} \\
&\quad - \frac{w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) \sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta}) z_{mq}}{[\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})]^2} \} \\
&= -\sum_{\Delta_i \epsilon_i = 1} \{ \frac{\sum_{s=1}^{n} z_{sh} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) z_{sq}}{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})} \\
&\quad - \frac{[\sum_{s=1}^{n} z_{sh} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta})][\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta}) z_{mq}]}{[\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})]^2} \} \\
&= -\sum_{\Delta_i \epsilon_i = 1} \left[ \frac{\sum_{s=1}^{n} z_{sh} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) z_{sq}}{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})} - \bar{z}_{ih}(\boldsymbol{\beta}) \bar{z}_{iq}(\boldsymbol{\beta}) \right] \\
&= -\sum_{\Delta_i \epsilon_i = 1} [ \frac{\sum_{s=1}^{n} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) z_{sh} z_{sq}}{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})} - \\
&\quad \frac{\bar{z}_{ih}(\boldsymbol{\beta}) \bar{z}_{iq}(\boldsymbol{\beta}) \sum_{s=1}^{n} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta})}{\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\mathbf{z}_m^T \boldsymbol{\beta})} ].
\end{aligned}
$$

Note that

$$
\begin{aligned}
&\sum_{s=1}^{n} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) [z_{sh} - \bar{z}_{ih}(\boldsymbol{\beta})][z_{sq} - \bar{z}_{iq}(\boldsymbol{\beta})] \\
&= \sum_{s=1}^{n} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) z_{sh} z_{sq} - \sum_{s=1}^{n} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) z_{sh} \bar{z}_{iq}(\boldsymbol{\beta}) \\
&\quad - \sum_{s=1}^{n} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) \bar{z}_{ih}(\boldsymbol{\beta}) z_{sq} + \sum_{s=1}^{n} w_s(T_i) Y_s(T_i) \exp(\mathbf{z}_s^T \boldsymbol{\beta}) \bar{z}_{ih}(\boldsymbol{\beta}) \bar{z}_{iq}(\boldsymbol{\beta}).
\end{aligned}
$$

Also note that

$$\begin{aligned}\bar{z}_{ih}(\boldsymbol{\beta}) &= \sum_{s=1}^{n} z_{sh} \frac{w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta})}{\sum_{m=1}^{n} w_m(T_i)Y_m(T_i)\exp(\mathbf{z}_m^T\boldsymbol{\beta})} \\ &= \frac{\sum_{s=1}^{n} z_{sh} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta})}{\sum_{m=1}^{n} w_m(T_i)Y_m(T_i)\exp(\mathbf{z}_m^T\boldsymbol{\beta})}.\end{aligned}$$

So $\sum_{s=1}^{n} z_{sh} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta}) = \bar{z}_{ih}(\boldsymbol{\beta})\sum_{m=1}^{n} w_m(T_i)Y_m(T_i)\exp(\mathbf{z}_m^T\boldsymbol{\beta})$.

Then

$$\begin{aligned}&\sum_{s=1}^{n} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta})[z_{sh} - \bar{z}_{ih}(\boldsymbol{\beta})][z_{sq} - \bar{z}_{iq}(\boldsymbol{\beta})] \\ =& \sum_{s=1}^{n} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta}) z_{sh} z_{sq} - \bar{z}_{iq}(\boldsymbol{\beta})\bar{z}_{ih}(\boldsymbol{\beta})\sum_{s=1}^{n} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta}) \\ &- \bar{z}_{ih}(\boldsymbol{\beta})\bar{z}_{iq}(\boldsymbol{\beta})\sum_{s=1}^{n} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta}) + \bar{z}_{ih}(\boldsymbol{\beta})\bar{z}_{iq}(\boldsymbol{\beta})\sum_{s=1}^{n} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta}) \\ =& \sum_{s=1}^{n} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta}) z_{sh} z_{sq} - \bar{z}_{ih}(\boldsymbol{\beta})\bar{z}_{iq}(\boldsymbol{\beta})\sum_{s=1}^{n} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta}).\end{aligned}$$

So

$$\begin{aligned}\frac{\partial^2 l}{\partial\beta_h\partial\beta_q} &= -\sum_{\Delta_i\epsilon_i=1} \frac{\sum_{s=1}^{n} w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta})[z_{sh} - \bar{z}_{ih}(\boldsymbol{\beta})][z_{sq} - \bar{z}_{iq}(\boldsymbol{\beta})]}{\sum_{m=1}^{n} w_m(T_i)Y_m(T_i)\exp(\mathbf{z}_m^T\boldsymbol{\beta})} \\ &= -\sum_{\Delta_i\epsilon_i=1}\sum_{s=1}^{n} \frac{w_s(T_i)Y_s(T_i)\exp(\mathbf{z}_s^T\boldsymbol{\beta})}{\sum_{m=1}^{n} w_m(T_i)Y_m(T_i)\exp(\mathbf{z}_m^T\boldsymbol{\beta})}[z_{sh} - \bar{z}_{ih}(\boldsymbol{\beta})][z_{sq} - \bar{z}_{iq}(\boldsymbol{\beta})] \\ &= -\sum_{\Delta_i\epsilon_i=1}\sum_{s=1}^{n} v_{is}(\boldsymbol{\beta})[z_{sh} - \bar{z}_{ih}(\boldsymbol{\beta})][z_{sq} - \bar{z}_{iq}(\boldsymbol{\beta})].\end{aligned}$$

Then the Hessian matrix of $l(\boldsymbol{\beta})$

$$\mathbf{H}_l(\boldsymbol{\beta}) = -\sum_{\Delta_i\epsilon_i=1}\sum_{s=1}^{n} v_{is}(\boldsymbol{\beta})[\mathbf{z}_s - \bar{\mathbf{z}}_i(\boldsymbol{\beta})][\mathbf{z}_s - \bar{\mathbf{z}}_i(\boldsymbol{\beta})]^T,$$

where $\bar{\mathbf{z}}_i(\boldsymbol{\beta}) = (\bar{z}_{i1}(\boldsymbol{\beta}), ..., \bar{z}_{ip}(\boldsymbol{\beta}))$. $[\mathbf{z}_s - \bar{\mathbf{z}}_i(\boldsymbol{\beta})][\mathbf{z}_s - \bar{\mathbf{z}}_i(\boldsymbol{\beta})]^T$ is positive-semidefinite, $v_{is}(\boldsymbol{\beta}) \geq 0$ for any $s = 1, ..., n$ and $i$ s.t. $\Delta_i \epsilon_i = 1$. Thus $-\mathbf{H}_l(\boldsymbol{\beta})$ is positive-semidefinite. Then $\mathbf{H}_l(\boldsymbol{\beta})$ is negative-semidefinite. Therefore $l(\boldsymbol{\beta})$ is concave. □

Then $\hat{\boldsymbol{\beta}}$ is a global maximum point of $l(\boldsymbol{\beta})$ if and only if $\nabla l(\boldsymbol{\beta}) = \mathbf{0}$, i.e. $\mathbf{U}(\boldsymbol{\beta}) = \mathbf{0}$. So to solve $\mathbf{U}(\boldsymbol{\beta}) = \mathbf{0}$ is equivalent to maximizing $l(\boldsymbol{\beta})$. To select important predictors, we maximize $l(\boldsymbol{\beta}) - n\sum_{h=1}^{p} p(\beta_h)$, where $p(\beta)$ is a penalty function. $l(\boldsymbol{\beta})$ is a relatively complicated function so it will be approximated by Taylor polynomial.

Given an approximate value of $\boldsymbol{\beta}$, $\tilde{\boldsymbol{\beta}}$, $l(\boldsymbol{\beta})$ can be approximated by $l(\tilde{\boldsymbol{\beta}}) + (\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})^T \nabla l(\tilde{\boldsymbol{\beta}}) + \frac{1}{2}(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})^T \boldsymbol{H}_l(\tilde{\boldsymbol{\beta}})(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})$. Define $l_\eta(\boldsymbol{\eta}) := \sum_{\Delta_i \epsilon_i = 1}\{\eta_i - \log[\sum_{m=1}^{n} w_m(T_i) Y_m(T_i) \exp(\eta_m)]\}$. Let $\mathbf{Z}$ be the design matrix. We can see that $l(\boldsymbol{\beta}) = l_\eta(\mathbf{Z}\boldsymbol{\beta})$. Using chain rule, we can show that $\nabla l(\tilde{\boldsymbol{\beta}}) = \mathbf{Z}^T \nabla l_\eta(\tilde{\boldsymbol{\eta}})$ and $\boldsymbol{H}_l(\tilde{\boldsymbol{\beta}}) = \mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}$, where $\tilde{\boldsymbol{\eta}} = \mathbf{Z}\tilde{\boldsymbol{\beta}}$, $\nabla l_\eta(\boldsymbol{\eta})$ is the gradient of $l_\eta(\boldsymbol{\eta})$ and $\boldsymbol{H}_{l_\eta}(\boldsymbol{\eta})$ is the Hessian matrix of $l_\eta(\boldsymbol{\eta})$. So $l(\boldsymbol{\beta})$ can be approximated by $l(\tilde{\boldsymbol{\beta}}) + (\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}})^T \nabla l_\eta(\tilde{\boldsymbol{\eta}}) + \frac{1}{2}(\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})(\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}})$.

To simplify the expression, notice that

$$
\begin{aligned}
&(\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}})^T \nabla l_\eta(\tilde{\boldsymbol{\eta}}) + \frac{1}{2}(\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})(\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}}) \\
&= (\boldsymbol{\beta}^T \mathbf{Z}^T - \tilde{\boldsymbol{\eta}}^T)\nabla l_\eta(\tilde{\boldsymbol{\eta}}) + \frac{1}{2}(\boldsymbol{\beta}^T \mathbf{Z}^T - \tilde{\boldsymbol{\eta}}^T)[\boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} - \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\tilde{\boldsymbol{\eta}}] \\
&= \boldsymbol{\beta}^T \mathbf{Z}^T \nabla l_\eta(\tilde{\boldsymbol{\eta}}) - \tilde{\boldsymbol{\eta}}^T \nabla l_\eta(\tilde{\boldsymbol{\eta}}) + \frac{1}{2}\boldsymbol{\beta}^T \mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} - \frac{1}{2}\boldsymbol{\beta}^T \mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\tilde{\boldsymbol{\eta}} - \frac{1}{2}\tilde{\boldsymbol{\eta}}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} \\
&\quad + \frac{1}{2}\tilde{\boldsymbol{\eta}}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\tilde{\boldsymbol{\eta}} \\
&= \frac{1}{2}\boldsymbol{\beta}^T \mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} + [\nabla l_\eta(\tilde{\boldsymbol{\eta}})^T \mathbf{Z} - \tilde{\boldsymbol{\eta}}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}]\boldsymbol{\beta} + \frac{1}{2}\tilde{\boldsymbol{\eta}}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\tilde{\boldsymbol{\eta}} - \tilde{\boldsymbol{\eta}}^T \nabla l_\eta(\tilde{\boldsymbol{\eta}}).
\end{aligned}
$$

When defining $\mathbf{y}(\tilde{\boldsymbol{\eta}}) = \tilde{\boldsymbol{\eta}} - \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})^{-1}\nabla l_\eta(\tilde{\boldsymbol{\eta}})$,

$$
\begin{aligned}
&\frac{1}{2}[\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{Z}\boldsymbol{\beta}]^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})[\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{Z}\boldsymbol{\beta}] \\
&= \frac{1}{2}[\mathbf{y}(\tilde{\boldsymbol{\eta}})^T - \boldsymbol{\beta}^T\mathbf{Z}^T][\boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta}] \\
&= \frac{1}{2}[\mathbf{y}(\tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{y}(\tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} - \boldsymbol{\beta}^T\mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{y}(\tilde{\boldsymbol{\eta}}) + \boldsymbol{\beta}^T\mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta}] \\
&= \frac{1}{2}\boldsymbol{\beta}^T\mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} - \mathbf{y}(\tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} + \frac{1}{2}\mathbf{y}(\tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{y}(\tilde{\boldsymbol{\eta}}) \\
&= \frac{1}{2}\boldsymbol{\beta}^T\mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} - [\tilde{\boldsymbol{\eta}} - \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})^{-1}\nabla l_\eta(\tilde{\boldsymbol{\eta}})]^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} + \frac{1}{2}\mathbf{y}(\tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{y}(\tilde{\boldsymbol{\eta}}) \\
&= \frac{1}{2}\boldsymbol{\beta}^T\mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} - [\tilde{\boldsymbol{\eta}}^T - \nabla l_\eta(\tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})^{-1}]\boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} + \frac{1}{2}\mathbf{y}(\tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{y}(\tilde{\boldsymbol{\eta}}) \\
&= \frac{1}{2}\boldsymbol{\beta}^T\mathbf{Z}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}\boldsymbol{\beta} + [\nabla l_\eta(\tilde{\boldsymbol{\eta}})^T\mathbf{Z} - \tilde{\boldsymbol{\eta}}^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{Z}]\boldsymbol{\beta} + \frac{1}{2}\mathbf{y}(\tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})\mathbf{y}(\tilde{\boldsymbol{\eta}}).
\end{aligned}
$$

Then $l(\tilde{\boldsymbol{\beta}}) + (\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}})^T\nabla l_\eta(\tilde{\boldsymbol{\eta}}) + \frac{1}{2}(\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}})^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})(\mathbf{Z}\boldsymbol{\beta} - \tilde{\boldsymbol{\eta}}) = \frac{1}{2}[\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{Z}\boldsymbol{\beta}]^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})[\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{Z}\boldsymbol{\beta}] + C(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\eta}})$, where $C(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\eta}})$ is a function of $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\eta}}$. So maximizing $l(\boldsymbol{\beta}) - n\sum_{h=1}^{p} p(\beta_h)$ is approximately equivalent to maximizing $\frac{1}{2}[\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{Z}\boldsymbol{\beta}]^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})[\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{Z}\boldsymbol{\beta}] - n\sum_{h=1}^{p} p(\beta_h)$. We replace $\boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})$ by a diagonal matrix with the $i$-th diagonal entry $h_i(\tilde{\boldsymbol{\eta}}) = (\boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}}))_{ii}$ to speed up the computation. Then $\frac{1}{2}[\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{Z}\boldsymbol{\beta}]^T \boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})[\mathbf{y}(\tilde{\boldsymbol{\eta}}) - \mathbf{Z}\boldsymbol{\beta}] - n\sum_{h=1}^{p} p(\beta_h)$ becomes $f(\boldsymbol{\beta}) := \frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T\boldsymbol{\beta}]^2 - n\sum_{h=1}^{p} p(\beta_h)$.

SCAD, MCP and SICA penalty functions will be used. SCAD, MCP and SICA are all nonconvex penalty functions and other nonconvex penalty functions could also be applied. In the following subsections, each penalty function is described.

### 2.1.1 SCAD

The SCAD penalty function is defined in [29] as

$$p(\beta) = \begin{cases} \frac{(a+1)\lambda^2}{2}, & \beta < -a\lambda \\ \frac{-\beta^2 - 2a\lambda\beta - \lambda^2}{2(a-1)}, & -a\lambda \leq \beta < -\lambda \\ -\lambda\beta, & -\lambda \leq \beta < 0 \\ \lambda\beta, & 0 \leq \beta < \lambda \\ \frac{-\beta^2 + 2a\lambda\beta - \lambda^2}{2(a-1)}, & \lambda \leq \beta < a\lambda \\ \frac{(a+1)\lambda^2}{2}, & \beta \geq a\lambda \end{cases}$$

for some $a > 2$. Then

$$p'(\beta) = \begin{cases} 0, & \beta < -a\lambda \\ -\frac{a\lambda+\beta}{a-1}, & -a\lambda \leq \beta < -\lambda \\ -\lambda, & -\lambda \leq \beta < 0 \\ \lambda, & 0 < \beta \leq \lambda \\ \frac{a\lambda-\beta}{a-1}, & \lambda < \beta \leq a\lambda \\ 0, & \beta > a\lambda \end{cases}$$

$$p''(\beta) = \begin{cases} 0, & \beta < -a\lambda \\ -\frac{1}{a-1}, & -a\lambda \leq \beta < -\lambda \\ 0, & -\lambda \leq \beta < 0 \\ 0, & 0 < \beta \leq \lambda \\ -\frac{1}{a-1}, & \lambda < \beta \leq a\lambda \\ 0, & \beta > a\lambda \end{cases}$$

### 2.1.2 MCP

The MCP penalty function is defined in [104] as

$$p(\beta) = \begin{cases} \frac{\gamma\lambda^2}{2}, & \beta < -\gamma\lambda \\ -\lambda(\beta + \frac{\beta^2}{2\gamma\lambda}), & -\gamma\lambda \leq \beta < 0 \\ \lambda(\beta - \frac{\beta^2}{2\gamma\lambda}), & 0 \leq \beta < \gamma\lambda \\ \frac{\gamma\lambda^2}{2}, & \beta \geq \gamma\lambda \end{cases}$$

for some $\gamma > 0$. Then

$$p'(\beta) = \begin{cases} 0, & \beta < -\gamma\lambda \\ -\lambda(1 + \frac{\beta}{\gamma\lambda}), & -\gamma\lambda \leq \beta < 0 \\ \lambda(1 - \frac{\beta}{\gamma\lambda}), & 0 < \beta \leq \gamma\lambda \\ 0, & \beta > \gamma\lambda \end{cases}$$

$$p''(\beta) = \begin{cases} 0, & \beta < -\gamma\lambda \\ -\frac{1}{\gamma}, & -\gamma\lambda \leq \beta < 0 \\ -\frac{1}{\gamma}, & 0 < \beta \leq \gamma\lambda \\ 0, & \beta > \gamma\lambda \end{cases}$$

### 2.1.3 SICA

The SICA penalty function is defined in [70] as

$$p(\beta) = \begin{cases} -\frac{\lambda(a+1)\beta}{a-\beta}, & \beta < 0 \\ \frac{\lambda(a+1)\beta}{a+\beta}, & \beta \geq 0 \end{cases}$$

for some $a > 0$. Then

$$p'(\beta) = \begin{cases} -\frac{\lambda a(a+1)}{(a-\beta)^2}, & \beta < 0 \\ \frac{\lambda a(a+1)}{(a+\beta)^2}, & \beta > 0 \end{cases}$$

and

$$p''(\beta) = \begin{cases} -\frac{2\lambda a(a+1)}{(a-\beta)^3}, & \beta < 0 \\ -\frac{2\lambda a(a+1)}{(a+\beta)^3}, & \beta > 0 \end{cases}$$

The coordinate ascent algorithm will be used to maximize $f(\boldsymbol{\beta})$. Specifically, for each $q = 1, ..., p$, fix $\beta_k,\ k \neq q$ and maximize $f(\boldsymbol{\beta})$. To find the global maximum points of $f(\boldsymbol{\beta})$, we need to find the local maximum points first. To find possible local maximum points in $(-\infty, 0) \cup (0, \infty)$, we take the partial derivative of $f(\boldsymbol{\beta})$ with respect to $\beta_q$

$$\frac{\partial f(\boldsymbol{\beta})}{\partial \beta_q} = -\sum_{i=1}^{n} z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}] - np'(\beta_q)$$

For SCAD

$$\frac{\partial f(\boldsymbol{\beta})}{\partial \beta_q} = \begin{cases} -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}], & \beta_q < -a\lambda \\ -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}] + \frac{n(a\lambda+\beta_q)}{a-1}, & -a\lambda \le \beta_q < -\lambda \\ -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}] + n\lambda, & -\lambda \le \beta_q < 0 \\ -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}] - n\lambda, & 0 < \beta_q \le \lambda \\ -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}] - \frac{n(a\lambda-\beta_q)}{a-1}, & \lambda < \beta_q \le a\lambda \\ -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}], & \beta_q > a\lambda \end{cases}$$

Set $\frac{\partial f(\boldsymbol{\beta})}{\partial \beta_q} = 0$ and solve for $\beta_q$ while fixing $\beta_k,\ k \neq q$ gives

$$\hat{\beta}_q = \begin{cases} \frac{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]}{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2}, & \beta_q < -a\lambda \\ \frac{(a-1)\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - na\lambda}{(a-1)\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2 + n}, & -a\lambda \le \beta_q < -\lambda \\ \frac{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lambda}{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2}, & -\lambda \le \beta_q < 0 \\ \frac{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + n\lambda}{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2}, & 0 < \beta_q \le \lambda \\ \frac{(a-1)\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + na\lambda}{(a-1)\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2 + n}, & \lambda < \beta_q \le a\lambda \\ \frac{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]}{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2}, & \beta_q > a\lambda \end{cases}$$

If there is at least one local maximum point in $(-\infty, 0) \cup (0, \infty)$, the second-order partial derivative must be checked

$$\frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_q^2} = \begin{cases} \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}), & \beta_q < -a\lambda \\ \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}) + \frac{n}{a-1}, & -a\lambda \leq \beta_q < -\lambda \\ \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}), & -\lambda \leq \beta_q < 0 \\ \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}), & 0 < \beta_q \leq \lambda \\ \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}) + \frac{n}{a-1}, & \lambda < \beta_q \leq a\lambda \\ \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}), & \beta_q > a\lambda \end{cases}$$

If $\frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_q^2}|_{\hat{\beta}_q} < 0$, then $\hat{\beta}_q$ is a local maximum point.

Define $f_q(\beta_q) = f(\boldsymbol{\beta})$ given fixed $\beta_l$, $l \neq q$. Note that $f_q(\beta_q)$ is not differentiable at 0 but is left and right differentiable at 0. If the left derivative of $f_q(\beta_q)$ at 0: $\partial_- f_q(0) := \lim_{\beta_q \to 0^-} \frac{f_q(\beta_q) - f_q(0)}{\beta_q} > 0$, then there exists a $\delta_1 > 0$ such that for any $-\delta_1 < \beta_q < 0$, $\frac{f_q(\beta_q) - f_q(0)}{\beta_q} > 0$. Then $f_q(\beta_q) - f_q(0) < 0$ or $f_q(0) > f_q(\beta_q)$. Similarly, if the right derivative of $f_q(\beta_q)$ at 0: $\partial_+ f_q(0) := \lim_{\beta_q \to 0^+} \frac{f_q(\beta_j) - f_q(0)}{\beta_q} < 0$, then there exists a $\delta_2 > 0$ such that for any

$0 < \beta_q < \delta_2,\ f_q(0) > f_q(\beta_q)$. Then 0 is a local maximum point.

$$\begin{aligned}
\partial_- f_q(0) = &\lim_{\beta_q \to 0^-} \frac{1}{\beta_q}(\{\frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k - z_{iq}\beta_q]^2 - n\sum_{k=1}^{p} p(\beta_k)\} \\
&- \{\frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]^2 - n\sum_{k\neq q} p(\beta_k) - np(0)\}) \\
= &\lim_{\beta_q \to 0^-} \frac{1}{\beta_q}[(\frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})\{[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]^2 - 2z_{iq}\beta_q[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + (z_{iq}\beta_q)^2\} \\
&- np(\beta_q)) - \{\frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]^2\}] \\
= &\lim_{\beta_q \to 0^-} \frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})\{-2z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + z_{iq}^2\beta_q\} - \frac{np(\beta_q)}{\beta_q} \\
= &-\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lim_{\beta_q \to 0^-} \frac{p(\beta_q)}{\beta_q}.
\end{aligned}$$

Similarly, $\partial_+ f_q(0) = -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lim_{\beta_q \to 0^+} \frac{p(\beta_q)}{\beta_q}$.

For SCAD,

$$\begin{aligned}
\partial_- f_q(0) &= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lim_{\beta_q \to 0^-} \frac{-\lambda\beta_q}{\beta_q} \\
&= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + n\lambda
\end{aligned}$$

$$\begin{aligned}
\partial_+ f_q(0) &= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lim_{\beta_q \to 0^+} \frac{\lambda\beta_q}{\beta_q} \\
&= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lambda
\end{aligned}$$

So if $-\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + n\lambda > 0$ and $-\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lambda < 0$, or $|\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]| < n\lambda$, 0 is a local maximum point.

For MCP,

$$\frac{\partial f(\boldsymbol{\beta})}{\partial \beta_q} = \begin{cases} -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}], & \beta_q < -\gamma\lambda \\ -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}] + n\lambda(1 + \frac{\beta_q}{\gamma\lambda}), & -\gamma\lambda \le \beta_q < 0 \\ -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}] - n\lambda(1 - \frac{\beta_q}{\gamma\lambda}), & 0 < \beta_q \le \gamma\lambda \\ -\sum_{i=1}^n z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T \boldsymbol{\beta}], & \beta_q > \gamma\lambda \end{cases}$$

$$\hat{\beta}_q = \begin{cases} \frac{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]}{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2}, & \beta_q < -\gamma\lambda \\ \frac{\gamma\{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lambda\}}{\gamma \sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2 + n} & -\gamma\lambda \le \beta_q < 0 \\ \frac{\gamma\{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + n\lambda\}}{\gamma \sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2 + n}, & 0 < \beta_q \le \gamma\lambda \\ \frac{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]}{\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2}, & \beta_q > \gamma\lambda \end{cases}$$

If there is at least one local maximum point in $(-\infty, 0) \cup (0, \infty)$, check the second-order partial derivative

$$\frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_q^2} = \begin{cases} \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}), & \beta_q < -\gamma\lambda \\ \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}) + \frac{n}{\gamma}, & -\gamma\lambda \le \beta_q < 0 \\ \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}) + \frac{n}{\gamma}, & 0 < \beta_q \le \gamma\lambda \\ \sum_{i=1}^n z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}), & \beta_q > \gamma\lambda \end{cases}$$

If $\frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_q^2}|_{\hat{\beta}_q} < 0$, then $\hat{\beta}_q$ is a local maximum point.

$$\begin{aligned} \partial_- f_q(0) &= -\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n \lim_{\beta_q \to 0^-} \frac{-\lambda(\beta_q + \frac{\beta_q^2}{2\gamma\lambda})}{\beta_q} \\ &= -\sum_{i=1}^n h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + n\lambda \end{aligned}$$

$$\begin{aligned}\partial_+ f_q(0) &= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n \lim_{\beta_q \to 0^+} \frac{\lambda(\beta_q - \frac{\beta_q^2}{2\gamma\lambda})}{\beta_q} \\ &= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n\lambda\end{aligned}$$

So if $|\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]| < n\lambda$, 0 is a local maximum point.

For SICA,

$$\frac{\partial f(\boldsymbol{\beta})}{\partial \beta_q} = \begin{cases} -\sum_{i=1}^{n} z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T\boldsymbol{\beta}] + \frac{n\lambda a(a+1)}{(a-\beta_q)^2}, & \beta_q < 0 \\ -\sum_{i=1}^{n} z_{iq} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T\boldsymbol{\beta}] - \frac{n\lambda a(a+1)}{(a+\beta_q)^2}, & \beta_q > 0 \end{cases}$$

$\hat{\beta}_q$ is the solution to

$$\begin{cases} [\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2]\beta_q^3 - \{2a\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2 + \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]\}\beta_q^2 + \\ \{a^2\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2 + 2a\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]\}\beta_q - & \beta_q < 0 \\ a^2\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + n\lambda a(a+1) = 0, \\ [\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2]\beta_q^3 + \{2a\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2 - \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]\}\beta_q^2 + \\ \{a^2\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2 - 2a\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]\}\beta_q - & \beta_q > 0 \\ \{a^2\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + n\lambda a(a+1)\} = 0. \end{cases}$$

If there is at least one local maximum point in $(-\infty, 0) \cup (0, \infty)$, check the second-order partial derivative

$$\frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_q^2} = \begin{cases} \sum_{i=1}^{n} z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}) + \frac{2n\lambda a(a+1)}{(a-\beta_q)^3}, & \beta_q < 0 \\ \sum_{i=1}^{n} z_{iq}^2 h_i(\tilde{\boldsymbol{\eta}}) + \frac{2n\lambda a(a+1)}{(a+\beta_q)^3}, & \beta_q > 0 \end{cases}$$

If $\frac{\partial^2 f(\boldsymbol{\beta})}{\partial \beta_q^2}|_{\hat{\beta}_q} < 0$, then $\hat{\beta}_q$ is a local maximum point.

$$\begin{aligned}\partial_- f_q(0) &= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n \lim_{\beta_q \to 0^-} \frac{-\frac{\lambda(a+1)\beta_q}{a-\beta_q}}{\beta_q} \\ &= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + \frac{n\lambda(a+1)}{a}\end{aligned}$$

$$\begin{aligned}\partial_+ f_q(0) &= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - n \lim_{\beta_q \to 0^+} \frac{\frac{\lambda(a+1)\beta_q}{a+\beta_q}}{\beta_q} \\ &= -\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}} - \sum_{k\neq q} z_{ik}\beta_k)] - \frac{n\lambda(a+1)}{a}\end{aligned}$$

So if $-\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] + \frac{n\lambda(a+1)}{a} > 0$ and $-\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k] - \frac{n\lambda(a+1)}{a} < 0$ or $|\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]| < \frac{n\lambda(a+1)}{a}$, 0 is a local maximum point.

If there are more than one local maximum points, calculate $f_q(\hat{\beta}_q)$ for each of the local maximum points and the one that has the largest value of $f_q(\hat{\beta}_q)$ is the global maximum point.

If one wishes to leave some predictor $\beta_q$ unpenalized, we can simply maximize $\frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})$ $[y_i(\tilde{\boldsymbol{\eta}}) - \mathbf{z}_i^T\boldsymbol{\beta}]^2$ fixing $\beta_k,\ k \neq q$, which is maximized at $\frac{\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})[y_i(\tilde{\boldsymbol{\eta}}) - \sum_{k\neq q} z_{ik}\beta_k]}{\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}) z_{iq}^2}$.

In each iteration, $\beta_q' s$ are updated successively. To eliminate the possible effect of the order of updates, the $\beta_q' s$ are randomly ordered to be updated. Once a $\hat{\beta}_q$ remains the same as its value in the previous iteration, it's considered to have converged and will not be updated again. To prevent the algorithm from running without converging in a timely fashion, the algorithm may be terminated after a predetermined number of iterations. 1,000 is used in subsequent simulation studies and the analysis of the AML data in Chapter 3.

Thus, the coordinate ascent algorithm of estimating penalized proportional subdistribution hazards model is:

1. Initialize $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}_0$, calculate $\mathbf{h}(\tilde{\boldsymbol{\eta}})$ and $\mathbf{y}(\tilde{\boldsymbol{\eta}})$. Set the index set of $\beta_q's$ to be updated $\mathcal{I} = \{1, ..., p\}$. Set the maximum number of iterations $N_{iter}$.

2. Let $\mathcal{I}_p$ be a random permutation of $\mathcal{I}$. In the order of $\mathcal{I}_p$:

   (a) Update $\tilde{\beta}_q$ by maximizing $f_q(\beta_q)$.

   (b) Remove $q$ from $\mathcal{I}$ if $\tilde{\beta}_q$ remains the same as its value in the previous iteration.

   (c) Update $\tilde{\boldsymbol{\eta}} = \mathbf{Z}\tilde{\boldsymbol{\beta}}$.

   (d) Update $\mathbf{h}(\tilde{\boldsymbol{\eta}})$ and $\mathbf{y}(\tilde{\boldsymbol{\eta}})$.

3. Repeat step 2 until $\mathcal{I} = \emptyset$ or the algorithm has run $N_{iter}$ iterations.

Cross-validation is recommended to determine an optimal choice for $\lambda$. For each of a proposed set of values of $\lambda$, we performed cross-validation and the value having the best cross-validated score was used. Two types of cross-validation scores have been proposed in the literature. In the Cox model setting, Simon et al. (2011) proposed the cross-validation score $CV = \sum_{fold=1}^{n_{fold}} l(\hat{\boldsymbol{\beta}}_{-fold}) - l_{-fold}(\hat{\boldsymbol{\beta}}_{-fold})$, where $n_{fold}$ is the number of folds used in the cross-validation, $\hat{\boldsymbol{\beta}}_{-fold}$ is the $\boldsymbol{\beta}$ estimated on the training data, $l_{-fold}$ is the log-partial likelihood defined on the training data and $l(\boldsymbol{\beta})$ is the log-partial likelihood defined on the complete data [85]. The value of $\lambda$ that yields the largest $CV$ will be selected. Bradic, Fan and Jiang (2011) proposed a sparse approximation to the generalized cross-validation score $SGCV = \sum_{fold=1}^{n_{fold}} \left[ \frac{l(\hat{\boldsymbol{\beta}}_{-fold})}{n(1-s/n)^2} - \frac{l_{-fold}(\hat{\boldsymbol{\beta}}_{-fold})}{n_{-fold}(1-s/n_{-fold})^2} \right]$, where $s$ is the number of nonzero elements in $\hat{\boldsymbol{\beta}}$ and $n_{-fold}$ is the sample size of the training data [10]. The value of $\lambda$ that yields the largest $SGCV$ will be selected. Both of these two types of scores will be used in the subsequent study. This method will be studied in a simulation study and applied to the AML dataset in the Chapter 3.

# Chapter 3: Simulation Study and Application to the AML Dataset

## 3.1 Simulation Study

### 3.1.1 Setup

In the simulation study, the sample size $n$ was set to 200, 300 and 400. The number of predictors $p$ was set to 5000. The predictors were grouped into 100 blocks of size 50. Within each block, the predictors were generated following a $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ distribution, where $\boldsymbol{\mu} = \mathbf{0}$ and $\Sigma_{iq} = 0.5^{|i-q|},\ i, q = 1, ..., 50$. $F_1(t|\mathbf{z})$ was set to $1 - \{1 - c[1 - \exp(-t)]\}^{\exp(\mathbf{z}^T\boldsymbol{\beta})}$ for some $0 < c < 1$. Then

$$\begin{aligned}
\lambda_1(t|\mathbf{z}) &= -d\log(\{1 - c[1 - \exp(-t)]\}^{\exp(\mathbf{z}^T\boldsymbol{\beta})})/dt \\
&= -\exp(\mathbf{z}^T\boldsymbol{\beta})d\log(\{1 - c[1 - \exp(-t)]\})/dt \\
&= -\exp(\mathbf{z}^T\boldsymbol{\beta})\frac{-c\exp(-t)}{1 - c[1 - \exp(-t)]} \\
&= \frac{c\exp(-t)}{1 - c[1 - \exp(-t)]}\exp(\mathbf{z}^T\boldsymbol{\beta})
\end{aligned}$$

$\lambda_1(t|\mathbf{z})$ follows the assumption made by the proportional subdistribution hazards model. Assume there are two competing risks. Then $P(\epsilon = 1) = \lim_{t\to\infty} F_1(t|\mathbf{z}) = 1 - (1 - c)^{\exp(\mathbf{z}^T\boldsymbol{\beta})}$

and $P(\epsilon = 2) = 1 - P(\epsilon = 1) = (1-c)^{\exp(\mathbf{z}^T\boldsymbol{\beta})}$.

For each observation $i$, $\epsilon_i$ was randomly drawn from the Bernoulli distribution with $P(\epsilon_i = 1) = 1-(1-c)^{\exp(\mathbf{z}_i^T\boldsymbol{\beta})}$ and $P(\epsilon_i = 2) = (1-c)^{\exp(\mathbf{z}_i^T\boldsymbol{\beta})}$. Ten predictors were randomly selected to have nonzero $\beta_q$ while for the remaining 4990 predictors, $\beta_q = 0$. Three values of $|\beta_q|$ were used for the 10 important predictors: $0.75, 1$ and $1.25$ with the sign randomly chosen as positive or negative. $c$ was set to 0.9999999 to mimic the motivating data, in which about 86% of the patients who were not censored relapsed. If $\epsilon_i = 1$, $T_i$ was drawn from the distribution $P(T_i \leq t) = F_1(t|\mathbf{z})/P(\epsilon_i = 1)$. If $\epsilon_i = 2$, $T_i$ was drawn from the $\exp(\lambda = 1)$ distribution. $C_i$ was drawn from an $\exp(0.4)$ distribution. The parameter was set to 0.4 to mimic the motivating data, in which about 30% of the patients were censored.

For SCAD, the parameter $a$ was set to 3.7 following the previous recommendation [29]. For MCP, $\gamma$ was set to 3 following a previous recommendation [11]. For SICA, $a$ was set to 0.1 since it worked well in a published simulation study [70]. In each simulation, 100 values of $\lambda$ were used. The values of $\lambda$ were chosen to form a geometric progression, so $\lambda_i = \max\lambda * ratio^{(i-1)/99}$ for $i = 1, ..., 100$. The minimum $\lambda = ratio * \max\lambda$.

To determine the candidate values of $\lambda$, a useful starting value is that which makes the algorithm select none of the predictors given $\boldsymbol{\beta}_0 = \mathbf{0}$. That means 0 is the global maximum point of $f_q(\beta_q)$ for any $q = 1, ..., p$ given $\beta_k = 0,\ k \neq q$. Then 0 must also be a local maximum point. Setting the initial parameter vector $\boldsymbol{\beta}_0 = \mathbf{0}$ such that $\boldsymbol{\eta}_0 = \mathbf{Z}\boldsymbol{\beta}_0 = \mathbf{0}$. For SCAD and MCP, that means $|\sum_{i=1}^n h_i(\mathbf{0})z_{iq}y_i(\mathbf{0})| < n\lambda$ so $\frac{|\sum_{i=1}^n h_i(\mathbf{0})z_{iq}y_i(\mathbf{0})|}{n}$ was used as the largest value. For SICA, that means $|\sum_{i=1}^n h_i(\mathbf{0})z_{iq}y_i(\mathbf{0})| < \frac{n\lambda(a+1)}{a}$, so $\frac{a|\sum_{i=1}^n h_i(\mathbf{0})z_{iq}y_i(\mathbf{0})|}{n(a+1)}$ was used as the largest value. In the simulation study, this method worked well for SCAD and MCP but not for SICA. When $\lambda = \frac{a|\sum_{i=1}^n h_i(\mathbf{0})z_{iq}y_i(\mathbf{0})|}{n(a+1)}$, $\mathbf{0}$ usually is not the global maximum point of $l(\boldsymbol{\beta})$ and the algorithm would select several predictors. So for SICA, trial and error was used to find the largest value.

For SCAD and MCP, the *ratio* was set to be 0.3 so that the algorithm selects more than

10 predictors but fewer than 100 when using the minimum value of $\lambda$. From previous genomic applications this range is a reasonable number of predictors and lends well to development of smaller assays. For SICA, different values of maximum $\lambda$ are used for each of the 9 settings and *ratio* was set to 0.1.

Since the solutions for the same dataset with slightly different values of $\lambda$ are expected to be similar, in cross-validation, the algorithm is run for each training set using $\lambda$ from the largest to the smallest. The $\hat{\boldsymbol{\beta}}$ obtained from using the last value of $\lambda$ will be set as the initial value for the next value of $\lambda$, known as the warm start approach. $\mathbf{0}$ was used as the initial value when using the largest value of $\lambda$.

Notice that

$$\frac{\partial l_\eta}{\partial \eta_i}(\tilde{\boldsymbol{\eta}}) = I(\Delta_i \epsilon_i = 1) - \sum_{\Delta_k \epsilon_k = 1} \frac{w_i(T_k) Y_i(T_k) \exp(\tilde{\eta}_i)}{\sum_{q=1}^n w_q(T_k) Y_q(T_k) \exp(\tilde{\eta}_q)}$$

and

$$\begin{aligned} h_i(\tilde{\boldsymbol{\eta}}) &= \frac{\partial^2 l_\eta}{\partial \eta_i^2}(\tilde{\boldsymbol{\eta}}) \\ &= -\sum_{\Delta_k \epsilon_k = 1} \frac{w_i(T_k) Y_i(T_k) \exp(\tilde{\eta}_i) \sum_{q=1}^n w_q(T_k) Y_q(T_k) \exp(\tilde{\eta}_q) - [w_i(T_k) Y_i(T_k) \exp(\tilde{\eta}_i)]^2}{[\sum_{q=1}^n w_q(T_k) Y_q(T_k) \exp(\tilde{\eta}_q)]^2} \\ &= \sum_{\Delta_k \epsilon_k = 1} [\frac{w_i(T_k) Y_i(T_k) \exp(\tilde{\eta}_i)}{\sum_{q=1}^n w_q(T_k) Y_q(T_k) \exp(\tilde{\eta}_q)}]^2 - \frac{w_i(T_k) Y_i(T_k) \exp(\tilde{\eta}_i)}{\sum_{q=1}^n w_q(T_k) Y_q(T_k) \exp(\tilde{\eta}_q)}. \end{aligned}$$

In the simulation study, some $\exp(\tilde{\eta}_i)$ can get so large that R treats them as infinity. In this case, the way that $\nabla l_\eta(\tilde{\boldsymbol{\eta}})$ and $\mathbf{h}(\tilde{\boldsymbol{\eta}})$ are calculated in the code needs to be modified. For each $k$ s.t. $\Delta_k \epsilon_k = 1$, let $\tilde{\eta}_m = \max\{\tilde{\eta}_q,\ q = 1, ..., n | w_q(T_k) Y_q(T_k) > 0\}$. If $\exp(\tilde{\eta}_m)$ is treated as finite in R, then $w_q(T_k) Y_q(T_k) \exp(\tilde{\eta}_q)$ can be directly calculated for $\tilde{\eta}_q \leq \tilde{\eta}_m$. Otherwise, subtract a factor from this $\tilde{\eta}_m$ and all smaller elements of $\tilde{\boldsymbol{\eta}}$. Now calculate $w_q(T_k) Y_q(T_k) \exp(\tilde{\eta}_q)$ for these $\tilde{\eta}_q$. For $\tilde{\eta}_q > \tilde{\eta}_m$, $w_q(T_k) Y_q(T_k) = 0$ so $w_q(T_k) Y_q(T_k) \exp(\tilde{\eta}_q) =$

0. Then we can calculate $\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)$ and $\frac{w_i(T_k)Y_i(T_k)\exp(\tilde{\eta}_i)}{\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)}$. Since the same factor is subtracted from each $\tilde{\eta}_q$, it's equivalent to dividing each of $\exp(\tilde{\eta}_q)$ by the same positive number. As this happens to both the numerator and the denominator, so they cancel out and the value of $\frac{w_i(T_k)Y_i(T_k)\exp(\tilde{\eta}_i)}{\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)}$ remains the same. Since R can compute $\exp(709)$ but treats $\exp(710)$ as infinity, the factor is set to be $\tilde{\eta}_m - 709$ so after subtracting the factor $\tilde{\eta}_m$ becomes 709. Occasionally, although $w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q),\ q = 1, ..., n$ are now all finite, $\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)$ was treated as $\infty$ by R for some $k$. In this case, define $ratio = \frac{\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)}{10^{308}}$, since $10^{308}$ is finite in R but $10^{309}$ is treated as $\infty$. Then divide each of $w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q),\ q = 1, ..., n$ by this ratio so that their sum after dividing this ratio is $10^{308}$. Now $\frac{w_i(T_k)Y_i(T_k)\exp(\tilde{\eta}_i)}{\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)}$ can be calculated.

To calculate $\mathbf{y}(\tilde{\eta})$, $\boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})^{-1}$ needs to be calculated. In the simulation study, $\boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})$ often has a determinant very close to 0 so R cannot invert it. So $\boldsymbol{H}_{l_\eta}(\tilde{\boldsymbol{\eta}})$ is replaced by its corresponding diagonal matrix. Then $y_i(\tilde{\boldsymbol{\eta}}) = \tilde{\eta}_i - \frac{1}{h_i(\tilde{\boldsymbol{\eta}})}\frac{\partial l_\eta}{\partial \eta_i}(\tilde{\boldsymbol{\eta}})$. Realizing that $y_i(\tilde{\boldsymbol{\eta}})$ only appears in $h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})$ when calculating $\hat{\beta}_q$, only $h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})$ is calculated instead of $y_i(\tilde{\boldsymbol{\eta}})$. $h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}}) = h_i(\tilde{\boldsymbol{\eta}})\tilde{\eta}_i - \frac{\partial l_\eta}{\partial \eta_i}(\tilde{\boldsymbol{\eta}})$.

Ten-fold cross-validation was performed to choose the optimal value of $\lambda$. After finding the optimal value of $\lambda$, the algorithm was applied to the the entire dataset using the optimal value with $\mathbf{0}$ as the initial value. 100 datasets were generated for each setting of sample size and effect size.

