Hindawi
Computational Intelligence and Neuroscience
Volume 2017, Article ID 2747431, 10 pages
https://doi.org/10.1155/2017/2747431

Research Article

Ensembling Variable Selectors by Stability Selection for the Cox Model

Qing-Yan Yin,[1] Jun-Li Li,[2] and Chun-Xia Zhang[2]

[1]*School of Science, Xi'an University of Architecture and Technology, Xi'an, Shaanxi 710055, China*
[2]*School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China*

Correspondence should be addressed to Qing-Yan Yin; qingyanyin@outlook.com

Received 13 May 2017; Revised 18 August 2017; Accepted 29 October 2017; Published 15 November 2017

Academic Editor: Paolo Gastaldo

As a pivotal tool to build interpretive models, variable selection plays an increasingly important role in high-dimensional data analysis. In recent years, variable selection ensembles (VSEs) have gained much interest due to their many advantages. Stability selection (Meinshausen and Bühlmann, 2010), a VSE technique based on subsampling in combination with a base algorithm like lasso, is an effective method to control false discovery rate (FDR) and to improve selection accuracy in linear regression models. By adopting lasso as a base learner, we attempt to extend stability selection to handle variable selection problems in a Cox model. According to our experience, it is crucial to set the regularization region Λ in lasso and the parameter $\lambda_{\min}$ properly so that stability selection can work well. To the best of our knowledge, however, there is no literature addressing this problem in an explicit way. Therefore, we first provide a detailed procedure to specify Λ and $\lambda_{\min}$. Then, some simulated and real-world data with various censoring rates are used to examine how well stability selection performs. It is also compared with several other variable selection approaches. Experimental results demonstrate that it achieves better or competitive performance in comparison with several other popular techniques.

1. Introduction

Variable selection is a classical problem in statistics and has enjoyed increased attention in recent years due to a massive growth of high-dimensional data across many scientific disciplines. In modern statistical applications, the number of variables or covariates p often exceeds the number of observations n. In such settings, the true model is often assumed to be sparse, in the sense that only a small proportion of the p variables actually relates to the response. Thus, variable selection is fundamentally important in statistical analysis of high-dimensional data. With a proper selection method and under suitable conditions, we are able to build a good model to interpret the relationship between covariates and our interested outcome more easily, to avoid overfitting in prediction and estimation, and to identify important variables for applications or further study.

For variable selection, many researchers focus on multiple linear regression models. To emphasize that variable selection methods are useful for other statistical models as well, we use a different statistical model, that is, a Cox's proportional hazards model (abbreviated as Cox model) [1], as the platform in this context. The Cox model was first proposed for exploring the relationship between the survival of a patient and some explanatory variables. As a matter of fact, the Cox model [2, 3] nowadays is one of the most commonly used forms in semiparametric models and it can not only solve the issues of censored data, but also analyze the influence of various factors on survival time simultaneously. A brief mathematical description of the Cox model is given as follows.

Suppose that there are n observations $\{(y_i, \mathbf{x}_i, \delta_i)\}_{i=1}^n$ of survival data. For an individual i, y_i denotes its survival time and $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$ stands for the observed data for the p covariates. At the same time, $\delta_i \in \{0, 1\}$ is a censoring indicator variable, where $\delta_i = 0$ means that y_i is right-censored. Let $h(t)$ be the hazard rate at time t; the generic form of a Cox's proportional hazards model can be expressed as

$$h(t \mid \mathbf{x}) = h_0(t) \exp\left(\mathbf{x}^T \boldsymbol{\beta}\right), \tag{1}$$

where $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)^T$ is a p-dimensional unknown coefficient vector and $h_0(t)$ is the baseline hazard function, that is, the hazard function at time t when all covariates take value zero. In general, $\boldsymbol{\beta}$ can be estimated by maximizing partial likelihood function. For convenience, we assume $h_0(t) = 1$ below.

Like linear regression models, traditional methods such as subset selection [4, 5], forward selection, backward elimination, and a combination of both are among the most common methods for selecting variables in a Cox model. However, these methods will have difficulty in computation when faced with high-dimensional data. Therefore, some other methods have been proposed to overcome this problem. After lasso (least absolute shrinkage and selection operator) [6] was first proposed for linear regression models, Tibshirani [7] extended it to the Cox model. Later on, many scholars [2, 3, 8–12] developed some penalized shrinkage techniques like SCAD [13] and adaptive lasso [14] specially for Cox models.

Although the above-mentioned variable selection methods have been shown to be successful in theoretical properties and numerous experiments, their performance strongly depends on the proper setup of the tuning parameter. On the other hand, these approaches may be unstable (especially in the situation of high-dimensional data). Breiman [15] proved that uncertainty can lead to more prediction loss. What is more important, small changes in data can result in that the same method selects different models. This makes the subsequent interpretation difficult and unreliable. In order to obtain more stable, accurate, and reliable variable selection results, ensemble learning [16, 17] is one kind of extremely potential technologies.

As a hot research topic in machine learning, ensemble learning is used more and more widely in many fields of natural science and social science in last two decades. The powerful advantages of ensemble learning lie in improving the generalization capacity and enhancing robustness in the process of learning. Its main idea is to obtain a number of different base learning machines by running some simple learning algorithm and then combine these base machines into an ensemble learning machine in some way. Generally, the base learning machines should have strong generalization capability on one side, and they should also complement each other on the other hand.

The ensemble approach for statistical modeling was first proposed for solving prediction problems, aiming to maximize *prediction accuracy*. Inspired by this idea, Zhu and Chipman [18] applied bagging ensemble approach to handle variable selection problems, aiming at maximizing *selection accuracy*. Meanwhile, they pointed out that there is much difference between "prediction ensembles" (PEs) and "variable selection ensembles" (VSEs). More recently, ensemble learning methods have attracted more attention on coping with variable selection problems since they can greatly improve the selection accuracy and lessen the risk to falsely select unimportant variables and simultaneously overcome the instability of traditional methods in the high-dimensional data analysis. Because of these benefits, there are more and more researches applying ensemble learning to variable selection and putting forward some novel approaches. As far as we know, existing VSE techniques mainly include PGA (parallel genetic algorithm) [18], stability selection [19], BSS (bagged stepwise search) [20], random lasso [21], ST2E (stochastic stepwise ensemble) [22], TCSL (tilted correlation screening learning) [23], RMSA (random splitting model averaging) [24], SCCE (stochastic correlation coefficient ensemble) [25], and PST2E (pruned stochastic stepwise ensemble) [26]. It is noteworthy that these algorithms are mainly designed for handling variable selection problems in linear regression models. Only Zhu and Fan [20] investigated the performance of BSS and PGA in the Cox model.

