Post-Selection Inference in Cox Proportional Hazards Models

by

Carla Louw

B.Sc., Simon Fraser University, 2020

Project Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
Department of Statistics and Actuarial Science
Faculty of Science

SIMON FRASER UNIVERSITY
Fall 2022

Declaration of Committee

Name:	**Carla Louw**
Degree:	**Master of Science**
Title:	**Post-Selection Inference in Cox Proportional Hazards Models**
Committee:	**Chair: Liangliang Wang** Professor, Statistics and Actuarial Science
	Richard Lockhart Co-Supervisor Professor, Statistics and Actuarial Science
	Gary Parker Co-Supervisor Associate Professor, Statistics and Actuarial Science
	Joan Hu Examiner Professor, Statistics and Actuarial Science

Abstract

Variable selection causes the distributions of parameter estimators to be unknown and difficult to determine. To do inference after selection, conditional distributions for parameter estimators given the selected model are needed. Taylor and Tibshirani (2018) call this post-selection inference and describe an estimator of regression parameters along with the corresponding conditional distribution, making post-selection inference possible. The Polyhedral Lemma (Lee et al., 2016) is used to determine the conditional distribution of this estimator given the model selected - a truncated normal distribution. We implement Taylor and Tibshirani's (2018) method in the Cox Proportional Hazards Regression setting and do a Monte Carlo study. The results are analyzed. The method controls the level of tests and coverage of confidence intervals well – much better than unadjusted Cox Proportional Hazards techniques. Numerical difficulties in the Cox Proportional Hazards software are identified and addressed in the post-selection inference context.

Keywords: Variable Selection; LASSO; Penalized Likelihood; Model Selection; Selective Inference.

To my family and loved ones,

No words could possibly describe how grateful I am for your support throughout this degree, and specifically this project. Thank you for being there for me every step of the way; for joining me on this journey of late nights of studying, chaotic deadlines, and adjusting when surprises show up in the work. Thank you so much for everything! I love you very much!

Acknowledgements

I would like to thank my supervisor, Dr. Richard Lockhart, for all the time, effort, and intriguing conversations that were vital to the process and completion of this project. I would also like to thank my co-supervisor, Dr. Gary Parker, for taking the time to go through this process and discussing areas that required further thought and clarification. Furthermore, I would like to thank the professors in the Statistics and Actuarial Science Department at SFU, for challenging me and teaching me the skills I would need to approach this project and to continue in my exploration of statistics and data analytics.

Table of Contents

List of Tables

List of Figures

Introduction:

In a world where data collection is the most efficient it has ever been, we find ourselves with the challenge of not only determining which measured characteristics are actually important, but also finding valid methods for inference after the process of selecting important characteristics has been done. When certain measured characteristics, or variables, are selected as predictors for a model, there are necessarily other variables that are dropped. This process of keeping some variables and dropping others forces the variables kept to account for any effect that would have been accounted for by a dropped variable, which can have major impacts on models chosen and estimates within those models.

Various model selection methods have been developed over time, such as the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC). While these two methods are very similar, differing only by the penalty utilized in the equations, the AIC will have a tendency to select larger models in an effort to ensure that the correct variables are included at the risk of overfitting and the BIC will have a tendency toward smaller models due to a stricter penalty term. While AIC and BIC both use Maximum Likelihood Estimates (MLEs), an alternative method known as the LASSO method considers the Residual Sum of Squares (RSS) with an L1 norm penalty term. The LASSO method attempts to balance model size and goodness of fit by selecting a small enough model to avoid overfitting but large enough to still achieve good fit.

Even though there are a number of model selection methods to choose from, it is difficult to find inference methods with credible coverage probabilities in a post-selection setting. This is due to the changes that occur in the distributions of parameter estimators when selection is done, and the fact that it is very difficult to compute these post-selection distributions accurately. Taylor and Tibshirani (2018) suggest a method for use in regression models with general likelihoods that utilizes the results of applying a LASSO penalty to the negative log-likelihood and using this to select important predictors. These results are then used in the approach to post-selection inference developed in Lee et al. (2016) to determine confidence intervals and test hypotheses with the selected model.

In this study, we present the details of Taylor and Tibshirani's (2018) method applied to the Cox partial likelihood function in proportional hazards modelling and implement the method on a real data set. We then create simulations that mimic the real data set to explore how well the method developed by Taylor and Tibshirani (2018) behaves in various scenarios; we then try to analyze the behaviour of the method with the results of the simulated samples. While Taylor and Tibshirani (2018) focused mainly on the application of the method in a general likelihood setting, this study will focus on extending the application of the method to a Cox Proportional Hazards Regression setting in which the likelihood becomes a partial likelihood.