For comparison, other competing methods were also run on 100 datasets generated under each of the 9 settings. The method proposed in Binder et al. (2009) [8] was implemented in the R package CoxBoost, available on GitHub. The penalty value was determined using the "optimCoxBoostPenalty" function with *minstepno* set to 0 and *maxstepno* set to 200. The method in Ishwaran et al. (2014) [50] is implemented in the R package randomForestSRC [47]. The "var.select" function was used to select variables. The "method" argument was set to "md" and the "conservative" argument was set to "medium". Those variables

that had positive "vimp.event.1" were taken to be the final set of selected predictors. The forward–backward scan algorithm in [53] is implemented in the R package fastcmprsk, available on GitHub. SCAD and MCP penalties are used. Values of $a$ in SCAD and $\gamma$ in MCP are the same and the $\lambda$ values were generated using the same method as in the proposed algorithm. The cyclic coordinate-wise BAR algorithm is implemented in the R package pshBAR, available on GitHub. However, the paper did not discuss how the tuning parameters $\xi_n$ and $\lambda_n$ should be chosen so it was not used in the simulation studies. The method in Fu, Parikh and Zhou (2017) [35] was implemented in the R package crrp but it has been removed from The Comprehensive R Archive Network. The method in Sun and Wang (2022) [89] is implemented in the R package RAEN, available on GitHub. However, since this method is not very well-established and the package's reference manual is not clearly written, it was not used.

The Unity cluster maintained by Arts and Sciences Technology Services and Ohio Supercomputer Center were used to perform all computations.

### 3.1.2 Results

The results are summarized in Table 3.1. In the table, $n$ stands for sample size, $|\beta|$ stands for the absolute value of the coefficients of the 10 predictors with nonzero coefficients, PPSH (penalized proportional subdistribution hazards model) indicates the method proposed in this chapter, "TP" indicates the number of true positives, "FP" indicates the number of false positives, "CV" indicates that the CV score was used as the criterion and "SGCV" indicates that the SGCV score was used as the criterion. All the numbers were averaged over 100 replications. The fastcmprsk package produced errors in some of the replications. "Errors" denotes the number of replications that gave error out of the 100 replications for fastcmprsk. The "TP" and "FP" were averaged over the rest of the replications. Note that

the CV and SGCV scores were not used for the randomForestSRC package but its results were also included in the same table for comparison.

Table 3.1: Mean of TP and FP over 100 replications (for fastcmprsk, runs with errors were excluded)

| $n$ | $\|\beta\|$ | Method | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) | Errors |
|---|---|---|---|---|---|---|---|
| 200 | 0.75 | PPSH SCAD | 8.18 | 4.15 | 8.53 | 5.52 | 0 |
| | | PPSH MCP | 6.17 | 1.09 | 4.34 | 0.54 | 0 |
| | | PPSH SICA | 4.75 | 0.52 | 2.31 | 0.13 | 0 |
| | | CoxBoost | 9.67 | 34.55 | 9.95 | 66.72 | 0 |
| | | fastcmprsk SCAD | 9.76 | 32.94 | 9.96 | 41.55 | 1 |
| | | fastcmprsk MCP | 9.62 | 7.79 | 9.11 | 4.58 | 47 |
| | | randomForestSRC | 3.02 | 704.6 | 3.02 | 704.6 | 0 |
| | 1 | PPSH SCAD | 8.59 | 1.59 | 9.05 | 2.94 | 0 |
| | | PPSH MCP | 6.52 | 0.47 | 5.69 | 0.44 | 0 |
| | | PPSH SICA | 5.44 | 0.46 | 4.33 | 0.25 | 0 |
| | | CoxBoost | 9.97 | 38.34 | 9.99 | 70.47 | 0 |
| | | fastcmprsk SCAD | 9.99 | 29.36 | 10 | 39.78 | 1 |
| | | fastcmprsk MCP | 9.96 | 6.75 | 9.16 | 3.29 | 31 |
| | | randomForestSRC | 3.26 | 734.72 | 3.26 | 734.72 | 0 |
| | 1.25 | PPSH SCAD | 8.62 | 0.6 | 9.15 | 1.79 | 0 |
| | | PPSH MCP | 6.28 | 0.3 | 5.89 | 0.14 | 0 |
| | | PPSH SICA | 5.66 | 0.33 | 4.51 | 0.17 | 0 |
| | | CoxBoost | 9.98 | 41.02 | 10 | 70.44 | 0 |
| | | fastcmprsk SCAD | 10 | 23.15 | 10 | 34.99 | 0 |

Table 3.1: Continued

| $n$ | $\|\beta\|$ | Method | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) | Errors |
|---|---|---|---|---|---|---|---|
| | | fastcmprsk MCP | 10 | 4.9 | 9.06 | 2.14 | 28 |
| | | randomForestSRC | 3.36 | 689.9 | 3.36 | 689.9 | 0 |
| 300 | 0.75 | PPSH SCAD | 9.17 | 0.66 | 9.87 | 6.04 | 0 |
| | | PPSH MCP | 7.57 | 0.1 | 8.43 | 0.26 | 0 |
| | | PPSH SICA | 6.82 | 0.14 | 5.01 | 0.11 | 0 |
| | | CoxBoost | 10 | 43 | 10 | 81.72 | 0 |
| | | fastcmprsk SCAD | 10 | 25.12 | 10 | 67.14 | 0 |
| | | fastcmprsk MCP | 10 | 5.29 | 10 | 13.32 | 18 |
| | | randomForestSRC | 4.2 | 927.98 | 4.2 | 927.98 | 0 |
| | 1 | PPSH SCAD | 9.20 | 0.15 | 9.95 | 3.05 | 0 |
| | | PPSH MCP | 7.02 | 0.05 | 8.18 | 0.09 | 0 |
| | | PPSH SICA | 6.37 | 0.13 | 4.77 | 0.04 | 0 |
| | | CoxBoost | 10 | 44.62 | 10 | 80.52 | 0 |
| | | fastcmprsk SCAD | 10 | 14.32 | 10 | 59.49 | 0 |
| | | fastcmprsk MCP | 10 | 2.18 | 10 | 10.33 | 8 |
| | | randomForestSRC | 4.46 | 897.8 | 4.46 | 897.8 | 0 |
| | 1.25 | PPSH SCAD | 9.10 | 0.07 | 9.94 | 1.38 | 0 |
| | | PPSH MCP | 7.25 | 0.02 | 7.48 | 0.04 | 0 |
| | | PPSH SICA | 5.55 | 0.03 | 4.39 | 0.05 | 0 |
| | | CoxBoost | 10 | 46.47 | 10 | 72.07 | 0 |
| | | fastcmprsk SCAD | 10 | 9.33 | 10 | 55.07 | 0 |
| | | fastcmprsk MCP | 10 | 1.44 | 9.91 | 7.23 | 6 |
| | | randomForestSRC | 4.63 | 930.63 | 4.63 | 930.63 | 0 |

Table 3.1: Continued

| $n$ | $\|\beta\|$ | Method | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) | Errors |
|---|---|---|---|---|---|---|---|
| 400 | 0.75 | PPSH SCAD | 9.58 | 0.1 | 9.97 | 8.52 | 0 |
| | | PPSH MCP | 7.96 | 0.04 | 9.79 | 0.19 | 0 |
| | | PPSH SICA | 8.99 | 0.13 | 5.31 | 0.04 | 0 |
| | | CoxBoost | 10 | 43.36 | 10 | 88.67 | 0 |
| | | fastcmprsk SCAD | 10 | 16.53 | 10 | 72.19 | 0 |
| | | fastcmprsk MCP | 10 | 3.28 | 10 | 21.03 | 1 |
| | | randomForestSRC | 5.57 | 976.76 | 5.57 | 976.76 | 0 |
| | 1 | PPSH SCAD | 9.58 | 0.09 | 9.99 | 4.08 | 0 |
| | | PPSH MCP | 7.83 | 0.03 | 9.31 | 0.06 | 0 |
| | | PPSH SICA | 8 | 0.09 | 4.76 | 0.01 | 0 |
| | | CoxBoost | 10 | 49.33 | 10 | 87.2 | 0 |
| | | fastcmprsk SCAD | 10 | 6.58 | 10 | 61.66 | 0 |
| | | fastcmprsk MCP | 10 | 1.18 | 10 | 17.39 | 0 |
| | | randomForestSRC | 5.79 | 1013.02 | 5.79 | 1013.02 | 0 |
| | 1.25 | PPSH SCAD | 9.35 | 0.07 | 10 | 1.61 | 0 |
| | | PPSH MCP | 7.72 | 0.01 | 8.58 | 0.01 | 0 |
| | | PPSH SICA | 6.75 | 0.02 | 4.52 | 0.01 | 0 |
| | | CoxBoost | 10 | 47.16 | 10 | 87.51 | 0 |
| | | fastcmprsk SCAD | 10 | 3.89 | 10 | 57.79 | 0 |
| | | fastcmprsk MCP | 10 | 0.82 | 10 | 14.01 | 0 |
| | | randomForestSRC | 6.15 | 986.24 | 6.15 | 986.24 | 0 |

Boxplots are also produced of TP and FP. randomForestSRC is not included because its large number of false positives would make the results of the other methods undistinguishable.