Through analyzing these VSE techniques, it can be found that their success primarily lies in producing multiple importance measures for each predictor. By simply averaging these measures across multiple trials, the noise variables can be more reliably distinguished from the informative ones. In this process, the strength to select important variables and the diversity between the importance measures need to be preserved simultaneously [20, 22]. Stability selection applies subsampling (or bootstrap) to a selection method like lasso to improve its performance. In fact, it is an extremely general ensemble learning technique for identifying important variables. Due to the characteristics of lasso, it is very efficient in high-dimensional situations. Another good property of stability selection is that it provides an effective way to control false discovery rate (FDR) in finite sample cases provided that its tuning parameters are set properly. Due to its versatility and flexibility, stability selection has been successfully applied in many domains such as gene expression analysis [24, 27–29]. Nevertheless, we have not found any literature about applying stability selection to a Cox model. Therefore, in this paper we would like to extend it to the situation of Cox models. At the same time, we also discuss how to set appropriate values for the involved parameters so that stability selection achieves its best performance.

The remainder of the paper is described as follows. In Section 2, the details for applying stability selection to the Cox model are described. We also provide an explicit way to set its involved parameters. In Section 3, some numerical experiments were conducted to study the impact of $\lambda_{\min}$ on the behavior of stability selection and to compare its performance with other variable selection approaches for the Cox model. In Section 4, some real examples are analyzed to further study the effectiveness of stability selection. Finally, some conclusions are offered in Section 5.

2. Stability Selection Algorithm for the Cox Model

In this paper, we consider stability selection with lasso as its base learner. Lasso [6] is one of the most effective techniques to deal with high-dimensional linear regression problems with $p > n$. With respect to its application in Cox models, the core idea is to maximize the partial likelihood minus the L_1 penalty function. For convenience, suppose that there are m unique failure times, say, $t_1 < t_2 < \cdots < t_m$, among the n observations $\{(y_i, \mathbf{x}_i, \delta_i)\}_{i=1}^n$. Let $j(i)$ denote the index of the

observation failing at time t_i. The lasso algorithm needs to maximize

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{m} \frac{\exp\left(\mathbf{x}_{j(i)}^T \boldsymbol{\beta}\right)}{\sum_{j \in R_i} \exp\left(\mathbf{x}_j^T \boldsymbol{\beta}\right)}, \tag{2}$$

under the constraint $\sum_{j=1}^{p} |\beta_j| \leq s$. In (2), R_i is the set of indices, j, with $y_j \geq t_i$ (i.e., the observations are at risk at time t_i). Equivalently, the estimate of $\boldsymbol{\beta}$ can be obtained as

$$\widehat{\boldsymbol{\beta}} = \arg\max_{\boldsymbol{\beta}} \left[\sum_{i=1}^{m} \mathbf{x}_{j(i)}^T \boldsymbol{\beta} - \log\left(\sum_{j \in R_i} \exp\left(\mathbf{x}_j^T \boldsymbol{\beta}\right) \right) - \lambda \sum_{j=1}^{p} \left|\beta_j\right| \right], \tag{3}$$

where λ is the regularization parameter which controls the trade-off between the model fitting and the coefficient shrinkage degree. At present, there are several efficient algorithms [7, 30] (such as cyclical coordinate descent) to get $\widehat{\boldsymbol{\beta}}$ in (3). We refer readers to the related literature for more details about the optimization strategy.

In applications, we need to first set a sensible region, say, $\Lambda = [\lambda_{\text{lower}}, \lambda_{\text{upper}}]$, for the regularization parameter λ in lasso. Notice that lasso will choose *all* variables (i.e., full model) for $\lambda \leq \lambda_{\text{lower}}$ while choosing *none* of the variables (i.e., null model) for $\lambda \geq \lambda_{\text{upper}}$. By taking K candidate values in Λ, that is, $\lambda_{\text{lower}} = \lambda_1 < \lambda_2 < \cdots < \lambda_K = \lambda_{\text{upper}}$, lasso generally employs 5-fold or 10-fold cross-validation to select an optimal value of λ, say λ_{opt}. Then, the variables which have nonzero coefficient estimation under λ_{opt} are determined as important variables. Although lasso with λ_{opt} being specified in this way has good prediction performance, much evidence [14, 19, 21] has shown that it tends to choose more variables than necessary (i.e., higher FDR).

To eliminate this drawback of lasso, Meinshausen and Bühlmann [19] developed stability selection which works by choosing variables whose *selection probabilities* are large as important ones. In reality, the selection probability can be estimated by running lasso on multiple different sets. These sets can be obtained via subsampling from the given set. Specifically, stability selection first estimates the probability that variable X_j ($j = 1, 2, \ldots, p$) is important for each regularization parameter $\lambda_1, \ldots, \lambda_K$, and then takes the maximum probability over $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_K\}$ as the important measure for X_j. Eventually, it selects important variables by a preset threshold π_{thr}. The detailed steps of stability selection algorithm for the Cox model are listed in Algorithm 1.

As argued by Meishausen and Bühlmann [19], the prominent advantage of stability selection is to control FDR under finite sample size and simultaneously to weaken the theoretical assumptions that are required to achieve variable selection consistency (i.e., the probability that the fitted model includes only truly important variables is tending to one when $n \to \infty$). Let V be the number of falsely selected variables with stability selection; Meinshausen and Bühlmann [19] have proved that, under some mild assumptions, for arbitrary $\pi_{\text{thr}} \in (1/2, 1)$, the expectation of V satisfies

$$E(V) \leq \frac{1}{2\pi_{\text{thr}} - 1} \cdot \frac{q_\Lambda^2}{p}, \tag{4}$$

where q_Λ represents the average number of variables selected by base learner. Roughly speaking, we can set any two parameters of $q_\Lambda, \pi_{\text{thr}}$, and $E(V)$ and determine the remaining one according to the above inequality. For example, let $E(V) \leq 4$ and $\pi_{\text{thr}} = 0.7$; then q_Λ can be specified as $q_\Lambda = \lceil (1.6p)^{1/2} \rceil$ in which $\lceil A \rceil$ denotes taking the smallest integer larger than or equal to A. As stated in [19], π_{thr} is recommended to take value in the range of $\pi_{\text{thr}} \in [0.6, 0.9]$ and the results tend to be similar. As far as $E(V)$ is concerned, it can be set by users according to the level of FDR that they would like to control. In general, small $E(V)$ means to control FDR strictly so that less noise variables are falsely included. Nevertheless, too small $E(V)$ may cause some truly important variables omitting in the final model. On the other hand, $E(V)$ can be larger if one can accept a little higher FDR to make sure that all important variables can be included. Regarding q_Λ, it should be no less than the number of truly important variables. Because we have no means to know the number of truly important variables in advance, however, one can first specify $E(V)$ and π_{thr} and let q_Λ be determined automatically.