In Chapter 1, we provide an overview of the fundamental concepts needed to understand Taylor and Tibshirani's (2018) new method. This is followed in Chapter 2 by a detailed explanation of our implementation of Taylor and Tibshirani's (2018) approach to the Cox Proportional Hazards Regression setting. In Chapter 3 we provide details about our Monte Carlo study design and our plan for analyzing the results. We detail results being tracked and methods for solving challenges that we faced in this study. Then, we visualize the results of the Monte Carlo study and make observations in Chapter 4. Final thoughts and our key findings are summarized at the end of Chapter 4. Complete sets of graphical visuals for each experimental setting are provided in an appendix.

Chapter 1.

Before a new method can be explored, it is important that the prerequisite techniques utilized in the method are understood. This chapter provides a basic overview of key methods and concepts underlying the proposal of Taylor and Tibshirani (2018) for post-selection inference in models with a general likelihood.

First, the standard linear regression model is defined and the traditional Ordinary Least Squares method for parameter estimation is briefly described. Then the LASSO method for variable selection is addressed, as this plays a vital role in the new method. We will be assessing the proposal in the context of the Cox Proportional Hazards Regression model; therefore, we outline the nature of survival data and a few basic concepts in survival analysis. These are then put together to describe the application of LASSO to the Cox Proportional Hazards Regression model.

1.1. Linear Regression Model

Consider the data $(y_i, \boldsymbol{x}_i)$ where $i = 1 \dots N$. For the i-th observation, y_i is the dependent variable of interest and $\boldsymbol{x}_i = [x_{i1} \quad \cdots \quad x_{ip}]$ contains the independent variables. The $\boldsymbol{x}_i$ vectors come together to form the design matrix $\boldsymbol{X}$ and the y_i combine to form $\boldsymbol{y}$ as follows

$$\boldsymbol{X} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{Np} \end{bmatrix}; \ \boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$$

(1)

In the linear regression model, the random variable y_i is related to fixed values for $\boldsymbol{x}_i$ with coefficients β_j through the following equation (Devore, 2016)

$$y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i$$

(2)

where $\boldsymbol{\beta} = [\beta_0 \quad \cdots \quad \beta_p]'$ and we assume the random error $\varepsilon_i \sim N(0, \sigma^2)$ combines to form $\boldsymbol{\varepsilon} = [\varepsilon_1 \quad \cdots \quad \varepsilon_N]'$. Equation (2) can be written in matrix form as $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$. The coefficients, $\boldsymbol{\beta}$, are estimated using Ordinary Least Squares (OLS). In OLS, the goal is to minimize the residual sum of squares (RSS), which is defined as

$$RSS = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 = (\boldsymbol{y} - \boldsymbol{X\beta})'(\boldsymbol{y} - \boldsymbol{X\beta}).$$

(3)

If $\boldsymbol{X'X}$ is invertible, then RSS is minimized by the OLS estimator. In the multivariate case, the OLS result is

$$\widehat{\boldsymbol{\beta}}_{OLS} = (\boldsymbol{X'X})^{-1}\boldsymbol{X'y}.$$

(4)

1.2. The Lasso Method

When the sample size N is smaller than the number of parameters p, the OLS estimate given above is not meaningful, because the $(p + 1)$ by $(p + 1)$ matrix $\boldsymbol{X'X}$ is not invertible. The rank of $\boldsymbol{X'X}$ cannot exceed the minimum of $(p + 1)$ and N. Furthermore, if $(p + 1) > N$ then the rank of $\boldsymbol{X'X}$ is less than the dimension and the model is not identifiable. Nevertheless, problems with $p > N$ are now commonly addressed by doing variable selection, that is, by finding a set of fewer than N variables which are hoped to predict $\boldsymbol{y}$ well. Even if $N > p$, not all p variables may be truly significant and including them increases the risk of overfitting the data. In both cases, it would be beneficial to select the important variables from those measured and include only those selected in the model. A variety of methods exist; we focus on the Least Absolute Shrinkage and Selection Operator proposed by Tibshirani (1996).