Figure 3.1: Boxplot of TP when $n = 200$ and $|\beta| = 0.75$ using CV score

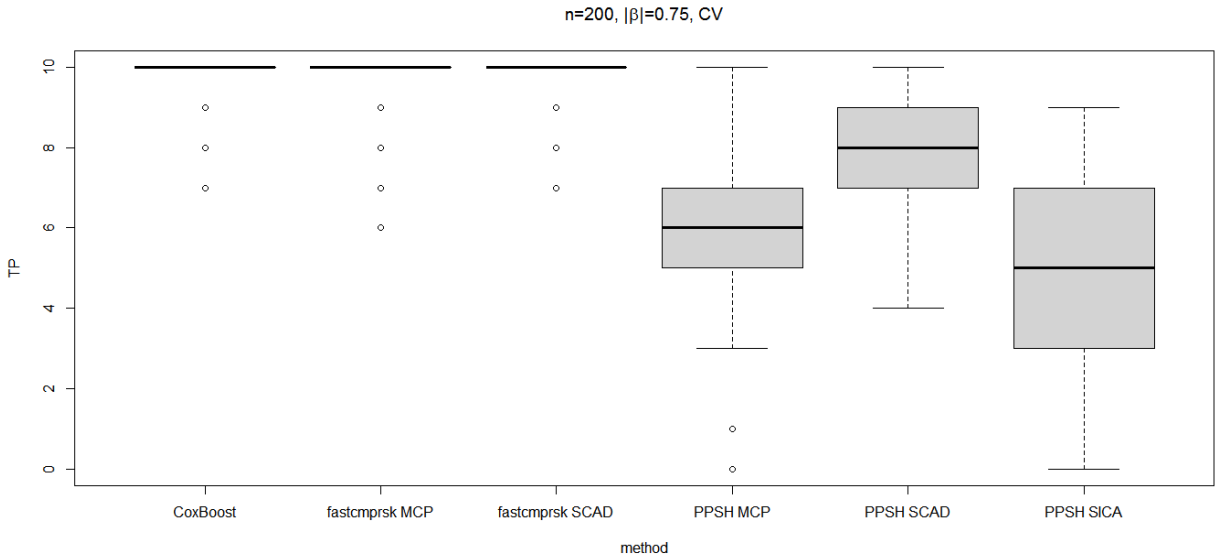



Figure 3.2: Boxplot of FP when $n = 200$ and $|\beta| = 0.75$ using CV score

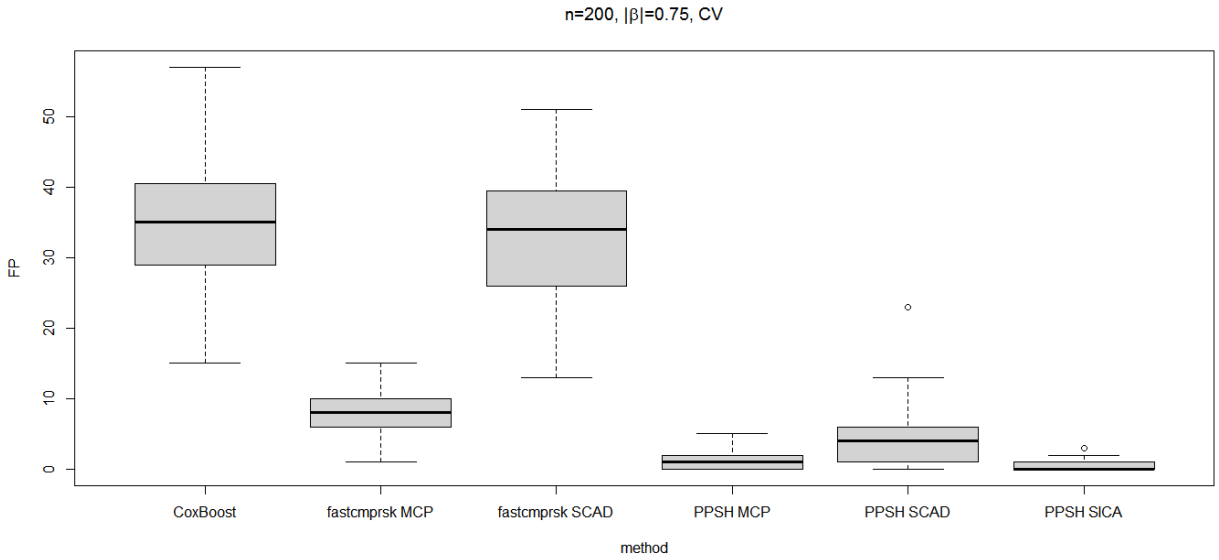


Figure 3.3: Boxplot of TP when $n = 200$ and $|\beta| = 0.75$ using SGCV score

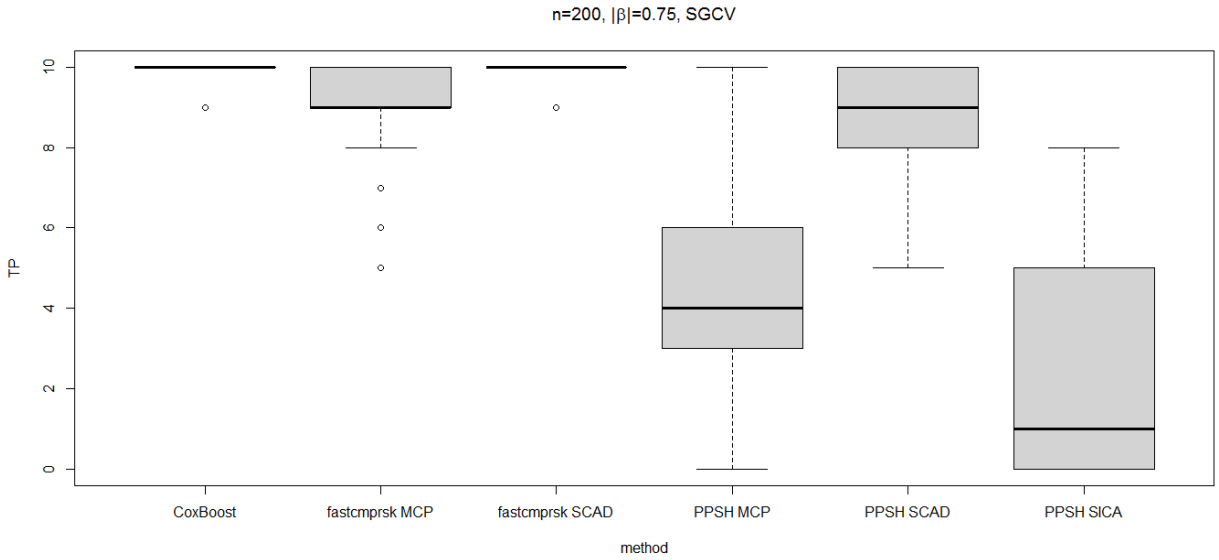



Figure 3.4: Boxplot of FP when $n = 200$ and $|\beta| = 0.75$ using SGCV score

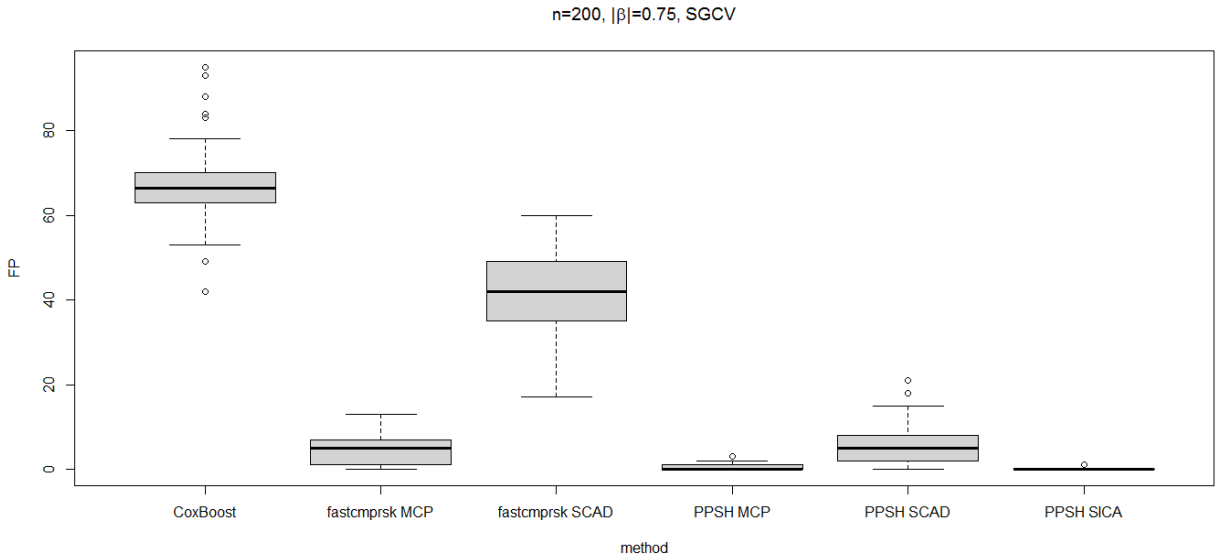


Figure 3.5: Boxplot of TP when $n = 200$ and $|\beta| = 1$ using CV score

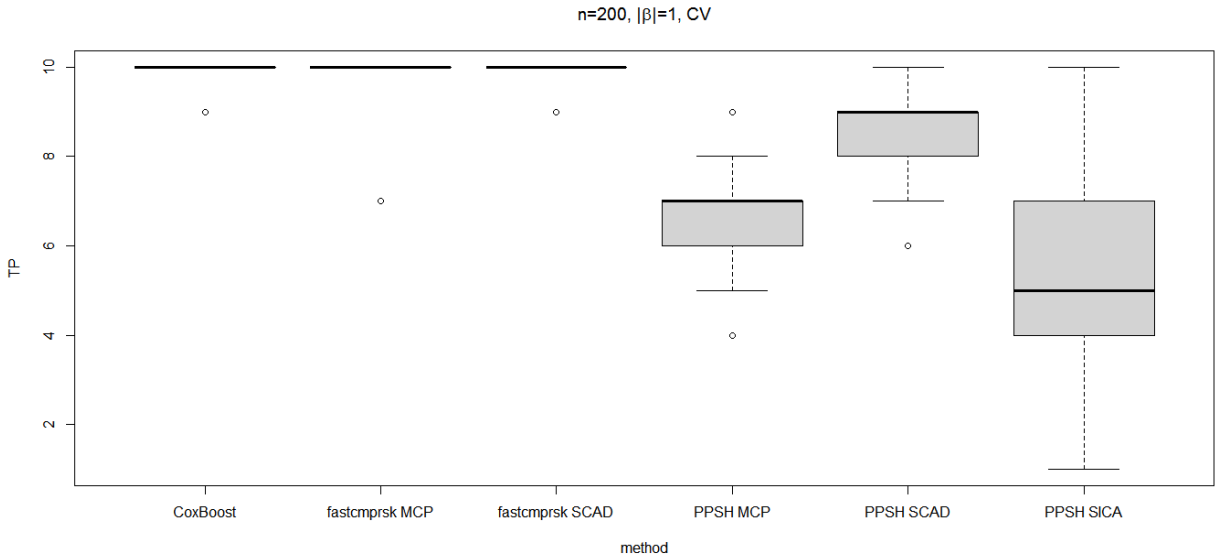



Figure 3.6: Boxplot of FP when $n = 200$ and $|\beta| = 1$ using CV score

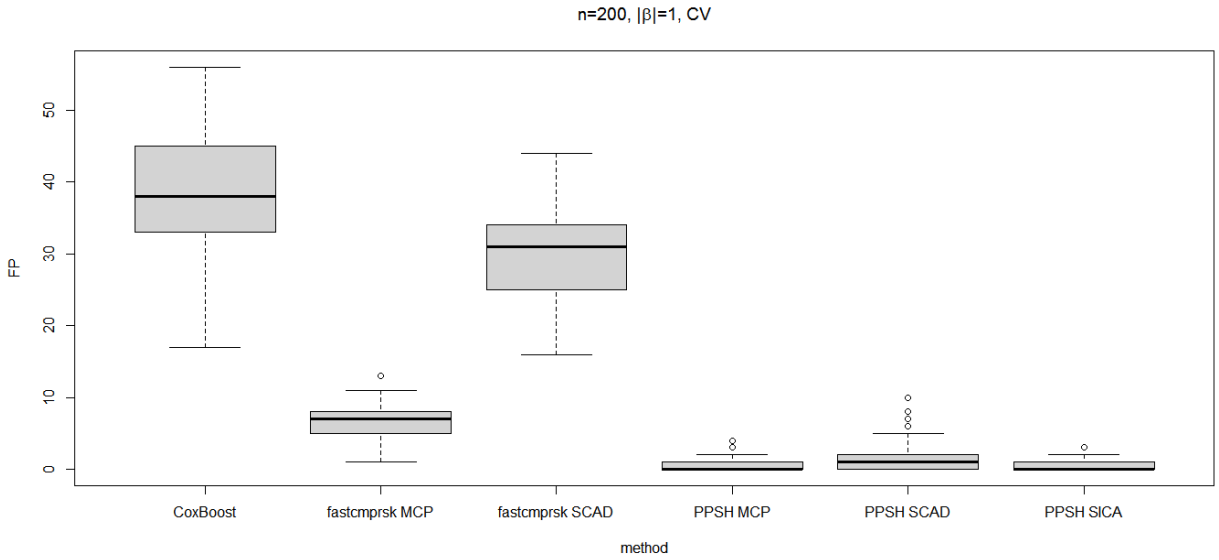


Figure 3.7: Boxplot of TP when $n = 200$ and $|\beta| = 1$ using SGCV score

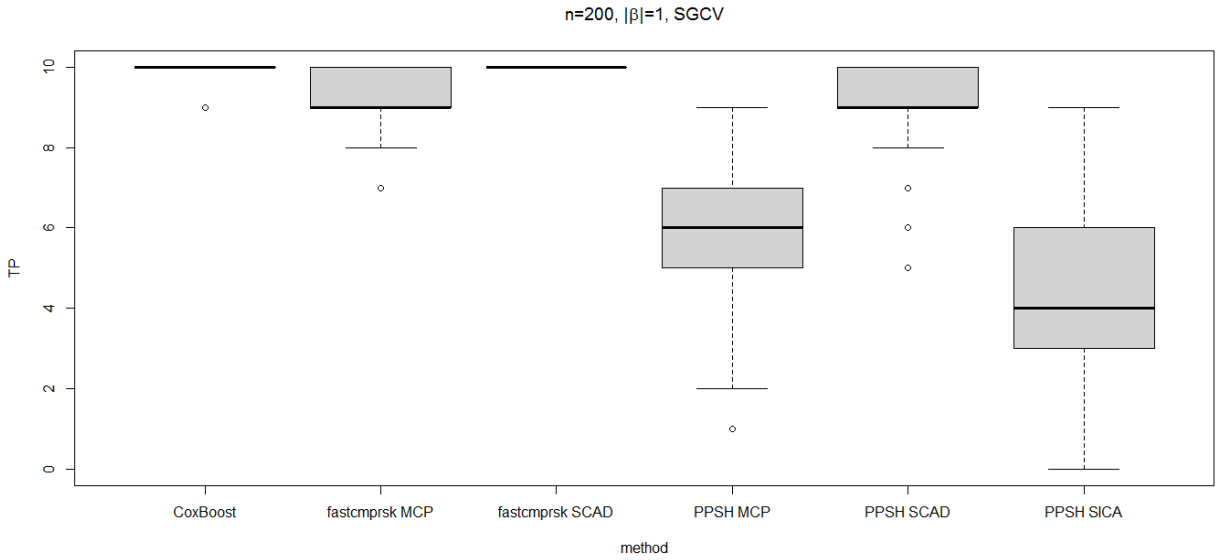



Figure 3.8: Boxplot of FP when $n = 200$ and $|\beta| = 1$ using SGCV score

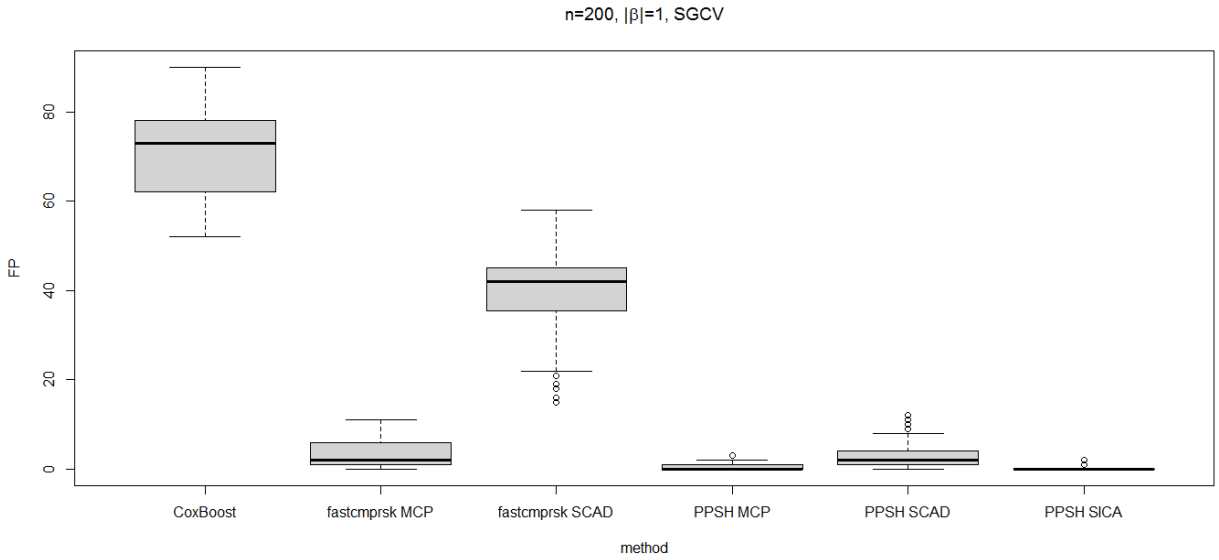


Figure 3.9: Boxplot of TP when $n = 200$ and $|\beta| = 1.25$ using CV score

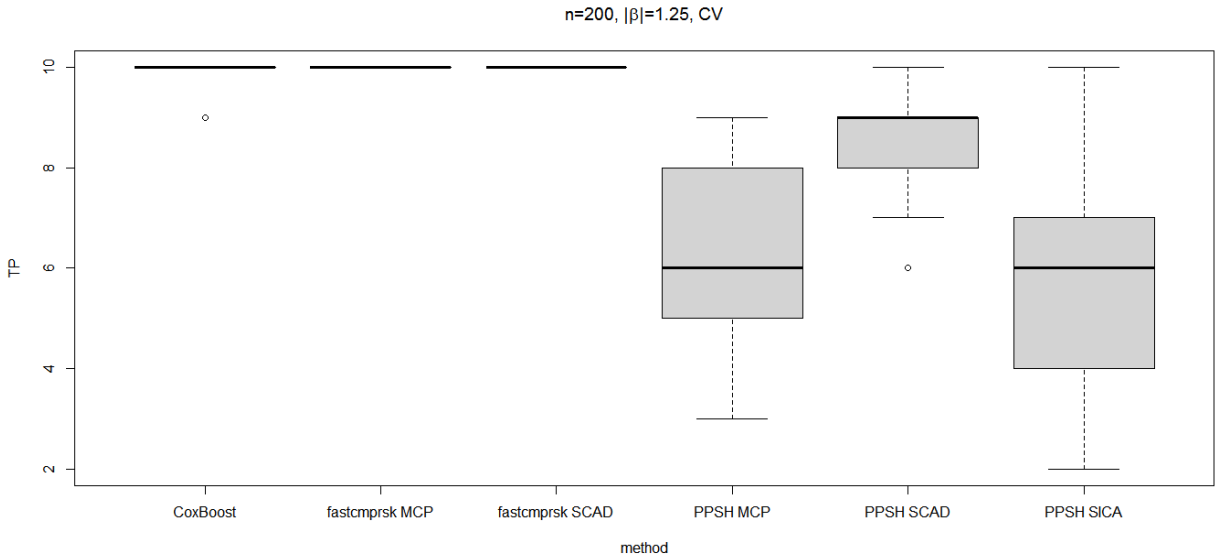



Figure 3.10: Boxplot of FP when $n = 200$ and $|\beta| = 1.25$ using CV score

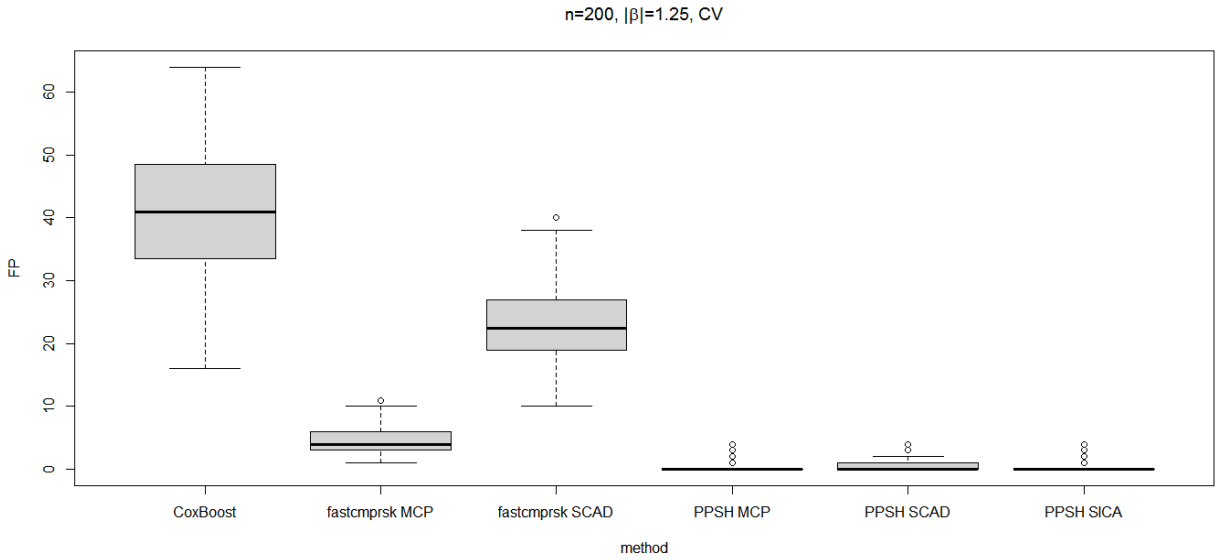


Figure 3.11: Boxplot of TP when $n = 200$ and $|\beta| = 1.25$ using SGCV score

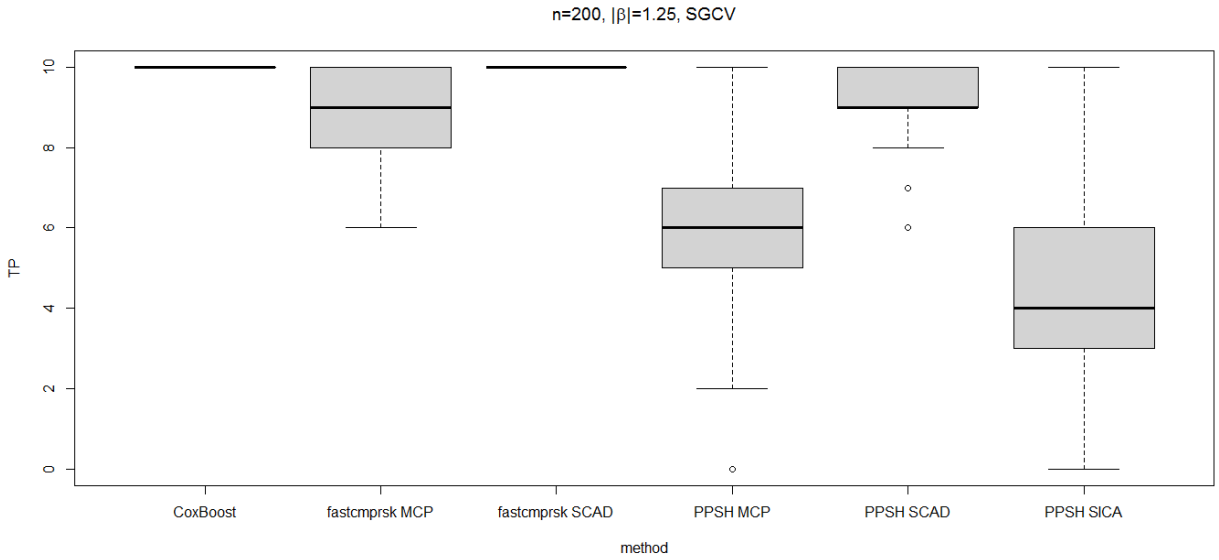



Figure 3.12: Boxplot of FP when $n = 200$ and $|\beta| = 1.25$ using SGCV score

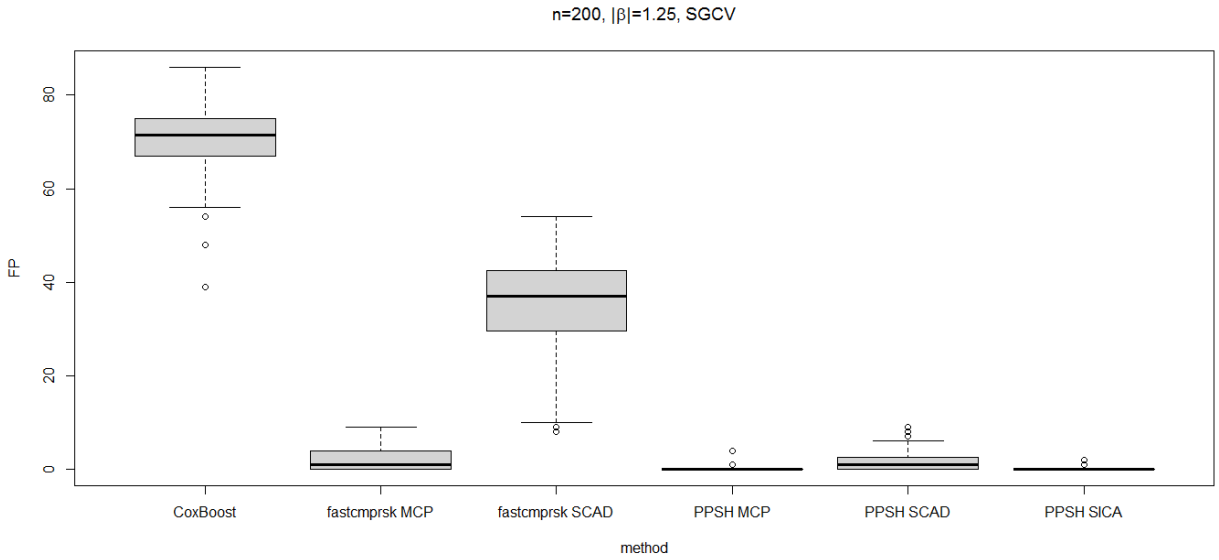


Figure 3.13: Boxplot of TP when $n = 300$ and $|\beta| = 0.75$ using CV score

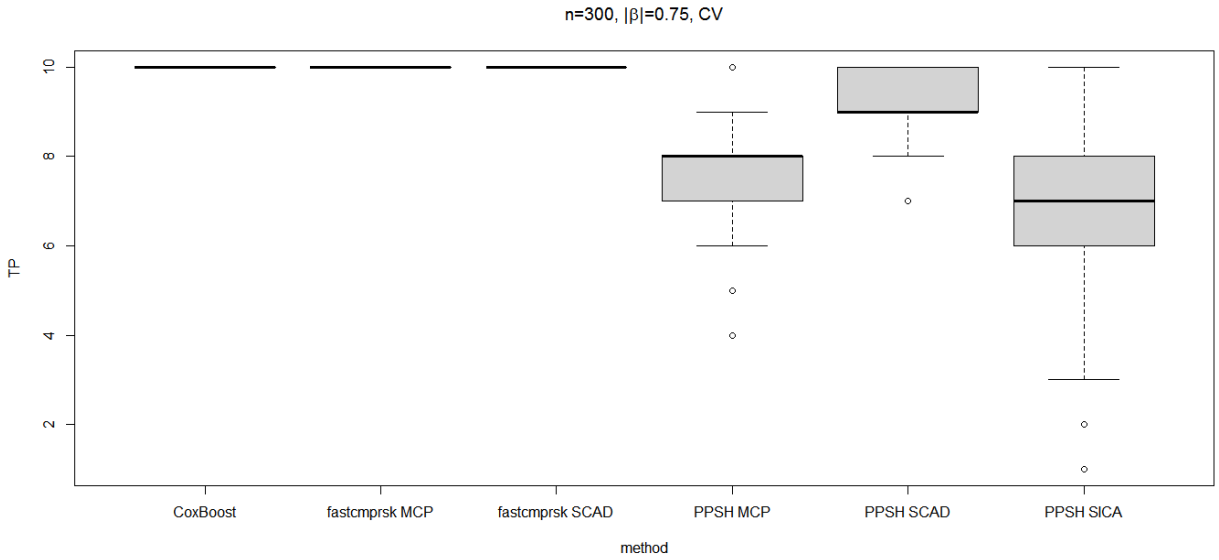



Figure 3.14: Boxplot of FP when $n = 300$ and $|\beta| = 0.75$ using CV score

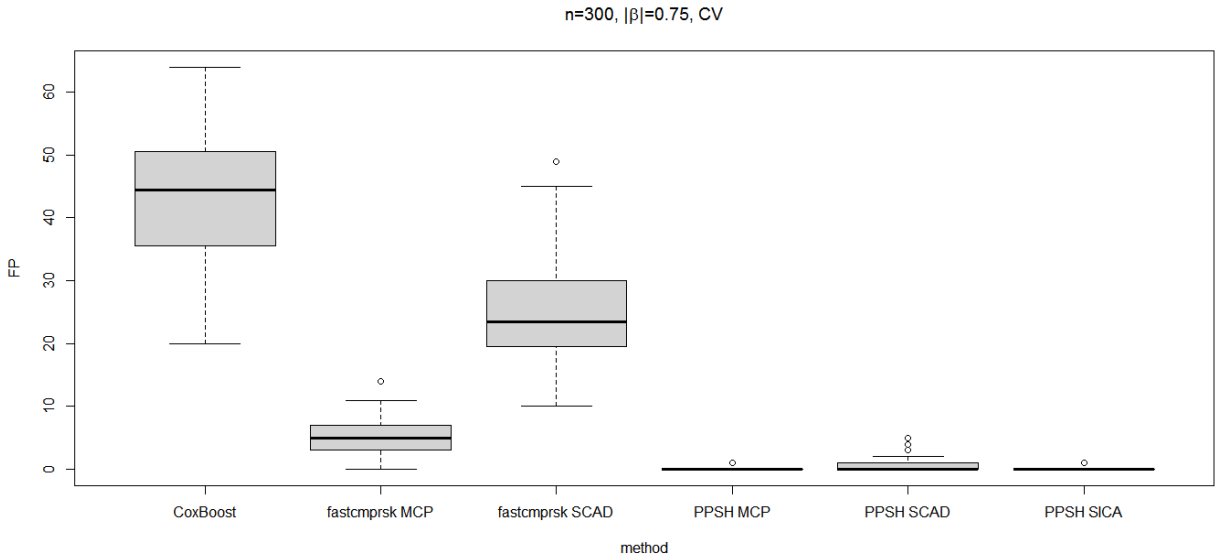


Figure 3.15: Boxplot of TP when $n = 300$ and $|\beta| = 0.75$ using SGCV score

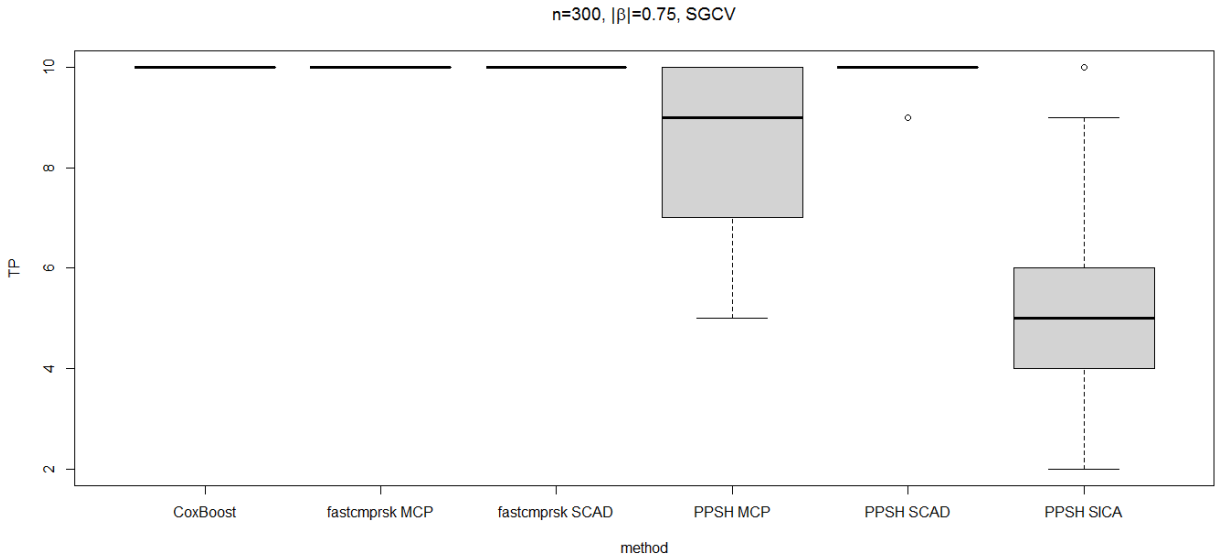



Figure 3.16: Boxplot of FP when $n = 300$ and $|\beta| = 0.75$ using SGCV score

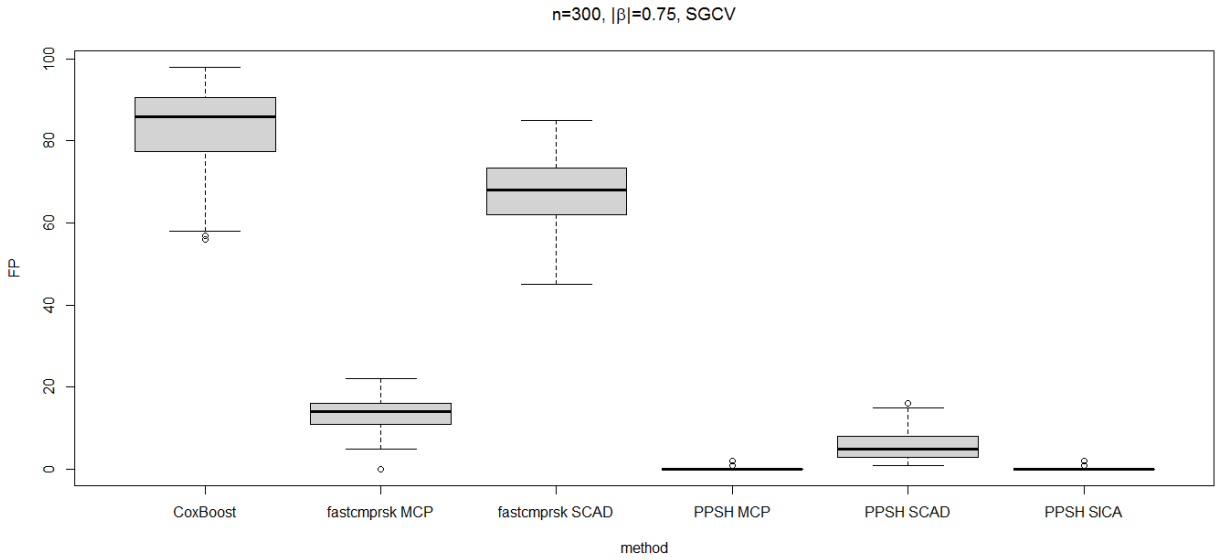


Figure 3.17: Boxplot of TP when $n = 300$ and $|\beta| = 1$ using CV score

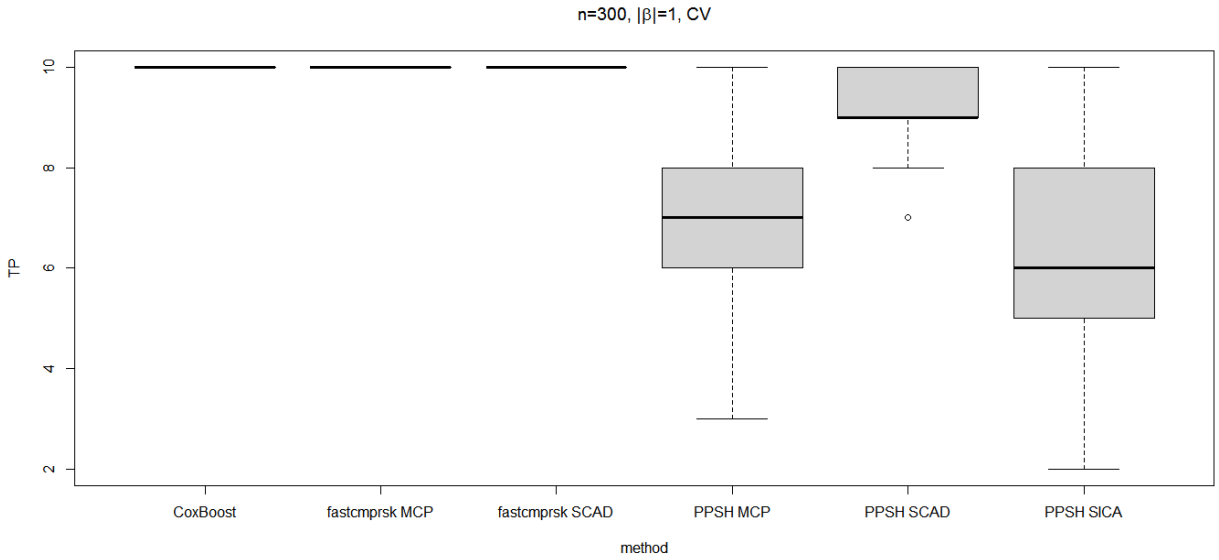



Figure 3.18: Boxplot of FP when $n = 300$ and $|\beta| = 1$ using CV score

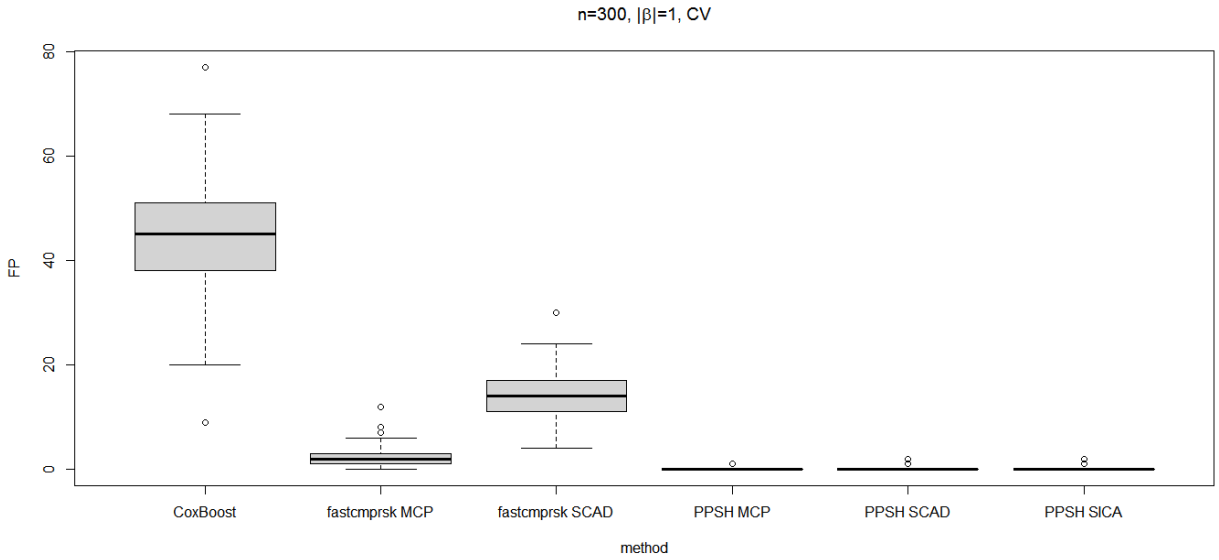


Figure 3.19: Boxplot of TP when $n = 300$ and $|\beta| = 1$ using SGCV score

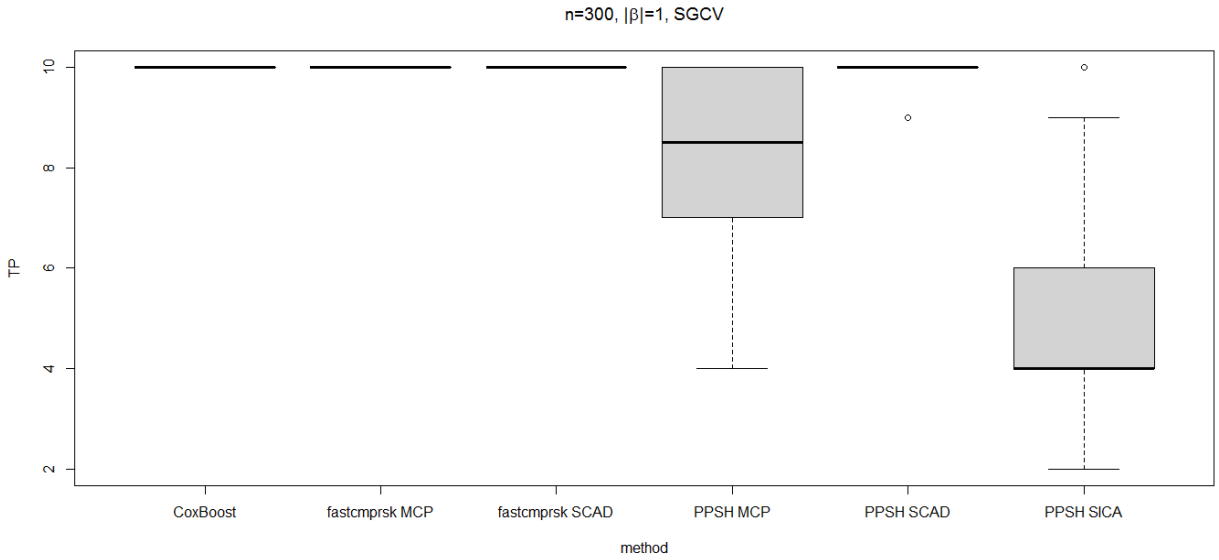



Figure 3.20: Boxplot of FP when $n = 300$ and $|\beta| = 1$ using SGCV score

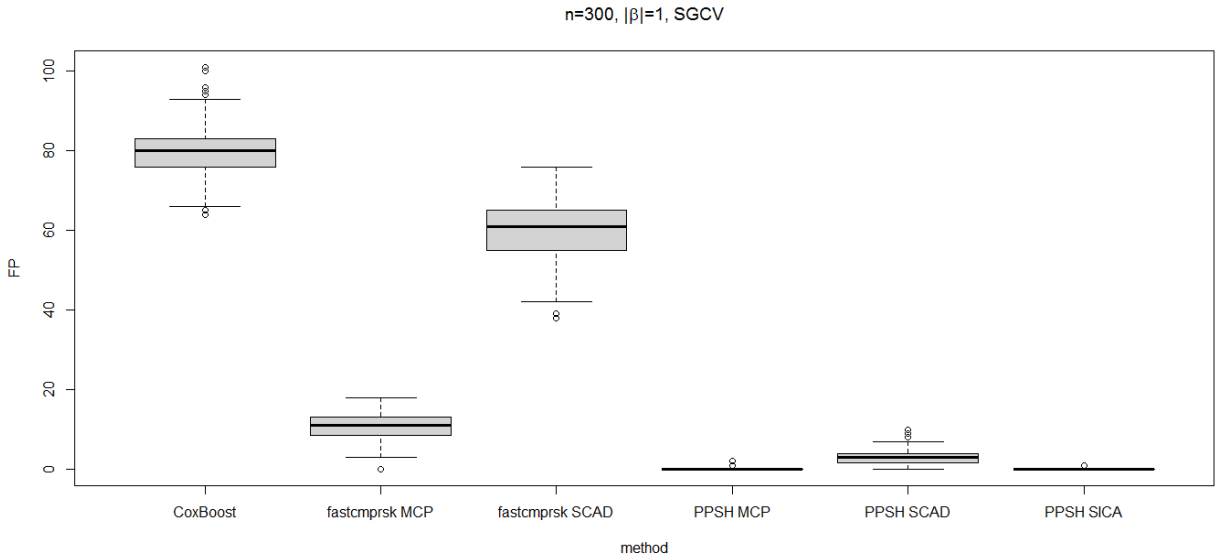


Figure 3.21: Boxplot of TP when $n = 300$ and $|\beta| = 1.25$ using CV score

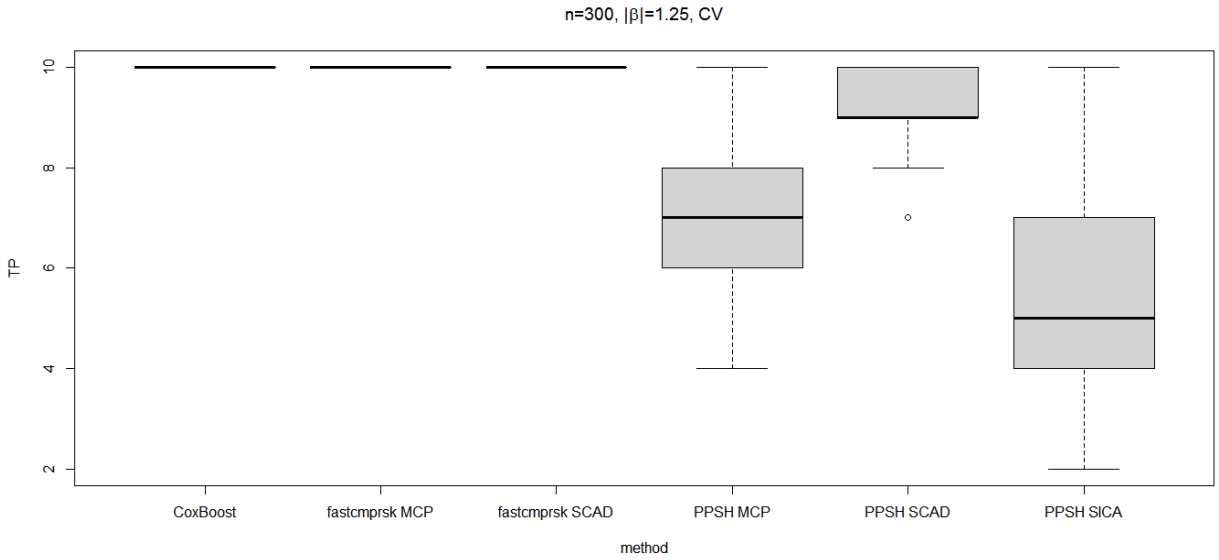



Figure 3.22: Boxplot of FP when $n = 300$ and $|\beta| = 1.25$ using CV score

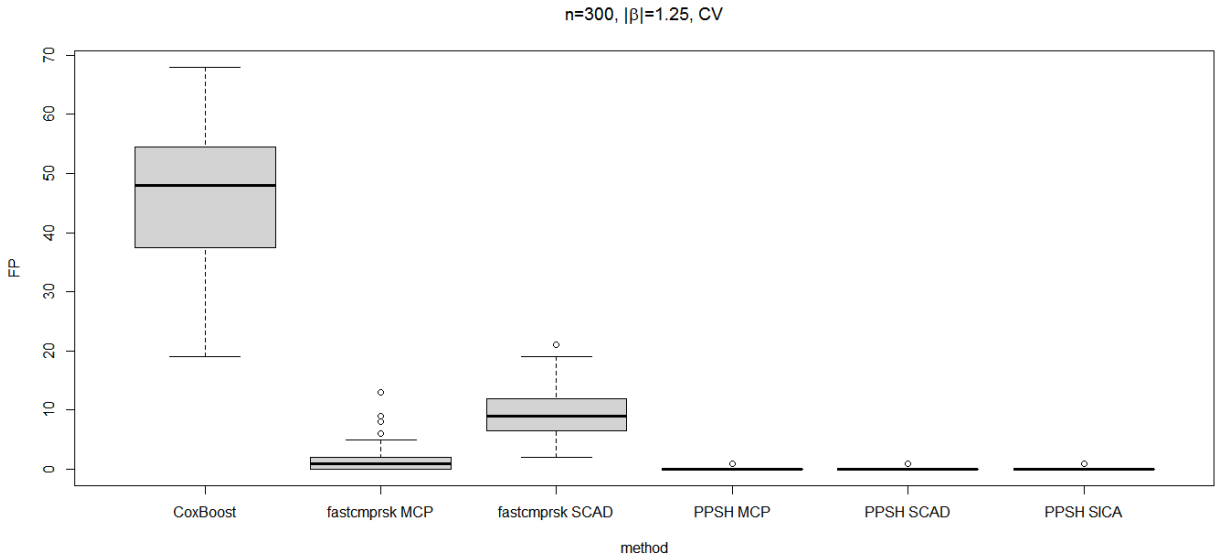


Figure 3.23: Boxplot of TP when $n = 300$ and $|\beta| = 1.25$ using SGCV score

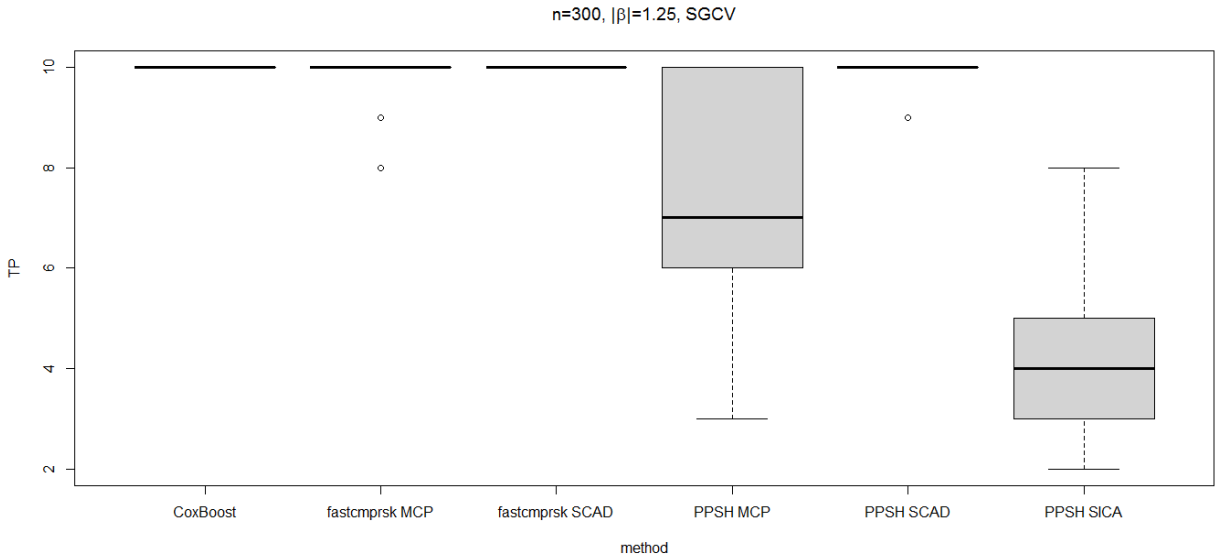



Figure 3.24: Boxplot of FP when $n = 300$ and $|\beta| = 1.25$ using SGCV score

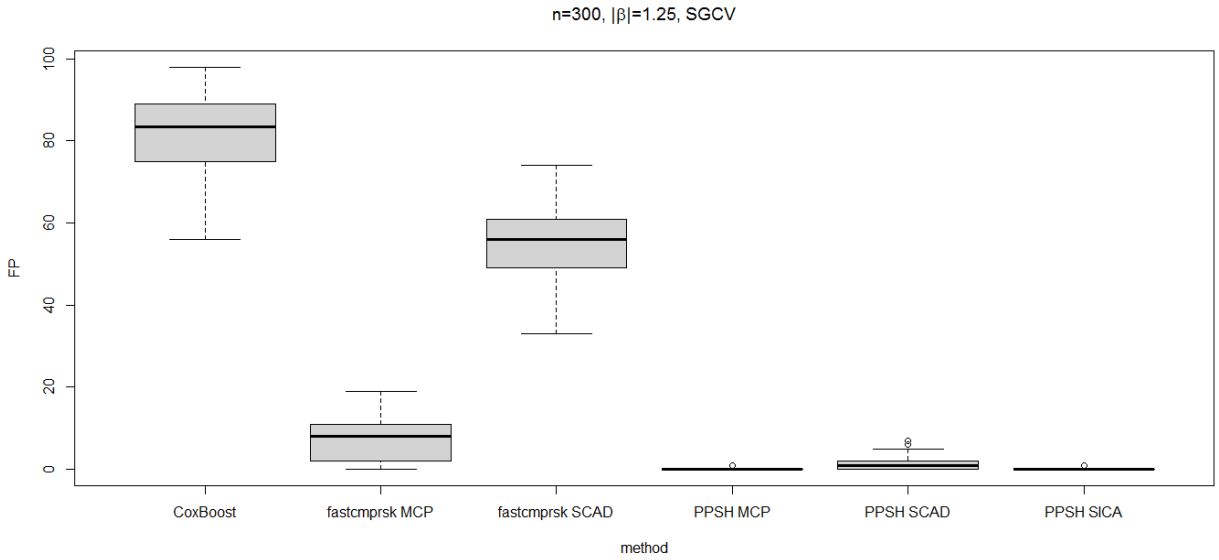


Figure 3.25: Boxplot of TP when $n = 400$ and $|\beta| = 0.75$ using CV score

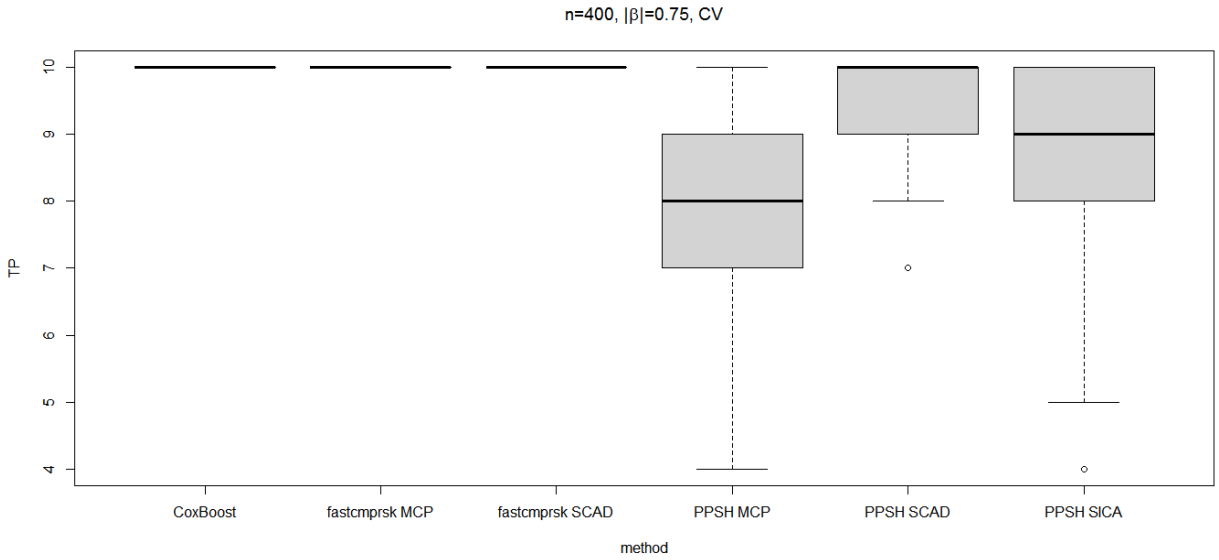



Figure 3.26: Boxplot of FP when $n = 400$ and $|\beta| = 0.75$ using CV score

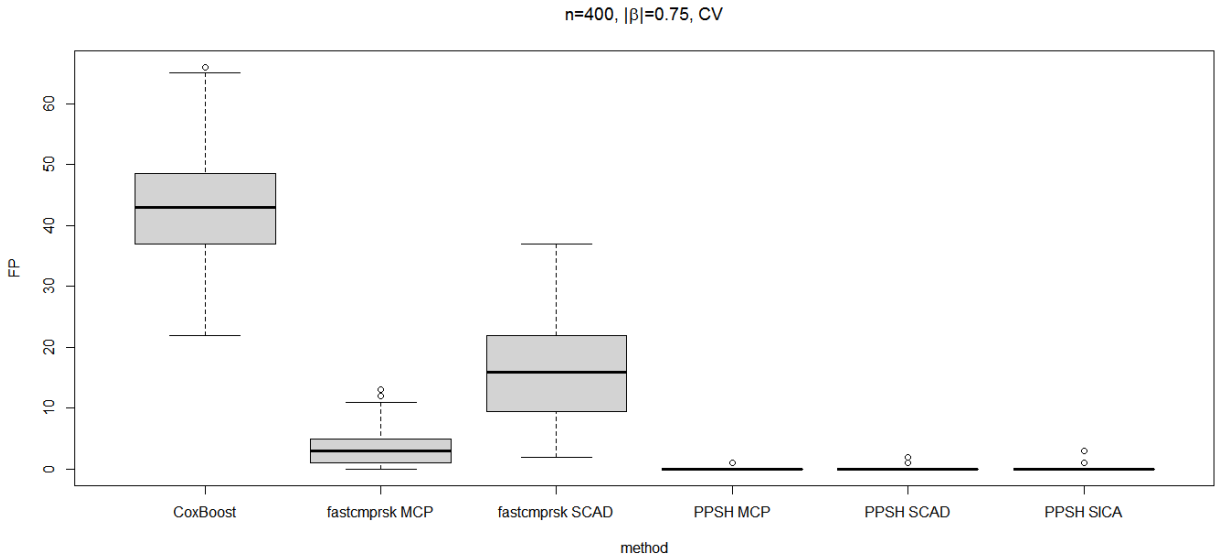


Figure 3.27: Boxplot of TP when $n = 400$ and $|\beta| = 0.75$ using SGCV score

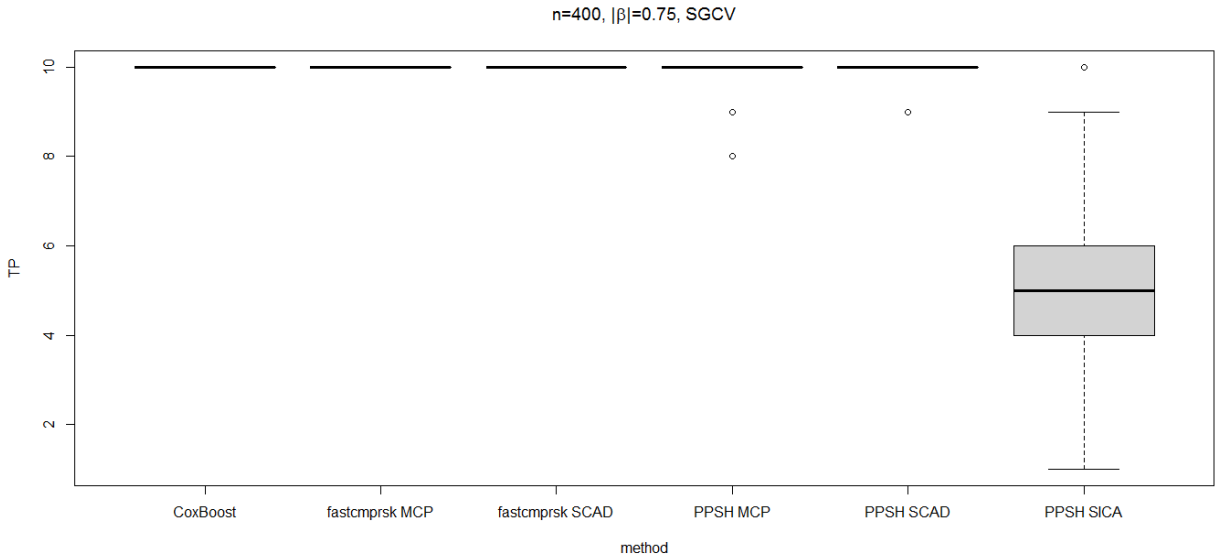



Figure 3.28: Boxplot of FP when $n = 400$ and $|\beta| = 0.75$ using SGCV score

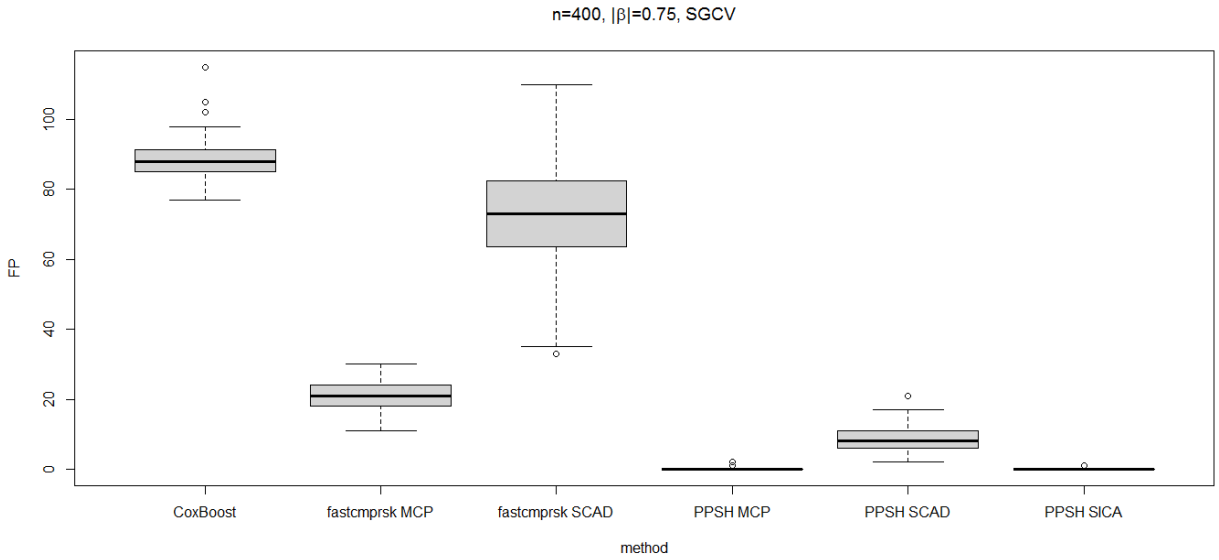


Figure 3.29: Boxplot of TP when $n = 400$ and $|\beta| = 1$ using CV score

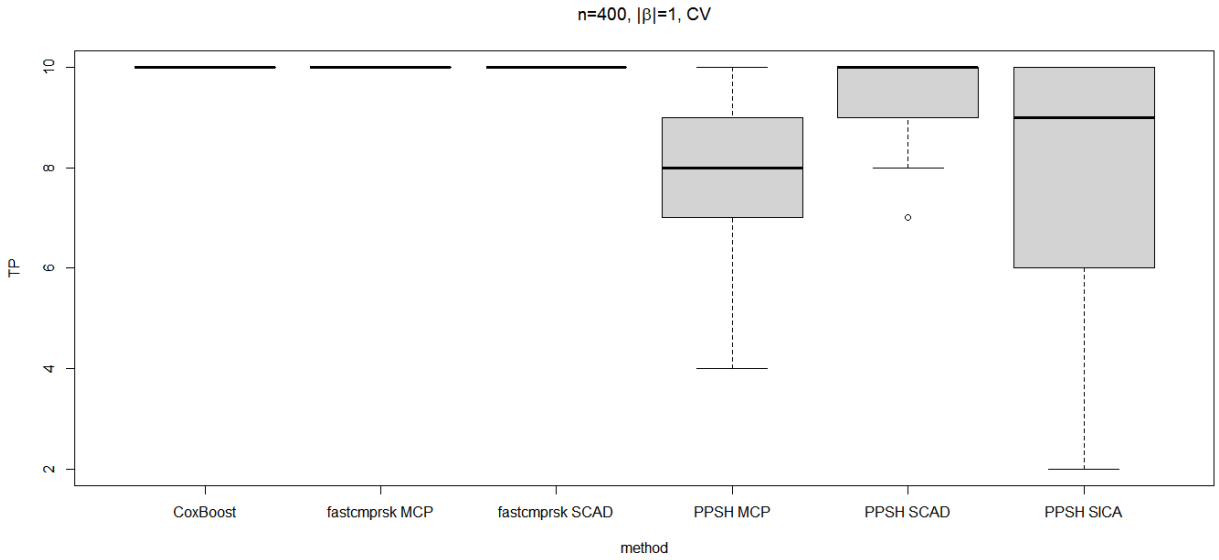



Figure 3.30: Boxplot of FP when $n = 400$ and $|\beta| = 1$ using CV score

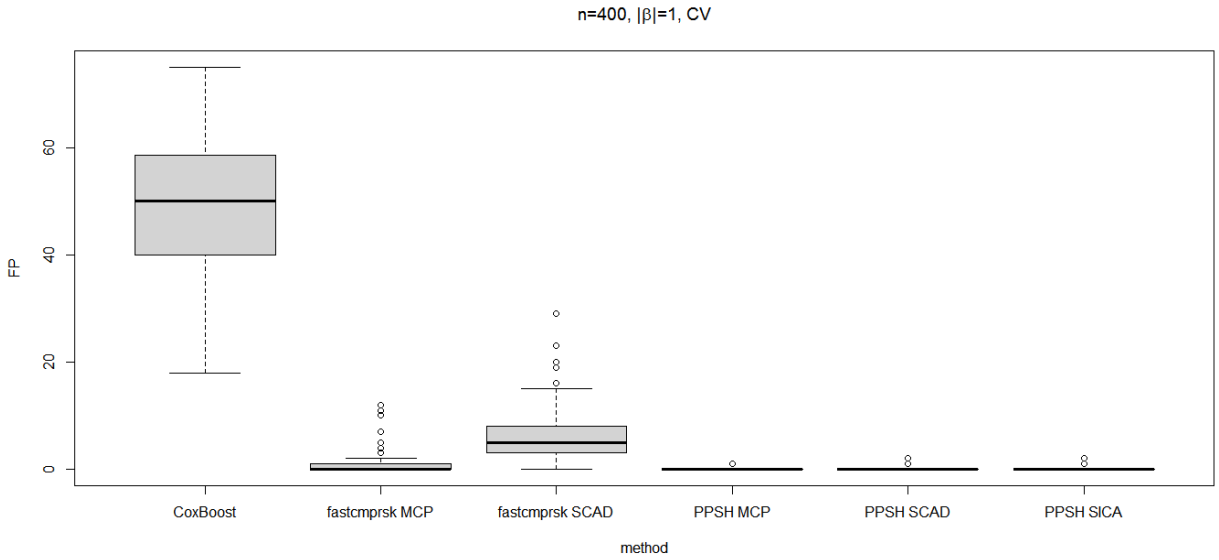


Figure 3.31: Boxplot of TP when $n = 400$ and $|\beta| = 1$ using SGCV score

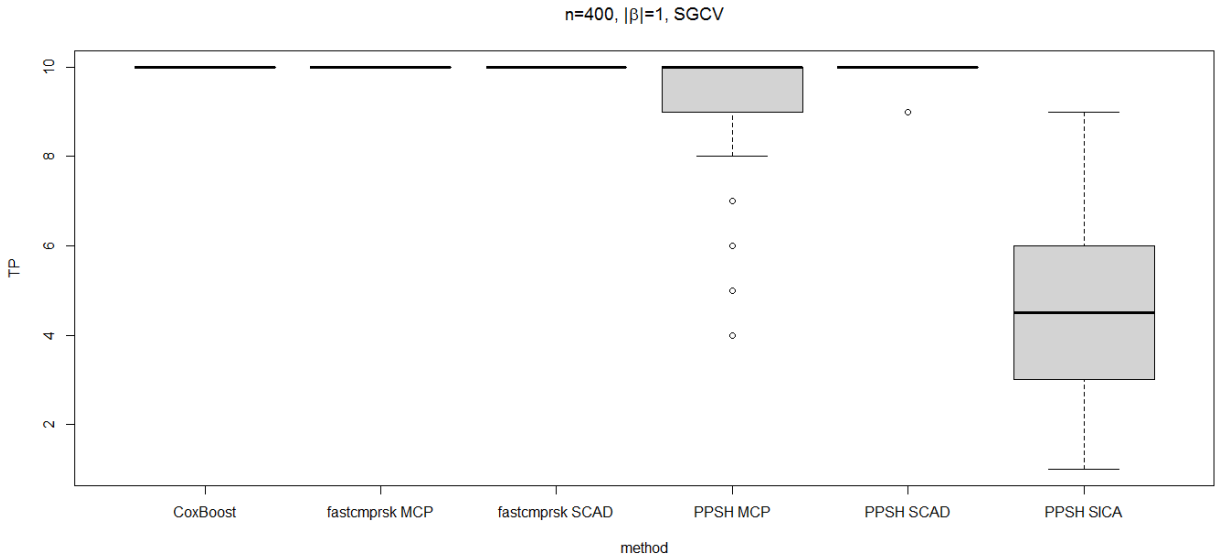



Figure 3.32: Boxplot of FP when $n = 400$ and $|\beta| = 1$ using SGCV score

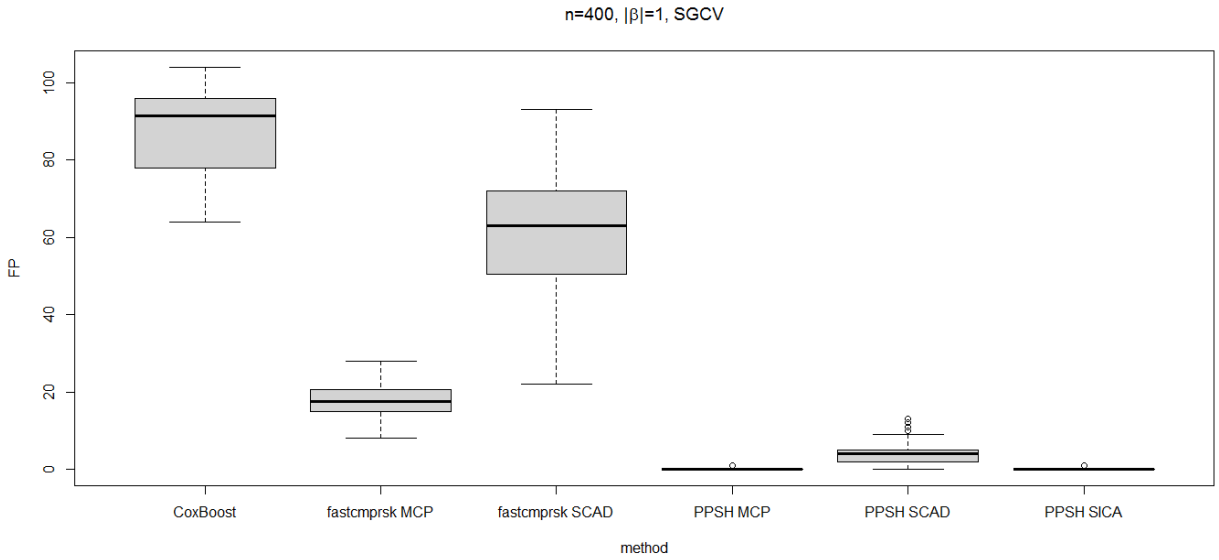


Figure 3.33: Boxplot of TP when $n = 400$ and $|\beta| = 1.25$ using CV score

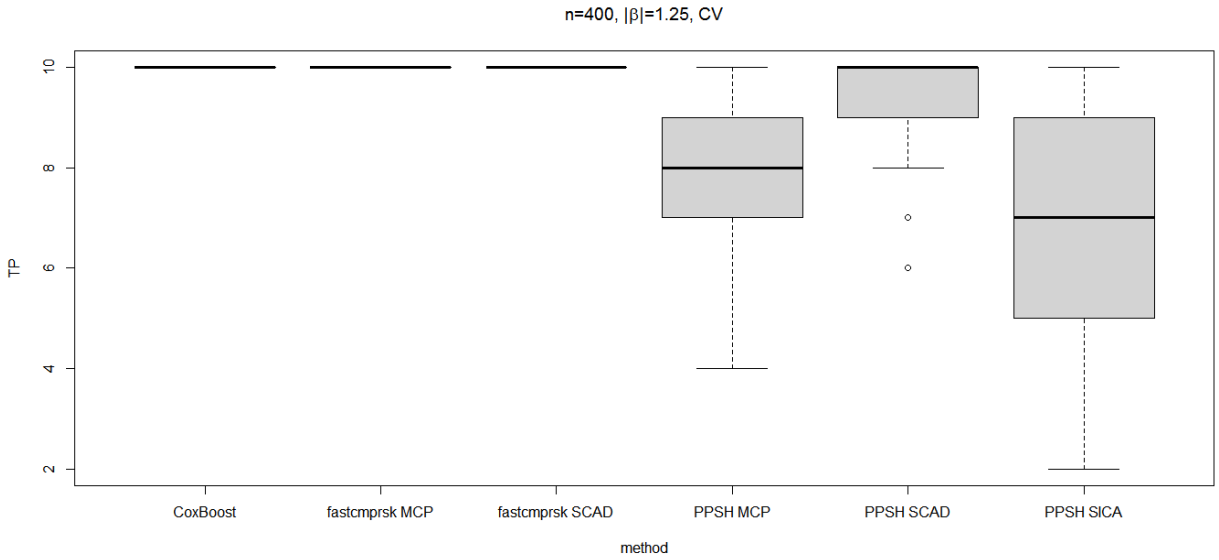



Figure 3.34: Boxplot of FP when $n = 400$ and $|\beta| = 1.25$ using CV score

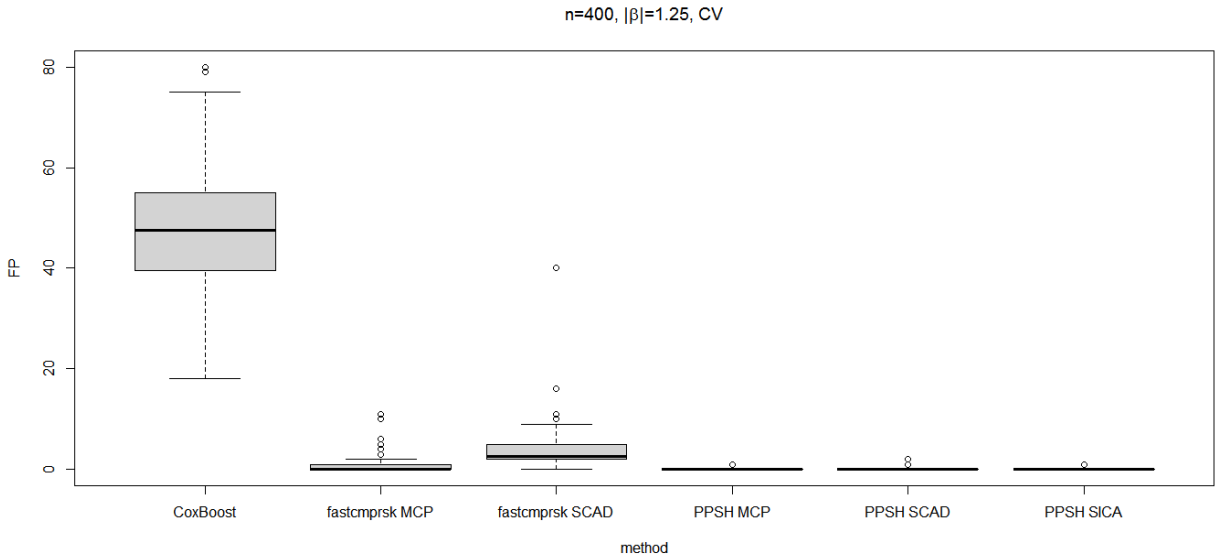


Figure 3.35: Boxplot of TP when $n = 400$ and $|\beta| = 1.25$ using SGCV score

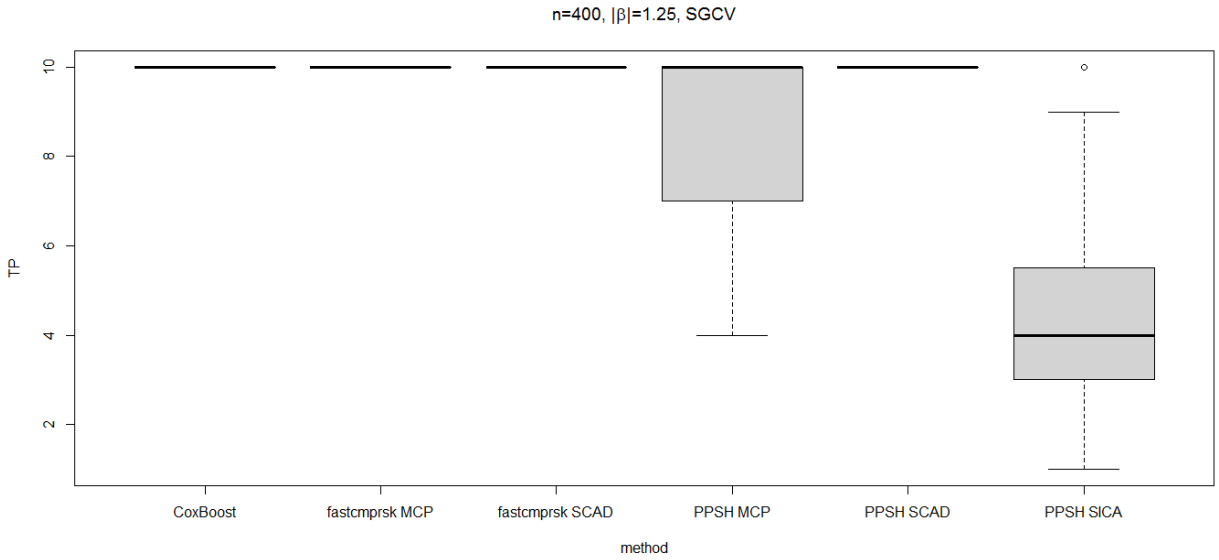
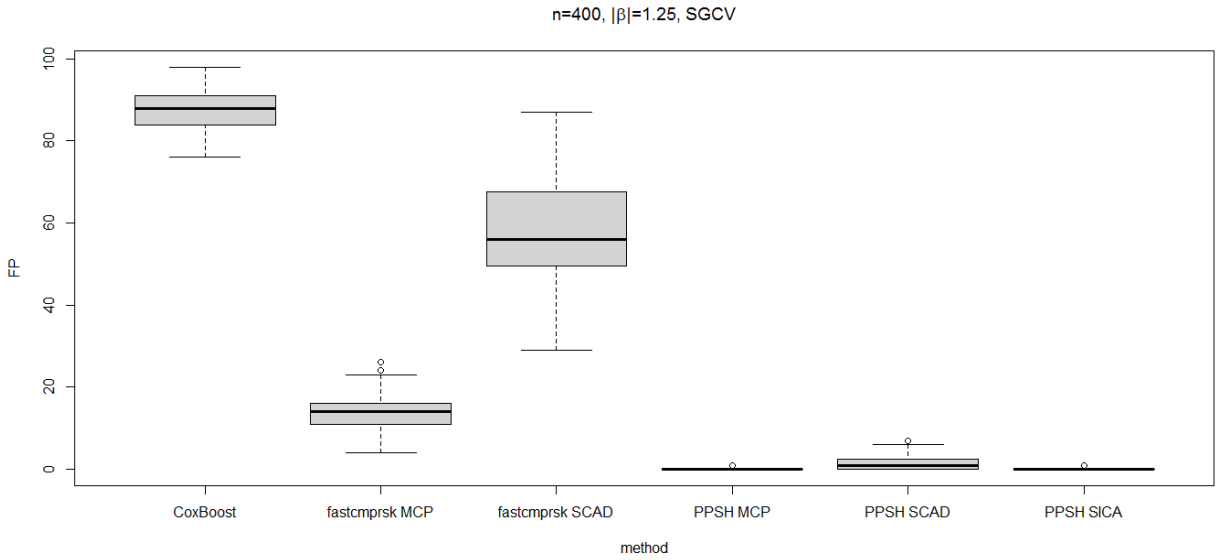



Figure 3.36: Boxplot of FP when $n = 400$ and $|\beta| = 1.25$ using SGCV score



We can see that using the proposed algorithm, SCAD always selects both more true positives and more false positives, than MCP. SCAD always selects more true positives when using the CV score as criterion and more true positives and false positives when using the SGCV score as criterion than SICA. MCP always selects more true positives than SICA when using the SGCV score as criterion. SCAD would select more true positives and fewer false positives, using the CV score as criterion, as sample size increases. As effect size increases, SCAD selects fewer false positives. MCP selects more true positives and fewer false positives, using either CV score or SGCV score as criterion, as sample size increases. SICA selects fewer false positives using the CV score as criterion, as effect size increases.