As mentioned earlier, the crucial role of stability selection is to reduce the FDR of lasso (i.e., to exclude noise variables more reliably). Intuitively, it is still difficult to identify the true sparse model if too much noise variables are falsely included every time. Thus, a minimum value of λ (or $\lambda_{\min}$) needs to be specified for stability selection so that every time at most q_Λ variables are chosen when $\lambda \geq \lambda_{\min}$. Subsequently, only the λ's lying in the interval $[\lambda_{\min}, \lambda_{\text{upper}}]$ are taken as candidate values of λ to implement lasso in each trial.

According to our experience, the setting of $\lambda_{\min}$ as well as Λ is crucial to the success of stability selection. However, we cannot find any detailed instruction in related literature [19, 27, 28] about how to set them. Moreover, all the existing literature related to stability selection has not discussed how to apply it in Cox models. Here, we would like to provide an explicit way to cope with this problem in the framework of Cox models. According to the proposal in [30], we can first set λ_{upper} for lasso in a Cox model as

$$\lambda_{\text{upper}} = \max_{1 \leq j \leq p} \frac{1}{n} \sum_{k=1}^{n} \omega_k x_{kj} z_k, \tag{5}$$

in which

$$\omega_k = \sum_{i \in C_k} \left[\frac{s_i - 1}{s_i^2} \right],$$

$$s_i = \sum_{j=1}^{n} \mathbb{I}\left(y_j > t_i\right), \quad C_k = \left\{ i \mid y_k > t_i,\ i = 1, 2, \ldots, n \right\}, \tag{6}$$

$$z_k = \frac{1}{\omega_k} \left[\delta_k - \sum_{i \in C_k} \frac{1}{s_i} \right].$$

Input
y: an $n \times 1$ response vector containing survival times for n observations.
$\boldsymbol{\delta}$: an $n \times 1$ vector containing censoring indicators for n observations.
X: $n \times p$ design matrix.
Λ: regularization parameter set, i.e., $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_K\}$.
B: ensemble size.
π_{thr}: a pre-set threshold.
Output: an index set $\mathcal{I}$ for selected variables.
Main process of stability selection
(1) For $b = 1, 2, \ldots, B$
(a) Randomly draw a subset $(\mathbf{X}^{(b)}, \mathbf{y}^{(b)}, \boldsymbol{\delta}^{(b)})$ of size $\lfloor n/2 \rfloor$ without replacement from $(\mathbf{X}, \mathbf{y}, \boldsymbol{\delta})$. Here, $\lfloor A \rfloor$ stands for the largest integer less than or equal to A.
(b) For each $\lambda_k \in \Lambda$, run lasso on $(\mathbf{X}^{(b)}, \mathbf{y}^{(b)}, \boldsymbol{\delta}^{(b)})$, and record the set for selected variables as $\widehat{\mathcal{S}}^{\lambda_k}_{\lfloor n/2 \rfloor, b}$ $(k = 1, 2, \ldots, K)$.
End For
(2) Estimate the probability of each variable being selected as

$$\widehat{\pi}_j = \max_{\lambda_k \in \Lambda} \{\widehat{\pi}_j^{\lambda_k}\}, \quad j = 1, 2, \ldots, p, \qquad (*)$$

where $\widehat{\pi}_j^{\lambda_k} = (1/B) \sum_{b=1}^{B} \mathbb{I}\{j \in \widehat{\mathcal{S}}^{\lambda_k}_{\lfloor n/2 \rfloor, b}\}$, and $\mathbb{I}(\cdot)$ is an indicator function, $\mathbb{I}(\cdot) = 1$ when its condition is satisfied and $\mathbb{I}(\cdot) = 0$ otherwise.
(3) Select variables which satisfy $\widehat{\pi}_j > \pi_{\text{thr}}$, i.e. $\mathcal{I} = \{j : \widehat{\pi}_j \geq \pi_{\text{thr}}\}$.

Algorithm 1: The stability selection algorithm for the Cox model.

Here, s_i is the number of subjects (observations) at risk at time t_i and C_k is the set of indices, i, with $t_i < y_k$ (i.e., the times for which observation k is still at risk). Subsequently, we can set $\lambda_{\text{lower}} = \epsilon \lambda_{\text{upper}}$ with $\epsilon = 0.05$ for $n < p$ and $\epsilon = 0.0001$ for $n \geq p$. In order to create $K + 1$ candidate values for $\lambda \in [\lambda_{\text{lower}}, \lambda_{\text{upper}}]$, we can set $\lambda_j = \lambda_{\text{upper}} (\lambda_{\text{lower}} / \lambda_{\text{upper}})^{j/K}$ for $j = 0, \ldots, K$.

Next, the parameter λ_{min} in stability selection can be determined by

$$\lambda_{\text{min}} = \arg\max_{\lambda} \left\{ \left| \lambda_{\text{upper}} - \lambda \right| : \lambda_{\text{lower}} \leq \lambda \leq \lambda_{\text{upper}}, \widehat{q}_{[\lambda, \lambda_{\text{upper}}]} = q_\Lambda \right\}. \qquad (7)$$

Equation (7) implies that λ_{min} must be chosen to ensure that lasso selects at most q_Λ variables for each $\lambda \in \Lambda = [\lambda_{\text{min}}, \lambda_{\text{upper}}]$. Specifically, one can begin with $\lambda = \lambda_{\text{upper}}$ and decrease λ gradually until lasso detects q_Λ variables as important (i.e., q_Λ variables having nonzero coefficients). The value of λ obtained at this point is exactly λ_{min} defined in (7). Then, only the candidate values lying in $[\lambda_{\text{min}}, \lambda_{\text{upper}}]$ are considered as the candidate values for λ in lasso to execute variable selection.

3. Experimental Studies

With simulated data, some experiments are conducted in this section to investigate the impact of λ_{min} on the behavior of stability selection in a Cox model and to compare it with several other variable selection approaches. In order to maintain consistency and comparability, we set ensemble size B as 200. Each simulation was run 100 times to estimate the evaluation of a method. To simplify notations, we abbreviated stability selection as StabSel. Regarding lasso, we made use of 10-fold cross-validation to determine its optimal regularization parameter.