The Least Absolute Shrinkage and Selection Operator (LASSO) utilizes the L1 norm as a penalty in convex optimization to fit a regression model that balances the "goodness of fit to the data… with the complexity of the model" (Taylor & Tibshirani, 2015). The penalty is implemented with a parameter λ which is "usually chosen by cross-

validation" (Taylor & Tibshirani, 2015). As stated by Taylor and Tibshirani (2015), in the general linear regression case, the estimates are given by

$$argmin_{\beta_0,\beta}\left\{\frac{1}{2}\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda\sum_{j=1}^{p}|\beta_j|\right\}. \quad (5)$$

The objective function, in braces above, balances the desire to have the RSS small while keeping the size of the parameter estimates small as well. The benefit of using the L1 norm ($\sum_{j=1}^{p}|\beta_j|$) is that, at the minimizer, some of the β_j are set exactly to 0, depending on how the penalty parameter λ is set (Tibshirani, 1997). If λ is set to a large value, then many of the β_j will be estimated as 0. Furthermore, if λ is set to a smaller, and thus less restrictive value, then fewer of the β_j will be estimated as 0.

1.3. Polyhedral Lemma and Post-Selection Inference

When the errors, ε_i, in equation (2) are normally distributed, then the OLS estimator $\widehat{\boldsymbol{\beta}}_{OLS}\sim MVN_p\left(\boldsymbol{\mu}_{\widehat{\boldsymbol{\beta}}_{OLS}},\boldsymbol{\Sigma}\right)$ where $\boldsymbol{\mu}_{\widehat{\boldsymbol{\beta}}_{OLS}}=\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}=\sigma^2(\boldsymbol{X}'\boldsymbol{X})^{-1}$. The coefficient σ^2 is estimated using the RSS evaluated at $\widehat{\boldsymbol{\beta}}_{OLS}$ divided by the degrees of freedom for error, $(N-p-1)$. These estimates can be used to form confidence intervals or test hypotheses about the true parameter vector $\boldsymbol{\beta}$ (Devore, 2016). However, model selection and penalization produce estimators of $\boldsymbol{\beta}$ which have complex distributions compared to those of OLS. When model selection is carried out using LASSO, it is much more difficult to use the resulting estimates as the basis for inference, as the distributions no longer follow the multivariate normal as described. Thus, an alternative approach to inference after selection is evidently needed. In Lockhart et al. (2014), significance testing of variables selected by LASSO in a linear regression setting is approached using, what they call, the covariance test statistic. This method considers the sequence of models selected along the path of the LASSO algorithm. Then in Tibshirani et al. (2016), this idea of post-selection inference after variables are selected through sequential procedures is expanded to include other variable selection tactics, such as forward stepwise regression, and methods of computing exact distributions for inference are discussed. In Tibshirani et al. (2016), the Polyhedral Lemma discussed by

Lee et al. (2016) is vital in these exact computations. Lee et al. (2016) refers to this as "Post-Selection Inference" and discusses how the Polyhedral Lemma can be utilized to achieve post-selection inference.

For a pre-specified value of λ, the estimator $\widehat{\boldsymbol{\beta}}$ which minimizes the LASSO objective function, equation (5), is found. The set of indices j for which the estimate of β_j is not 0 is called the (estimated) 'active' set, denoted $\widehat{M}$. Lee et al. (2016) propose to give confidence intervals and test hypotheses about the vector $\boldsymbol{\beta}^*_{\widehat{M}}$, which minimizes the mean squared error in approximating the mean vector $\boldsymbol{\mu} = E(\boldsymbol{y})$ over all $\boldsymbol{\beta}$ whose non-zero entries are a subset of the estimated 'active' set. They achieve this by considering the least squares estimate of β_j if $\boldsymbol{y}$ were regressed only on the selected variables, which are contained in the $\widehat{M}$ columns of the design matrix $\boldsymbol{X}$. The least squares estimate of β_j takes the form $\boldsymbol{\gamma}^T\boldsymbol{y}$ where the vector $\boldsymbol{\gamma}^T$ gives the row of $\left(\boldsymbol{X}_{\widehat{M}}^T\boldsymbol{X}_{\widehat{M}}\right)^{-1}\boldsymbol{X}_{\widehat{M}}^T$ corresponding to variable j. Lee et al. (2016) then show that the conditional distribution of this estimator, $\boldsymbol{\gamma}^T\boldsymbol{y}$, given a certain event, is normal, truncated to an interval which can be computed from the design matrix and the conditioning information. To be specific, they condition on the event that LASSO, with the given λ, selects the particular model chosen and also on the orthogonal complement $\left(I - \frac{\boldsymbol{\gamma}\boldsymbol{\gamma}^T}{\boldsymbol{\gamma}^T\boldsymbol{\gamma}}\right)\boldsymbol{y}$. A normal distribution with mean μ and standard deviation σ, truncated to an interval $[a, b]$, has cumulative distribution function (CDF)

$$F^{a,b}_{\mu,\sigma^2}(x) = \frac{\Phi\left(\frac{x-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)} \qquad (6)$$

where Φ is the CDF of a $N(0,1)$ random variable (Lee et al., 2016). Taylor and Tibshirani (2018) suggest an extension of this method to more general regression models. One such model is the Cox Proportional Hazards model which we describe next.