CoxBoost tends to select more true positives than the proposed method but at the cost of much more false positives. randomForestSRC performs poorly at variable selection, selecting few true positives and a rather large number of false positives. fastcmprsk has difficulty fitting models to very high-dimensional, low signal data, especially with MCP. For those replications where fastcmprsk could successfully run, like CoxBoost, it selects more true positives but many more false positives too.

## 3.2 Application to the AML Dataset

### 3.2.1 Setup

The AML patients are from a nationally representative, well-phenotyped cohort whereby all patients were enrolled onto Cancer and Leukemia Group B (CALGB) or Alliance for Clinical Trials in Oncology (Alliance) clinical trials and companion studies. RNA-sequencing assays were performed through The Ohio State University Comprehensive Cancer Center's Genomics Shared Resource, which used ribosomal RNA-depleted RNA-seq protocols to capture RNA transcripts independent of polyadenylation status. Quality of total RNA was

assessed on an Agilent 2100 Bioanalyzer (BioA) using the RNA 6000 Nanochip and quantity was assessed on a Qubit 2.0 Fluorometer (Agilent Technologies, Santa Clara, CA) using the RNA HS Assay Kit. Samples with an RNA Integrity Number (RIN) greater than four, with no visible sign of genomic DNA (gDNA) contamination and a concentration of $> 40$ ng/L were used for total RNA library generation. RNA-seq libraries were prepared using the Illumina TruSeq Stranded Total RNA Sample Prep Kit with RiboZero Gold (#RS1222201) according to the manufacturer's instructions. Sequencing was performed with the Illumina HiSeq 2500 system using the HiSeq version 3 sequencing reagents to an approximate cluster density of 800,000/$mm^2$. Image analysis, base calling, error estimation, and quality thresholds were performed using the HiSeq Controller Software (version 2.2.38) and the Real Time Analyzer software (version 1.18.64). Martin (2011) [72] was used for adapter trimming and FastQC was used for quality control of the FASTQ files. After removing reads that aligned to repeats, mitochondria, rRNAs, and other sequences that are not of interest, paired-end reads were aligned to the human genome (GENECODE ver22) using STAR for aligning the short reads [21]. Thereafter, Htseq was used to quantify mRNA expression [4]. Data were then voom normalized and log2 transformed [62].

Variables used as predictors included sex, white blood cell count (wbc), hemoglobin, platelet count, percent of blasts in bone marrow (bmblasts), percent of blasts in peripheral blood (pbblasts), age at complete remission, cytogenetic group (cyto_group), mutation status of ASXL1, mutation status of BCOR, indicator of whether there is a double mutation in CEBP$\alpha$ (cebpa_double), mutation status of DNMT3A anywhere but the R882 position (DNMT3A_nonR882), mutation status of DNMT3A in the R882 position (DNMT3A_R882), indicator of presence of FLT3-internal tandem duplication (FLT3-ITD), mutation status of FLT3-TKD, mutation status of GATA2, mutation status of IDH1, mutation status of IDH2, mutation status of NPM1, mutation status of NRAS, mutation status of PTPN11, mutation status of RUNX1, mutation status of SRSF2, mutation status of TET2, mutation status

of TP53, mutation status of WT1 and 35226 mRNA expression variables. Race was not used because it has so many levels with many having too few patients included. European LeukemiaNet prognostic group membership based on [23] was not used because it is not a primary measure on the patients, rather it is derived using cytogenetic and the selected mutation data. For categorical variables, the most frequently observed category serves as the reference. 583 patients achieved complete remission out of 816 patients. There were three patients who relapsed but their relapse dates were not recorded so they were not included in the analysis. Thus 580 patients were included in the subsequent analysis. Among them, 348 patients relapsed, 57 patients died without relapse, and 175 patients were lost to follow-up without relapse or death. The baseline characteristics of these 580 patients are summarized in Table 3.2. "Overall" denotes the number of patients in each level and its percentage for a categorical variable, and mean and standard deviation for a continuous variable. "Missing" denotes the percentage of missing values.

Table 3.2: Summary of baseline characteristics of the patients who achieved complete remission

| | level | Overall | Missing |
|---|---|---|---|
| n | | 580 | |
| cyto_group (%) | 2 unbalanced rearrangements | 24 ( 4.1) | 0 |
| | complex | 32 ( 5.5) | |
| | inv(16) | 65 (11.2) | |
| | inv(3) | 0 ( 0.0) | |
| | normal | 292 (50.3) | |
| | other 11q23 | 14 ( 2.4) | |
| | other balanced rearrangements | 6 ( 1.0) | |

Table 3.2: Continued

| | level | Overall | Missing |
|---|---|---|---|
| | other possible similar to inv(3) | 2 ( 0.3) | |
| | other sole deletions | 7 ( 1.2) | |
| | other sole trisomies and monosomies | 7 ( 1.2) | |
| | other translocaton and inversions | 18 ( 3.1) | |
| | other unbalanced | 5 ( 0.9) | |
| | sole deletion/loss 20q | 1 ( 0.2) | |
| | sole deletion/loss 5q | 3 ( 0.5) | |
| | sole deletion/loss 7q | 5 ( 0.9) | |
| | sole deletion/loss 9q | 6 ( 1.0) | |
| | sole loss of Y | 5 ( 0.9) | |
| | sole monosomy 7 | 7 ( 1.2) | |
| | sole trisomy 11 | 2 ( 0.3) | |
| | sole trisomy 13 | 3 ( 0.5) | |
| | sole trisomy 21 | 3 ( 0.5) | |
| | sole trisomy 4 | 2 ( 0.3) | |
| | sole trisomy 8 | 20 ( 3.4) | |
| | t(6;9) | 1 ( 0.2) | |
| | t(8;21) | 33 ( 5.7) | |
| | t(9;11) | 16 ( 2.8) | |
| | t(9;22) | 1 ( 0.2) | |
| Sex (%) | Female | 249 (42.9) | 0 |
| | Male | 331 (57.1) | |
| bmblasts (mean (SD)) | | 63.15 (20.29) | 0.2 |

Table 3.2: Continued

| | level | Overall | Missing |
|---|---|---|---|
| pbblasts (mean (SD)) | | 51.46 (28.06) | 0.7 |
| wbc (mean (SD)) | | 43.40 (52.34) | 1.2 |
| platelet (mean (SD)) | | 74.05 (65.48) | 1.6 |
| ASXL1 (%) | No | 544 (93.8) | 1.6 |
| | Yes | 27 ( 4.7) | |
| | <NA> | 9 ( 1.6) | |
| BCOR (%) | No | 550 (94.8) | 1.6 |
| | Yes | 21 ( 3.6) | |
| | <NA> | 9 ( 1.6) | |
| DNMT3A_nonR882 (%) | No | 530 (91.4) | 1.6 |
| | Yes | 41 ( 7.1) | |
| | <NA> | 9 ( 1.6) | |
| DNMT3A_R882 (%) | No | 466 (80.3) | 1.6 |
| | Yes | 105 (18.1) | |
| | <NA> | 9 ( 1.6) | |
| GATA2 (%) | No | 536 (92.4) | 1.6 |
| | Yes | 35 ( 6.0) | |
| | <NA> | 9 ( 1.6) | |
| IDH1 (%) | No | 533 (91.9) | 1.6 |
| | Yes | 38 ( 6.6) | |
| | <NA> | 9 ( 1.6) | |
| IDH2 (%) | No | 522 (90.0) | 1.6 |
| | Yes | 49 ( 8.4) | |

Table 3.2: Continued

| | level | Overall | Missing |
|---|---|---|---|
| | <NA> | 9 ( 1.6) | |
| NRAS (%) | No | 477 (82.2) | 1.6 |
| | Yes | 94 (16.2) | |
| | <NA> | 9 ( 1.6) | |
| PTPN11 (%) | No | 528 (91.0) | 1.6 |
| | Yes | 43 ( 7.4) | |
| | <NA> | 9 ( 1.6) | |
| RUNX1 (%) | No | 530 (91.4) | 1.6 |
| | Yes | 41 ( 7.1) | |
| | <NA> | 9 ( 1.6) | |
| TET2 (%) | No | 512 (88.3) | 1.6 |
| | Yes | 59 (10.2) | |
| | <NA> | 9 ( 1.6) | |
| TP53 (%) | No | 549 (94.7) | 1.6 |
| | Yes | 22 ( 3.8) | |
| | <NA> | 9 ( 1.6) | |
| WT1 (%) | No | 532 (91.7) | 1.6 |
| | Yes | 39 ( 6.7) | |
| | <NA> | 9 ( 1.6) | |
| NPM1 (%) | No | 343 (59.1) | 1.7 |
| | Yes | 227 (39.1) | |
| | <NA> | 10 ( 1.7) | |
| SRSF2 (%) | No | 543 (93.6) | 2.2 |

Table 3.2: Continued

| | level | Overall | Missing |
|---|---|---|---|
| | Yes | 24 ( 4.1) | |
| | <NA> | 13 ( 2.2) | |
| hglobin (mean (SD)) | | 9.36 (2.03) | 2.8 |
| FLT3.TKD (%) | No | 513 (88.4) | 2.8 |
| | Yes | 51 ( 8.8) | |
| | <NA> | 16 ( 2.8) | |
| FLT3.ITD (%) | No | 435 (75.0) | 3.3 |
| | Yes | 126 (21.7) | |
| | <NA> | 19 ( 3.3) | |
| cebpa_double (%) | No | 387 (66.7) | 23.8 |
| | Yes | 55 ( 9.5) | |
| | <NA> | 138 (23.8) | |
| age (mean (SD)) | | 46.26 (13.67) | 0 |

The histogram of correlations between the mRNA expression variables is plotted in Figure 3.37.

The cumulative incidence functions of relapse and death without relapsed are estimated and plotted using the cmprsk package, shown in Figure 3.38.

The mRNA expression data are complete but there are some missing values in some of the other variables thus we need to impute them first. The VIM package could not be installed on Unity or Ohio Supercomputer Center. The mice package can run on low-dimensional data but gives an error on our high-dimensional data. The rMIDAS package could not be installed on Unity and though it could be installed it gave an error when running on Ohio

Supercomputer Center. The only available R package that can handle our high-dimensional data is the missForest package so it was used to impute the missing data. The "mtry" argument, the number of variables randomly sampled at each split, was set to 100 per the package's recommendation and all the other arguments were set to the default values, in the "missForest" function. Thirty copies of imputed datasets were obtained from running the function.

Categorical variables were transformed into dummy variables using the R package fastDummies. The maximum values of $\lambda$ for SCAD and MCP were set using the same formula used in the simulation study. The maximum values of $\lambda$ for SICA was set to be 0.05. The minimum value of $\lambda$ was set to be $0.5\times$ the maximum $\lambda$ for SCAD, $0.4\times$ the maximum $\lambda$ for MCP and $0.2\times$ the maximum $\lambda$ for SICA. 100 candidate values of $\lambda$ were used and generated in the same way as in the simulation study. 10-fold cross-validation was performed to choose the optimal value of $\lambda$. Different folds were used in the cross-validation for each of the 30 copies of imputed datasets but the same 10 folds were used for each of the three penalty functions in each dataset.

Mathematically, $h_i(\tilde{\boldsymbol{\eta}}) < 0,\ i = 1, ..., n$ but when running my code on the AML dataset, when $\lambda$ gets smaller, the range of $\tilde{\beta}_q,\ q = 1, ..., p$ gets larger and some values of $\tilde{\eta}_i,\ i = 1, ..., n$ can get very large. Then $\frac{w_i(T_k)Y_i(T_k)\exp(\tilde{\eta}_i)}{\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)}$ gets very close to 1 or 0 for each $k$ s.t. $\Delta_k\epsilon_k = 1$. Then $[\frac{w_i(T_k)Y_i(T_k)\exp(\tilde{\eta}_i)}{\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)}]^2 - \frac{w_i(T_k)Y_i(T_k)\exp(\tilde{\eta}_i)}{\sum_{q=1}^{n} w_q(T_k)Y_q(T_k)\exp(\tilde{\eta}_q)}$ gets very close to 0. Hence $h_i(\tilde{\boldsymbol{\eta}})$ gets very close to 0. Sometimes, unfortunately, $h_i(\tilde{\boldsymbol{\eta}})$ is too close to 0 that R treats it as 0. Occasionally, even more unfortunately, all $h_i(\tilde{\boldsymbol{\eta}}),\ i = 1, ..., n$ might be treated as 0 by R.

This happened to MCP on most of the 30 imputed datasets but not to SCAD or SICA.

$$f(\boldsymbol{\beta}) = \frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})[y_i^2(\tilde{\boldsymbol{\eta}}) - 2y_i(\tilde{\boldsymbol{\eta}})\mathbf{z}_i^T\boldsymbol{\beta} + (\mathbf{z}_i^T\boldsymbol{\beta})^2] - n\sum_{q=1}^{p} p(\beta_q)$$
$$= \frac{1}{2}[\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i^2(\tilde{\boldsymbol{\eta}}) - 2\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})\mathbf{z}_i^T\boldsymbol{\beta} + \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})(\mathbf{z}_i^T\boldsymbol{\beta})^2] - n\sum_{q=1}^{p} p(\beta_q)$$

Fix $\beta_k,\ k \neq q$,

$$f_q(\beta_q) = \frac{1}{2}[\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i^2(\tilde{\boldsymbol{\eta}}) - 2\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})\sum_{k\neq q} z_{ik}\beta_k - 2\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})z_{iq}\beta_q$$
$$+ \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})(\mathbf{z}_i^T\boldsymbol{\beta})^2] - n\sum_{k=1}^{p} p(\beta_k)$$
$$= -[\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})z_{iq}]\beta_q + \frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})(\mathbf{z}_i^T\boldsymbol{\beta})^2 - np(\beta_q) + c,$$

where $c$ does not depend on $\beta_q$. When $h_i(\tilde{\boldsymbol{\eta}}),\ i = 1, ...n$ are treated as 0 by R,

$$f_q(\beta_q) = -[\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})z_{iq}]\beta_q - np(\beta_q) + c.$$

Remember that only $h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}}) = h_i(\tilde{\boldsymbol{\eta}})\tilde{\eta}_i - \frac{\partial l_\eta}{\partial \eta_i}(\tilde{\boldsymbol{\eta}})$ is calculated so even if $h_i(\tilde{\boldsymbol{\eta}})$ is treated as 0 in R, $h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})$ may not necessarily equal 0. If $\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})z_{iq} = 0,\ f_q(\beta_q)$ is maximized when $\beta_q = 0$. Otherwise $f_q(\beta_q)$ doesn't have a maximum point because $[\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}})y_i(\tilde{\boldsymbol{\eta}})z_{iq}]\beta_q$ can go to $\infty$ when $\beta_q$ goes to $\infty$ and $p(\beta_q)$ is bounded for SCAD, MCP and SICA so $f_q(\beta_q)$ can go to $\infty$ when $\beta_q$ goes to $\infty$.

To address this issue, the algorithm was rerun without using the warm start approach. Instead, $\mathbf{0}$ is used as the initial value for each value of $\lambda$. This issue was then avoided in most cases but in a few cases this still happened, in which case the algorithm was terminated and the current value of $\tilde{\boldsymbol{\beta}}$ was used as the estimate of $\hat{\boldsymbol{\beta}}$ for the current value of $\lambda$. The

resulting $\hat{\boldsymbol{\eta}}$ usually included large elements treated as $\infty$ by R so the way $l(\hat{\boldsymbol{\beta}})$ is calculated also required modification. Note $l(\hat{\boldsymbol{\beta}}) = l_\eta(\hat{\boldsymbol{\eta}})$, where $\hat{\boldsymbol{\eta}} = \mathbf{Z}\hat{\boldsymbol{\beta}}$. For $i$ s.t. $\Delta_i\epsilon_i = 1$, let $\hat{\eta}_m = \max\{\hat{\eta}_q,\ q = 1, ..., n | w_q(T_i)Y_q(T_i) > 0\}$. If $\exp(\hat{\eta}_m)$ is treated as finite in R, then $w_q(T_i)Y_q(T_i)\exp(\hat{\eta}_q)$ can be directly calculated for $\hat{\eta}_q \leq \hat{\eta}_m$. Otherwise, calculate

$$
\begin{aligned}
&(\hat{\eta}_i - factor) - \log[\sum_{q=1}^{n} w_q(T_i)Y_q(T_i)\exp(\hat{\eta}_q - factor)] \\
&= (\hat{\eta}_i - factor) - \log[\sum_{q=1}^{n} w_q(T_i)Y_q(T_i)\exp(\hat{\eta}_q)/\exp(factor)] \\
&= (\hat{\eta}_i - factor) - \{\log[\sum_{q=1}^{n} w_q(T_i)Y_q(T_i)\exp(\hat{\eta}_q)] - factor\} \\
&= \hat{\eta}_i - \log[\sum_{q=1}^{n} w_q(T_i)Y_q(T_i)\exp(\hat{\eta}_q)].
\end{aligned}
$$

where $factor$ is chosen to be $\hat{\eta}_m - 709$. For $\hat{\eta}_q > \hat{\eta}_m,\ w_q(T_i)Y_q(T_i)\exp(\hat{\eta}_q) = 0$.

### 3.2.2 Results

The predictors that were at least selected once and their number of times of being selected using SICA, MCP and SCAD are listed in Tables 3.4-3.6, respectively. Times_CV denotes the number of times being selected using the CV score as the criterion in cross-validation and Times_SGCV denotes it when using the SGCV score as the criterion. "cyto_group_complex" is a dummy variable created from the categorical variable "cyto_group". The reference category is "normal".

Table 3.3: Number of times predictors selected using SICA on 30 imputed datasets

| Predictor | Times_CV | Times_SGCV |
|---|---|---|
| SCN9A | 12 | 0 |
| ENSG00000223528 | 11 | 0 |

Figure 3.37: Histogram of correlations between mRNA expression variables

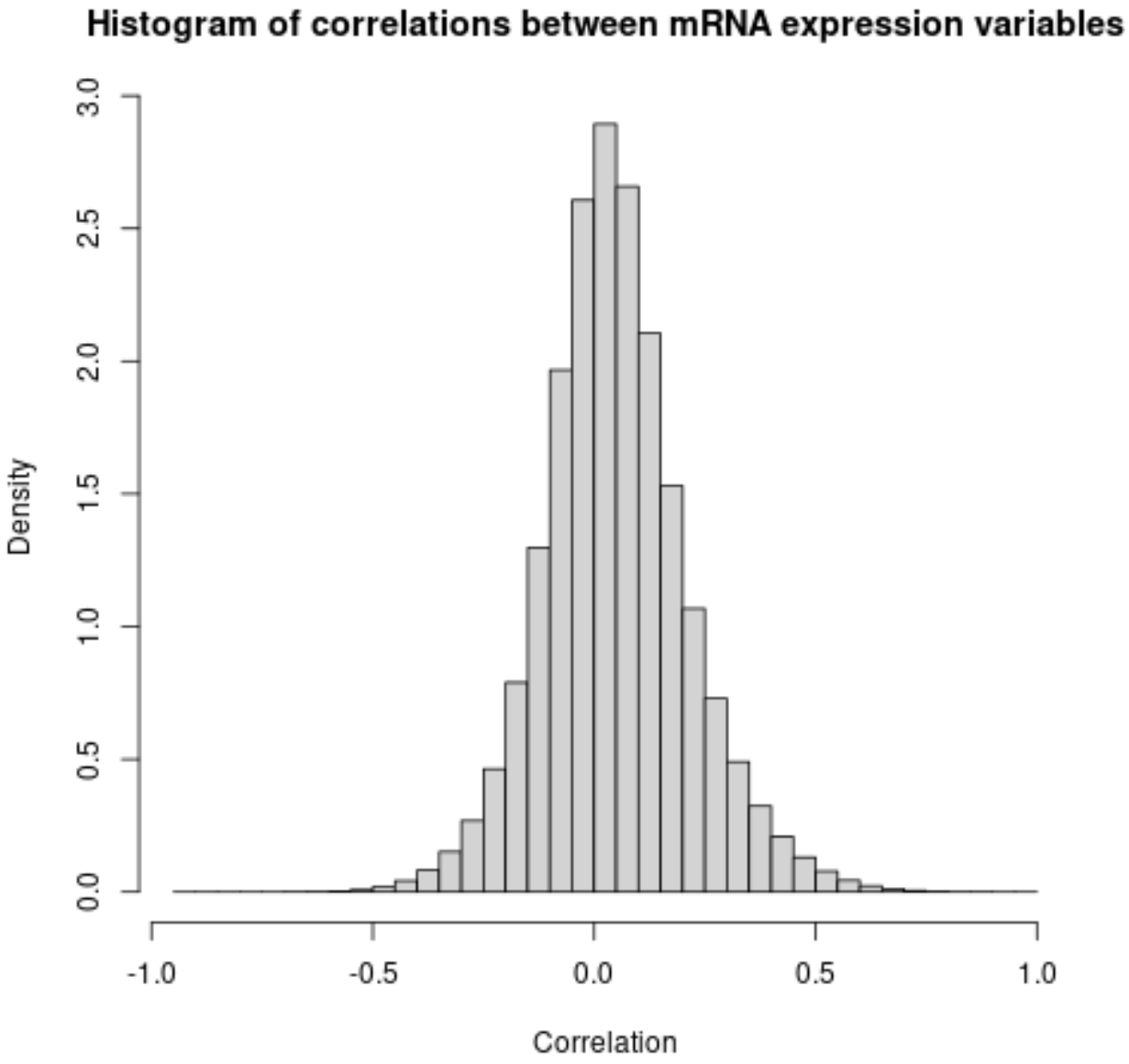


Figure 3.38: Estimated cumulative incidence functions

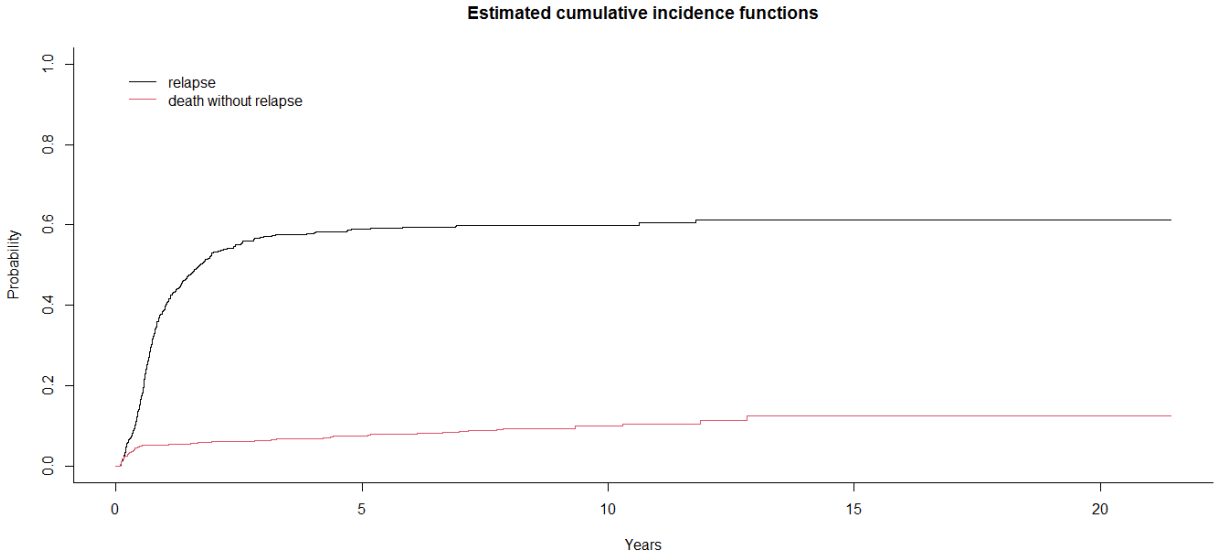


Table 3.4: Number of times predictors selected using MCP on 30 imputed datasets

| Predictor | Times_CV | Times_SGCV |
|---|---|---|
| ENSG00000233451 | 29 | 21 |
| ENSG00000271857 | 16 | 9 |
| ENSG00000248347 | 14 | 11 |
| ENSG00000223528 | 12 | 12 |
| RPS3AP41 | 12 | 4 |
| CD109 | 11 | 9 |
| PTMAP5 | 10 | 13 |
| SCN9A | 10 | 10 |
| SSPN | 9 | 3 |
| RPS2P21 | 8 | 3 |
| LINC01770 | 7 | 4 |
| ALDH2 | 7 | 2 |
| LINC01979 | 5 | 3 |
| GARRE1 | 4 | 1 |
| EGFEM1P | 3 | 4 |
| RN7SKP32 | 3 | 1 |
| PLCB4 | 2 | 4 |
| HOPX | 2 | 1 |
| GAS6 | 2 | 1 |
| SDHAP3 | 1 | 3 |
| PTMAP3 | 1 | 3 |
| ENSG00000262172 | 1 | 3 |

Table 3.4: Continued

| | | |
|---|---|---|
| DDX19B | 1 | 2 |
| CLIC4 | 1 | 2 |
| CALCRL | 1 | 1 |
| CIBAR1 | 1 | 1 |
| PTMAP4 | 1 | 1 |
| SPAG6 | 1 | 0 |
| KIF1A | 1 | 0 |
| IL2RA | 1 | 0 |
| CIBAR1P1 | 1 | 0 |
| CENPV | 1 | 0 |
| FRG2C | 1 | 0 |
| CD34 | 1 | 0 |
| PTMAP9 | 1 | 0 |
| ENSG00000228201 | 1 | 0 |
| EIF5AL1 | 1 | 0 |
| LINC02421 | 1 | 0 |
| LINC01415 | 1 | 0 |
| ENSG00000255232 | 0 | 3 |
| EREG | 0 | 2 |
| C1QL1 | 0 | 2 |
| ZNF355P | 0 | 2 |
| INSYN2A | 0 | 2 |
| LINC00676 | 0 | 2 |
| PCDHGC5 | 0 | 2 |

Table 3.4: Continued

| | | |
|---|---|---|
| CCDC179 | 0 | 2 |
| MADCAM1-AS1 | 0 | 2 |
| LINC02080 | 0 | 2 |
| NXF2 | 0 | 2 |
| ENSG00000279024 | 0 | 2 |
| LINC02359 | 0 | 2 |
| cyto_group_complex | 0 | 2 |
| WT1 | 0 | 2 |
| APOL4 | 0 | 1 |
| FOXP2 | 0 | 1 |
| ITGB4 | 0 | 1 |
| MIP | 0 | 1 |
| THRB | 0 | 1 |
| IRX1 | 0 | 1 |
| PLEKHD1 | 0 | 1 |
| TIMM22 | 0 | 1 |
| PPP1R42 | 0 | 1 |
| SAGE1 | 0 | 1 |
| OR1J2 | 0 | 1 |
| ENSG00000203334 | 0 | 1 |
| SNORD51 | 0 | 1 |
| GPR166P | 0 | 1 |
| C10orf55 | 0 | 1 |
| LOC112268293 | 0 | 1 |

Table 3.4: Continued

| | | |
|---|---|---|
| PSMD2P1 | 0 | 1 |
| CCT5P2 | 0 | 1 |
| LOC101927143 | 0 | 1 |
| ZNRF3-IT1 | 0 | 1 |
| OLMALINC | 0 | 1 |
| RPL36AP26 | 0 | 1 |
| ENSG00000237301 | 0 | 1 |
| RNU7-169P | 0 | 1 |
| ENSG00000242795 | 0 | 1 |
| RPSAP3 | 0 | 1 |
| ENSG00000243744 | 0 | 1 |
| RPS24P17 | 0 | 1 |
| SCARF2 | 0 | 1 |
| TARS1-DT | 0 | 1 |
| ENSG00000251293 | 0 | 1 |
| ENSG00000274303 | 0 | 1 |
| IGHV2-70 | 0 | 1 |
| ENSG00000275216 | 0 | 1 |
| GXYLT1P5 | 0 | 1 |

Table 3.5: Number of times predictors selected using SCAD on 30 imputed datasets

| Predictor | Times_CV | Times_SGCV |
|---|---|---|
| GARRE1 | 30 | 30 |

Table 3.5: Continued

| | | |
|---|---|---|
| RPS2P21 | 30 | 30 |
| ENSG00000233451 | 30 | 30 |
| RPS3AP41 | 30 | 30 |
| ENSG00000271857 | 30 | 28 |
| SSPN | 28 | 26 |
| CD109 | 28 | 26 |
| SDHAP3 | 27 | 28 |
| ENSG00000262172 | 27 | 28 |
| SCN9A | 27 | 24 |
| ENSG00000248347 | 25 | 26 |
| ENSG00000223528 | 24 | 17 |
| ENSG00000228303 | 23 | 27 |
| ENSG00000275216 | 23 | 26 |
| CLIC4 | 21 | 25 |
| ALDH2 | 21 | 16 |
| PTMAP5 | 21 | 13 |
| C1QL1 | 19 | 28 |
| LINC01979 | 19 | 17 |
| TUBB2BP1 | 18 | 23 |
| GAS6 | 17 | 16 |
| LINC01770 | 16 | 11 |
| HOPX | 14 | 16 |
| TMEM217 | 14 | 8 |
| PLCB4 | 13 | 13 |

Table 3.5: Continued

| | | |
|---|---|---|
| KIF1A | 12 | 18 |
| cyto_group_complex | 12 | 16 |
| CENPV | 12 | 10 |
| STAR | 11 | 10 |
| PTMAP3 | 10 | 17 |
| LINC00676 | 10 | 17 |
| CLEC3B | 10 | 15 |
| ENSG00000228201 | 10 | 13 |
| MSLN | 10 | 9 |
| CYCSP23 | 9 | 12 |
| EGFEM1P | 9 | 10 |
| IL2RA | 9 | 9 |
| ZNF355P | 8 | 22 |
| DDX19B | 8 | 9 |
| LINC01415 | 8 | 6 |
| ENSG00000229664 | 7 | 14 |
| MADCAM1-AS1 | 7 | 11 |
| ENSG00000251293 | 7 | 9 |
| ENSG00000251467 | 7 | 7 |
| ENSG00000261346 | 7 | 5 |
| CASP10 | 6 | 14 |
| RBM3 | 5 | 10 |
| MPZ | 5 | 9 |
| MIR155HG | 5 | 8 |

Table 3.5: Continued

| | | |
|---|---|---|
| RN7SKP32 | 5 | 5 |
| CIBAR1P1 | 5 | 3 |
| TPM3P1 | 5 | 3 |
| INTS6-AS1 | 5 | 2 |
| OLMALINC | 4 | 4 |
| DDIT4 | 4 | 3 |
| HSPE1P22 | 4 | 3 |
| VSTM4 | 3 | 10 |
| ENSG00000243744 | 3 | 9 |
| PTMAP4 | 3 | 6 |
| PTMAP9 | 3 | 5 |
| CD82 | 2 | 2 |
| TRIM9 | 2 | 7 |
| APOL4 | 1 | 5 |
| PTMAP12 | 2 | 4 |
| CIBAR1 | 2 | 2 |
| TMEM273 | 2 | 1 |
| PTMA | 2 | 0 |
| EREG | 1 | 7 |
| DOCK1 | 1 | 7 |
| SDCBP2 | 1 | 3 |
| ENSG00000242951 | 1 | 3 |
| INSYN2A | 1 | 2 |
| MIR1244-3 | 1 | 2 |

Table 3.5: Continued

| | | |
|---|---|---|
| CALCRL | 1 | 1 |
| SMAD5 | 1 | 1 |
| GAPDHP59 | 1 | 1 |
| C10orf55 | 0 | 6 |
| IL17RE | 0 | 3 |
| PSG2 | 0 | 3 |
| PSMC1P9 | 0 | 3 |
| LINC01978 | 0 | 3 |
| ST7/ST7-OT3 | 0 | 2 |
| CCDC68 | 0 | 2 |
| ACTBP11 | 0 | 2 |
| C10orf105 | 0 | 2 |
| ZNF460-AS1 | 0 | 2 |
| LINC02359 | 0 | 2 |
| RBM23 | 0 | 1 |
| THG1L | 0 | 1 |
| ENPP2 | 0 | 1 |
| CD34 | 0 | 1 |
| DMRTC1B | 0 | 1 |
| PTMAP2 | 0 | 1 |
| CHP1P1 | 0 | 1 |
| HNRNPA1P55 | 0 | 1 |
| RAC1P2 | 0 | 1 |
| EIF5AL1 | 0 | 1 |

Table 3.5: Continued

| | | |
|---|---|---|
| PRORP | 0 | 1 |
| ENSG00000271882 | 0 | 1 |
| PACERR | 0 | 1 |
| FLT3.ITD | 0 | 1 |

We can see that SCAD selects the most predictors while SICA selects the fewest predictors.

For comparison, the proposed method and other competing methods were also applied to the complete dataset, containing only those patients who had no missing value on any of the predictors. The randomForestSRC package can be downloaded on the Ohio Supercomputer Center but not on Unity. But the AML data can't be loaded on the Ohio Supercomputer Center so it was not used. The fastcmprsk package did not select any predictor, using either MCP or SCAD as the penalty function under either the CV or SGCV score as criterion. The proposed method selected 4, 8, 21 predictors, using SICA, MCP and SCAD, respectively, under the CV criterion, and 1, 4 and 19 predictors, using SICA, MCP and SCAD, respectively, under the SGCV criterion. The CoxBoost package selected 58 predictors under the CV criterion, including all predictors selected by the proposed method using SICA, and 6 and 18 predictors selected by MCP and SCAD, respectively. It selected 90 predictors under the SGCV criterion, including all predictors selected by the proposed method using SICA and MCP, and 17 predictors selected by SCAD. The Venn diagrams of the predictors selected by the proposed method using SCAD and CoxBoost, under the CV and SGCV criteria, respectively were plotted in Figures 3.39-3.40.

Figure 3.39: Venn diagram of the predictors selected by the proposed method using SCAD and CoxBoost under the CV criterion

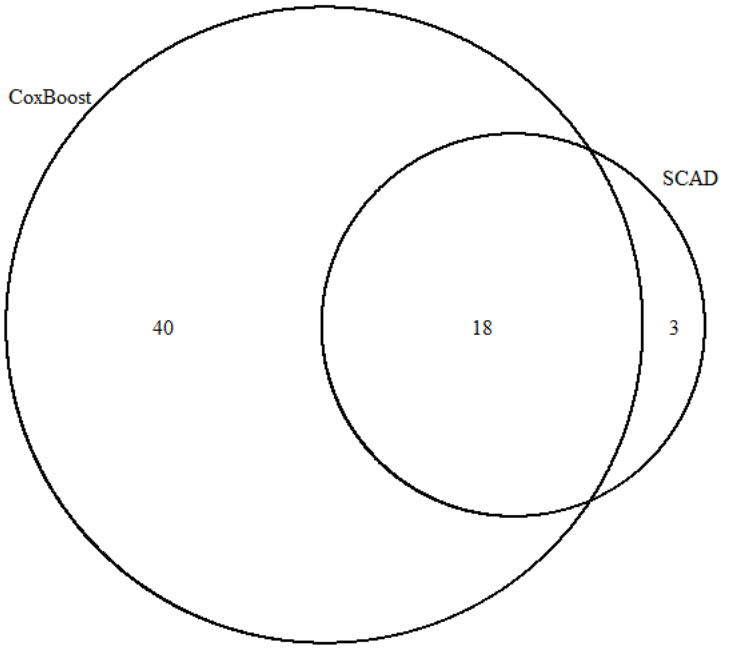
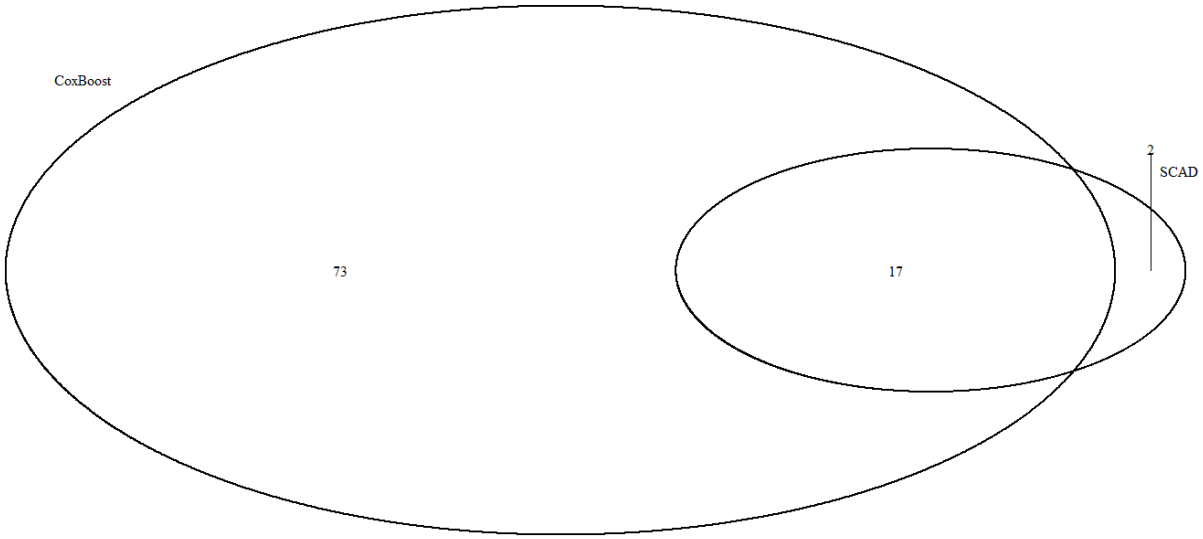



Figure 3.40: Venn diagram of the predictors selected by the proposed method using SCAD and CoxBoost under the SGCV criterion



To evaluate the predictive performance of the methods, the complete dataset was randomly partitioned into two parts of roughly equal sizes. One of them was used as the training data, on which each of the methods was applied, while the other was the testing data. The fastcmprsk package gave error using SCAD and did not select any predictor using MCP under either the CV or SGCV score. Our proposed method did not select any predictor using SICA under either the CV or SGCV score. Then the probabilities of having relapsed by 3 and 5 years since complete remission were predicted for each patient in the testing data, respectively. The cumulative baseline subdistribution hazard function was estimated using the recommended formula in Fine and Gray (1999) [32]. Then the performance of the predictions was compared using Area Under the ROC Curve (AUC) and the expected Brier score (BS) proposed by Blanche et al. (2015) [9], implemented in the riskRegression package [39]. The results are summarized in Table 3.6.

Table 3.6: AUC and BS for each method at 3 and 5 years on the testing data

| Method | AUC (3 years) | BS (3 years) | AUC (5 years) | BS (5 years) |
|---|---|---|---|---|
| fastcmprsk_MCP | 0.5 | 0.244 | 0.5 | 0.241 |
| PPSH_SICA | 0.5 | 0.244 | 0.5 | 0.241 |
| PPSH_MCP_CV | 0.577 | 0.257 | 0.593 | 0.255 |
| PPSH_MCP_SGCV | 0.577 | 0.269 | 0.593 | 0.26 |
| PPSH_SCAD_CV | 0.645 | 0.237 | 0.653 | 0.233 |
| PPSH_SCAD_SGCV | 0.645 | 0.237 | 0.653 | 0.233 |
| CoxBoost_CV | 0.653 | 0.235 | 0.667 | 0.23 |
| CoxBoost_SGCV | 0.671 | 0.267 | 0.675 | 0.266 |

The prediction is more accurate if the AUC score is higher and the BS score is lower. The CoxBoost package under the SGCV score gave the highest AUC scores of 0.671 and 0.675 for 3 and 5 years, respectively, while the fastcmprsk package using MCP and our proposed method using SICA, which did not use any predictor thus predicted the same probability for

each patient, gave the lowest AUC score of 0.5. The CoxBoost package under the CV score gave the lowest BS scores of 0.235 and 0.23 for 3 and 5 years, respectively. Our proposed method using MCP under the SGCV score gave the highest BS score of 0.269 at 3 years and the CoxBoost package under the SGCV score gave the highest BS score of 0.266 at 5 years. Note that the null model, which used the Aalen-Johansen estimator [1] for all patients, gave BS scores of 0.243 and 0.24 for 3 and 5 years, respectively.

## 3.3 Discussion

SCAD, MCP and SICA all require the specification of a tuning parameter other than $\lambda$. How to properly choose the values of these tuning parameters deserves further research. The result is influenced by the choice of initial value for the parameter vector $\boldsymbol{\beta}$, $\boldsymbol{\beta}_0$. Setting $\boldsymbol{\beta}_0$ to be a vector other than $\mathbf{0}$ might select more predictors. The choice of the optimal value of $\lambda$ highly depends on the folds used in the cross-validation. To get more convincing results, multiple runs of the algorithm using different folds might be necessary. When $\lambda$ gets small, $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\eta}}$ could have very wide ranges, causing trouble to the coordinate ascent algorithm. How to deal with very large magnitudes of $\tilde{\boldsymbol{\beta}}$ might be worth more consideration.

# Chapter 4: Variable Selection for High-dimensional Competing Risk Data With Missing Data

When there are missing values in the data and multiple complete datasets are obtained from imputation, the same method may not select the same set of predictors when applied to each complete dataset. However, it is usually desirable to select a single set of predictors from all the imputed datasets.

Let $m$ represent the number of imputed datasets. Let $l_d(\boldsymbol{\beta})$ be the log-partial likelihood function defined on the $d$-th imputed dataset. We propose to maximize $\sum_{d=1}^{m} l_d(\boldsymbol{\beta}) - mn\sum_{q=1}^{p} p(\beta_q)$ to select predictors. For the $d$-th imputed dataset $\mathbf{Z}_d$, define $\tilde{\boldsymbol{\eta}}_d = \mathbf{Z}_d\tilde{\boldsymbol{\beta}}$. $l_d(\boldsymbol{\beta})$ can be approximated by $\frac{1}{2}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}]^2 + C_d(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\eta}}_d)$, where $C_d(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\eta}}_d)$ is a function of $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\eta}}_d$. So maximizing $\sum_{d=1}^{m} l_d(\boldsymbol{\beta}) - mn\sum_{q=1}^{p} p(\beta_q)$ is approximately equivalent to maximizing $f_m(\boldsymbol{\beta}) := \frac{1}{2}\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}]^2 - mn\sum_{q=1}^{p} p(\beta_q)$.