3.1. Simulation 1: Influence of λ_{min}. Meinshausen and Bühlmann [19] stated that the threshold value π_{thr} is a tuning parameter whose influence is small as long as it is in the range of $(0.6, 0.9)$. According to our experience, λ_{min} has more significant effect in comparison with the parameter π_{thr}. When V and π_{thr} are fixed, small λ_{min} will make lasso select more variables in each path. As a result, some noise variables may be falsely considered as important ones (i.e., high false positive rate). On the other hand, the noise variables can be safely filtered out by setting a large λ_{min}. However, this may lead us to miss some signal variables (i.e., high false negative rate). Thus, λ_{min} plays a role in controlling the trade-off between false positive rate and false negative rate of StabSel. Due to this consideration, we fixed $\pi_{\text{thr}} = 0.6$ and report results for several values of λ_{min} in the first experiment.

Suppose that there are $p = 8$ variables, $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_8$, with each generated from the standard normal distribution $N(0, 1)$. Furthermore, the variables are correlated with $\rho(x_i, x_j) = 0.5^{|i-j|}$ for all $i \neq j$ $(i, j = 1, \ldots, 8)$. The response **y** was generated from an exponential distribution whose hazard function is

$$h(t \mid \mathbf{x}) = h_0(t) \exp\left(\mathbf{x}^T \boldsymbol{\beta}\right), \qquad (8)$$

where the true coefficient vector $\boldsymbol{\beta} = (3, 1.5, 0, 0, 2, 0, 0, 0)^T$. Clearly, only three variables $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_5$ are truly important

Table 1: Selection frequencies of StabSel to identify IV and UIV.

	$x_j \in$ IV ($j = 1, 2, 5$)			$x_j \in$ UIV ($j = 3, 4, 6, 7, 8$)		
	Min	Med	Max	Min	Med	Max
0% *censoring*						
$\lambda_{\min} = 0.3$	67	69	73	0	0	1
$\lambda_{\min} = 0.2$	75	77	81	0	1	3
$\lambda_{\min} = 0.1$	77	81	84	3	7	15
20% *censoring*						
$\lambda_{\min} = 0.3$	85	88	91	0	0	3
$\lambda_{\min} = 0.2$	93	99	100	1	3	6
$\lambda_{\min} = 0.1$	100	100	100	3	8	20
40% *censoring*						
$\lambda_{\min} = 0.3$	49	76	98	0	0	2
$\lambda_{\min} = 0.2$	94	98	100	0	1	6
$\lambda_{\min} = 0.1$	100	100	100	3	6	14

and the remaining ones are unimportant. We took $n = 50$ and conducted three experiments with censoring rates 0%, 20%, and 40%, respectively. For the censoring mechanism, a censoring time t_i is generated independently and uniformly from $[0, \eta]$ for each observation. If $y_i > t_i$, we replaced y_i with t_i and then let $\delta_i = 0$. Here, the parameter η was chosen to achieve some desired censoring rates. For example, $\eta = 45$ corresponds to 20% censoring rate and $\eta = 4$ corresponds to 40% censoring rate. Aiming at evaluating the performance of StabSel for a given $\lambda_{\min}$, we computed the *selection frequency* of StabSel in each case. Specifically, the selection frequency was calculated as, among 100 simulations, the minimum, median, and maximum number of times that the important and unimportant variables (*IV* and *UIV*) are selected by StabSel, respectively. Interested readers can refer to [26] for the detailed definition of selection frequency. Table 1 summarizes the results for the cases with different centering rates.

The results in Table 1 demonstrate that StabSel using a relatively large $\lambda_{\min}$ performs slightly better in excluding unimportant variables. However, the side effect is that it more likely misses some truly important variables. In other words, StabSel controls false discovery rate (or false positive rate) quite effectively with a relatively large $\lambda_{\min}$, but this will cause it to behave poorly in terms of catching important variables. To improve its selection accuracy, we must reduce $\lambda_{\min}$. Nevertheless, this inevitably allows more false discoveries. In practice, it is worthy of choosing an appropriate value for $\lambda_{\min}$ depending on whether our emphasis is more on false positive rate or false negative rate. Moreover, we need to pay more attention to the tuning of $\lambda_{\min}$ if the censoring rate is high.

3.2. Simulation 2: Performance Comparison on a Cox Model with High-Dimensional Data. In this subsection, we concentrated on applying StabSel and lasso to a Cox model with high-dimensional data. To generate the design matrix, the following two simulated datasets were generated by following the strategy in [19].

Case 1. $\mathbf{x}_k \sim N(\mathbf{0}, \mathbf{I}_n)$, where $k = 1, 2, \ldots, p$ and $p = 1000$, $n = 100$.

Case 2. $\mathbf{x}_k = f_{k,1}\phi_1 + f_{k,2}\phi_2 + \eta_k$, for $k = 1, 2, \ldots, p$, where $\phi_1, \phi_2, f_{k,1}, f_{k,2}, \eta_k \sim N(\mathbf{0}, \mathbf{I})$, and $p = 1000$, $n = 200$.

Moreover, we created sparse regression vectors by setting $\beta_k = 0$ for all $k = 1, \ldots, p$, except for a small variable set S. For all $k \in S$, we chose the coefficient β_k independently and uniformly in $[0, 1]$ and let the size $s = |S|$ varying between 4 and 10. Here, we employed the method used in Section 3.1 to achieve the censoring rates 0% and 20%. Then, a Cox model was constructed by (8).

To compare the power of StabSel and lasso to ranking variables, we adopted the strategy utilized by [19], that is, focusing on the probability that γs variables in S can be recovered correctly, where $\gamma \in \{0.1, 0.3\}$. For lasso, this means that there is a regularization parameter such that at least $\lceil \gamma s \rceil$ variables in S are selected while all variables in $N = \{1, \ldots, p\} \setminus S$ are not selected. For stability selection, it stands for the fact that $\lceil \gamma s \rceil$ variables with highest selection frequency are all in S. In this example, we fixed the threshold value $\pi_{\text{thr}} = 0.6$ and $q_\Lambda = \lceil (0.8p)^{1/2} \rceil$ to determine a proper value for $\lambda_{\min}$.