Similarly, the coordinate ascent algorithm will be used to maximize $f_m(\boldsymbol{\beta})$. We take the partial derivative of $f_m(\boldsymbol{\beta})$ with respect to $\beta_q$

$$\frac{\partial f_m(\boldsymbol{\beta})}{\partial \beta_q} = -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] - mnp'(\beta_q)$$

For SCAD

$$\frac{\partial f_m(\boldsymbol{\beta})}{\partial \beta_q} = \begin{cases} -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}], & \beta_q < -a\lambda \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] + \frac{mn(a\lambda+\beta_q)}{a-1}, & -a\lambda \le \beta_q < -\lambda \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] + mn\lambda, & -\lambda \le \beta_q < 0 \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] - mn\lambda, & 0 < \beta_q \le \lambda \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] - \frac{mn(a\lambda-\beta_q)}{a-1}, & \lambda < \beta_q \le a\lambda \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}], & \beta_q > a\lambda \end{cases}$$

Set $\frac{\partial f_m(\boldsymbol{\beta})}{\partial \beta_q} = 0$ and solve for $\beta_q$ while fixing $\beta_k,\ k \neq q$ gives

$$\hat{\beta}_q = \begin{cases} \frac{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]}{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2}, & \beta_q < -a\lambda \\ \frac{(a-1)\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mna\lambda}{(a-1)\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2 + mn}, & -a\lambda \le \beta_q < -\lambda \\ \frac{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn\lambda}{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2}, & -\lambda \le \beta_q < 0 \\ \frac{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + mn\lambda}{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{dij}^2}, & 0 < \beta_q \le \lambda \\ \frac{(a-1)\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + mna\lambda}{(a-1)\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2 + mn}, & \lambda < \beta_q \le a\lambda \\ \frac{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]}{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2}, & \beta_j > a\lambda \end{cases}$$

If there is at least one local maximum point in $(-\infty, 0) \cup (0, \infty)$, the second-order partial derivative is checked

$$\frac{\partial^2 f_m(\boldsymbol{\beta})}{\partial \beta_q^2} = \begin{cases} \sum_{d=1}^{m} \sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d), & \beta_q < -a\lambda \\ \sum_{d=1}^{m} \sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d) + \frac{mn}{a-1}, & -a\lambda \le \beta_q < -\lambda \\ \sum_{d=1}^{m} \sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d), & -\lambda \le \beta_q < 0 \\ \sum_{d=1}^{m} \sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d), & 0 < \beta_q \le \lambda \\ \sum_{d=1}^{m} \sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d) + \frac{mn}{a-1}, & \lambda < \beta_q \le a\lambda \\ \sum_{d=1}^{m} \sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d), & \beta_q > a\lambda \end{cases}$$

If $\frac{\partial^2 f_m(\boldsymbol{\beta})}{\partial \beta_q^2}|_{\hat{\beta}_q} < 0$, then $\hat{\beta}_q$ is a local maximum point.

Define $f_{mq}(\beta_q) = f_m(\boldsymbol{\beta})$ given fixed $\beta_k,\ k \neq q$. $f_{mq}(\beta_q)$ is not differentiable at 0 but is left and right differentiable at 0. If its left derivative at 0 $\partial_- f_{mq}(0) > 0$, and its right derivative at 0 $\partial_+ f_{mq}(0) < 0$, then 0 is a local maximum point.

$$\begin{aligned} \partial_- f_{mq}(0) = & \lim_{\beta_q \to 0^-} \frac{1}{\beta_q} (\{\frac{1}{2} \sum_{d=1}^{m} \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k \neq q} z_{dik}\beta_k - z_{diq}\beta_q]^2 - mn \sum_{k=1}^{p} p(\beta_k)\} \\ & - \{\frac{1}{2} \sum_{d=1}^{m} \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k \neq q} z_{dik}\beta_k]^2 - mn \sum_{k \neq q} p(\beta_k) - mnp(0)\}) \\ = & \lim_{\beta_q \to 0^-} \frac{1}{\beta_q} [(\frac{1}{2} \sum_{d=1}^{m} \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d)\{[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k \neq q} z_{dik}\beta_k]^2 - 2z_{diq}\beta_q[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k \neq q} z_{dik}\beta_k] \\ & + (z_{diq}\beta_q)^2\} - mnp(\beta_q)) - \{\frac{1}{2} \sum_{d=1}^{m} \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k \neq q} z_{dik}\beta_k]^2\}] \\ = & \lim_{\beta_q \to 0^-} \frac{1}{2} \sum_{d=1}^{m} \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d)\{-2z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k \neq q} z_{dik}\beta_k] + z_{diq}^2\beta_q\} - \frac{mnp(\beta_q)}{\beta_q} \\ = & -\sum_{d=1}^{m} \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k \neq q} z_{dik}\beta_k] - mn \lim_{\beta_q \to 0^-} \frac{p(\beta_q)}{\beta_q}. \end{aligned}$$

Similarly, $\partial_+ f_{mq}(0) = -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn \lim_{\beta_q \to 0^+} \frac{p(\beta_q)}{\beta_q}$.

For SCAD,

$$\begin{aligned}\partial_- f_{mq}(0) &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn \lim_{\beta_q \to 0^-} \frac{-\lambda\beta_q}{\beta_q} \\ &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + mn\lambda\end{aligned}$$

$$\begin{aligned}\partial_+ f_{mq}(0) &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{l\neq q} z_{dil}\beta_l] - mn \lim_{\beta_q \to 0^+} \frac{\lambda\beta_q}{\beta_q} \\ &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{l\neq q} z_{dik}\beta_l] - mn\lambda\end{aligned}$$

So if $-\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + mn\lambda > 0$ and
$-\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn\lambda < 0$, or
$|\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]| < mn\lambda$, 0 is a local maximum point.

For MCP,

$$\frac{\partial f_m(\boldsymbol{\beta})}{\partial \beta_q} = \begin{cases} -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}], & \beta_q < -\gamma\lambda \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] + mn\lambda(1 + \frac{\beta_q}{\gamma\lambda}), & -\gamma\lambda \leq \beta_q < 0 \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] - mn\lambda(1 - \frac{\beta_q}{\gamma\lambda}), & 0 < \beta_q \leq \gamma\lambda \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}], & \beta_q > \gamma\lambda \end{cases}$$

$$\hat{\beta}_q = \begin{cases} \frac{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]}{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2}, & \beta_q < -\gamma\lambda \\ \frac{\gamma\{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn\lambda\}}{\gamma\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2 + mn} & -\gamma\lambda \leq \beta_q < 0 \\ \frac{\gamma\{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + mn\lambda\}}{\gamma\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2 + mn}, & 0 < \beta_q \leq \gamma\lambda \\ \frac{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]}{\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2}, & \beta_q > \gamma\lambda \end{cases}$$

If there is at least one local maximum point in $(-\infty, 0) \cup (0, \infty)$, check the second-order partial derivative

$$\frac{\partial^2 f_m(\boldsymbol{\beta})}{\partial \beta_q^2} = \begin{cases} \sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d), & \beta_q < -\gamma\lambda \\ \sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d) + \frac{mn}{\gamma}, & -\gamma\lambda \leq \beta_q < 0 \\ \sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d) + \frac{mn}{\gamma}, & 0 < \beta_q < \gamma\lambda \\ \sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d), & \beta_q \geq \gamma\lambda \end{cases}$$

If $\frac{\partial^2 f_m(\boldsymbol{\beta})}{\partial \beta_q^2}|_{\hat{\beta}_q} < 0$, then $\hat{\beta}_q$ is a local maximum point.

$$\begin{aligned} \partial_- f_{mq}(0) &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn \lim_{\beta_q \to 0^-} \frac{-\lambda(\beta_q + \frac{\beta_q^2}{2\gamma\lambda})}{\beta_q} \\ &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + mn\lambda \end{aligned}$$

$$\begin{aligned} \partial_+ f_{mq}(0) &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn \lim_{\beta_q \to 0^+} \frac{\lambda(\beta_q - \frac{\beta_q^2}{2\gamma\lambda})}{\beta_q} \\ &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn\lambda \end{aligned}$$

So if $|\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]| < mn\lambda$, 0 is a local maximum point.

For SICA,

$$\frac{\partial f_m(\boldsymbol{\beta})}{\partial \beta_q} = \begin{cases} -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] + \frac{mn\lambda a(a+1)}{(a-\beta_q)^2}, & \beta_q < 0 \\ -\sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq} h_i(\tilde{\boldsymbol{\eta}}_d)[y_i(\tilde{\boldsymbol{\eta}}_d) - \mathbf{z}_{di}^T\boldsymbol{\beta}] - \frac{mn\lambda a(a+1)}{(a+\beta_q)^2}, & \beta_q > 0 \end{cases}$$

$\hat{\beta}_q$ is the solution to

$$\begin{cases} [\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2]\beta_q^3 - \{2a\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2 + \\ \sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]\}\beta_q^2 + \\ \{a^2\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2 + 2a\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]\}\beta_q - & \beta_q < 0 \\ a^2\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + mn\lambda a(a+1) = 0, \\ [\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2]\beta_q^3 + \{2a\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2 - \\ \sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]\}\beta_q^2 & \beta_q > 0 \\ +\{a^2\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}^2 - 2a\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]\}\beta_q \\ -\{a^2\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + mn\lambda a(a+1)\} = 0. \end{cases}$$

If there is at least one local maximum point in $(-\infty, 0) \cup (0, \infty)$, check the second-order partial derivative

$$\frac{\partial^2 f_m(\boldsymbol{\beta})}{\partial \beta_q^2} = \begin{cases} \sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d) + \frac{2mn\lambda a(a+1)}{(a-\beta)^3}, & \beta_q < 0 \\ \sum_{d=1}^{m}\sum_{i=1}^{n} z_{diq}^2 h_i(\tilde{\boldsymbol{\eta}}_d) + \frac{2mn\lambda a(a+1)}{(a+\beta)^3}, & \beta_q > 0 \end{cases}$$

If $\frac{\partial^2 f_m(\boldsymbol{\beta})}{\partial \beta_q^2}|_{\hat{\beta}_q} < 0$, then $\hat{\beta}_q$ is a local maximum point.

$$\begin{aligned}\partial_- f_{mq}(0) &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn \lim_{\beta_q \to 0^-} \frac{-\frac{\lambda(a+1)\beta_q}{a-\beta_q}}{\beta_q} \\ &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + \frac{mn\lambda(a+1)}{a}\end{aligned}$$

$$\begin{aligned}\partial_+ f_{mq}(0) &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - mn \lim_{\beta_q \to 0^-} \frac{\frac{\lambda(a+1)\beta_q}{a+\beta_q}}{\beta_q} \\ &= -\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - \frac{mn\lambda(a+1)}{a}\end{aligned}$$

So if $-\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] + \frac{mn\lambda(a+1)}{a} > 0$ and
$-\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k] - \frac{mn\lambda(a+1)}{a} < 0$ or
$|\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\tilde{\boldsymbol{\eta}}_d) z_{diq}[y_i(\tilde{\boldsymbol{\eta}}_d) - \sum_{k\neq q} z_{dik}\beta_k]| < \frac{mn\lambda(a+1)}{a}$, 0 is a local maximum point.

# Chapter 5: Simulation Study and Application to the AML Dataset

## 5.1 Simulation Study

### 5.1.1 Setup

Simulations were conducted following the same setting in Chapter 3. Missing values were generated in the design matrix with probability of missing set to 1% and 10%, respectively. 10 imputed datasets were generated for each dataset using the MissForest package. For each imputed dataset, each covariate was standardized to have mean 0 and variance 1. The maximum number of iterations in the algorithm was set to 100 due to time constraints. 100 replications were used in each scenario. Again, a value that would make the algorithm select none of the predictors given $\boldsymbol{\beta}_0 = \mathbf{0}$ was used as the largest value of $\lambda$. That means 0 is the global maximum point of $f_{mq}(\beta_q)$ for any $q = 1, ..., p$ given $\beta_k = 0,\ k \neq q$. $\boldsymbol{\eta}_{d,0} = \mathbf{Z}_d\boldsymbol{\beta}_0 = \mathbf{0}$. For SCAD and MCP, that means $|\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\mathbf{0})z_{diq}y_i(\mathbf{0})| < mn\lambda$. So $\frac{|\sum_{d=1}^{m}\sum_{i=1}^{n} h_i(\mathbf{0})z_{diq}y_i(\mathbf{0})|}{mn}$ was used as the largest value. For SICA, the same values of $\lambda$ were used as in Chapter 3.

### 5.1.2 Results

The simulation results are summaried in the following Tables 5.1-5.27. The results from Chapter 3 without missing values are also included for comparison. Boxplots are also provided for TP and FP.

Table 5.1: Mean of TP and FP using SCAD over 100 replications when $n = 200$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 8.18 | 4.15 | 8.53 | 5.52 |
| 1% | 8.28 | 4.11 | 8.34 | 5.84 |
| 10% | 7.05 | 6.74 | 7.7 | 9.65 |

Table 5.2: Mean of TP and FP using SCAD over 100 replications when $n = 200$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 8.59 | 1.59 | 9.05 | 2.94 |
| 1% | 8.53 | 1.6 | 8.87 | 2.99 |
| 10% | 7.94 | 4.3 | 7.98 | 5.42 |

Table 5.3: Mean of TP and FP using SCAD over 100 replications when $n = 200$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 8.62 | 0.6 | 9.15 | 1.79 |
| 1% | 8.57 | 0.85 | 9.05 | 1.9 |
| 10% | 8.31 | 2.83 | 8.35 | 4.11 |

Table 5.4: Mean of TP and FP using SCAD over 100 replications when $n = 300$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 9.17 | 0.66 | 9.87 | 6.04 |
| 1% | 9.24 | 0.54 | 9.9 | 5.8 |
| 10% | 9.35 | 2.58 | 9.73 | 7.91 |

Table 5.5: Mean of TP and FP using SCAD over 100 replications when $n = 300$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 9.2 | 0.15 | 9.95 | 3.05 |
| 1% | 9.33 | 0.2 | 9.95 | 2.77 |
| 10% | 9.24 | 0.58 | 9.79 | 5.13 |

Table 5.6: Mean of TP and FP using SCAD over 100 replications when $n = 300$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 9.1 | 0.07 | 9.94 | 1.38 |
| 1% | 9.19 | 0.12 | 9.93 | 1.54 |
| 10% | 9.2 | 0.31 | 9.88 | 3.88 |

Table 5.7: Mean of TP and FP using SCAD over 100 replications when $n = 400$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 9.58 | 0.1 | 9.97 | 8.52 |
| 1% | 9.62 | 0.15 | 10 | 8.16 |
| 10% | 9.71 | 0.63 | 9.97 | 11.72 |

Table 5.8: Mean of TP and FP using SCAD over 100 replications when $n = 400$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 9.58 | 0.09 | 9.99 | 4.08 |
| 1% | 9.63 | 0.1 | 10 | 4.54 |
| 10% | 9.59 | 0.12 | 9.96 | 7.38 |

Table 5.9: Mean of TP and FP using SCAD over 100 replications when $n = 400$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 9.35 | 0.07 | 10 | 1.61 |
| 1% | 9.62 | 0.04 | 9.99 | 2.51 |
| 10% | 9.45 | 0.1 | 9.96 | 5.22 |

Table 5.10: Mean of TP and FP using MCP over 100 replications when $n = 200$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 6.17 | 1.09 | 4.34 | 0.54 |
| 1% | 6.08 | 0.86 | 4.74 | 0.5 |
| 10% | 4.92 | 1.02 | 3.27 | 0.71 |

Table 5.11: Mean of TP and FP using MCP over 100 replications when $n = 200$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 6.52 | 0.47 | 5.69 | 0.44 |
| 1% | 6.42 | 0.38 | 5.62 | 0.16 |
| 10% | 5.63 | 0.74 | 3.65 | 0.34 |

Table 5.12: Mean of TP and FP using MCP over 100 replications when $n = 200$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 6.28 | 0.3 | 5.89 | 0.14 |
| 1% | 6.26 | 0.23 | 5.7 | 0.19 |
| 10% | 5.93 | 0.52 | 3.63 | 0.25 |

Table 5.13: Mean of TP and FP using MCP over 100 replications when $n = 300$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 7.57 | 0.1 | 8.43 | 0.26 |
| 1% | 7.54 | 0.12 | 8.33 | 0.22 |
| 10% | 7.85 | 0.46 | 8.01 | 0.46 |

Table 5.14: Mean of TP and FP using MCP over 100 replications when $n = 300$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 7.02 | 0.05 | 8.18 | 0.09 |
| 1% | 7.17 | 0.02 | 7.83 | 0.06 |
| 10% | 7.31 | 0.07 | 7.63 | 0.14 |

Table 5.15: Mean of TP and FP using MCP over 100 replications when $n = 300$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 7.25 | 0.02 | 7.48 | 0.04 |
| 1% | 7.29 | 0.02 | 7.53 | 0.03 |
| 10% | 7.08 | 0.05 | 6.84 | 0.05 |

Table 5.16: Mean of TP and FP using MCP over 100 replications when $n = 400$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 7.96 | 0.04 | 9.79 | 0.19 |
| 1% | 7.9 | 0.02 | 9.67 | 0.13 |
| 10% | 8.84 | 0.09 | 9.54 | 0.28 |

Table 5.17: Mean of TP and FP using MCP over 100 replications when $n = 400$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 7.83 | 0.03 | 9.31 | 0.06 |
| 1% | 7.8 | 0.01 | 9.21 | 0.02 |
| 10% | 7.67 | 0.01 | 8.98 | 0.01 |

Table 5.18: Mean of TP and FP using MCP over 100 replications when $n = 400$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 7.72 | 0.01 | 8.58 | 0.01 |
| 1% | 7.57 | 0.01 | 8.51 | 0.01 |
| 10% | 7.54 | 0.02 | 8.31 | 0.03 |

Table 5.19: Mean of TP and FP using SICA over 100 replications when $n = 200$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 4.75 | 0.52 | 2.31 | 0.13 |
| 1% | 4.36 | 0.48 | 1.93 | 0.11 |
| 10% | 2.69 | 0.45 | 0.8 | 0.04 |

Table 5.20: Mean of TP and FP using SICA over 100 replications when $n = 200$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 5.44 | 0.46 | 4.33 | 0.25 |
| 1% | 5.3 | 0.41 | 3.74 | 0.2 |
| 10% | 4.16 | 0.48 | 1.59 | 0.12 |

Table 5.21: Mean of TP and FP using SICA over 100 replications when $n = 200$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 5.66 | 0.33 | 4.51 | 0.17 |
| 1% | 5.53 | 0.35 | 4.33 | 0.23 |
| 10% | 4.61 | 0.45 | 2.34 | 0.08 |

Table 5.22: Mean of TP and FP using SICA over 100 replications when $n = 300$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 6.82 | 0.14 | 5.01 | 0.11 |
| 1% | 7.17 | 0.2 | 5.68 | 0.18 |
| 10% | 6.68 | 0.35 | 5.61 | 0.21 |

Table 5.23: Mean of TP and FP using SICA over 100 replications when $n = 300$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 6.37 | 0.13 | 4.77 | 0.04 |
| 1% | 6.37 | 0.1 | 5.15 | 0.05 |
| 10% | 6.34 | 0.16 | 5.29 | 0.08 |

Table 5.24: Mean of TP and FP using SICA over 100 replications when $n = 300$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 5.55 | 0.03 | 4.39 | 0.05 |
| 1% | 6.37 | 0.1 | 5.15 | 0.05 |
| 10% | 5.71 | 0.05 | 4.47 | 0.03 |

Table 5.25: Mean of TP and FP using SICA over 100 replications when $n = 400$ and $|\beta|$=0.75 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 8.99 | 0.13 | 5.31 | 0.04 |
| 1% | 8.91 | 0.13 | 4.83 | 0.03 |
| 10% | 7.97 | 0.12 | 5.86 | 0.1 |

Table 5.26: Mean of TP and FP using SICA over 100 replications when $n = 400$ and $|\beta|$=1 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 8 | 0.09 | 4.76 | 0.01 |
| 1% | 7.84 | 0.1 | 4.67 | 0 |
| 10% | 7.64 | 0.15 | 5.12 | 0.02 |

Table 5.27: Mean of TP and FP using SICA over 100 replications when $n = 400$ and $|\beta|$=1.25 without and with missing data

| Missing probability | TP(CV) | FP(CV) | TP(SGCV) | FP(SGCV) |
|---|---|---|---|---|
| 0 | 6.75 | 0.02 | 4.52 | 0.01 |
| 1% | 6.6 | 0.02 | 4.5 | 0 |
| 10% | 7.01 | 0.06 | 5.14 | 0.02 |

Figure 5.1: Boxplot of TP using SCAD when $n = 200$ and $|\beta|$=0.75 without and with missing data

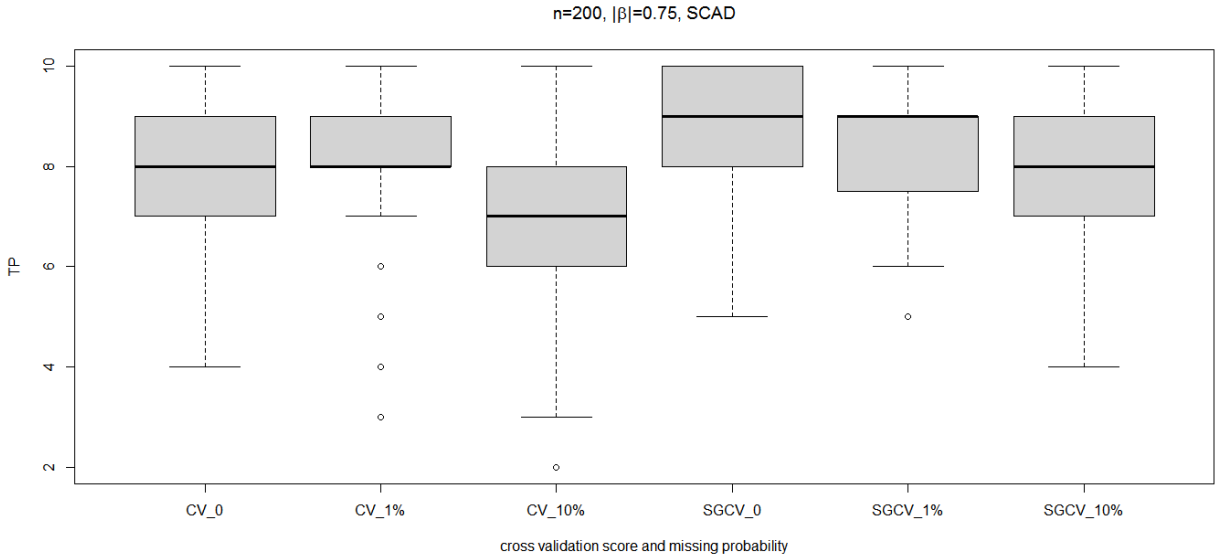



Figure 5.2: Boxplot of FP using SCAD when $n = 200$ and $|\beta|$=0.75 without and with missing data

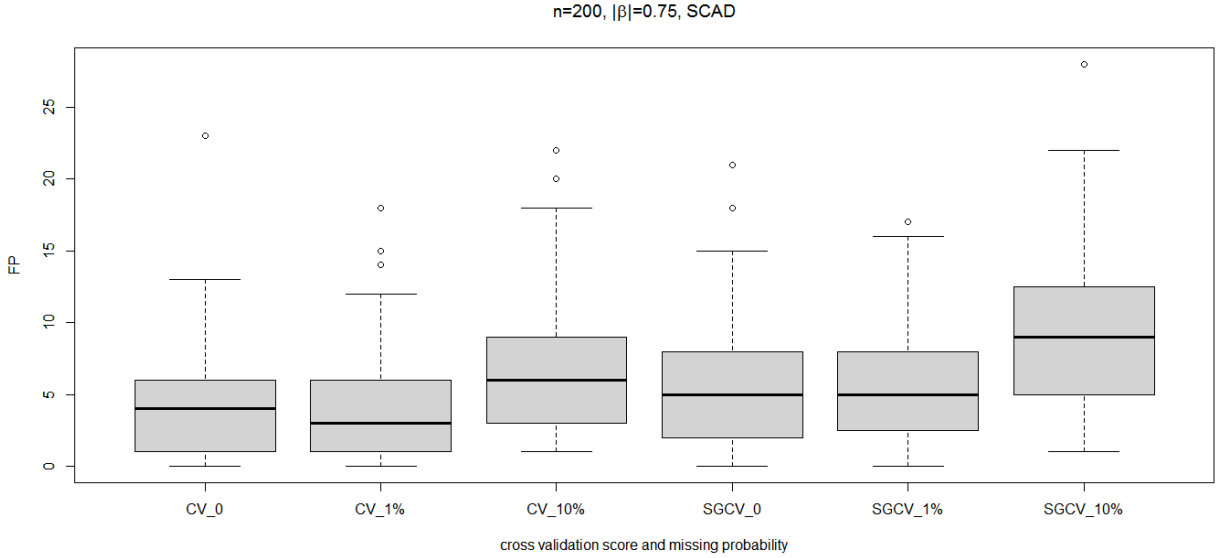


Figure 5.3: Boxplot of TP using SCAD when $n = 200$ and $|\beta|$=1 without and with missing data

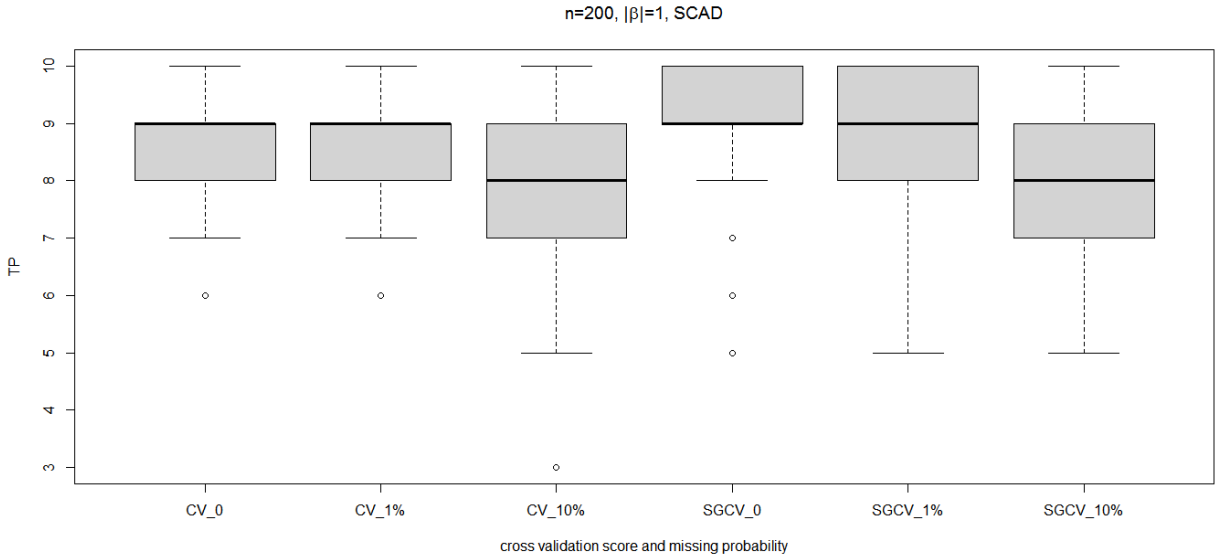



Figure 5.4: Boxplot of FP using SCAD when $n = 200$ and $|\beta|$=1 without and with missing data

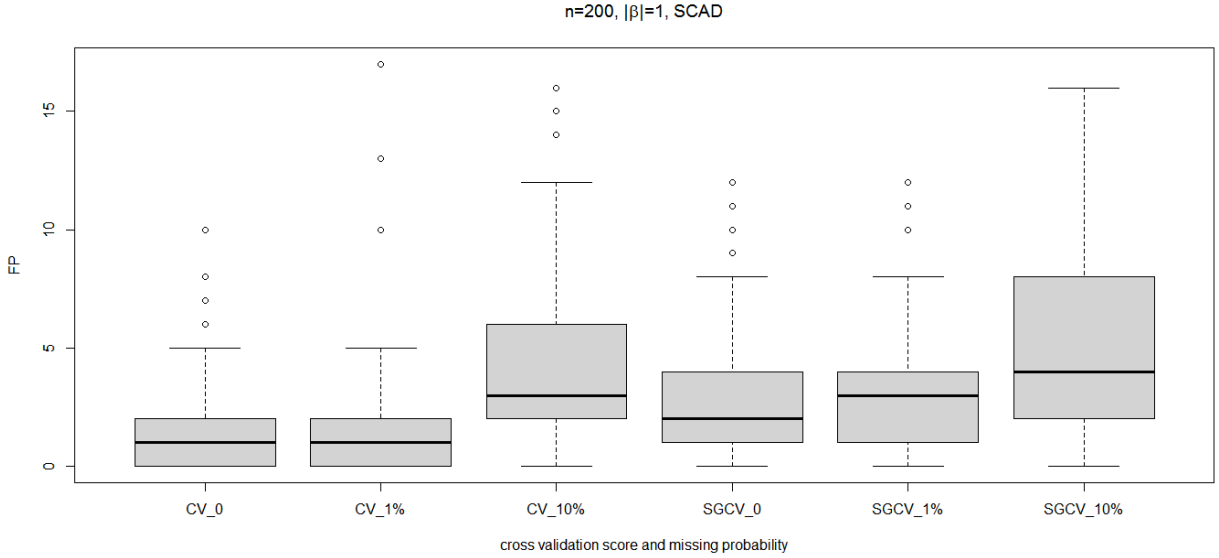


Figure 5.5: Boxplot of TP using SCAD when $n = 200$ and $|\beta|$=1.25 without and with missing data

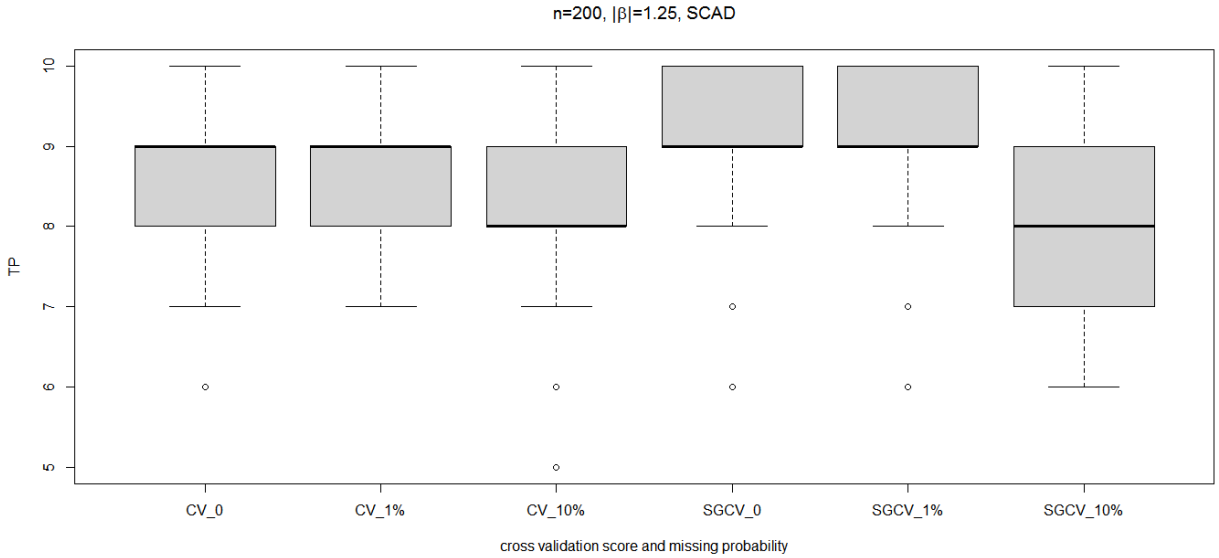



Figure 5.6: Boxplot of FP using SCAD when $n = 200$ and $|\beta|$=1.25 without and with missing data

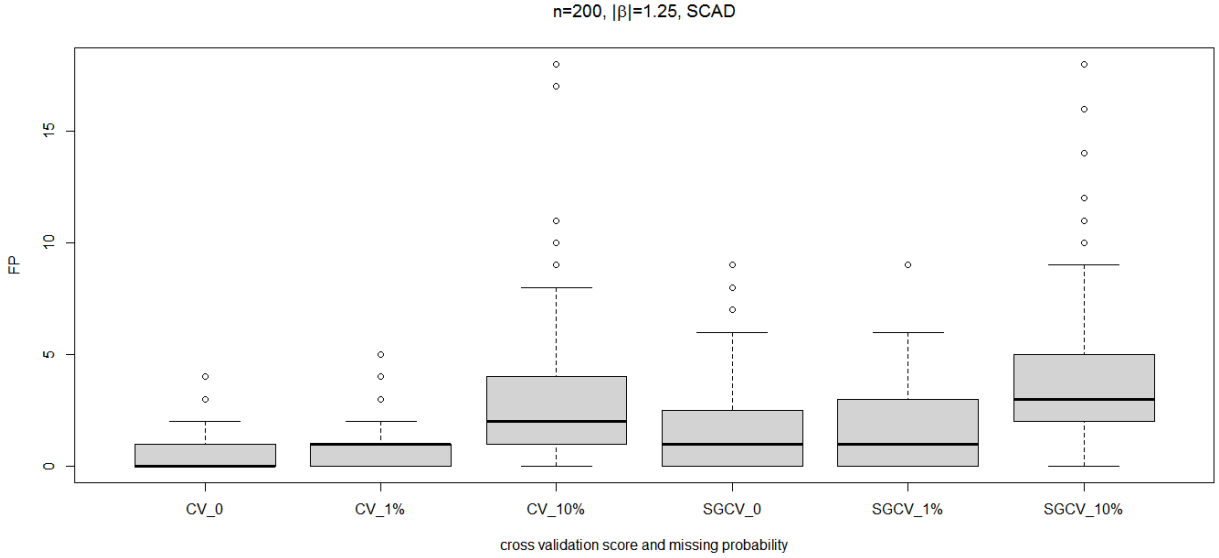


Figure 5.7: Boxplot of TP using SCAD when $n = 300$ and $|\beta|$=0.75 without and with missing data

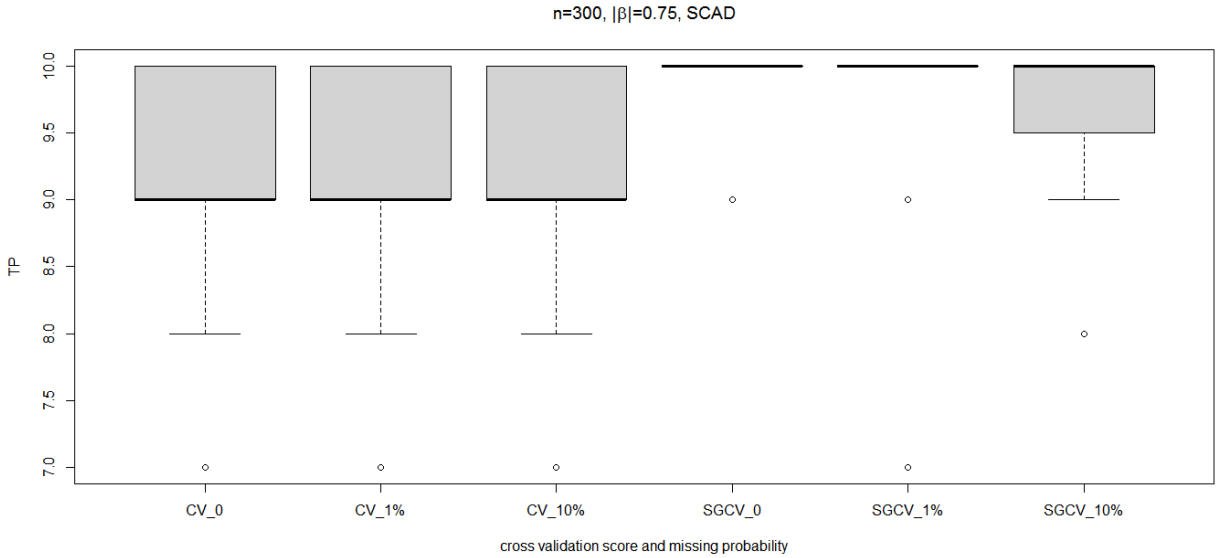



Figure 5.8: Boxplot of FP using SCAD when $n = 300$ and $|\beta|$=0.75 without and with missing data

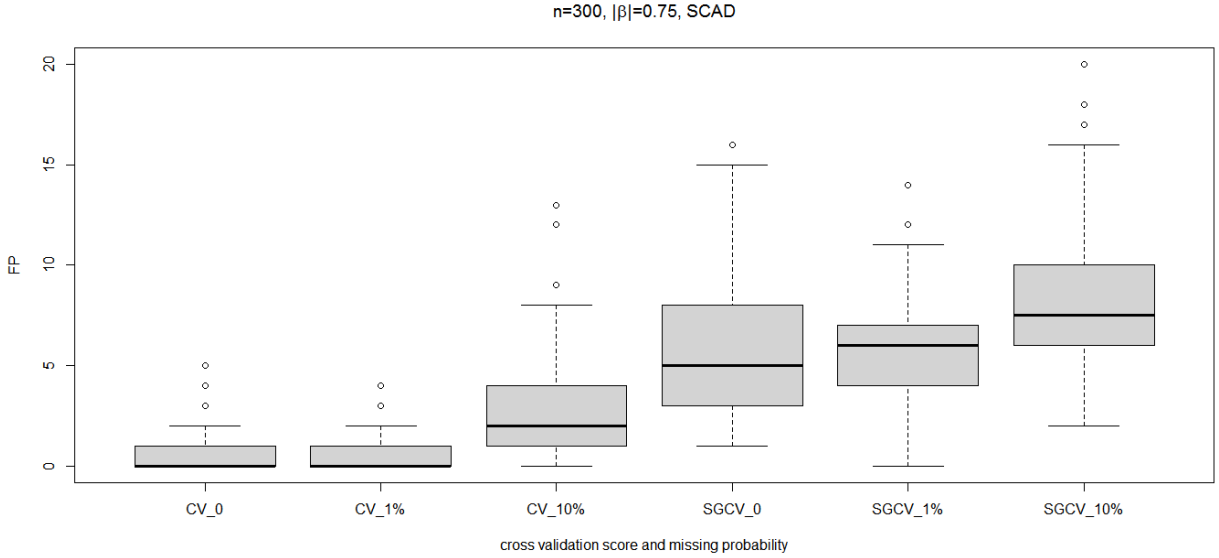


Figure 5.9: Boxplot of TP using SCAD when $n = 300$ and $|\beta|$=1 without and with missing data

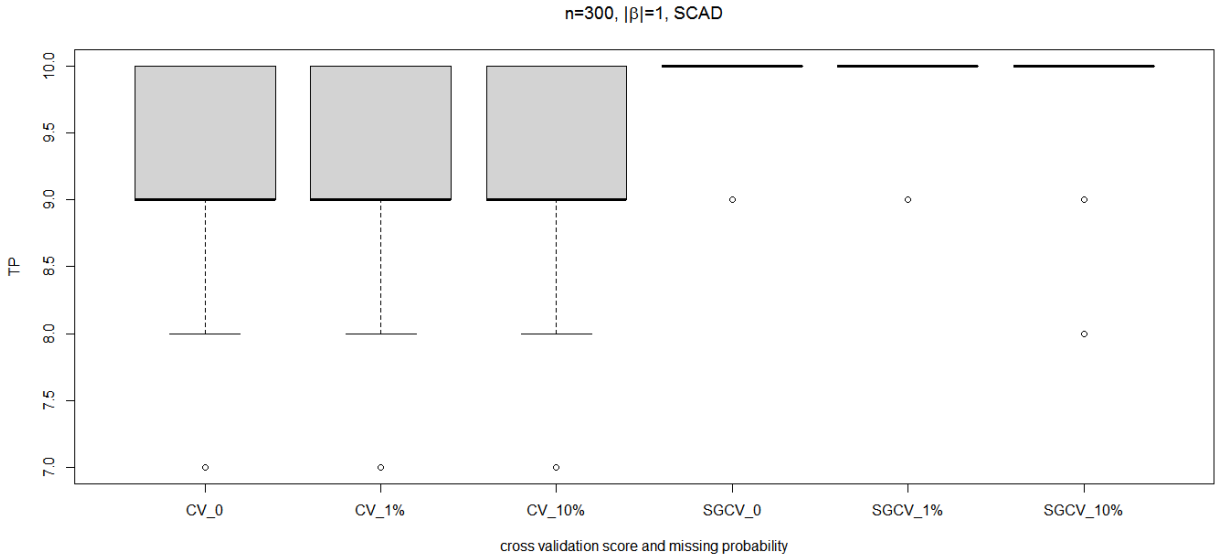



Figure 5.10: Boxplot of FP using SCAD when $n = 300$ and $|\beta|$=1 without and with missing data

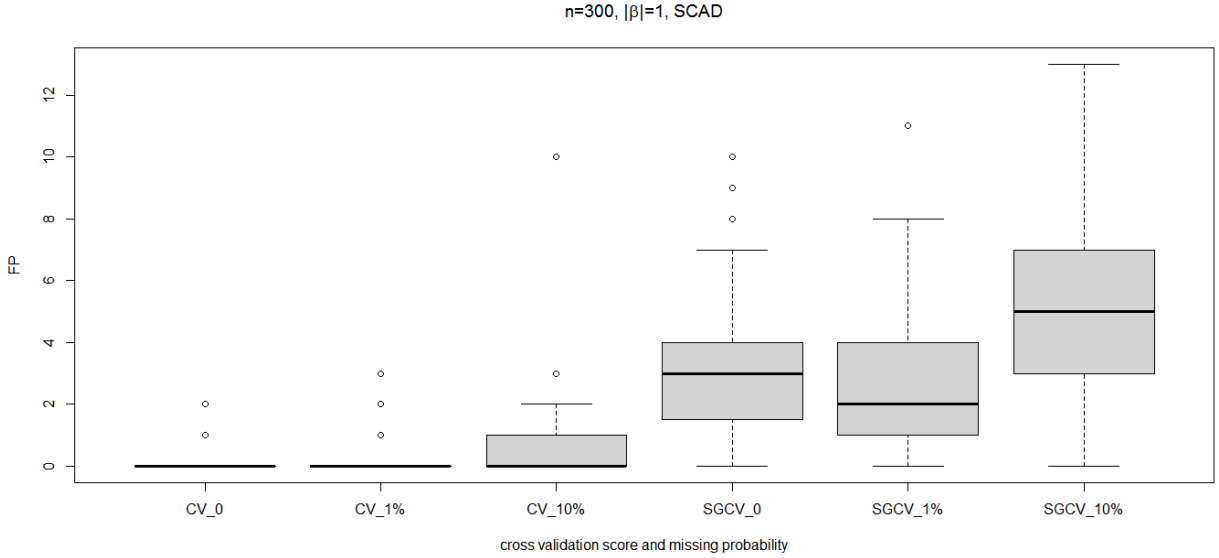


Figure 5.11: Boxplot of TP using SCAD when $n = 300$ and $|\beta|$=1.25 without and with missing data

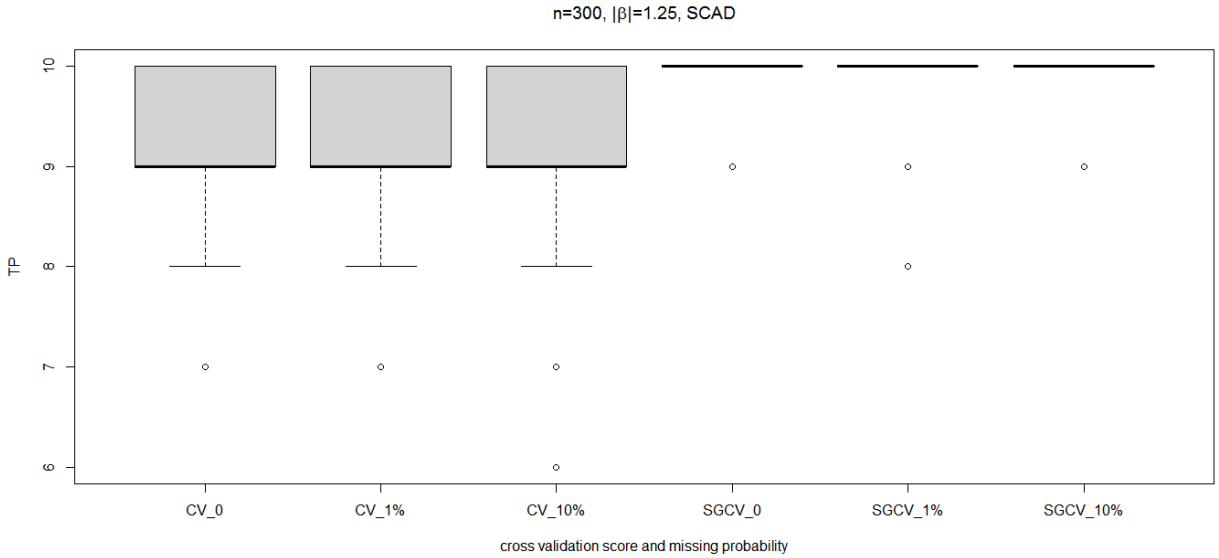



Figure 5.12: Boxplot of FP using SCAD when $n = 300$ and $|\beta|$=1.25 without and with missing data