The top two subplots in Figure 1 correspond to the situation of $\gamma = 0.1$ while the bottom two subplots illustrate the results for $\gamma = 0.3$. Notice that the latter task is more challenging than the former one. When the covariates are independent in Case 1, lasso performs satisfactorily and the advantage of StabSel is not significant. In Case 2, the dominance of StabSel over lasso to identify important variables more correctly can be clearly seen, especially when faced with censored data. In the more challenging task in which more important variables are required to be ranked ahead (i.e., $\gamma = 0.3$), the superiority of StabSel is more significant. In conclusion, this experiment shows that StabSel is indeed helpful to enhance the ranking ability of lasso.

3.3. Simulation 3: Performance Comparison with Several Other Methods. Finally, we considered a simulated dataset used in [20]. There are $n = 80$ observations and $p = 20$ predictor variables. Each predictor was generated according to

$$\mathbf{x}_j = \mathbf{z} + \boldsymbol{\epsilon}_j, \quad j = 1, 2, \ldots, 20, \ \boldsymbol{\epsilon}_j, \mathbf{z} \overset{iid}{\sim} N_{80}(\mathbf{0}, \mathbf{I}). \tag{9}$$

The response vector $\mathbf{y}$ was generated from an exponential distribution with hazard function

$$h_i(t) = h_0(t) \exp\left(0.5x_{i,5} + x_{i,10} + 1.5x_{i,15}\right). \tag{10}$$

As for the variables other than $\mathbf{x}_5$, $\mathbf{x}_{10}$, $\mathbf{x}_{15}$, the coefficient is zero. Altogether, three simulation studies were conducted with censoring rates 0%, 20%, and 40%, respectively. For StabSel, we fixed $\pi_{\text{thr}} = 0.6$ and $q_\Lambda = \lceil (1.6p)^{1/2} \rceil$. As mentioned in Section 2, the number of variables that lasso selects in each trial should be at least larger than the number of truly important variables. Thus, we increased the factor multiplying p in q_Λ because p is small in this simulation. We compared it with traditional stepwise search as well as some

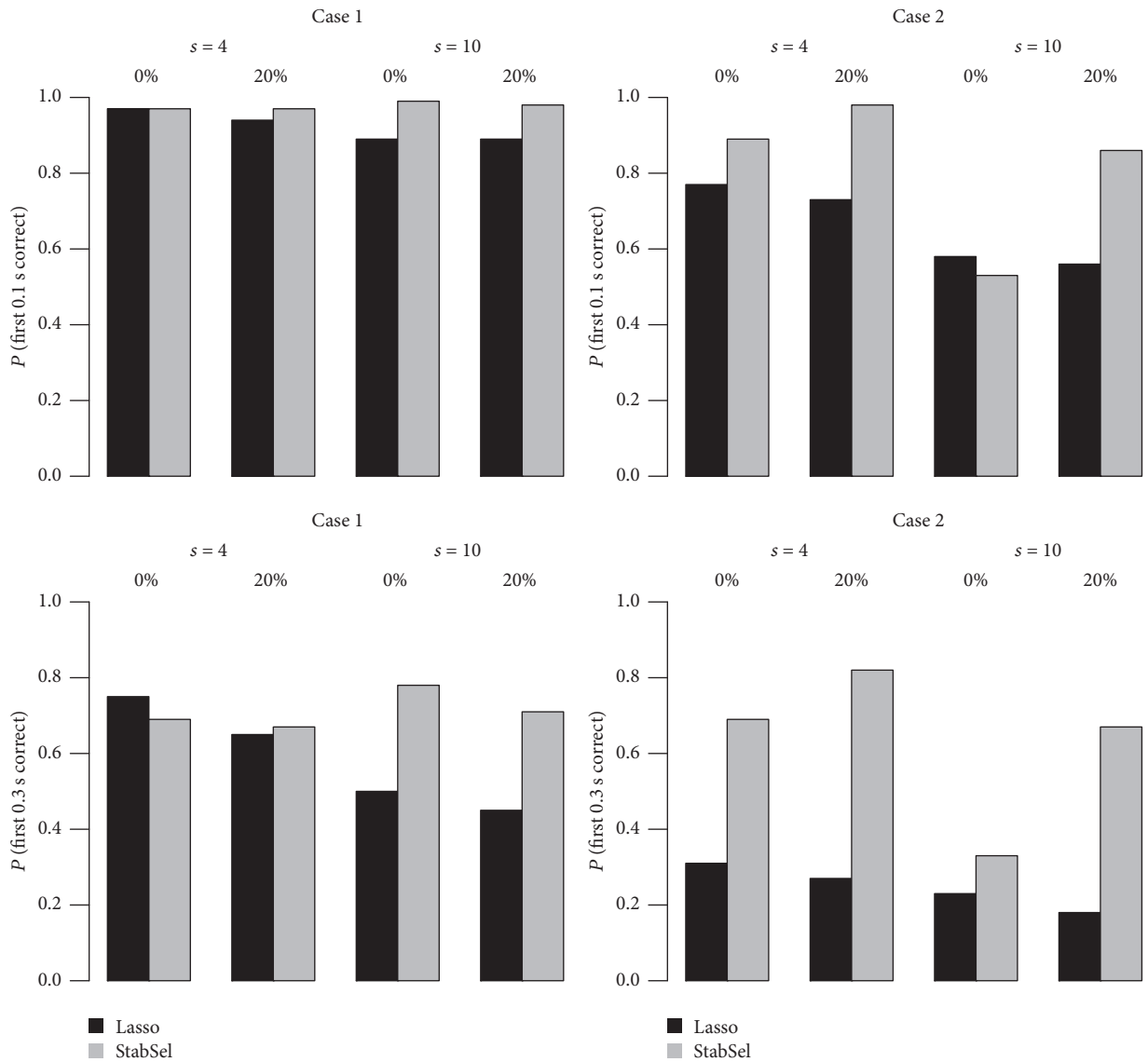

Figure 1: Selection probabilities of StabSel and lasso.

VSE techniques including BSS [20], PGA [18], RSMA [24], and ST2E [22]. The parameters involved in these methods were set according to the related literature.

Table 2 summarizes the selection frequencies of IV and UIV for each approach. The results demonstrate that although PGA performs better to exclude unimportant variables, it may miss some truly important variables. On the other hand, RSMA, ST2E, and StabSel can identify almost the same number of important variables; the difference only lies in the exclusion of unimportant ones. In this aspect, StabSel is observed to behave the best. As for BSS, its ability to guard against noise variables seems to be worse than the others although it works well to identify IVs.