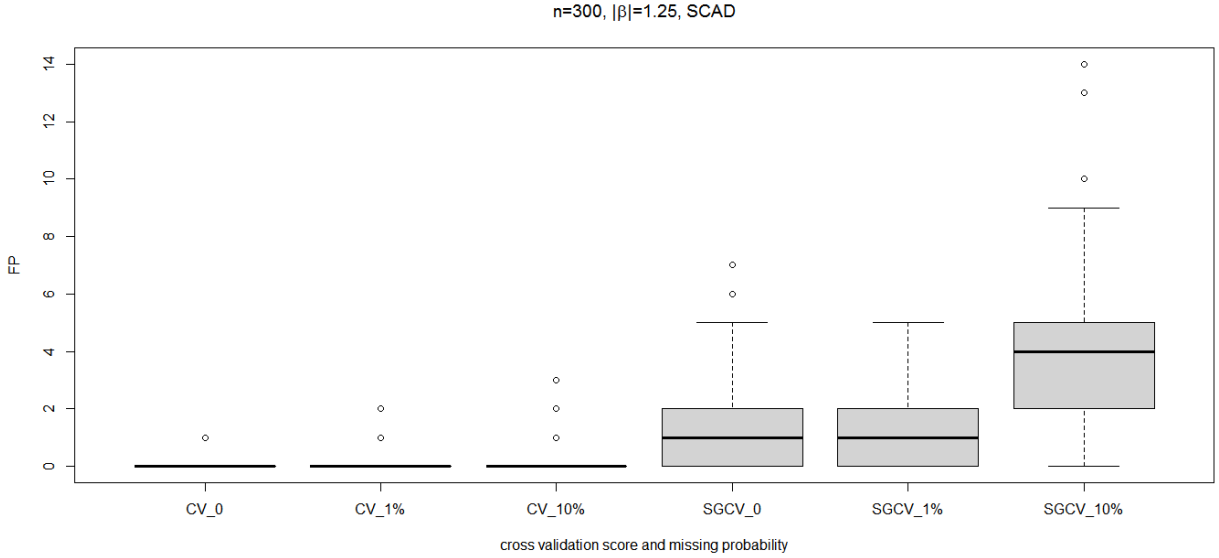


Figure 5.13: Boxplot of TP using SCAD when $n = 400$ and $|\beta|$=0.75 without and with missing data

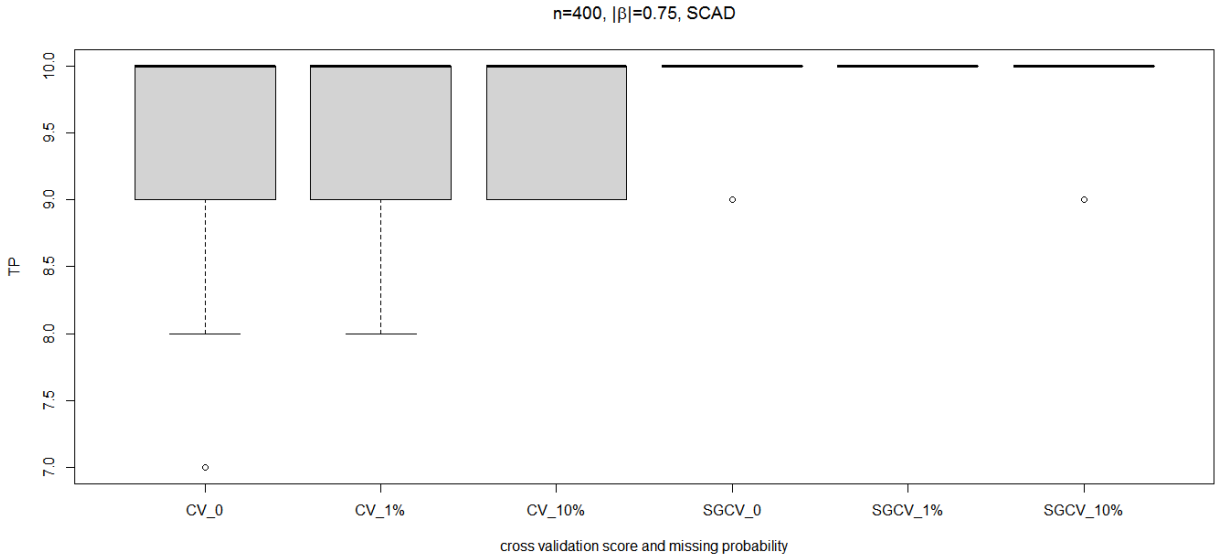



Figure 5.14: Boxplot of FP using SCAD when $n = 400$ and $|\beta|$=0.75 without and with missing data

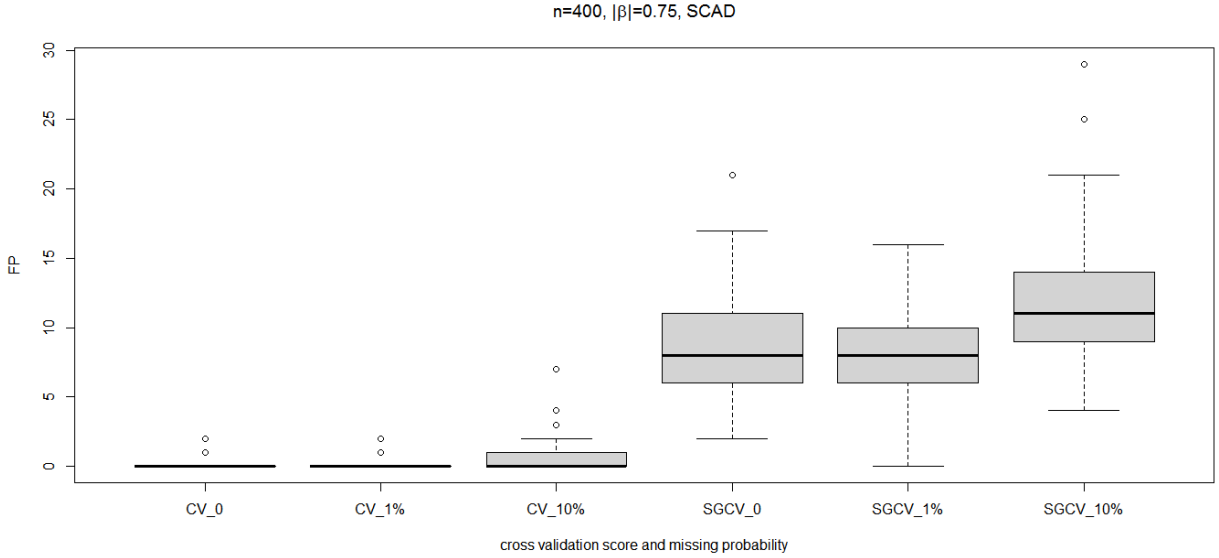


Figure 5.15: Boxplot of TP using SCAD when $n = 400$ and $|\beta|$=1 without and with missing data

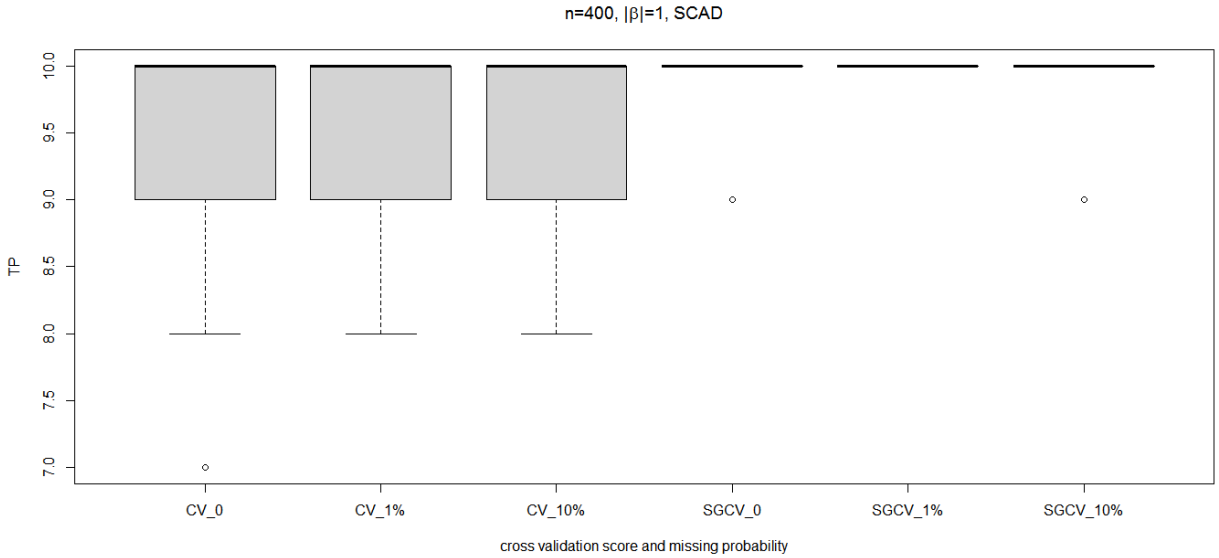



Figure 5.16: Boxplot of FP using SCAD when $n = 400$ and $|\beta|$=1 without and with missing data

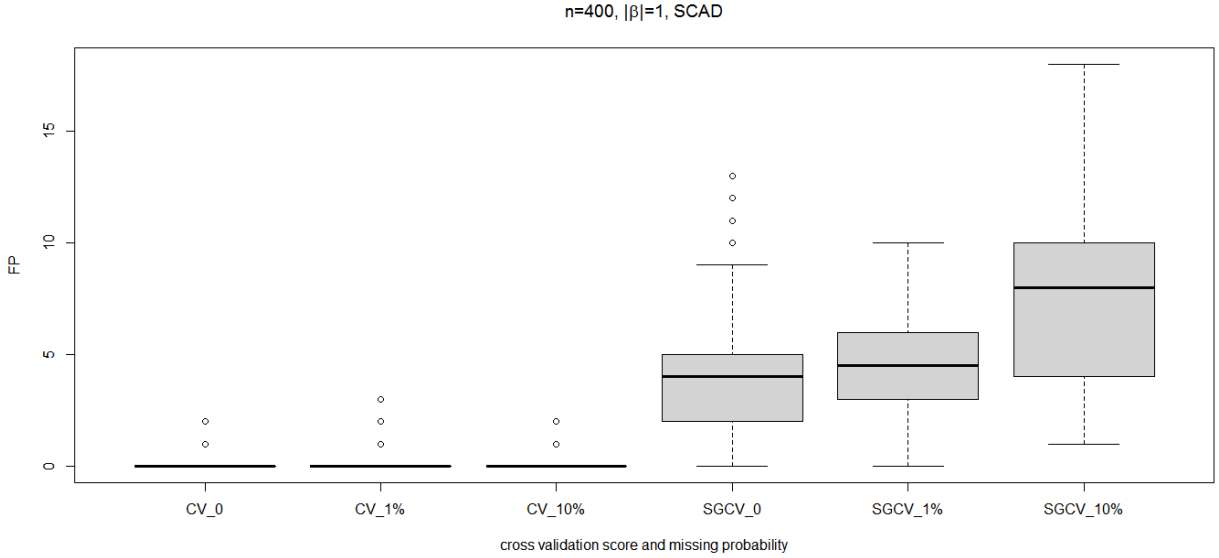


Figure 5.17: Boxplot of TP using SCAD when $n = 400$ and $|\beta|$=1.25 without and with missing data

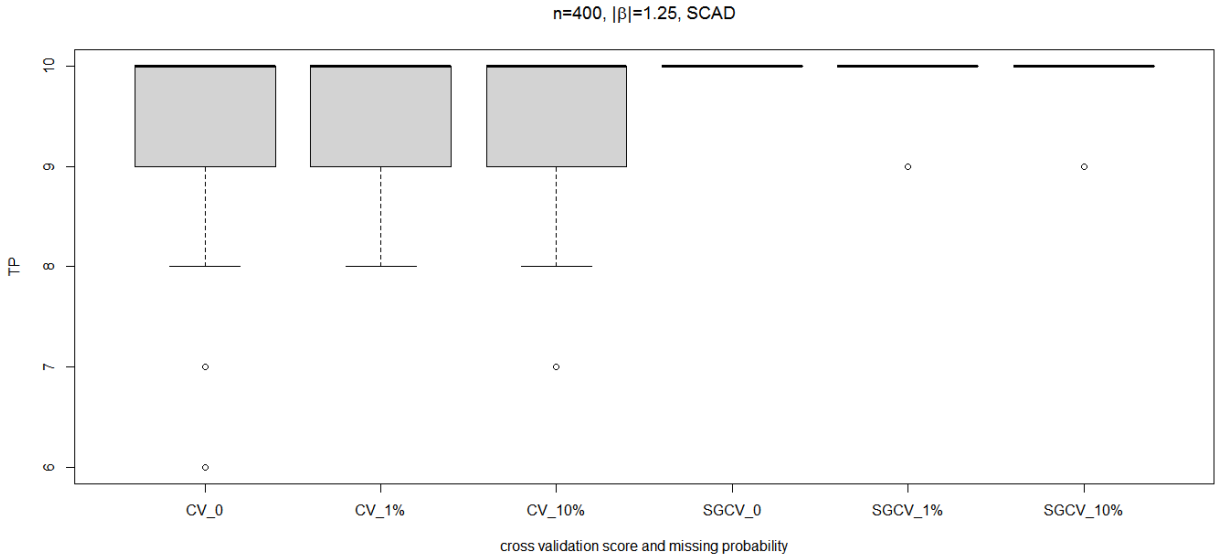



Figure 5.18: Boxplot of FP using SCAD when $n = 400$ and $|\beta|$=1.25 without and with missing data

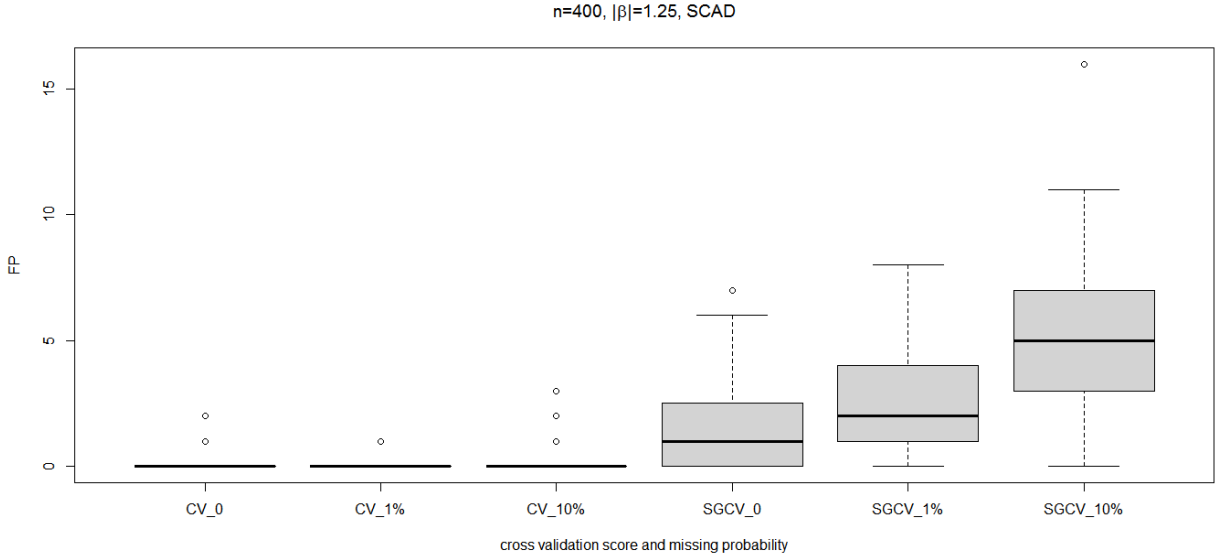


Figure 5.19: Boxplot of TP using MCP when $n = 200$ and $|\beta|$=0.75 without and with missing data

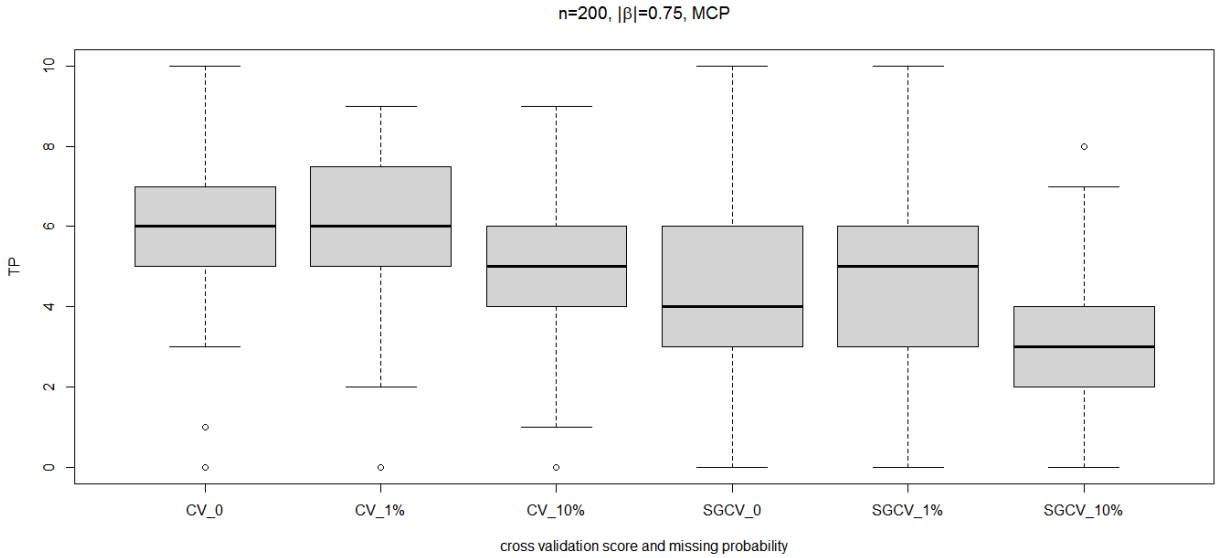



Figure 5.20: Boxplot of FP using MCP when $n = 200$ and $|\beta|$=0.75 without and with missing data

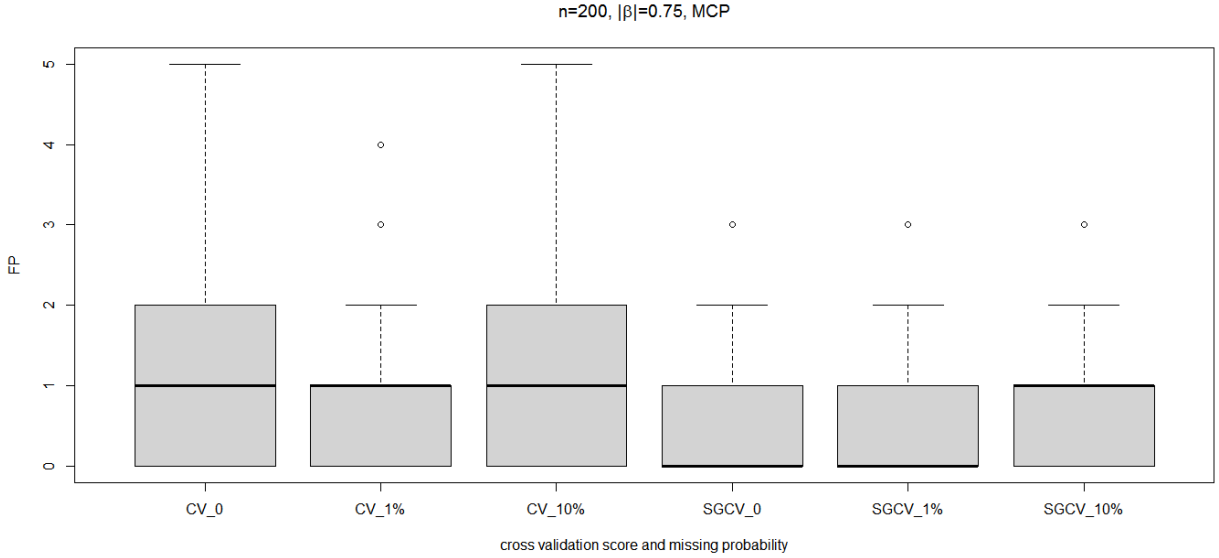


Figure 5.21: Boxplot of TP using MCP when $n = 200$ and $|\beta|$=1 without and with missing data

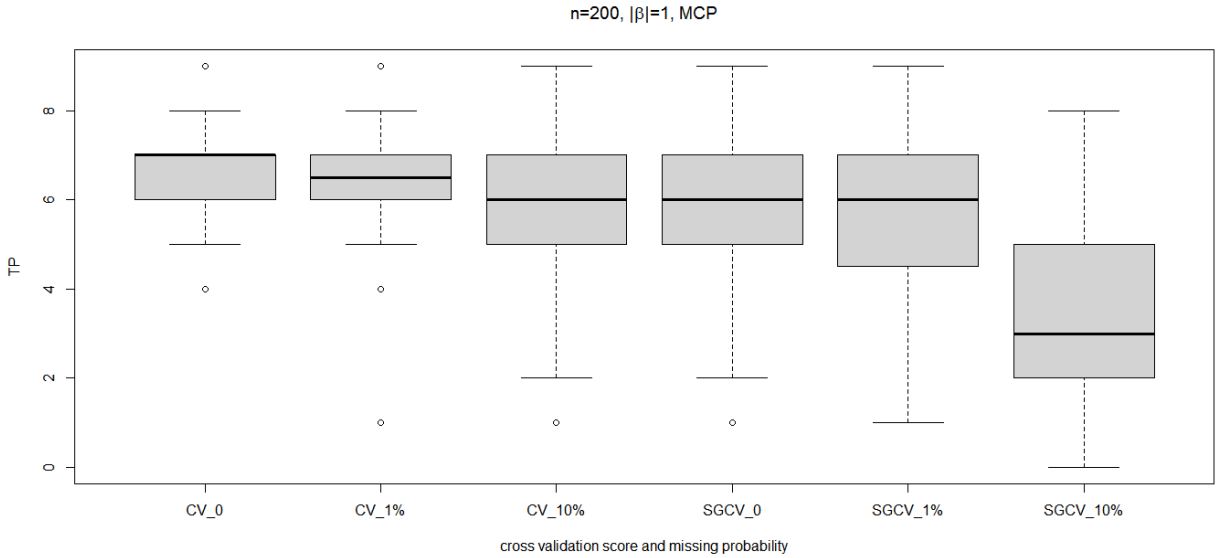



Figure 5.22: Boxplot of FP using MCP when $n = 200$ and $|\beta|$=1 without and with missing data

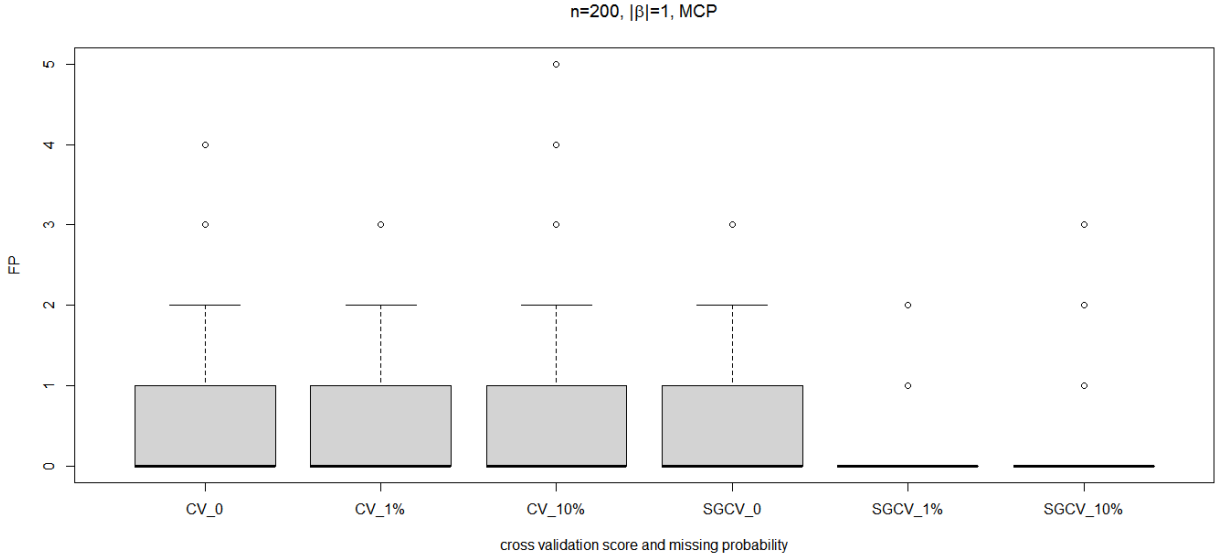


Figure 5.23: Boxplot of TP using MCP when $n = 200$ and $|\beta|$=1.25 without and with missing data

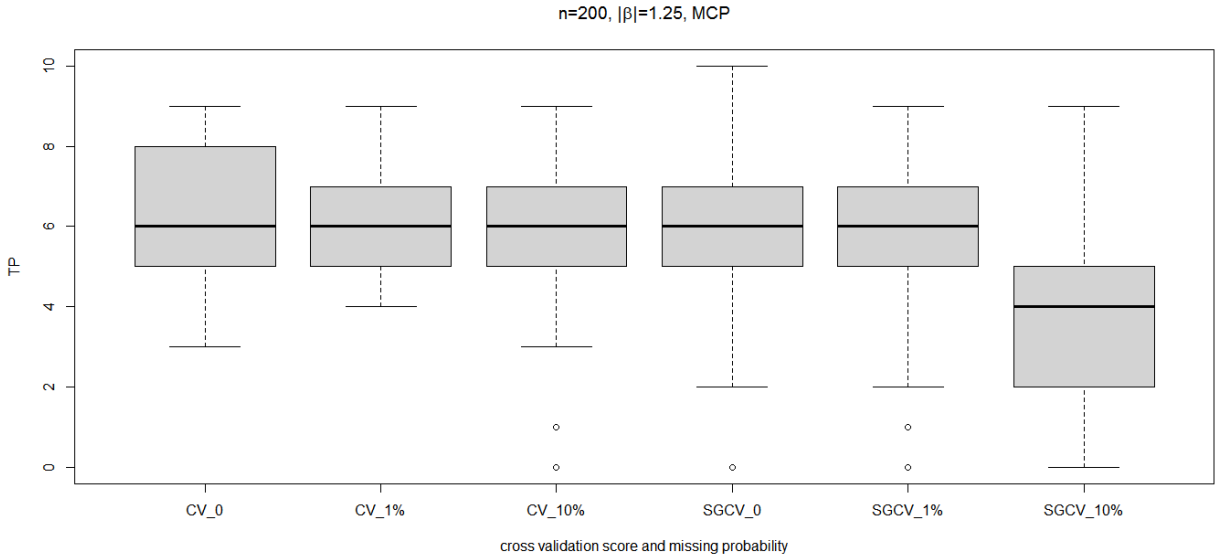



Figure 5.24: Boxplot of FP using MCP when $n = 200$ and $|\beta|$=1.25 without and with missing data

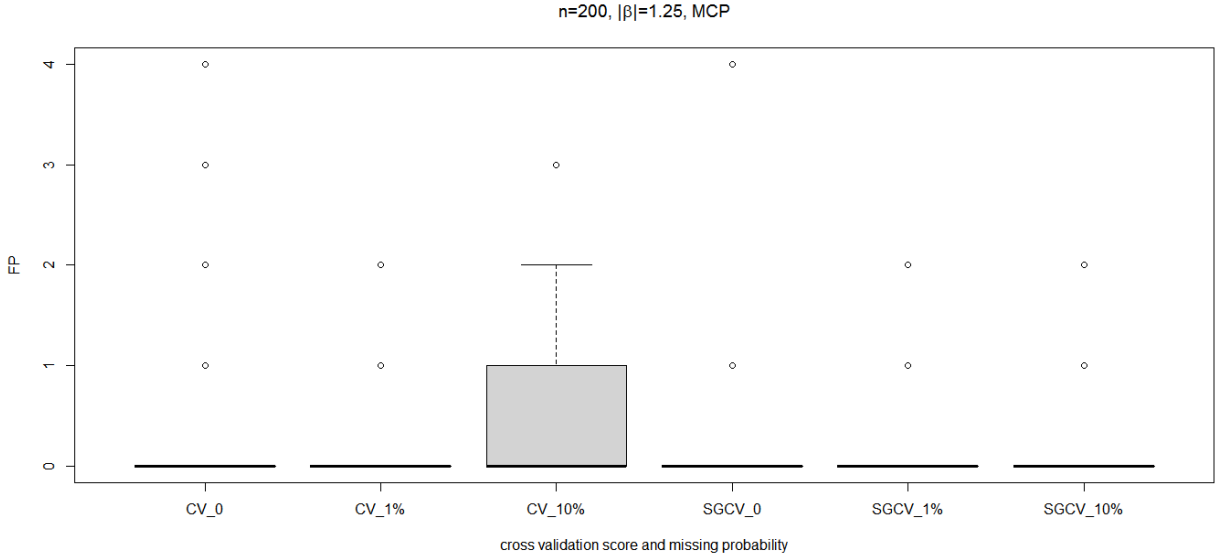


Figure 5.25: Boxplot of TP using MCP when $n = 300$ and $|\beta|$=0.75 without and with missing data

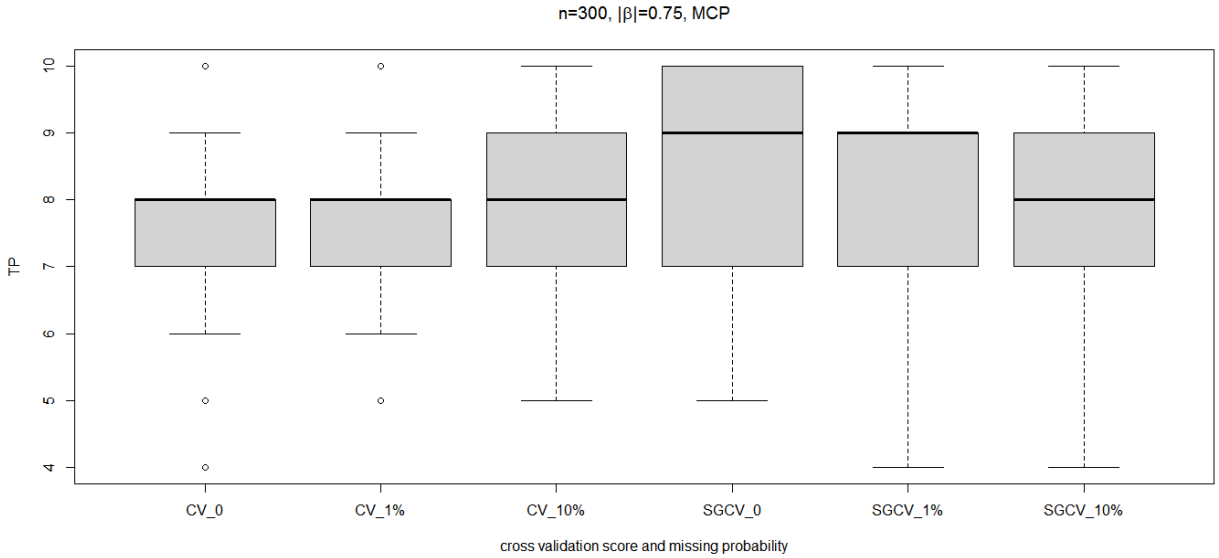



Figure 5.26: Boxplot of FP using MCP when $n = 300$ and $|\beta|$=0.75 without and with missing data

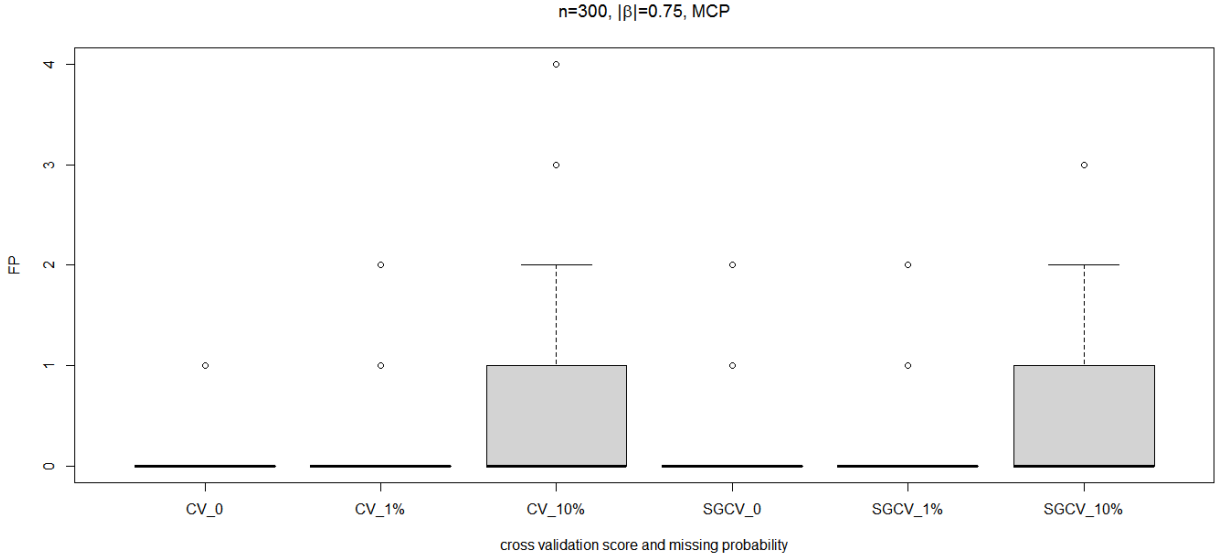


Figure 5.27: Boxplot of TP using MCP when $n = 300$ and $|\beta|$=1 without and with missing data

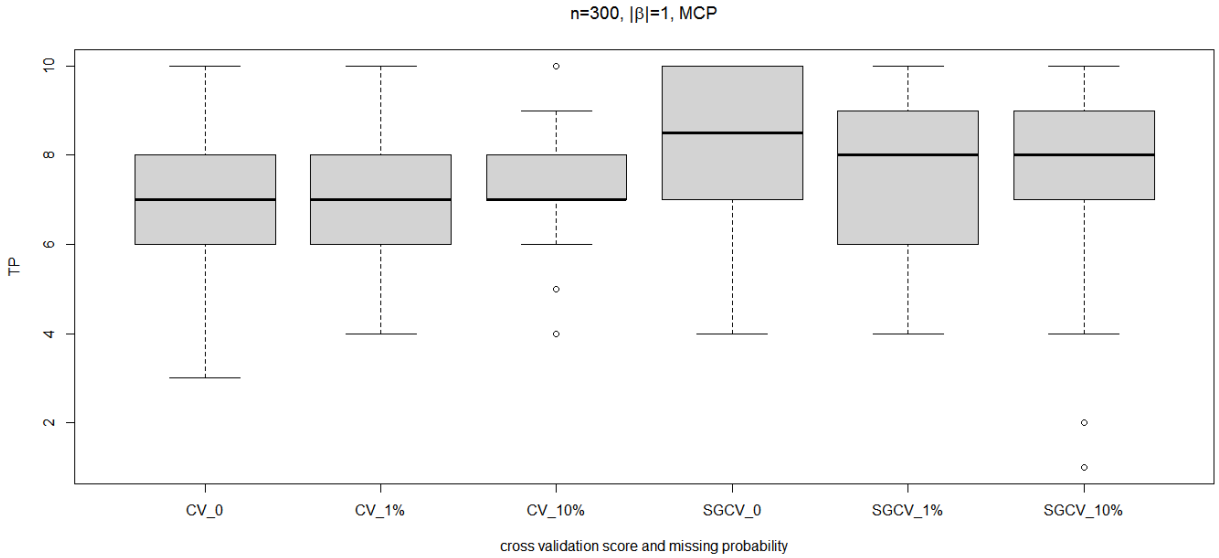



Figure 5.28: Boxplot of FP using MCP when $n = 300$ and $|\beta|$=1 without and with missing data

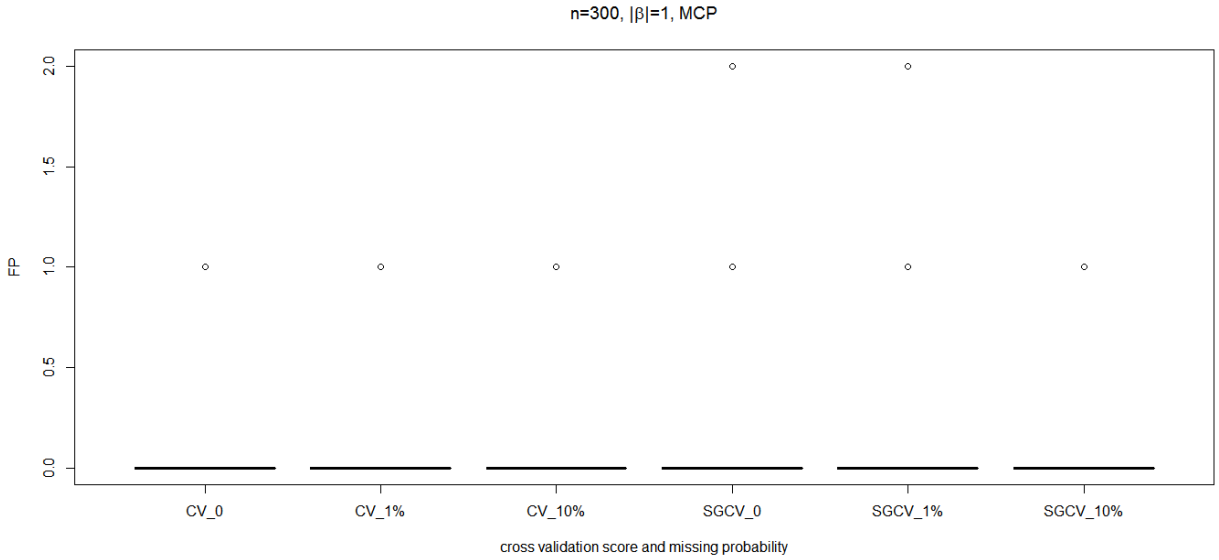


Figure 5.29: Boxplot of TP using MCP when $n = 300$ and $|\beta|$=1.25 without and with missing data

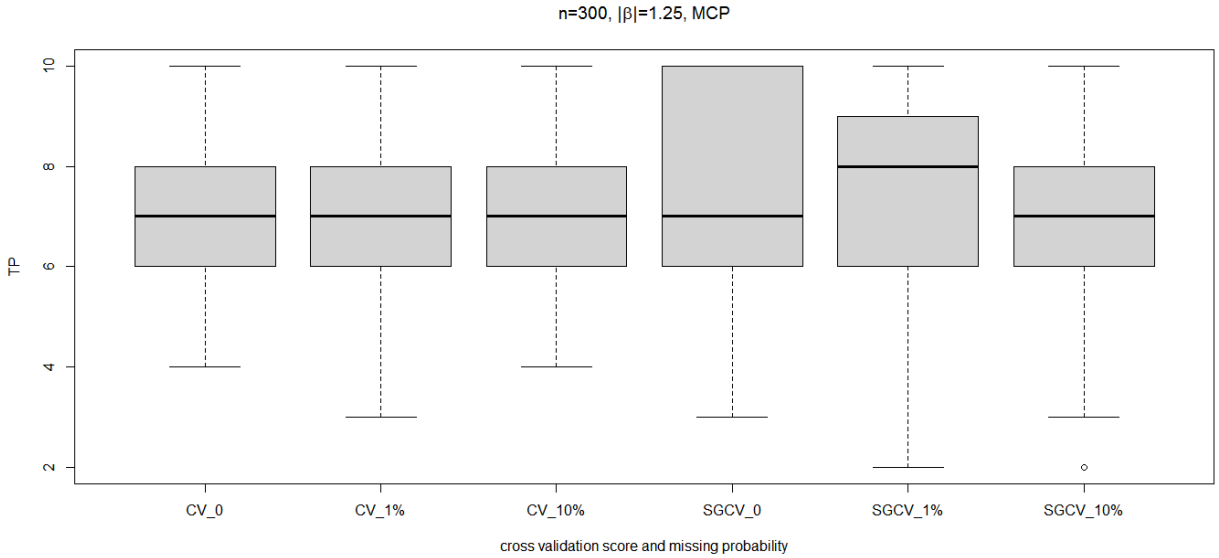



Figure 5.30: Boxplot of FP using MCP when $n = 300$ and $|\beta|$=1.25 without and with missing data

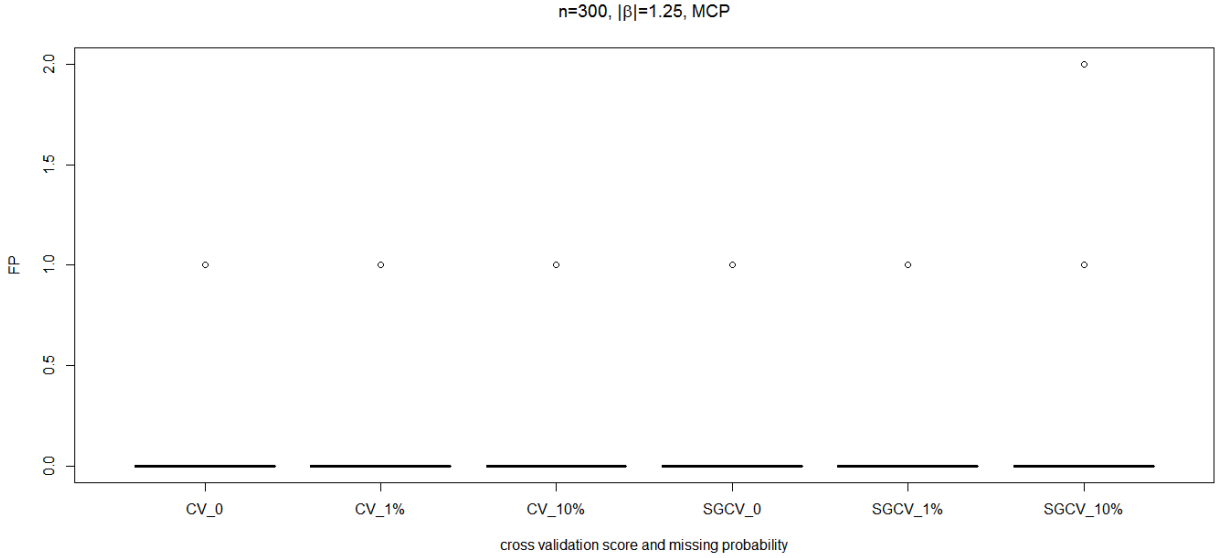


Figure 5.31: Boxplot of TP using MCP when $n = 400$ and $|\beta|$=0.75 without and with missing data

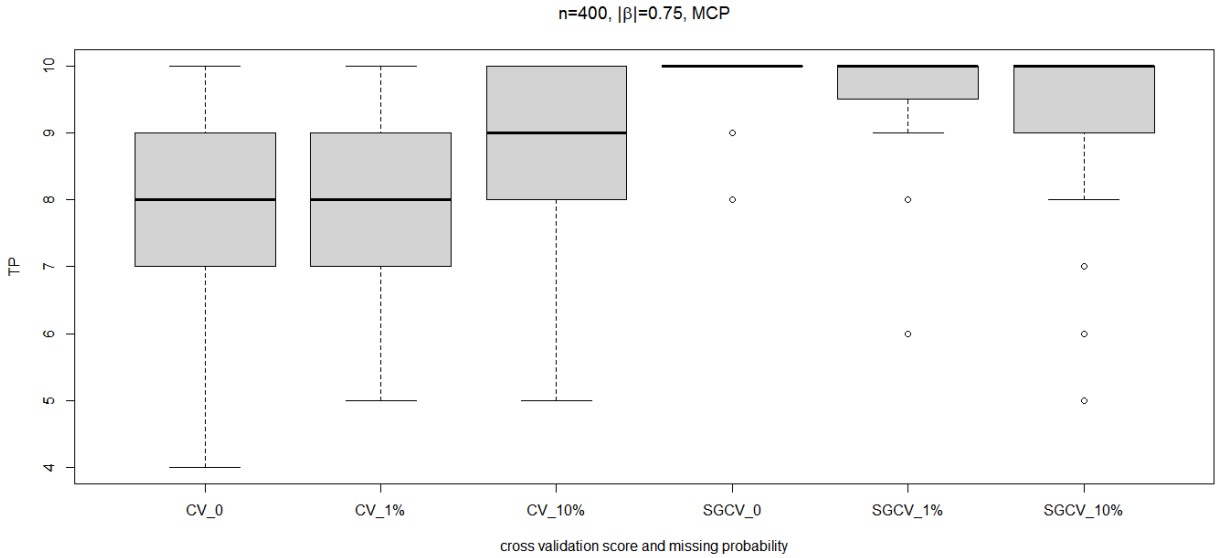



Figure 5.32: Boxplot of FP using MCP when $n = 400$ and $|\beta|$=0.75 without and with missing data

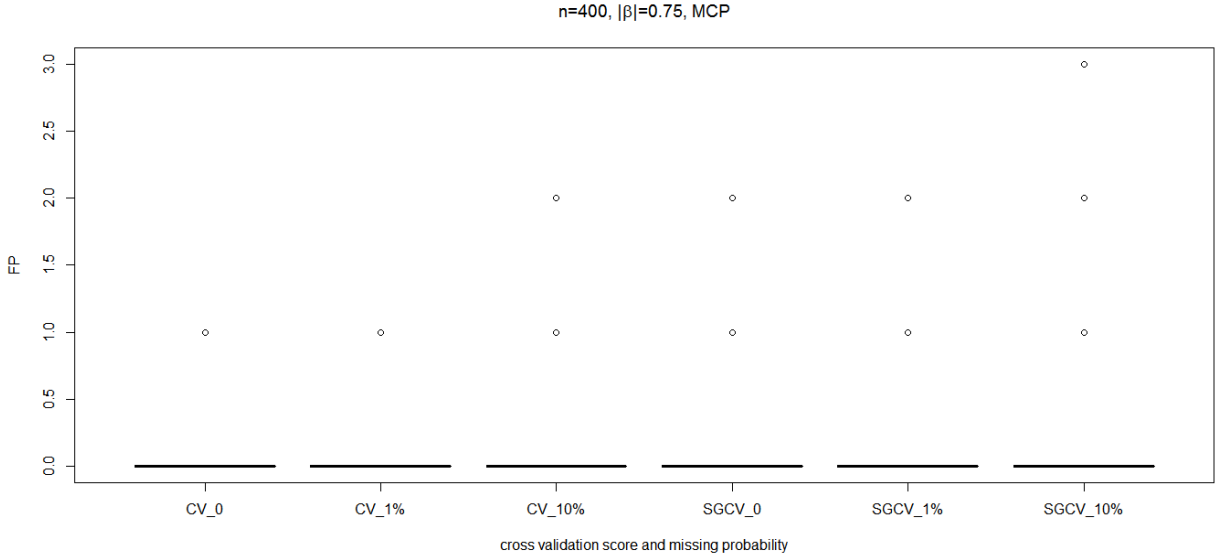


Figure 5.33: Boxplot of TP using MCP when $n = 400$ and $|\beta|$=1 without and with missing data

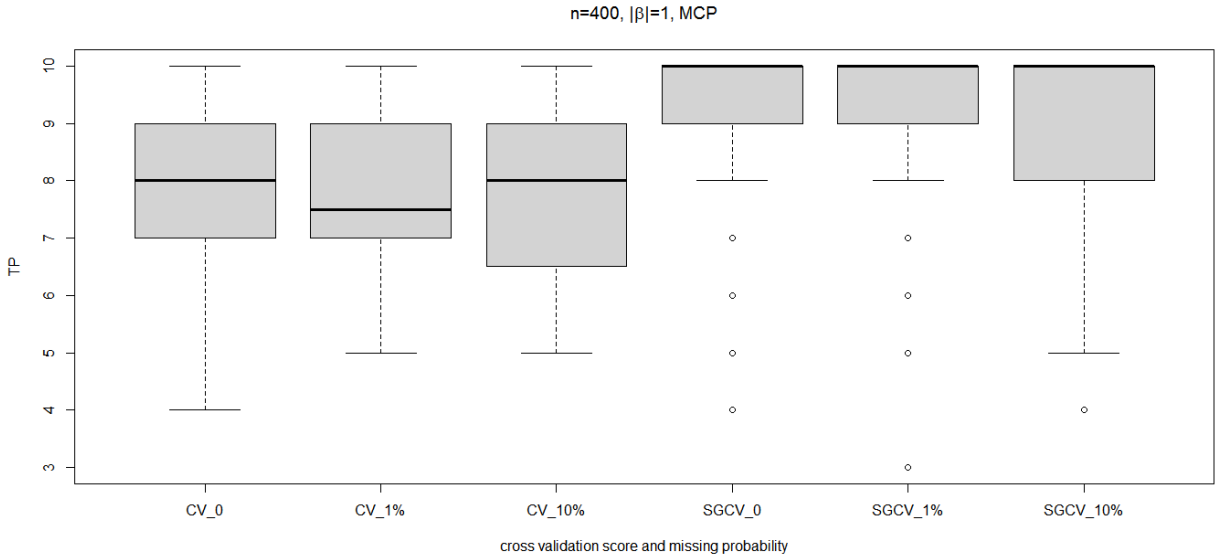



Figure 5.34: Boxplot of FP using MCP when $n = 400$ and $|\beta|$=1 without and with missing data

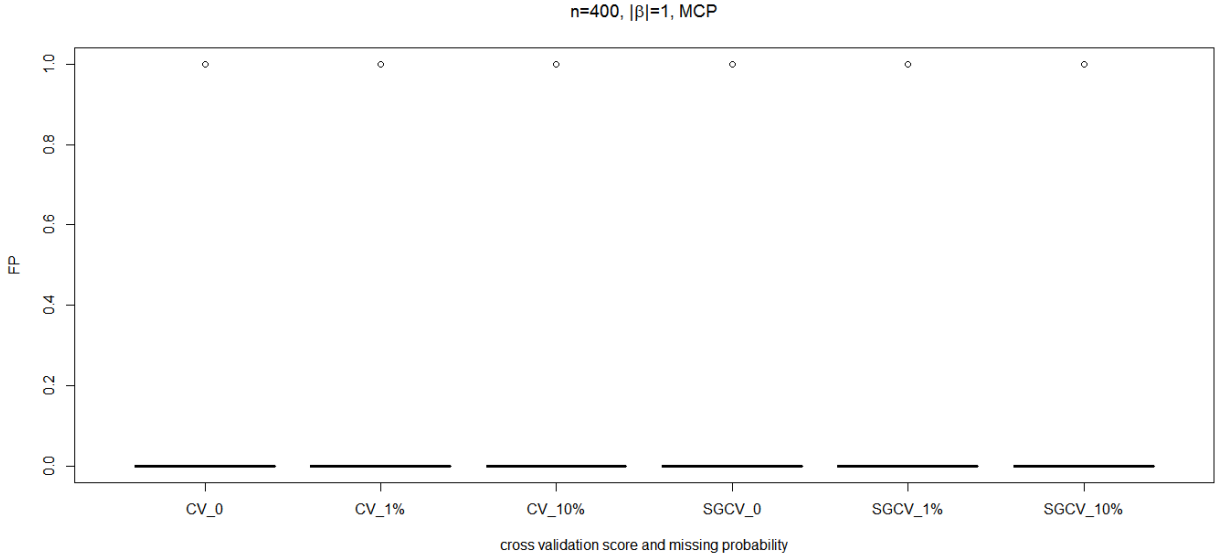


Figure 5.35: Boxplot of TP using MCP when $n = 400$ and $|\beta|$=1.25 without and with missing data

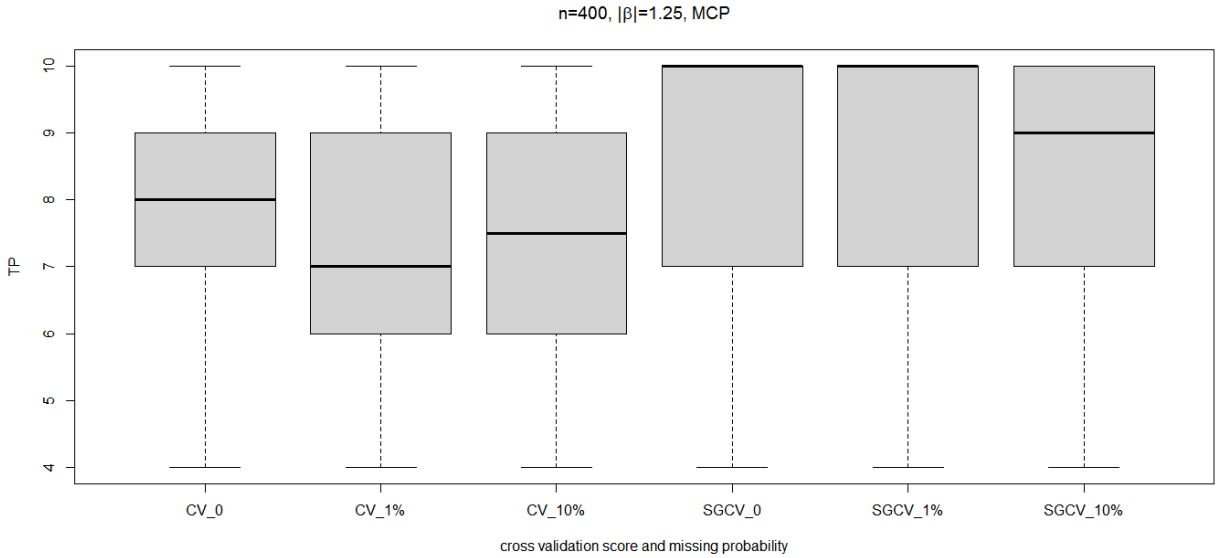



Figure 5.36: Boxplot of FP using MCP when $n = 400$ and $|\beta|$=1.25 without and with missing data

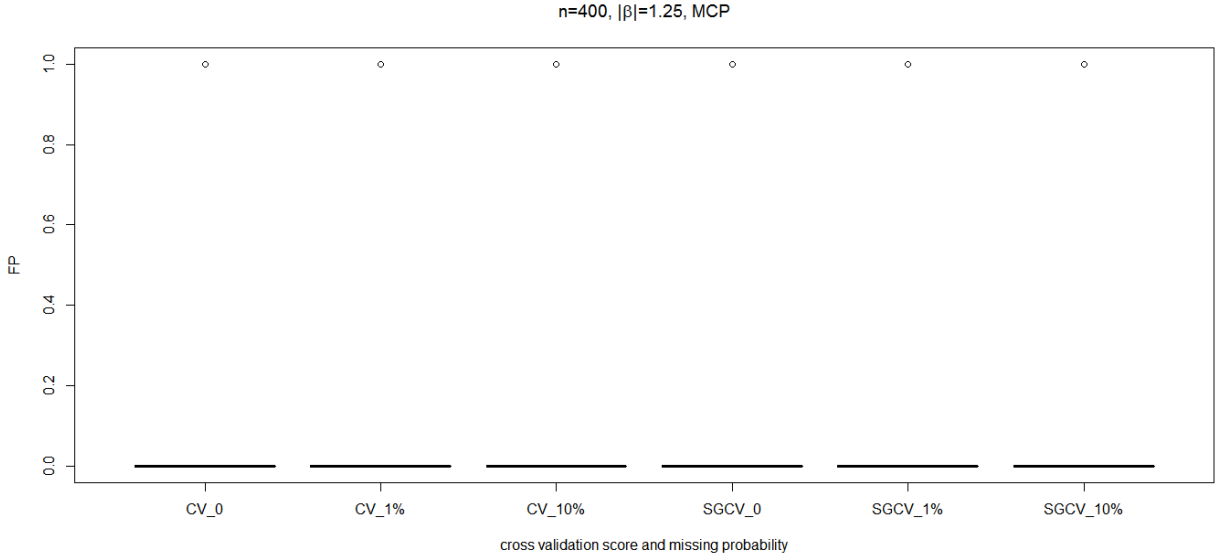


Figure 5.37: Boxplot of TP using SICA when $n = 200$ and $|\beta|$=0.75 without and with missing data

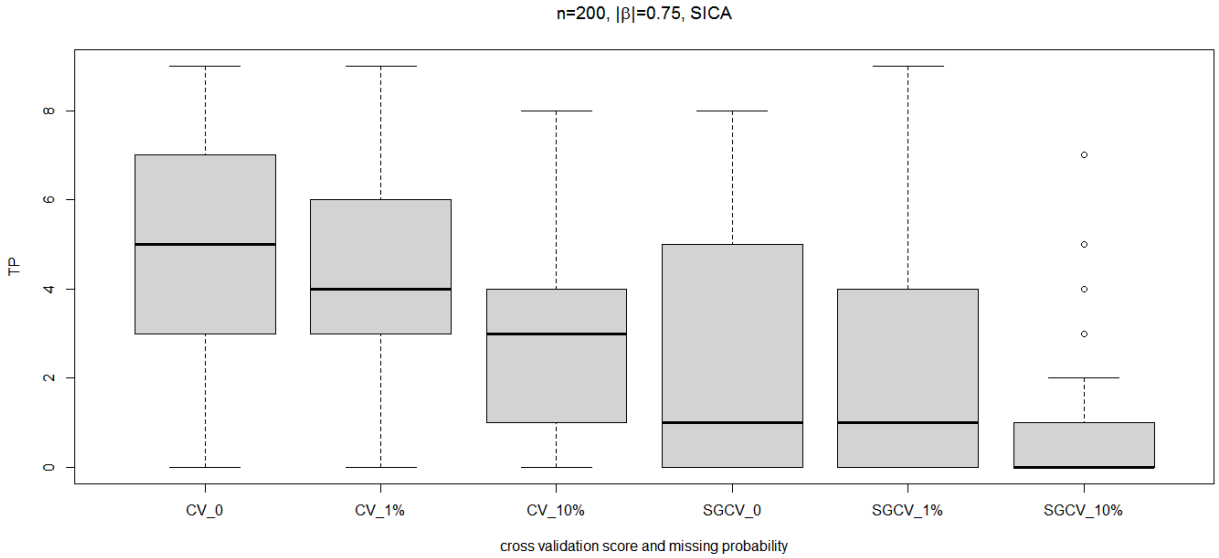



Figure 5.38: Boxplot of FP using SICA when $n = 200$ and $|\beta|$=0.75 without and with missing data

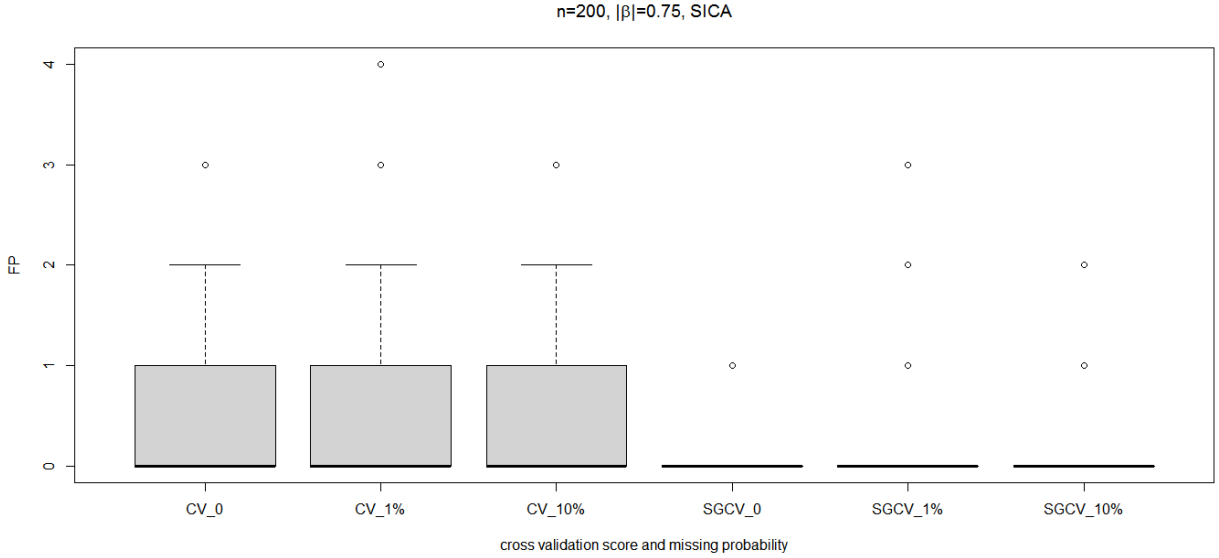


Figure 5.39: Boxplot of TP using SICA when $n = 200$ and $|\beta|$=1 without and with missing data

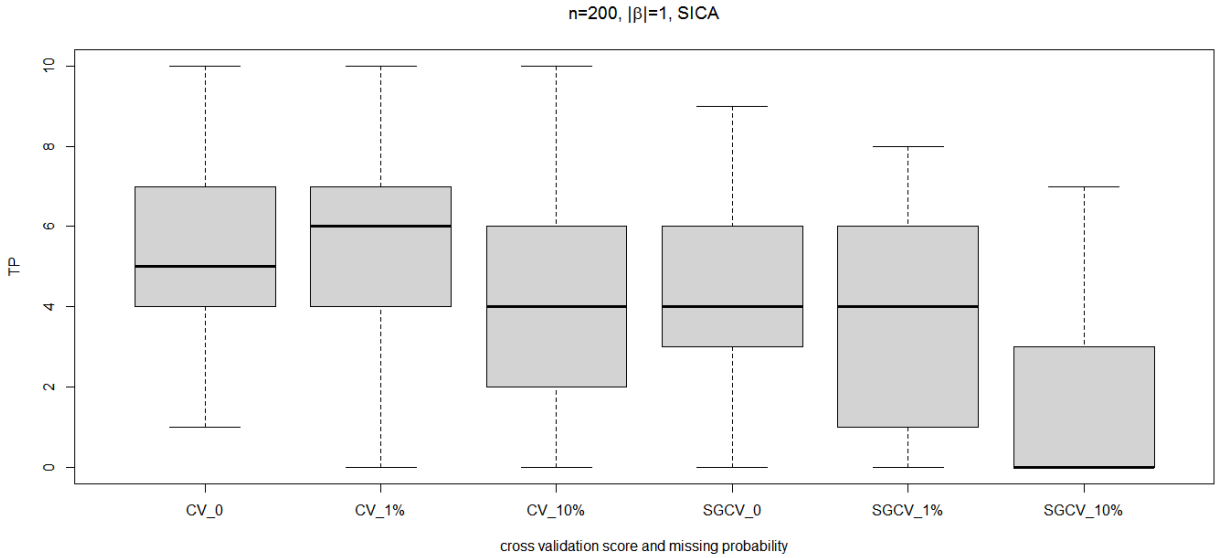



Figure 5.40: Boxplot of FP using SICA when $n = 200$ and $|\beta|$=1 without and with missing data

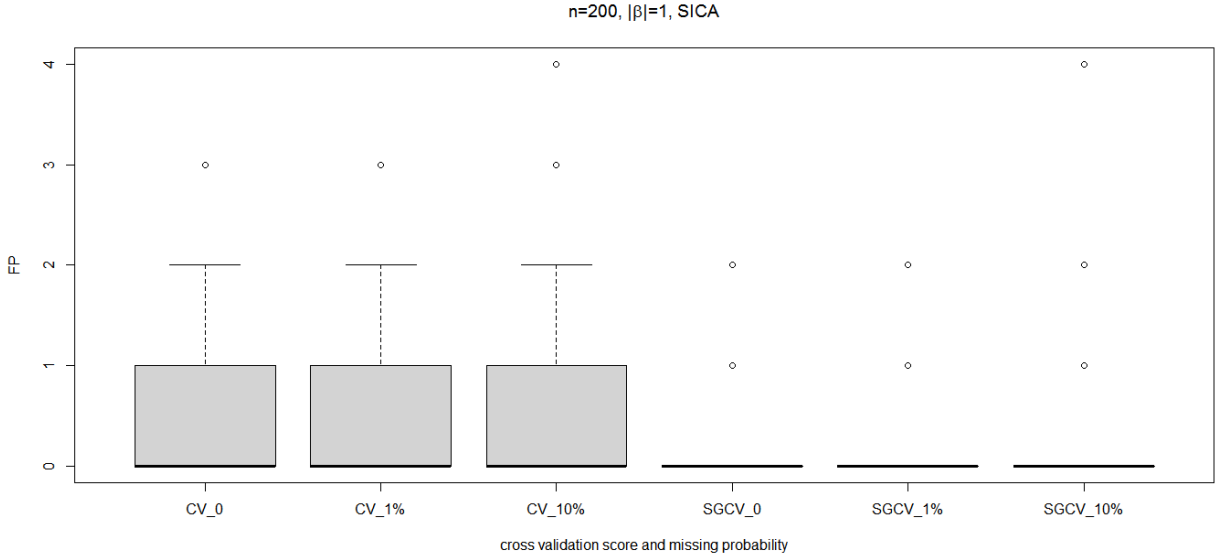


Figure 5.41: Boxplot of TP using SICA when $n = 200$ and $|\beta|$=1.25 without and with missing data

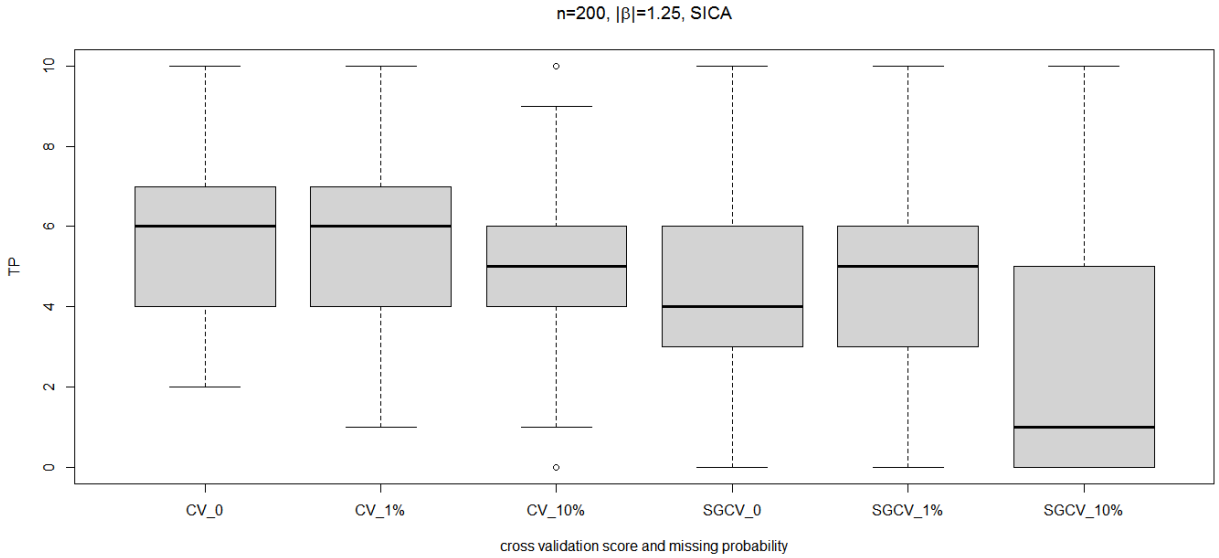



Figure 5.42: Boxplot of FP using SICA when $n = 200$ and $|\beta|$=1.25 without and with missing data

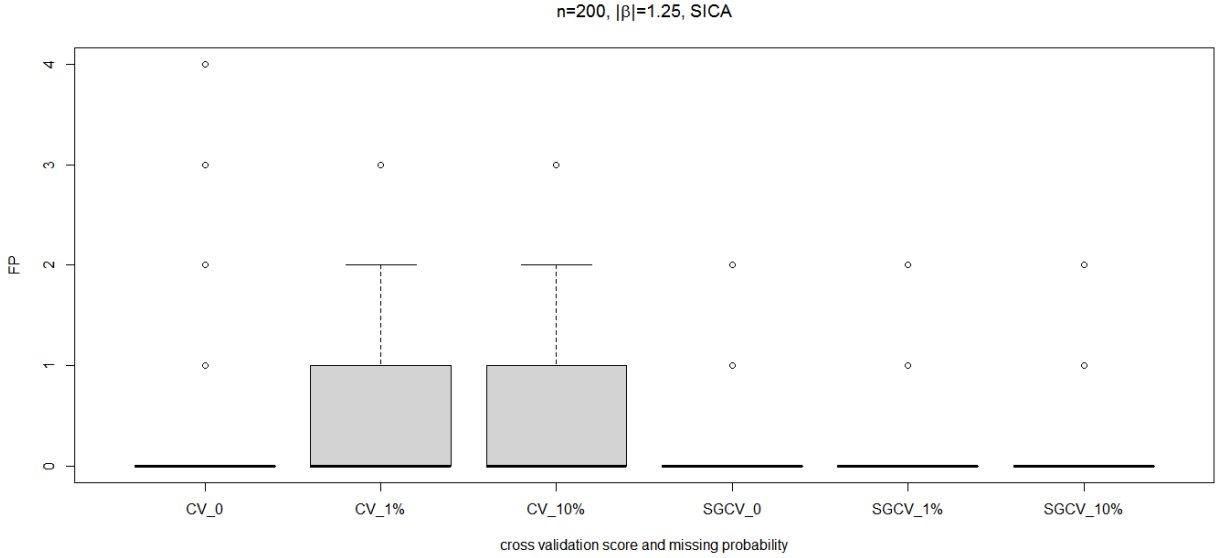


Figure 5.43: Boxplot of TP using SICA when $n = 300$ and $|\beta|$=0.75 without and with missing data

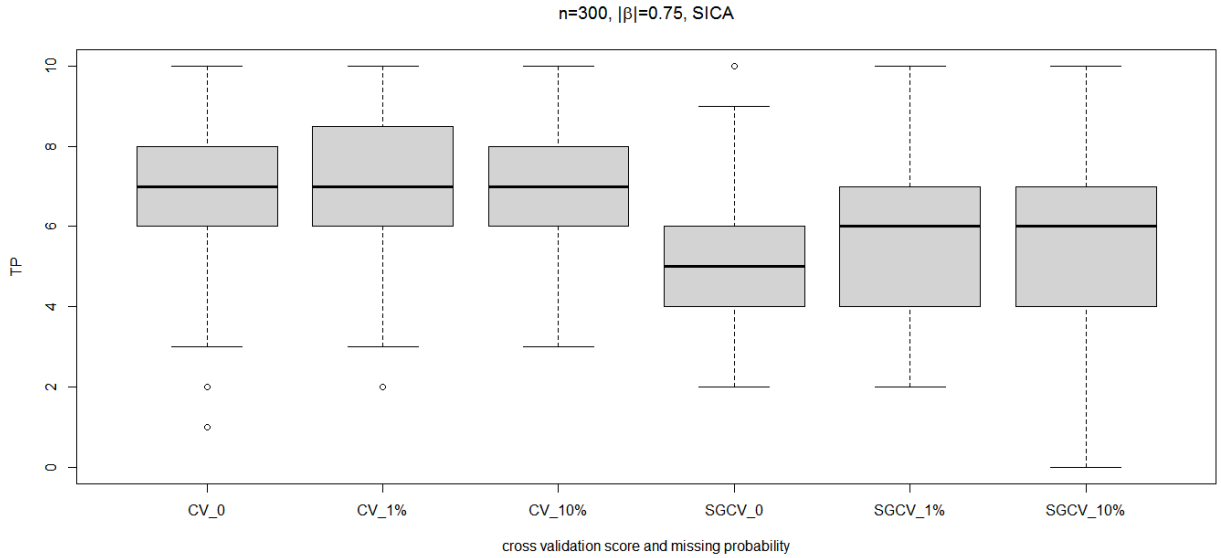



Figure 5.44: Boxplot of FP using SICA when $n = 300$ and $|\beta|$=0.75 without and with missing data

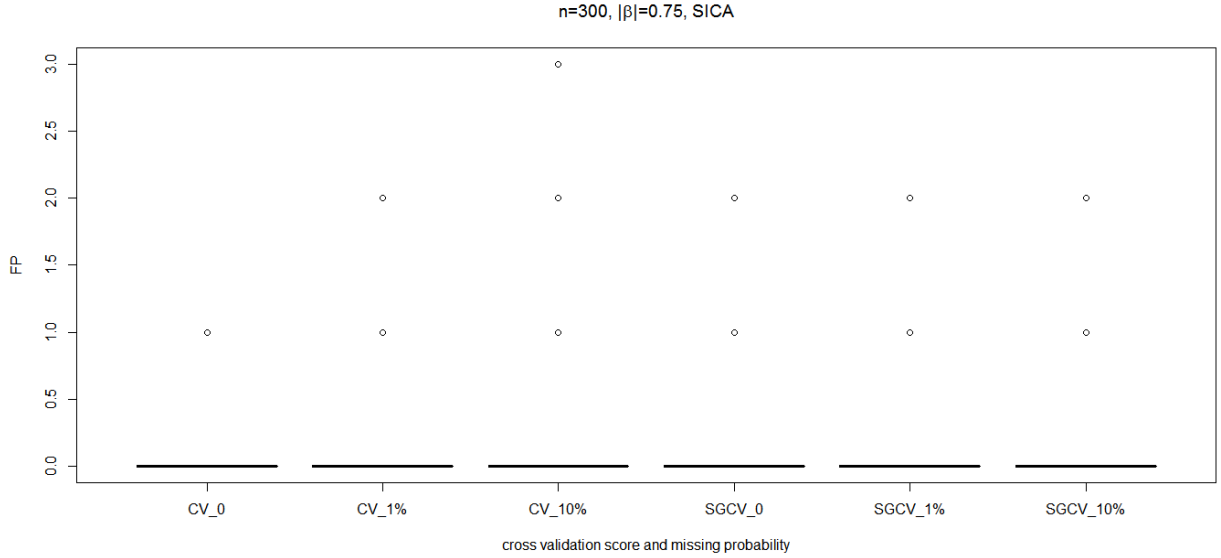


Figure 5.45: Boxplot of TP using SICA when $n = 300$ and $|\beta|$=1 without and with missing data

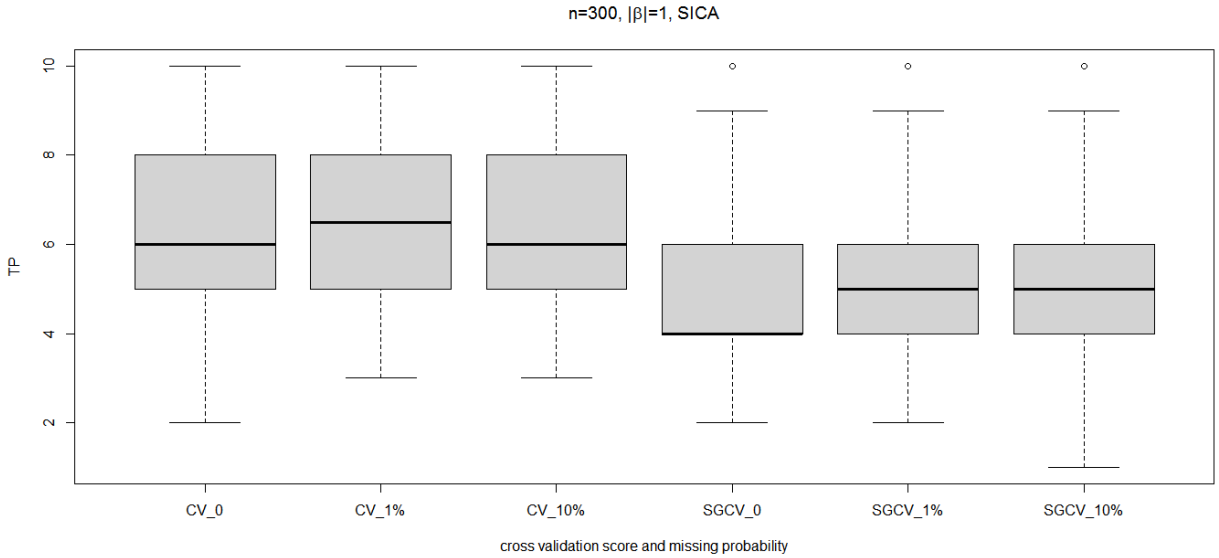



Figure 5.46: Boxplot of FP using SICA when $n = 300$ and $|\beta|$=1 without and with missing data

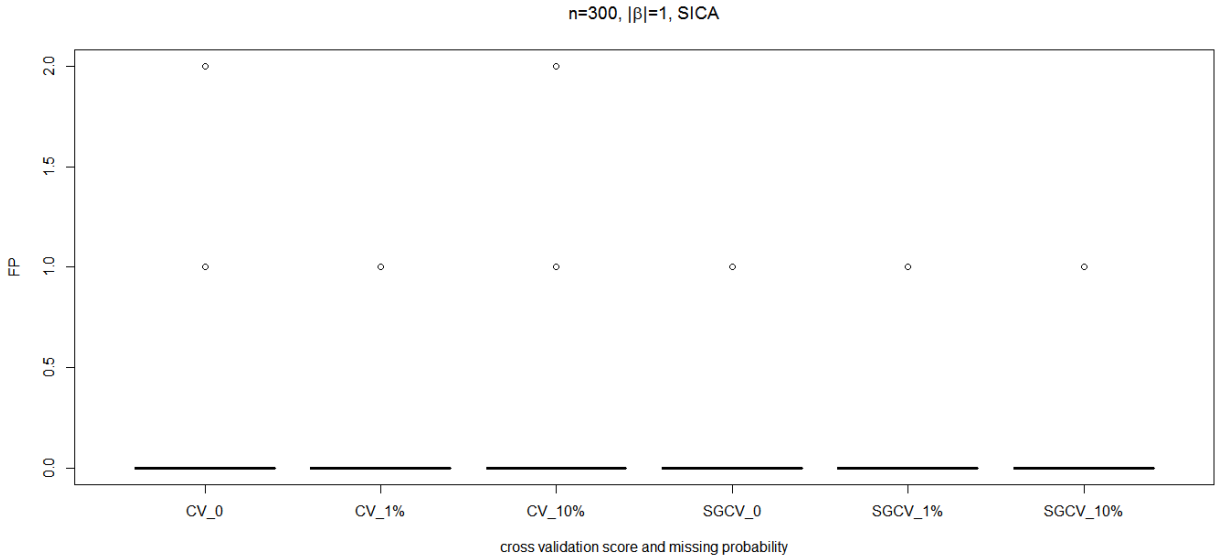


Figure 5.47: Boxplot of TP using SICA when $n = 300$ and $|\beta|$=1.25 without and with missing data

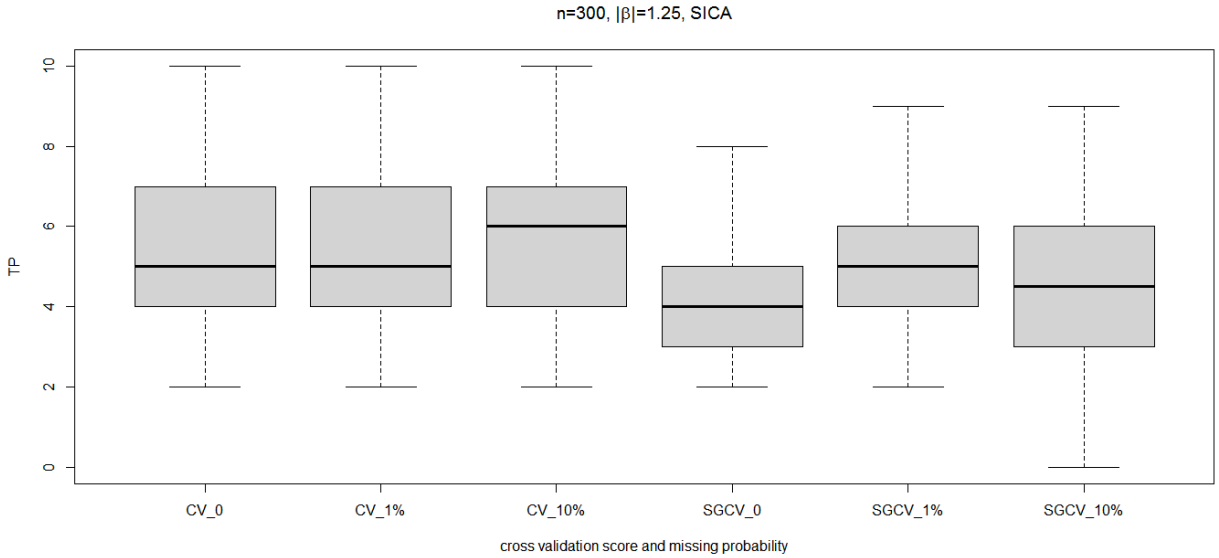



Figure 5.48: Boxplot of FP using SICA when $n = 300$ and $|\beta|$=1.25 without and with missing data

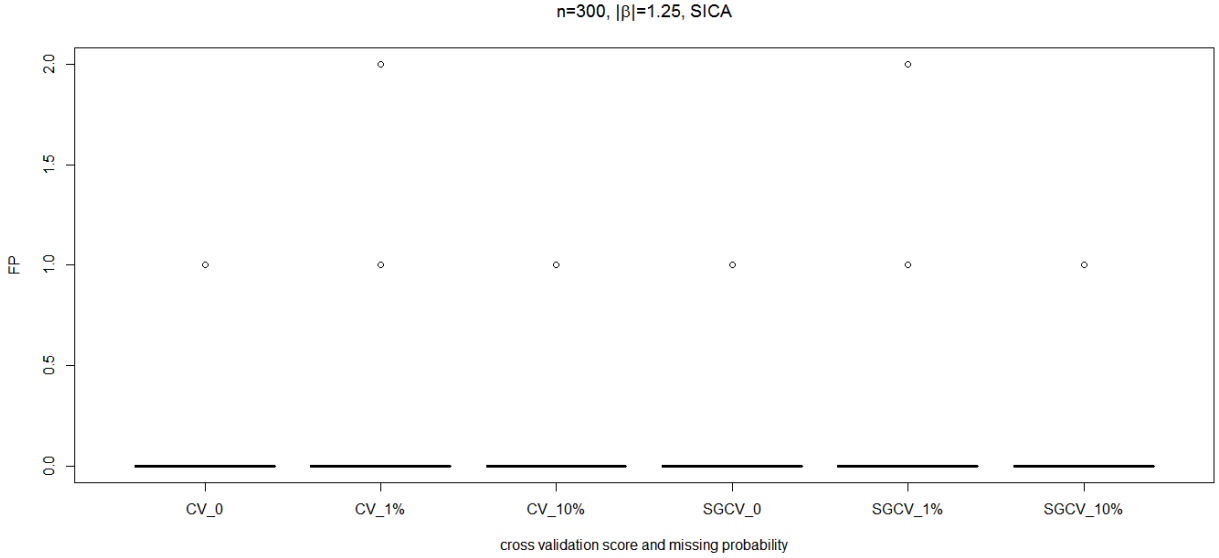


Figure 5.49: Boxplot of TP using SICA when $n = 400$ and $|\beta|$=0.75 without and with missing data

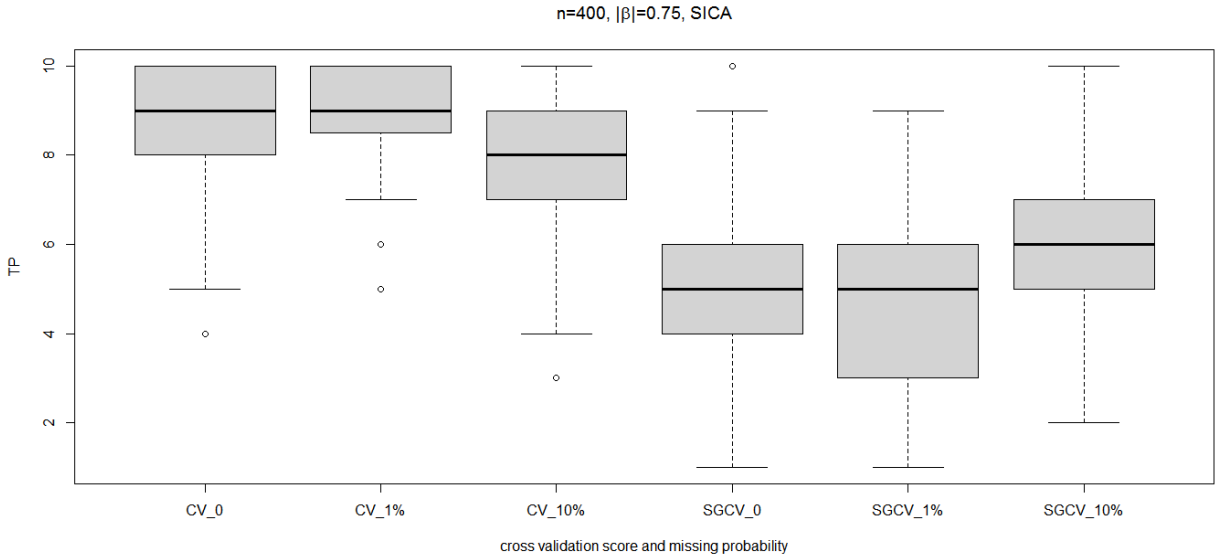



Figure 5.50: Boxplot of FP using SICA when $n = 400$ and $|\beta|$=0.75 without and with missing data

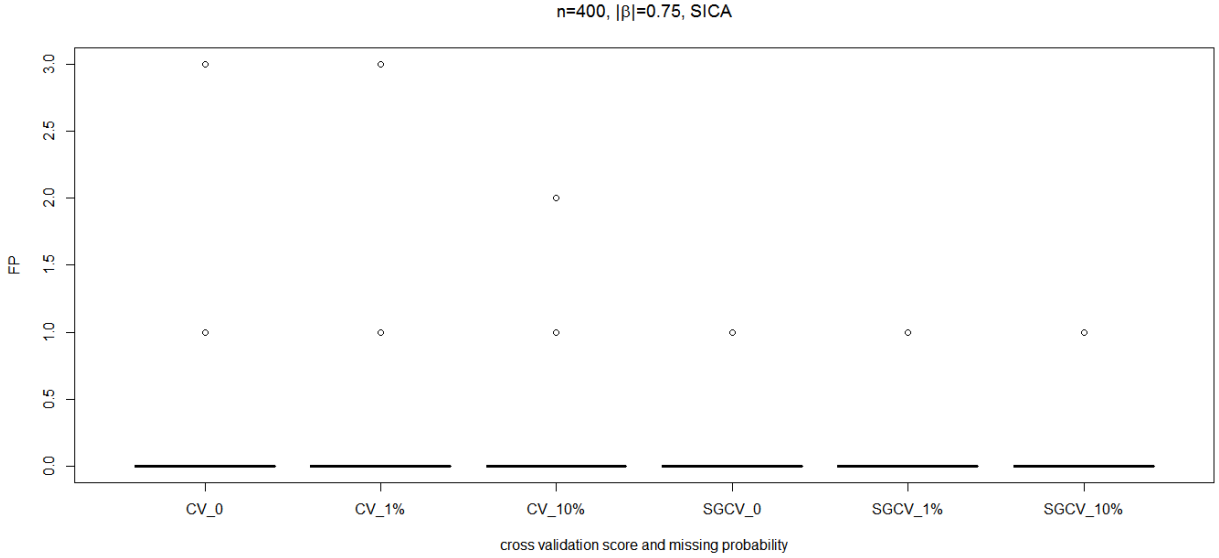


Figure 5.51: Boxplot of TP using SICA when $n = 400$ and $|\beta|$=1 without and with missing data