In order to see more clearly the differences among the considered approaches, we computed the average *selection rate* of IV and UIV. For IV, it was computed as the selection probabilities averaged over all important variables. The metric was similarly estimated for UIV. The results are illustrated in Figure 2. The top three subplots are IVs while the bottom three ones are for UIVs. From Figure 2, we can come to some conclusions similar to those drawn from Table 2.

At the same time, we also utilized several other metrics to extensively evaluate each method. First, we computed the *selection success rate* [13]. Given an algorithm, it refers to the fraction of times among 100 runs that the algorithm correctly identifies the true model (i.e., the model only includes $\{\mathbf{x}_5, \mathbf{x}_{10}, \mathbf{x}_{15}\}$). Second, the *true positive rate* (TPR) and *true negative rate* (TNR) of each method were considered. In particular, TPR and TNR are as follows:

$$\begin{aligned} \text{TPR} &= \frac{1}{100 \cdot |\text{IV}|} \sum_{t=1}^{100} \sum_{j \in \text{IV}} \mathbb{I}\left(\widehat{\beta}_{j,t} \neq 0\right), \\ \text{TNR} &= \frac{1}{100 \cdot |\text{UIV}|} \sum_{t=1}^{100} \sum_{j \in \text{UIV}} \mathbb{I}\left(\widehat{\beta}_{j,t} = 0\right), \end{aligned} \tag{11}$$

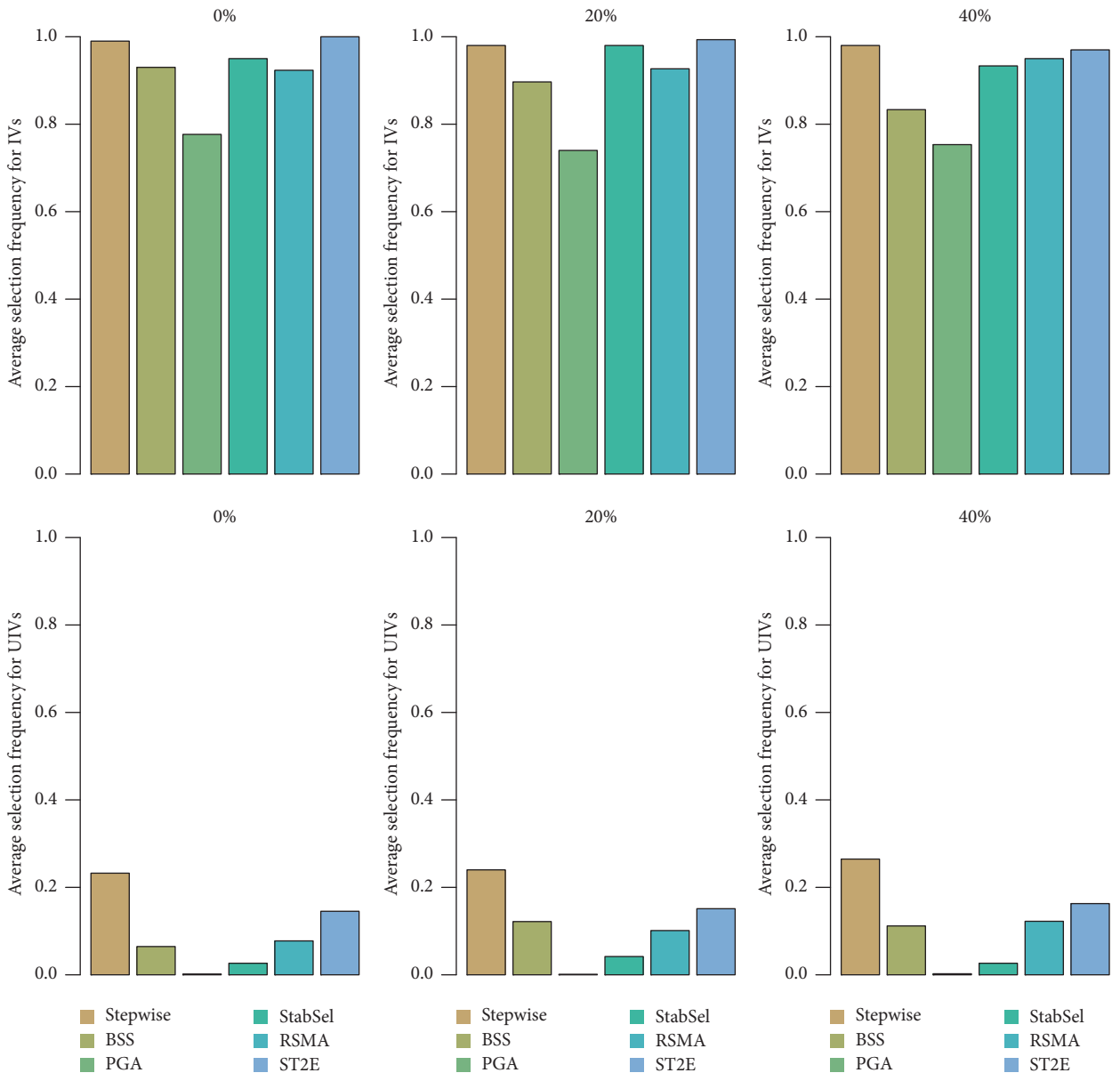

Figure 2: Average selection rate for different ensemble approaches.

where $\widehat{\boldsymbol{\beta}} = (\widehat{\beta}_{1,t}, \widehat{\beta}_{2,t}, \ldots, \widehat{\beta}_{p,t})^T$ is the estimated coefficient vector in the tth simulation. In addition, |IV| and |UIV| represent the size of IV and UIV, respectively. The method "Oracle" corresponds to fitting a Cox model with only variables $\mathbf{x}_5$, $\mathbf{x}_{10}$, and $\mathbf{x}_{15}$. Usually, a good variable selection method should produce results as close as possible to those of Oracle.

It can be seen from Table 3 that stepwise method is hopeless to select variables since it can hardly find the true model. Among the VSE algorithms, StabSel always reaches the largest selection success rate, especially when the censoring rate is high. On the other hand, StabSel tends to achieve a model size closest to that of Oracle. As far as the prediction performance is concerned, StabSel almost always outperforms the other approaches.

4. Real-World Applications

In this section, we applied the compared VSE techniques to three real-world datasets, that is, PBC [31], Lung [32], and Rats [33]. These real datasets were taken from the R package `survival`. For the original PBC and lung sets, we simply ignored the observations containing missing data. In these situations, there are no means to know which variables are truly important or not. Aiming at evaluating the selection behavior of each method, we took the original variables as truly important ones (i.e., IVs). Then, some irrelevant variables were artificially added to these sets by following the strategy used in [25, 34]. These irrelevant variables were generated from a uniform distribution on the interval [0, 1]. Table 4 lists the main characteristics of the used three datasets.

Table 2: Selection frequencies of each method in Simulation 3.

Method	$x_j \in$ IV			$x_j \in$ UIV		
	Min	Med	Max	Min	Med	Max
0% *censoring*						
Stepwise	97	100	100	13	22	30
BSS	79	100	100	3	7	10
PGA	40	93	100	0	0	1
StabSel	91	97	97	0	3	5
RSMA	79	98	100	4	8	13
ST2E	100	100	100	10	15	18
20% *censoring*						
Stepwise	94	100	100	19	24	31
BSS	70	100	100	6	12	17
PGA	29	94	100	0	0	1
StabSel	94	96	97	1	3	5
RSMA	80	98	100	4	9	17
ST2E	94	100	100	8	15	23
40% *censoring*						
Stepwise	94	100	100	22	26	38
BSS	65	89	96	8	11	15
PGA	31	95	100	0	0	1
StabSel	97	100	100	1	3	7
RSMA	80	99	100	7	13	18
ST2E	91	100	100	11	15	25

Table 3: Results for each method in Simulation 3.

Method	Succ. rate	Size	TNR	TPR
0% *censoring*				
Stepwise	0.02	6.92	0.768	0.990
BSS	0.51	3.89	0.935	0.930
PGA	0.37	2.36	0.998	0.777
StabSel	0.55	3.30	0.973	0.950
RSMA	0.21	4.09	0.922	0.923
ST2E	0.01	5.47	0.855	1.000
20% *censoring*				
Stepwise	0.04	7.02	0.760	0.980
BSS	0.31	4.76	0.879	0.900
PGA	0.28	2.25	0.999	0.743
StabSel	0.57	3.33	0.914	0.957
RSMA	0.14	4.50	0.899	0.927
ST2E	0.05	5.55	0.849	0.993
40% *censoring*				
Stepwise	0.02	7.44	0.735	0.980
BSS	0.15	4.55	0.878	0.823
PGA	0.30	2.30	0.998	0.753
StabSel	0.61	3.51	0.968	0.990
RSMA	0.06	4.87	0.878	0.930
ST2E	0.03	5.68	0.837	0.970
Oracle	1.00	3.00	1.00	1.00

Analogous to the situation of simulation studies, the ensemble size was set as $B = 200$. The parameters involved in each method were set similarly to those used in simulations. For each dataset, the experiment was repeated 100 times. In each replication, a training set was randomly drawn from the given set with size being specified in Table 4. The rest of observations was then used as a test set to evaluate the prediction performance measured with C-index [35] (i.e., concordance index). In particular, we applied each algorithm to the training set to perform variable selection. Based on the selected variables, the parameters in the corresponding model were estimated and the C-index was estimated on the test set. Table 5 shows the results obtained with each algorithm.

Table 4: Main characteristics of the used real-world datasets.

Dataset	Number of variables	Number of samples	Training size
PBC	15 (original covariates) +20 (random uniform)	276	200
Lung	8 (original covariates) +20 (random uniform)	167	100
Rats	3 (original covariates) +20 (random uniform)	300	250

In terms of selection rate, it can be observed from Table 5 that BSS performs well to identify IVs. Nevertheless, it behaves worse to exclude UIVs. On the contrary, PGA shows the lowest selected rate of UIVs while it has the lowest selected rate of IVs. Therefore, BSS and PGA are not ideal selection methods. For the remaining methods, StabSel, RSMA, and ST2E behave similarly in identifying IVs. But when compared with StabSel, RSMA and ST2E include more irrelevant variables. In conclusion, StabSel achieves better performance on variable selection when being evaluated with selection rate.

Furthermore, the results of C-index in Table 5 reveal that the prediction performance of StabSel is competitive although it is not the best one. Furthermore, almost all ensemble methods tend to have low TPR values in these three real datasets. This is largely due to the fact that we directly consider all the original covariates as IVs among which some are actually uninformative.

5. Conclusions

As an ensemble method, StabSel [19] is the marriage of subsampling with a variable selection algorithm such as lasso. Due to its property of controlling false discovery rate, StabSel has a flexible manner to choose a proper amount of regularization. Another superiority of StabSel over lasso is that it requires less assumptions to achieve variable selection consistency. In this article, we extended StabSel to the Cox model. The specification of $\lambda_{\min}$ significantly affects the performance of StabSel since it controls the balance between false positive rate and false negative rate. We provide an explicit way to set a proper value for $\lambda_{\min}$ in the situation of Cox models. In comparison with other VSE techniques including PGA, BSS, RSMA, and ST2E, StabSel exhibits better selection ability to correctly identify important variables in a high-dimensional Cox model. At the same time, StabSel

Table 5: The performance of each method on three real datasets.

Dataset	Metric	PGA	BSS	StabSel	RSMA	ST2E
PBC	Sel. rate					
	IVs (1–15)	0.299	0.597	0.518	0.543	0.605
	UIVs (26–35)	0.015	0.291	0.053	0.074	0.120
	C-index	0.792	0.812	0.819	0.826	0.835
	TPR	0.24	0.50	0.42	0.54	0.63
	TNR	0.98	0.60	0.99	0.96	0.96
Lung	Sel. rate					
	IVs (1–8)	0.284	0.607	0.477	0.476	0.466
	UIVs (9–28)	0.077	0.426	0.097	0.289	0.200
	C-index	0.631	0.703	0.695	0.680	0.695
	TPR	0.27	0.61	0.41	0.51	0.54
	TNR	0.92	0.55	0.83	0.71	0.74
Rats	Sel. rate					
	IVs (1–3)	0.627	0.890	0.850	0.893	0.997
	UIVs (4–23)	0.043	0.332	0.101	0.160	0.159
	C-index	0.800	0.870	0.853	0.869	0.693
	TPR	0.60	0.89	0.70	0.89	1.00
	TNR	0.91	0.67	0.90	0.84	0.84

has satisfactory prediction performance. When the censoring rate is high, its advantage is even more significant. Therefore, StabSel can be considered as an alternative to explore the relationship between covariates and survival times in survival analysis.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported in part by the National Natural Science Foundation of China (Grant nos. 11671317, 11601412, and 11501438), the Natural Science Basic Research Plan in Shaanxi Province of China (Grant no. 2017JQ1034), and the Science Foundation of Xi'an University of Architecture and Technology (Grant nos. RC1438 and QN1508).