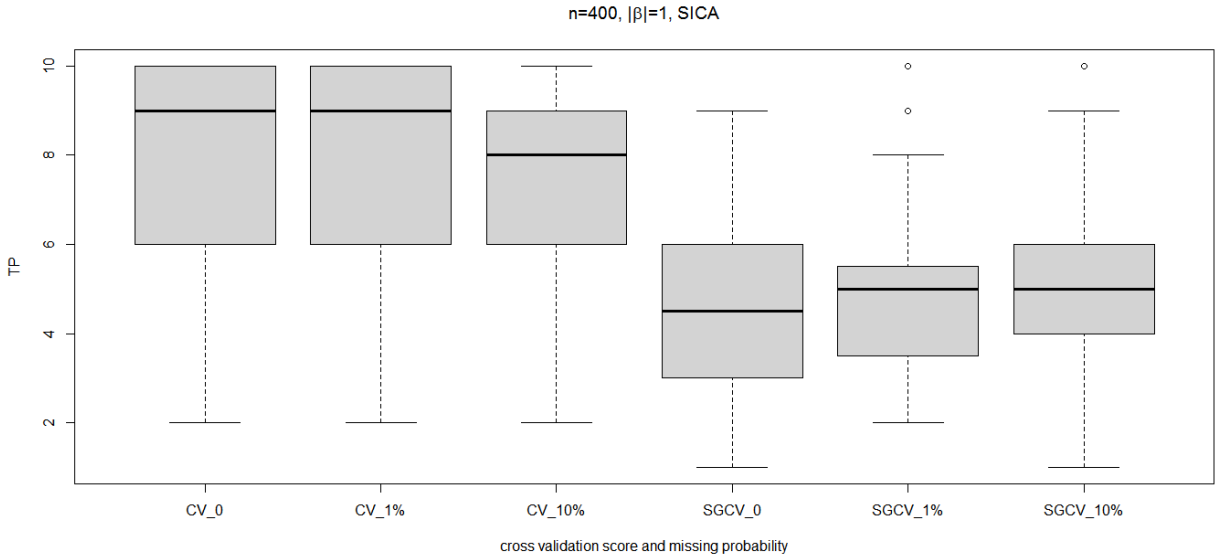



Figure 5.52: Boxplot of FP using SICA when $n = 400$ and $|\beta|$=1 without and with missing data

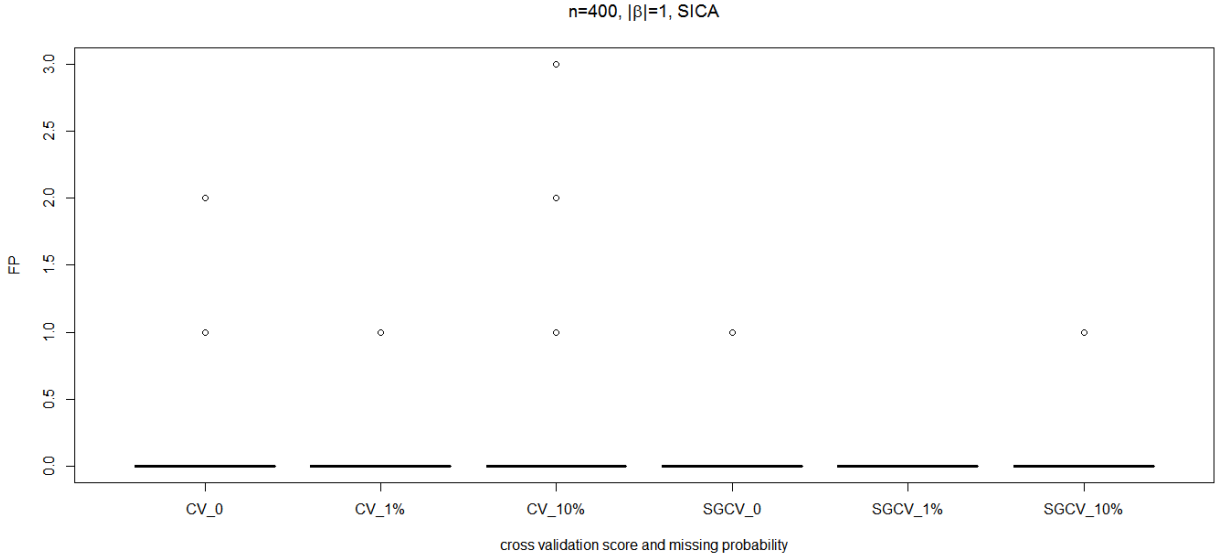


Figure 5.53: Boxplot of TP using SICA when $n = 400$ and $|\beta|$=1.25 without and with missing data

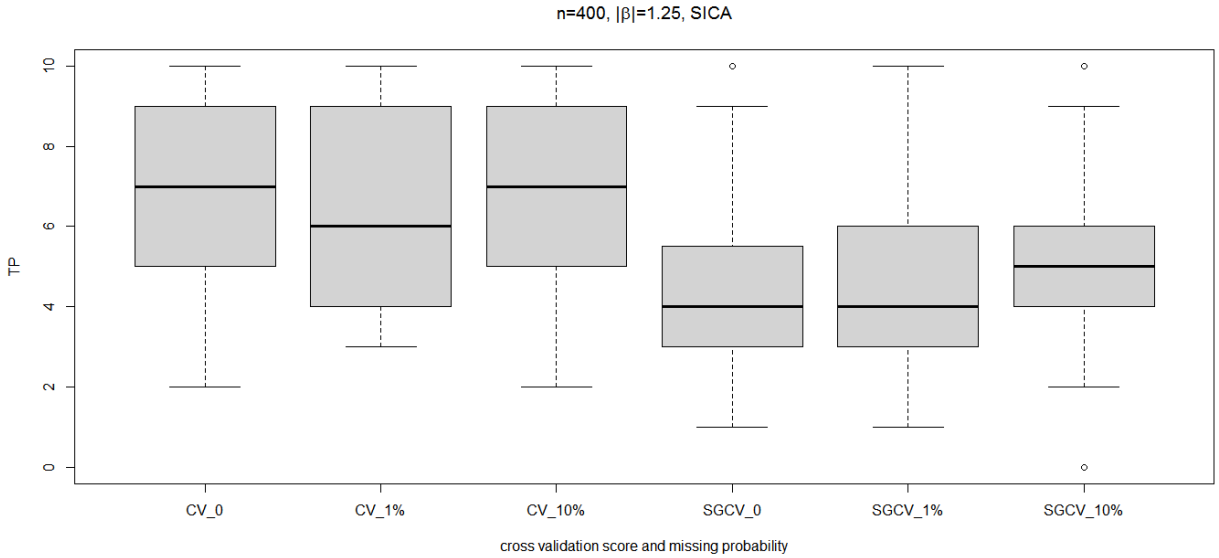
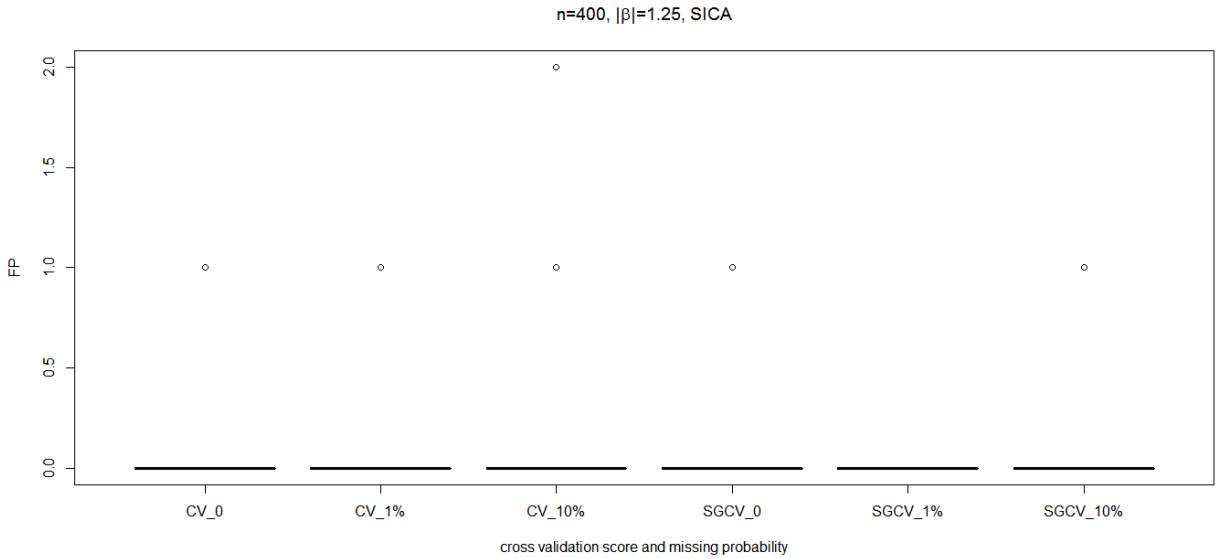



Figure 5.54: Boxplot of FP using SICA when $n = 400$ and $|\beta|$=1.25 without and with missing data




This method works reasonably well compared to the method for complete data in Chapter

3. In some cases, when the probability of missing was set to 1%, it was even better than the case with no missing data, selecting more true positives and fewer false positives.

## 5.2 Application to the AML Dataset

The method proposed in Chapter 4 was applied to the AML data. 10 copies of imputed datasets were obtained using the MissForest package and analyzed. Cross-validation was performed 10 times with a different set of folds used each time. The maximum number of iterations in the algorithm was set to 1000. 100 was used instead if the job could not be finished within the maximum time allowed: 14 days. The warm start approach resulted in numerical issues for some of the 10 cross-validations for MCP and SICA so $\mathbf{0}$ was used as the initial value for each value of $\lambda$ instead for these jobs. The algorithm could not finish running for a couple of training data in the cross-validations when using SICA and SCAD. In this case, $\mathbf{0}$ was used as the initial value for each value of $\lambda$ for these training data. The predictors that were at least selected once and their number of times of being selected using SICA, MCP and SCAD are listed in Table 5.28-5.30, respectively. Times_CV denotes the number of times being selected using the CV score as the criterion in cross-validation and Times_SGCV denotes it when using the SGCV score as the criterion.

Table 5.28: Number of times predictors selected using SICA on 10 imputed datasets with 10 times of cross-validation

| Predictor | Times_CV | Times_SGCV |
|---|---|---|
| ENSG00000233451 | 4 | 2 |
| ENSG00000271857 | 2 | 1 |
| CD109 | 2 | 0 |

Table 5.28: Continued

| | | |
|---|---|---|
| cyto_group_complex | 1 | 2 |
| SDHAP3 | 1 | 2 |
| SPAG1 | 1 | 1 |
| SSPN | 1 | 1 |
| SCN9A | 1 | 1 |
| TMEM217 | 1 | 1 |
| C10orf55 | 1 | 1 |
| RN7SKP32 | 1 | 1 |
| ENSG00000251293 | 1 | 1 |
| ENSG00000275216 | 1 | 1 |
| ENSG00000223528 | 1 | 0 |
| ENSG00000229418 | 1 | 0 |
| PTMAP4 | 1 | 0 |
| INTS6-AS1 | 1 | 0 |
| PTMAP2 | 1 | 0 |
| TPM3P1 | 1 | 0 |
| LINC01979 | 1 | 0 |
| PTMAP12 | 1 | 0 |
| FHL1 | 1 | 0 |
| METTL16 | 1 | 0 |
| DDX19B | 1 | 0 |
| DDIT4 | 1 | 0 |
| ENSG00000223528 | 0 | 3 |
| B4GALNT3 | 0 | 2 |

Table 5.28: Continued

| | | |
|---|---|---|
| CLEC3B | 0 | 2 |
| RNU6-31P | 0 | 2 |
| HMGB3P22 | 0 | 2 |
| ENSG00000228303 | 0 | 2 |
| C1QTNF9 | 0 | 2 |
| RPS24P17 | 0 | 2 |
| TARS1-DT | 0 | 2 |
| ENSG00000255232 | 0 | 2 |
| cyto_group_2 unbalanced rearrangements | 0 | 1 |
| cyto_group_sole deletion/loss 9q | 0 | 1 |
| CASP10 | 0 | 1 |
| CDK11A | 0 | 1 |
| PREX2 | 0 | 1 |
| BCAR1 | 0 | 1 |
| SPAG6 | 0 | 1 |
| PTPRH | 0 | 1 |
| MECOM | 0 | 1 |
| CUX2 | 0 | 1 |
| IL1R1 | 0 | 1 |
| PTGIS | 0 | 1 |
| EREG | 0 | 1 |
| WNT1 | 0 | 1 |
| MYO1B | 0 | 1 |
| C1QL1 | 0 | 1 |

Table 5.28: Continued

| | | |
|---|---|---|
| STAR | 0 | 1 |
| KCTD15 | 0 | 1 |
| PPP4R1 | 0 | 1 |
| GPR78 | 0 | 1 |
| RALGDS | 0 | 1 |
| IL17RE | 0 | 1 |
| GARRE1 | 0 | 1 |
| GLOD4 | 0 | 1 |
| ZNF355P | 0 | 1 |
| ENSG00000170165 | 0 | 1 |
| GTSF1 | 0 | 1 |
| RXFP1 | 0 | 1 |
| LINC02904 | 0 | 1 |
| LOC102723701 | 0 | 1 |
| SCN5A | 0 | 1 |
| ZFTA | 0 | 1 |
| OR1J2 | 0 | 1 |
| MIR367 | 0 | 1 |
| SNORD51 | 0 | 1 |
| APTR | 0 | 1 |
| PTCHD3P3 | 0 | 1 |
| DAZAP2P1 | 0 | 1 |
| AKR1C5P | 0 | 1 |
| TMA7B | 0 | 1 |

Table 5.28: Continued

| | | |
|---|---|---|
| ZNF277-AS1 | 0 | 1 |
| MYL12BP2 | 0 | 1 |
| HMGN2P3 | 0 | 1 |
| C1DP1 | 0 | 1 |
| ENSG00000231868 | 0 | 1 |
| MGAT2P1 | 0 | 1 |
| RPL36AP26 | 0 | 1 |
| ENSG00000236391 | 0 | 1 |
| ENSG00000237301 | 0 | 1 |
| RPL14P3 | 0 | 1 |
| ENSG00000242795 | 0 | 1 |
| ENSG00000242951 | 0 | 1 |
| ENSG00000243744 | 0 | 1 |
| HNRNPA1P55 | 0 | 1 |
| ENSG00000250362 | 0 | 1 |
| RN7SKP57 | 0 | 1 |
| ENSG00000255487 | 0 | 1 |
| PSMC1P9 | 0 | 1 |
| ENSG00000256139 | 0 | 1 |
| PSMA2 | 0 | 1 |
| PRORP | 0 | 1 |
| RN7SL45P | 0 | 1 |
| MADCAM1-AS1 | 0 | 1 |
| LINC02080 | 0 | 1 |

Table 5.28: Continued

| | | |
|---|---|---|
| ATXN7L3-AS1 | 0 | 1 |
| ENSG00000272343 | 0 | 1 |
| ENSG00000272400 | 0 | 1 |
| ENSG00000272849 | 0 | 1 |
| ENSG00000272865 | 0 | 1 |
| ENSG00000276026 | 0 | 1 |
| ENSG00000276248 | 0 | 1 |
| ENSG00000279024 | 0 | 1 |
| LINC02341 | 0 | 1 |
| LINC02767 | 0 | 1 |

Table 5.29: Number of times predictors selected using MCP on 10 imputed datasets with 10 times of cross-validation

| Predictor | Times_CV | Times_SGCV |
|---|---|---|
| ENSG00000233451 | 8 | 4 |
| ENSG00000271857 | 8 | 2 |
| PTMAP5 | 7 | 5 |
| SCN9A | 6 | 6 |
| ALDH2 | 4 | 3 |
| CD109 | 3 | 1 |
| SSPN | 3 | 0 |
| GARRE1 | 3 | 0 |
| RN7SKP32 | 3 | 0 |
| RPS3AP41 | 3 | 0 |

Table 5.29: Continued

| | | |
|---|---|---|
| ENSG00000223528 | 2 | 7 |
| SPAG6 | 2 | 0 |
| RPS2P21 | 2 | 0 |
| ENSG00000248347 | 1 | 3 |
| LINC01770 | 1 | 2 |
| CASP10 | 1 | 0 |
| CALCRL | 1 | 0 |
| SDHAP3 | 1 | 0 |
| ENSG00000228201 | 1 | 0 |
| ENSG00000242951 | 1 | 0 |
| ENSG00000251293 | 1 | 0 |
| ENSG00000262172 | 1 | 0 |
| PLCB4 | 0 | 2 |
| INSYN2A | 0 | 1 |

Table 5.30: Number of times predictors selected using SCAD on 10 imputed datasets with 10 times of cross-validation

| Predictor | Times_CV | Times_SGCV |
|---|---|---|
| RPS2P21 | 10 | 10 |
| ENSG00000233451 | 10 | 10 |
| RPS3AP41 | 10 | 10 |
| ENSG00000271857 | 10 | 10 |
| CD109 | 10 | 9 |
| SDHAP3 | 10 | 9 |

Table 5.30: Continued

| | | |
|---|---|---|
| GARRE1 | 10 | 9 |
| GAS6 | 10 | 5 |
| SSPN | 9 | 10 |
| SCN9A | 9 | 10 |
| ALDH2 | 9 | 9 |
| PTMAP5 | 9 | 7 |
| ENSG00000262172 | 8 | 8 |
| ENSG00000248347 | 7 | 7 |
| LINC01979 | 7 | 5 |
| ENSG00000228303 | 6 | 10 |
| C1QL1 | 6 | 9 |
| ENSG00000275216 | 6 | 9 |
| IL2RA | 6 | 3 |
| PLCB4 | 5 | 7 |
| RBM3 | 5 | 6 |
| KIF1A | 5 | 5 |
| STAR | 5 | 4 |
| ENSG00000228201 | 5 | 4 |
| CASP10 | 4 | 9 |
| ENSG00000223528 | 4 | 6 |
| CLIC4 | 4 | 6 |
| CENPV | 4 | 5 |
| TUBB2BP1 | 4 | 5 |
| HOPX | 4 | 4 |

Table 5.30: Continued

| | | |
|---|---|---|
| MSLN | 4 | 1 |
| INTS6-AS1 | 4 | 0 |
| TRIM9 | 3 | 5 |
| ACTBP11 | 3 | 3 |
| RN7SKP32 | 3 | 3 |
| LINC01770 | 3 | 1 |
| ENSG00000229664 | 3 | 1 |
| PTMAP3 | 3 | 1 |
| TPM3P1 | 3 | 0 |
| DDX19B | 2 | 5 |
| CLEC3B | 2 | 5 |
| LINC00676 | 2 | 4 |
| TMEM217 | 2 | 3 |
| EGFEM1P | 2 | 3 |
| LINC01415 | 2 | 1 |
| ZNF355P | 1 | 8 |
| cyto_group_complex | 1 | 6 |
| ENSG00000243744 | 1 | 3 |
| SDCBP2 | 1 | 2 |
| CYCSP23 | 1 | 2 |
| MPZ | 1 | 1 |
| VSTM4 | 1 | 1 |
| TMEM273 | 1 | 1 |
| ENSG00000251467 | 1 | 1 |

Table 5.30: Continued

| | | |
|---|---|---|
| CIBAR1 | 1 | 0 |
| ENSG00000242951 | 1 | 0 |
| CIBAR1P1 | 1 | 0 |
| ENSG00000261346 | 0 | 4 |
| MADCAM1-AS1 | 0 | 4 |
| RBM23 | 0 | 3 |
| MIR155HG | 0 | 3 |
| CCDC68 | 0 | 2 |
| ENSG00000251293 | 0 | 2 |
| PTMAP12 | 0 | 2 |
| FLT3.ITD | 0 | 2 |
| ST7/ST7-OT3 | 0 | 1 |
| THG1L | 0 | 1 |
| SMAD5 | 0 | 1 |
| IL17RE | 0 | 1 |
| DMRTC1B | 0 | 1 |
| INSYN2A | 0 | 1 |
| PTMAP2 | 0 | 1 |
| PTMAP9 | 0 | 1 |
| C10orf55 | 0 | 1 |
| PTMAP4 | 0 | 1 |
| HSPE1P22 | 0 | 1 |
| PSMC1P9 | 0 | 1 |

Consistent with the previous results, SICA selects the smallest number of predictors while SCAD selects the most. We focused on the 11 predictors that were always selected by SCAD, using either the CV or the SGCV score as criterion. Expression of the SCN9A gene has been shown to be related to prostate, gastric and ovarian cancer [2, 20, 74, 101, 105]. SSPN has been shown to be related to childhood acute lymphoblastic leukemia [7, 25]. Expression of the GAS6 gene has been reported to be an adverse prognostic marker in cytogenetically normal AML [99]. Expression of the CD109 gene has been reported to be a prognostic factor in acute myeloid leukaemia [73, 97, 76, 22].

We also tried to validate the results on a dataset on the Gene Expression Omnibus with GEO accession: GSE146173 [5]. Of the 202 patients in the dataset, 132 achieved complete remission. One of them lacks information on whether the patient relapsed so was excluded from analysis. Of these 131 patients, 66 relapsed and 65 were censored without relapse. 5 of the 11 predictors that we focused on are present in this dataset. One of them, ENSG00000228303 takes the value of 0 for all patients so it was not analyzed. Because for this dataset death was not a competing risk given all patients who died also relapsed before death, we fit univariate Cox models for time from complete remission to relapse using each of the 4 predictors and list their p-values in Table 5.31. "LRT" stands for likelihood ratio test. "Wald" stands for Wald test. "Score" stands for score test.

Table 5.31: p-values for 4 predictors when fit on the GSE146173 data

| Predictors | p-value (LRT) | p-value (Wald) | p-value (Score) |
|---|---|---|---|
| RPS2P21 | 0.08 | 0.05 | 0.05 |
| ENSG00000233451 | 0.6 | 0.6 | 0.6 |
| RPS3AP41 | 0.5 | 0.6 | 0.5 |
| ENSG00000271857 | 0.009 | 0.001 | 0.0003 |

ENSG00000233451 and RPS3AP41 were not confirmed. RPS2P21 was marginally significant. ENSG00000271857 was highly significant. ENSG00000271857 is a long non-coding

RNA, located on chromosome 6 and is antisense to the RUNX2 gene. This might be worth further research.

## 5.3 Discussion

Because the method proposed in Chapter 4 is fit on multiple imputed datasets, it takes much longer to run than the method proposed in Chapter 2. When the dataset is too large, the algorithm may not be able to finish running within the maximum allowed time on a computing cluster. In this case, it would be necessary to reduce the number of predictors by screening before running the algorithm.

# Chapter 6: Conclusions and Future Research

In this thesis, we developed methods to select variables for high-dimensional competing-risks data without or with missing data, based on the Fine-Gray model. Simulation studies show that our methods performed fairly well when identifying truly associated predictors and identified fewer false positive predictors in comparison to competing methods: fastcmprsk, CoxBoost, and randomForestSRC. Then we applied the developed methods to an AML dataset which contains missing values.

Other than the Fine-Gray model, competing risks data are also oftened analyzed using the proportional cause-specific hazards model. This dissertation focused on the Fine-Gray model because, unlike cause-specific hazards model, it directly models the cumulative incidence function thus the predictor effects can be more simply interpreted. In real data analysis, one can use both models to gain more insight into the data.

When the tuning parameter $\lambda$ gets very small, the warm start approach might encounter numerical issues. This is due to the large number of predictors. When the dataset is too big, the algorithm may not be able to finish running within the maximum allowed time on a computing cluster. To avoid these numerical issues and to be able to finish in time, one might consider screening the predictors to get a smaller number of candidates before selecting variables. One could fit a univariate Fine-Gray model on each of the predictors, calculate its p-value and select those predictors with smaller p-values, possibly controlling

the false discovery rate, for subsequent variable selection. Li, Mei and Tian (2018), Chen et. al. (2020) and Chen et al. (2022) proposed screening methods for competing risk data, which could also be used [63, 15, 16].

After selecting variables, it would be nice if we could perform inference on their corresponding coefficients in the model, constructing confidence intervals and testing hypotheses. Fang, Ning and Liu (2017), Taylor and Tibshirani (2018), Yu, Bradic and Samworth (2021) and Kong et al. (2021) studied post-selection inference in Cox model [30, 91, 102, 56]. Hou, Bradic and Xu (2019) studied post-selection inference in Fine-Gray model with LASSO penalty [46]. Future work should include developing a method for nonconvex penalty functions.

This dissertation focused on variable selection. As demonstrated by application to the complete AML data, none of the proposed methods, CoxBoost package and fastcmprsk package seems very accurate in terms of prediction. If one is more interested in prediction, the randomForestSRC package or some other machine learning-based method might be worth trying.

Note that a patient will still die after relapse. Time to relapse and time to death are collectively known as semi-competing risks data. If both events are of interest, one could employ one of the models developed for semi-competing risks data in the literature to analyze them. Future work should also include development of a variable selection method for semi-competing risks data.

Note that in the AML dataset, there are some patients who did not achieve complete remission. Their ultimate fate would be death without complete remission. One could also include them for an analysis using a multi-state model. Variable selection could also be performed for each transition of states.

References

[1] Odd O Aalen and Søren Johansen. "An empirical transition matrix for non-homogeneous Markov chains based on censored observations". In: *Scandinavian Journal of Statistics* (1978), pp. 141–150.

[2] Hatice Gumushan Aktas and Huda Ayan. "Oleuropein: A potential inhibitor for prostate cancer cell motility by blocking voltage-gated sodium channels". In: *Nutrition and Cancer* 73.9 (2021), pp. 1758–1767.

[3] Federico Ambrogi and Thomas H Scheike. "Penalized estimation for competing risks regression with applications to high-dimensional covariates". In: *Biostatistics* 17.4 (2016), pp. 708–721.

[4] Simon Anders, Paul Theodor Pyl, and Wolfgang Huber. "HTSeq—a Python framework to work with high-throughput sequencing data". In: *Bioinformatics* 31.2 (2015), pp. 166–169.

[5] Stefanos A Bamopoulos et al. "Clinical presentation and differential splicing of SRSF2, U2AF1 and SF3B1 mutations in patients with acute myeloid leukemia". In: *Leukemia* 34.10 (2020), pp. 2621–2634.

[6] Anna Bellach et al. "Weighted NPMLE for the subdistribution of a competing risk". In: *Journal of the American Statistical Association* 114.525 (2019), pp. 259–270.

[7] Deepa Bhojwani et al. “Methotrexate-Induced Neurotoxicity and Leukoencephalopathy in Childhood Acute Lymphoblastic Leukemia”. In: *Journal of Clinical Oncology* 32.9 (2014), pp. 949–959.

[8] Harald Binder et al. “Boosting for high-dimensional time-to-event data with competing risks”. In: *Bioinformatics* 25.7 (2009), pp. 890–896.

[9] Paul Blanche et al. “Quantifying and comparing dynamic predictive accuracy of joint models for longitudinal marker and time-to-event in presence of censoring and competing risks”. In: *Biometrics* 71.1 (2015), pp. 102–113.

[10] Jelena Bradic, Jianqing Fan, and Jiancheng Jiang. “Regularization for Cox’s proportional hazards model with NP-dimensionality”. In: *Annals of Statistics* 39.6 (2011), p. 3092.

[11] Patrick Breheny and Jian Huang. “Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection”. In: *The Annals of Applied Statistics* 5.1 (2011), p. 232.

[12] Leo Breiman. “Random forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32.

[13] I-Shou Chang et al. “Non-parametric maximum-likelihood estimation in a semiparametric mixture model for competing-risks data”. In: *Scandinavian Journal of Statistics* 34.4 (2007), pp. 870–895.

[14] Qixuan Chen and Sijian Wang. “Variable selection for multiply-imputed data with application to dioxin exposure study”. In: *Statistics in Medicine* 32.21 (2013), pp. 3646–3659.

[15] Xiaolin Chen et al. “Model-free feature screening for ultra-high dimensional competing risks data”. In: *Statistics & Probability Letters* 164 (2020), p. 108815.

[16] Xiaolin Chen et al. "On correlation rank screening for ultra-high dimensional competing risks data". In: *Journal of Applied Statistics* 49.7 (2022), pp. 1848–1864.

[17] Sangbum Choi and Xuelin Huang. "Maximum likelihood estimation of semiparametric mixture component models for competing risks data". In: *Biometrics* 70.3 (2014), pp. 588–598.

[18] Sangbum Choi et al. "Weighted least-squares regression with competing risks data". In: *Statistics in Medicine* 41.2 (2022), pp. 227–241.

[19] David R Cox. "Regression models and life-tables". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 34.2 (1972), pp. 187–202.

[20] James KJ Diss et al. "Expression profiles of voltage-gated Na+ channel $\alpha$-subunit genes in rat and human prostate cancer cell lines". In: *The Prostate* 48.3 (2001), pp. 165–178.

[21] Alexander Dobin et al. "STAR: ultrafast universal RNA-seq aligner". In: *Bioinformatics* 29.1 (2013), pp. 15–21.

[22] T Roderick Docking et al. "A clinical transcriptome approach to patient stratification and therapy selection in acute myeloid leukemia". In: *Nature Communications* 12.1 (2021), p. 2474.

[23] Hartmut Döhner et al. "Diagnosis and management of AML in adults: 2017 ELN recommendations from an international expert panel". In: *Blood, The Journal of the American Society of Hematology* 129.4 (2017), pp. 424–447.

[24] Jiacong Du et al. "Variable selection with multiply-imputed datasets: choosing between stacked and grouped methods". In: *Journal of Computational and Graphical Statistics* (2022), pp. 1–13.

[25] Thomas Dunwell et al. “A Genome-Wide Screen Identifies Frequently Methylated Genes in Haematological and Epithelial Cancers”. In: *Molecular Cancer* 9 (Feb. 2010), p. 44.

[26] Frank Eriksson et al. “The proportional odds cumulative incidence model for competing risks”. In: *Biometrics* 71.3 (2015), pp. 687–695.

[27] Gabriel Escarela and Russell J Bowater. “Fitting a semi-parametric mixture model for competing risks in survival data”. In: *Communications in Statistics—Theory and Methods* 37.2 (2008), pp. 277–293.

[28] Jianqing Fan and Runze Li. “Variable selection for Cox’s proportional hazards model and frailty model”. In: *The Annals of Statistics* 30.1 (2002), pp. 74–99.

[29] Jianqing Fan and Runze Li. “Variable selection via nonconcave penalized likelihood and its oracle properties”. In: *Journal of the American Statistical Association* 96.456 (2001), pp. 1348–1360.

[30] Ethan X Fang, Yang Ning, and Han Liu. “Testing and confidence intervals for high dimensional proportional hazards models”. In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* (2017), pp. 1415–1437.

[31] Jason P Fine. “Regression modeling of competing crude failure probabilities”. In: *Biostatistics* 2.1 (2001), pp. 85–97.

[32] Jason P Fine and Robert J Gray. “A proportional hazards model for the subdistribution of a competing risk”. In: *Journal of the American Statistical Association* 94.446 (1999), pp. 496–509.

[33] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. “Regularization paths for generalized linear models via coordinate descent”. In: *Journal of Statistical Software* 33.1 (2010), p. 1.

[34] Jerome Friedman et al. "Pathwise coordinate optimization". In: *The Annals of Applied Statistics* 1.2 (2007), pp. 302–332.

[35] Zhixuan Fu, Chirag R Parikh, and Bingqing Zhou. "Penalized variable selection in competing risks regression". In: *Lifetime Data Analysis* 23.3 (2017), pp. 353–376.

[36] Miaomiao Ge and Ming-Hui Chen. "Bayesian inference of the fully specified subdistribution model for survival data with competing risks". In: *Lifetime Data Analysis* 18.3 (2012), pp. 339–363.

[37] Thomas A Gerds, Thomas H Scheike, and Per K Andersen. "Absolute risk regression for competing risks: interpretation, link functions, and prediction". In: *Statistics in Medicine* 31.29 (2012), pp. 3921–3930.

[38] Thomas A Gerds and Martin Schumacher. "Consistent estimation of the expected Brier score in general survival models with right-censored event times". In: *Biometrical Journal* 48.6 (2006), pp. 1029–1040.

[39] Thomas A. Gerds and Michael W. Kattan. *Medical Risk Prediction Models: With Ties to Machine Learning (1st ed.)* Chapman and Hall/CRC, 2021.

[40] Jelle J Goeman. "L1 penalized estimation in the Cox proportional hazards model". In: *Biometrical Journal* 52.1 (2010), pp. 70–84.

[41] Erika Graf et al. "Assessment and comparison of prognostic classification schemes for survival data". In: *Statistics in Medicine* 18.17-18 (1999), pp. 2529–2545.

[42] Robert J Gray. "A class of K-sample tests for comparing the cumulative incidence of a competing risk". In: *The Annals of Statistics* (1988), pp. 1141–1154.

[43] Peng He et al. "A proportional hazards regression model for the subdistribution with covariates-adjusted censoring weight for competing risks data". In: *Scandinavian Journal of Statistics* 43.1 (2016), pp. 103–122.

[44] Sally R Hinchliffe and Paul C Lambert. "Flexible parametric modelling of cause-specific hazards to estimate cumulative incidence functions". In: *BMC Medical Research Methodology* 13.1 (2013), pp. 1–14.

[45] JD Holt. "Competing risk analyses with special reference to matched pair experiments". In: *Biometrika* 65.1 (1978), pp. 159–165.

[46] Jue Hou, Jelena Bradic, and Ronghui Xu. "Inference under Fine-Gray competing risks model with high-dimensional covariates". In: *Electronic Journal of Statistics* 13.2 (2019), pp. 4449–4507.

[47] H. Ishwaran and U.B. Kogalur. *Fast Unified Random Forests for Survival, Regression, and Classification (RF-SRC)*. R package version 3.1.0. manual, 2022. URL: `https://cran.r-project.org/package=randomForestSRC`.

[48] Hemant Ishwaran et al. "High-dimensional variable selection for survival data". In: *Journal of the American Statistical Association* 105.489 (2010), pp. 205–217.

[49] Hemant Ishwaran et al. "Random survival forests". In: *The Annals of Applied Statistics* 2.3 (2008), pp. 841–860.

[50] Hemant Ishwaran et al. "Random survival forests for competing risks". In: *Biostatistics* 15.4 (2014), pp. 757–773.

[51] Jong-Hyeon Jeong and Jason P Fine. "Parametric regression on cumulative incidence function". In: *Biostatistics* 8.2 (2007), pp. 184–196.

[52] John D Kalbfleisch and Ross L Prentice. "Marginal likelihoods based on Cox's regression and life model". In: *Biometrika* 60.2 (1973), pp. 267–278.

[53] Eric S Kawaguchi et al. "Scalable algorithms for large competing risks data". In: *Journal of Computational and Graphical Statistics* 30.3 (2021), pp. 685–693.

[54] John P Klein. "Modelling competing risks in cancer studies". In: *Statistics in Medicine* 25.6 (2006), pp. 1015–1034.

[55] John P Klein and Per Kragh Andersen. "Regression modeling of competing risks data based on pseudovalues of the cumulative incidence function". In: *Biometrics* 61.1 (2005), pp. 223–229.

[56] Shengchun Kong et al. "High-dimensional robust inference for Cox regression models using desparsified Lasso". In: *Scandinavian Journal of Statistics* 48.3 (2021), pp. 1068–1095.

[57] Alexander Kowarik and Matthias Templ. "Imputation with the R Package VIM". In: *Journal of Statistical Software* 74 (2016), pp. 1–16.

[58] Stephan W Lagakos. "A covariate model for partially censored data subject to competing causes of failure". In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 27.3 (1978), pp. 235–241.

[59] Ranjit Lall and Thomas Robinson. "The MIDAS touch: Accurate and scalable missing-data imputation with deep learning". In: *Political Analysis* 30.2 (2022), pp. 179–196.

[60] Paul C Lambert, Sally R Wilkes, and Michael J Crowther. "Flexible parametric modelling of the cause-specific cumulative incidence function". In: *Statistics in Medicine* 36.9 (2017), pp. 1429–1446.

[61] Martin G Larson and Gregg E Dinse. "A mixture model for the regression analysis of competing risks data". In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 34.3 (1985), pp. 201–211.

[62] Charity W Law et al. "voom: Precision weights unlock linear model analysis tools for RNA-seq read counts". In: *Genome Biology* 15.2 (2014), pp. 1–17.

[63] E Li, B Mei, and M Tian. "Feature screening based on ultrahigh dimensional competing risks models (in Chinese)". In: *Sci. Sinica Math* 48 (2018), pp. 1061–1086.

[64] Erqian Li, Maozai Tian, and Man-Lai Tang. "Variable selection in competing risks models based on quantile regression". In: *Statistics in Medicine* 38.23 (2019), pp. 4670–4685.

[65] Erqian Li et al. "Weighted Competing Risks Quantile Regression Models and Variable Selection". In: *Mathematics* 11.6 (2023), p. 1295.

[66] Wanxing Li, Xiaoming Xue, and Yonghong Long. "An additive subdistribution hazard model for competing risks data". In: *Communications in Statistics-Theory and Methods* 46.23 (2017), pp. 11667–11687.

[67] Ying Liu et al. "Variable selection and prediction with incomplete high-dimensional data". In: *The Annals of Applied Statistics* 10.1 (2016), p. 418.

[68] Qi Long and Brent A Johnson. "Variable selection in the presence of missing data: resampling and imputation". In: *Biostatistics* 16.3 (2015), pp. 596–610.

[69] Wenbin Lu and Limin Peng. "Semiparametric analysis of mixture regression models with competing risks data". In: *Lifetime Data Analysis* 14.3 (2008), pp. 231–252.

[70] Jinchi Lv and Yingying Fan. "A unified approach to model selection and sparse recovery using regularized least squares". In: *The Annals of Statistics* 37.6A (2009), pp. 3498–3528.

[71] Lu Mao and DY Lin. "Efficient estimation of semiparametric transformation models for the cumulative incidence of competing risks". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 79.2 (2017), pp. 573–587.

[72] Marcel Martin. "Cutadapt removes adapter sequences from high-throughput sequencing reads". In: *EMBnet. journal* 17.1 (2011), pp. 10–12.

[73] Klaus H Metzeler et al. “An 86-probe-set gene-expression signature predicts survival in cytogenetically normal acute myeloid leukemia”. In: *Blood, The Journal of the American Society of Hematology* 112.10 (2008), pp. 4193–4201.

[74] T Nakajima et al. “Eicosapentaenoic acid inhibits voltage-gated sodium channels and invasiveness in prostate cancer cells”. In: *British Journal of Pharmacology* 156.3 (2009), pp. 420–431.

[75] SK Ng and GJ McLachlan. “An EM-based semi-parametric mixture model approach to the regression analysis of competing-risks data”. In: *Statistics in Medicine* 22.7 (2003), pp. 1097–1111.

[76] Ahmadreza Niavarani et al. “A 4-gene expression score associated with high levels of Wilms Tumor-1 (WT 1) expression is an adverse prognostic factor in acute myeloid leukaemia”. In: *British Journal of Haematology* 172.3 (2016), pp. 401–411.

[77] Mee Young Park and Trevor Hastie. “L1-regularization path algorithm for generalized linear models”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 69.4 (2007), pp. 659–677.

[78] Limin Peng and Jason P Fine. “Competing risks quantile regression”. In: *Journal of the American Statistical Association* 104.488 (2009), pp. 1440–1453.

[79] Ross L Prentice et al. “The analysis of failure times in the presence of competing risks”. In: *Biometrics* (1978), pp. 541–554.

[80] Thomas Robinson, Ranjit Lall, and Alex Stenlake. *rMIDAS: Multiple Imputation using Denoising Autoencoders*. R package version 0.4.1. 2022.

[81] Maral Saadati et al. “Prediction accuracy and variable selection for penalized cause-specific hazards models”. In: *Biometrical Journal* 60.2 (2018), pp. 288–306.

[82] Mahmood Salesi et al. “An expectation-conditional maximization-based Weibull-Gompertz mixture model for analyzing competing-risks data: Using post-transplant malignancy data”. In: *Journal of Biostatistics and Epidemiology* 2.1 (2016), pp. 1–8.

[83] Rotraut Schoop et al. “Quantifying the predictive accuracy of time-to-event models in the presence of competing risks”. In: *Biometrical Journal* 53.1 (2011), pp. 88–112.

[84] Haiwen Shi, Yu Cheng, and Jong-Hyeon Jeong. “Constrained parametric model for simultaneous inference of two cumulative incidence functions”. In: *Biometrical Journal* 55.1 (2013), pp. 82–96.

[85] Noah Simon et al. “Regularization paths for Cox’s proportional hazards model via coordinate descent”. In: *Journal of Statistical Software* 39.5 (2011), p. 1.

[86] Daniel J Stekhoven and Peter Bühlmann. “MissForest—non-parametric missing value imputation for mixed-type data”. In: *Bioinformatics* 28.1 (2012), pp. 112–118.

[87] Daniel J. Stekhoven. *missForest: Nonparametric Missing Value Imputation using Random Forest*. R package version 1.5. 2022.

[88] Yu-Sung Su et al. “Multiple imputation with diagnostics (mi) in R: Opening windows into the black box”. In: *Journal of Statistical Software* 45 (2011), pp. 1–31.

[89] Han Sun and Xiaofeng Wang. “High-dimensional feature selection in competing risks modeling: A stable approach using a split-and-merge ensemble algorithm”. In: *Biometrical Journal* (2022).

[90] Leili Tapak et al. “Regularized Weighted Nonparametric Likelihood Approach for High-Dimension Sparse Subdistribution Hazards Model for Competing Risk Data”. In: *Computational and Mathematical Methods in Medicine* 2021 (2021).

[91] Jonathan Taylor and Robert Tibshirani. “Post-selection inference for l1-penalized likelihood models”. In: *Canadian Journal of Statistics* 46.1 (2018), pp. 41–61.

[92] Matthias Templ, Alexander Kowarik, and Peter Filzmoser. "Iterative stepwise regression imputation using standard and robust methods". In: *Computational Statistics & Data Analysis* 55.10 (2011), pp. 2793–2806.

[93] Robert Tibshirani. "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1 (1996), pp. 267–288.

[94] Robert Tibshirani. "The lasso method for variable selection in the Cox model". In: *Statistics in Medicine* 16.4 (1997), pp. 385–395.

[95] Stef Van Buuren and Karin Groothuis-Oudshoorn. "mice: Multivariate imputation by chained equations in R". In: *Journal of Statistical Software* 45 (2011), pp. 1–67.

[96] Alejandro R Vásquez and Gabriel Escarela. "Parametric and semiparametric copula-based models for the regression analysis of competing risks". In: *Communications in Statistics-Theory and Methods* 50.12 (2021), pp. 2831–2847.

[97] Sarah Wagner et al. "A parsimonious 3-gene signature predicts clinical outcomes in an acute myeloid leukemia multicohort study". In: *Blood Advances* 3.8 (2019), pp. 1330–1346.

[98] Y Wan et al. "Variable selection models based on multiple imputation with an application for predicting median effective dose and maximum effect". In: *Journal of Statistical Computation and Simulation* 85.9 (2015), pp. 1902–1916.

[99] Susan P Whitman et al. "GAS6 expression identifies high-risk adult AML patients: potential implications for therapy". In: *Leukemia* 28.6 (2014), pp. 1252–1258.

[100] Marcel Wolbers et al. "Concordance for prognostic models with competing risks". In: *Biostatistics* 15.3 (2014), pp. 526–539.

[101] Jianling Xia et al. “Voltage-gated sodium channel Nav1. 7 promotes gastric cancer progression through MACC1-mediated upregulation of NHE1”. In: *International Journal of Cancer* 139.11 (2016), pp. 2553–2569.

[102] Yi Yu, Jelena Bradic, and Richard J Samworth. “Confidence intervals for high-dimensional Cox models”. In: *Statistica Sinica* 31.1 (2021), pp. 243–267.

[103] Faisal Maqbool Zahid, Shahla Faisal, and Christian Heumann. “Variable selection techniques after multiple imputation in high-dimensional data”. In: *Statistical Methods & Applications* 29.3 (2020), pp. 553–580.

[104] Cun-Hui Zhang. “Nearly unbiased variable selection under minimax concave penalty”. In: *The Annals of Statistics* 38.2 (2010), pp. 894–942.

[105] Hao Zhang et al. “Effects of Intravenous Infusion of Lidocaine on Short-Term Outcomes and Survival in Patients Undergoing Surgery for Ovarian Cancer: A Retrospective Propensity Score Matching Study”. In: *Frontiers in Oncology* 11 (2022), p. 5609.