References

[1] D. R. Cox, "Regression models and life-tables," *Journal of the Royal Statistical Society*, vol. 34, no. 2, pp. 187–220, 1972.

[2] J. Fan and R. Li, "Variable selection for Cox's proportional hazards model and frailty model," *The Annals of Statistics*, vol. 30, no. 1, pp. 74–99, 2002.

[3] J. Q. Fan, Y. Feng, and Y. C. Wu, "High-dimensional variable selection for Cox's proportional hazards model," *IMS Collections*, vol. 6, pp. 70–86, 2010.

[4] D. R. Wang and Z. Z. Zhang, "Variable selection for linear regression models: a survey," *Journal of Applied Statistics and Management*, vol. 29, no. 4, pp. 615–627, 2010.

[5] A. Miller, *Subset Selection in Regression*, Chapman and Hall/CRC Press, New Work, NY, USA, 2nd edition, 2002.

[6] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[7] R. Tibshirani, "The lasso method for variable selection in the cox model," *Statistics in Medicine*, vol. 16, no. 4, pp. 385–395, 1997.

[8] H. H. Zhang and W. Lu, "Adaptive Lasso for Cox's proportional hazards model," *Biometrika*, vol. 94, no. 3, pp. 691–703, 2007.

[9] J. Huang, S. Ma, and C.-H. Zhang, "Adaptive Lasso for sparse high-dimensional regression models," *Statistica Sinica*, vol. 18, no. 4, pp. 1603–1618, 2008.

[10] A. Antoniadis, P. Fryzlewicz, and F. Letué, "The Dantzig Selector in Cox's Proportional Hazards Model," *Scandinavian Journal of Statistics*, vol. 37, no. 4, pp. 531–552, 2010.

[11] P. Du, S. Ma, and H. Liang, "Penalized variable selection procedure for Cox models with semiparametric relative risk," *The Annals of Statistics*, vol. 38, no. 4, pp. 2092–2117, 2010.

[12] C. Liu, Y. Liang, X.-Z. Luan et al., "The L1/2 regularization method for variable selection in the Cox model," *Applied Soft Computing*, vol. 14, pp. 498–503, 2014.

[13] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American Statistical Association*, vol. 96, no. 456, pp. 1348–1360, 2001.

[14] H. Zou, "The adaptive lasso and its oracle properties," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1418–1429, 2006.

[15] L. Breiman, "Heuristics of instability and stabilization in model selection," *The Annals of Statistics*, vol. 24, no. 6, pp. 2350–2383, 1996.

[16] L. I. Kuncheva, *Combining Pattern Classifiers, Methods and Algorithms*, John Wiley and Sons, New Jersey, NJ, USA, 2014.

[17] Z. H. Zhou, *Machine Learning*, Qinghua University Press, Beijing, China, 2016.

[18] M. Zhu and H. A. Chipman, "Darwinian evolution in parallel universes: a parallel genetic algorithm for variable selection," *Technometrics*, vol. 48, no. 4, pp. 491–502, 2006.

[19] N. Meinshausen and P. Bühlmann, "Stability selection," *Journal of the Royal Statistical Society: Series B*, vol. 72, no. 4, pp. 417–473, 2010.

[20] M. Zhu and G. Z. Fan, "Variable selection by ensembles for the Cox model," *Journal of Statistical Computation and Simulation*, vol. 81, no. 12, pp. 1983–1992, 2011.

[21] S. J. Wang, B. Nan, S. Rosset et al., "Random Lasso," *The Annals of Applied Statistics*, vol. 5, no. 1, pp. 468–485, 2011.

[22] L. Xin and M. Zhu, "Stochastic stepwise ensembles for variable selection," *Journal of Computational and Graphical Statistics*, vol. 21, no. 2, pp. 275–294, 2012.

[23] B. Lin and Z. Pang, "Tilted correlation screening learning in high-dimensional data analysis," *Journal of Computational and Graphical Statistics*, vol. 23, no. 2, pp. 478–496, 2014.

[24] B. Lin, Q. Wang, J. Zhang, and Z. Pang, "Stable prediction in high-dimensional linear models," *Statistics and Computing*, vol. 27, no. 5, pp. 1401–1412, 2017.

[25] J. Che and Y. Yang, "Stochastic correlation coefficient ensembles for variable selection," *Journal of Applied Statistics*, vol. 44, no. 10, pp. 1721–1742, 2017.

[26] C. Zhang, J. Zhang, and Q. Yin, "A ranking-based strategy to prune variable selection ensembles," *Knowledge-Based Systems*, vol. 125, pp. 13–25, 2017.

[27] B. Hofner, L. Boccuto, and M. Göker, "Controlling false discoveries in high-dimensional situations: boosting with stability selection," *BMC Bioinformatics*, vol. 16, no. 1, article 144, 2015.

[28] A. Beinrucker, Ü. Dogan, and G. Blanchard, "Extensions of stability selection using subsamples of observations and covariates," *Statistics and Computing*, vol. 26, no. 5, pp. 1059–1077, 2016.

[29] K. He, Y. Li, J. Zhu et al., "Component-wise gradient boosting and false discovery control in survival analysis with high-dimensional covariates," *Bioinformatics*, vol. 32, no. 1, pp. 50–57, 2015.

[30] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for Cox's proportional hazards model via coordinate descent," *Journal of Statistical Software* , vol. 39, no. 5, pp. 1–13, 2011.

[31] T. Therneau and P. Grambsch, *Modeling Survival Data: Extending the Cox Model*, Springer-Verlag, New York, NY, USA, 2000.

[32] C. L. Loprinzi, J. A. Laurie, H. S. Wieand et al., "Prospective evaluation of prognostic variables from patient-completed questionnaires," *Journal of Clinical Oncology*, vol. 12, no. 3, pp. 601–607, 1994.

[33] N. Mantel, N. R. Bohidar, and J. L. Ciminera, "Mantel-Haenszel analyses of litter-matched time to response data, with modifications for recovery of interlitter information," *Cancer Research*, vol. 37, no. 11, pp. 3863–3868, 1977.

[34] A. Mkhadri and M. Ouhourane, "A group VISA algorithm for variable selection," *Statistical Methods and Applications*, vol. 24, no. 1, pp. 41–60, 2015.

[35] H. Uno, T. Cai, M. J. Pencina, R. B. D'Agostino, and L. J. Wei, "On the C-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data," *Statistics in Medicine*, vol. 30, no. 10, pp. 1105–1117, 2011.