1.4. Cox Proportional Hazards Regression Model

In this study, we will be focusing specifically on implementing these post-selection inference procedures in the survival analysis setting. Survival analysis focuses

on creating, fitting, and examining models of the time it takes for an event to occur. As the name suggests, a common event in these types of analyses is death, though the theory can be applied to other types of timed events as well. The event of interest is sometimes labelled as either a hard endpoint or a soft endpoint. An example of a hard endpoint is death; an event that has a specific time stamp of occurrence, measured with little or no error. A soft endpoint has an approximate time stamp of event occurrence, such as the time noted for the recurrence of a disease; by the time the disease is detected, the time of recurrence is now approximate due to the delay.

Regardless of the type of endpoint, the reality is that these studies rely on waiting an extended period of time to observe the occurrences of these events. In addition to the challenges that are typically present in studies, there is the challenge of unexpected events interfering with results. Remaining in the healthcare setting with the event of interest being death, it is possible for patients to die from other causes, known as competing risks. For example, if the patient is killed in a car accident, then even though they have died, it was not the disease being studied that caused the death. Another possible, and relatively common, challenge is known as censoring. Censoring, usually known more specifically as right-censoring, occurs when patients leave the study or the timeline of the study comes to an end before the event of interest has an opportunity to occur. A patient may leave a study simply due to moving to a new home, which is considered to be independent of the event of death; but they could also leave the study due to negative side effects of a treatment, which may or may not be considered as independent of the event.

While it is possible to apply general linear methods on this data, these methods do not use the key portion of information gained from the aspect of time. Sir David Cox (1972) created the Cox Proportional Hazards Regression model as a way of modeling the data while utilizing as much of the relevant data as possible. Let T represent the random variable of time that has the cumulative distribution function (CDF)

$$F(t) = Pr(T \leq t) \tag{7}$$

and probability density function (PDF)

$$f(t) = \frac{d(F(t))}{dt}.$$

(8)

Using the CDF, the corresponding survival function, which is the complement of the above CDF, is defined as

$$S(t) = Pr(T > t) = 1 - F(t).$$

(9)

This survival function is used as the denominator in the hazard function. The hazard function assesses the instantaneous rate of the event (in this case death) at time t, given that the patient has survived until at least that time, as shown below

$$h(t) = \lim_{\Delta t \to 0} \frac{Pr((t \leq T \leq t + \Delta t) | T \geq t)}{\Delta t} = \frac{f(t)}{S(t)}$$

(10)

Various hazard functions are possible in order to correspond with various survival functions that may be present in data sets.

Now consider the survival data $(\boldsymbol{x}_i, y_i, \delta_i)$ where $i = 1, \ldots, N$. The vector $\boldsymbol{x}_i = [x_{i1} \quad \cdots \quad x_{ip}]$ contains the features (also known as the covariates) for the *i-th* individual. This leads to the design matrix

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Np} \end{bmatrix}$$

To accommodate right censoring we let δ_i be 1 to denote that subject i's time of death was observed and 0 if subject i was censored (Tibshirani, 1997). Note that it is assumed that the censoring time is independent of death time. The corresponding y_i denotes the end time for each individual, whether that be time of death or time of

censoring. Allow D to denote the indices of the failure times (when $\delta_i = 1$) and R_i to denote the risk set at death time t_i. This risk set is composed of the indices of individuals who have the potential for death at time t_i, which includes the individual who died at that specified time and all other individuals who have neither died nor been censored prior to time t_i.

A key assumption for the Cox Proportional Hazards Regression model is that the hazard function for subject i at time *t* takes the form

$$h_i(t|\boldsymbol{x}) = h_0(t)e^{\sum_j x_{ij}\beta_j} = h_0(t)e^{\boldsymbol{x_i\beta}} \tag{11}$$

where $h_0(t)$ is an arbitrary baseline hazard function and $\boldsymbol{\beta} = [\beta_1 \quad \cdots \quad \beta_p]'$ (Tibshirani, 1997). These hazard functions across individuals are used to create the partial likelihood function. A partial likelihood function is similar to a likelihood function, but it does not depend on all the parameters which are typically needed to fully describe a distribution; in particular it omits the model for censoring and the part of the likelihood which includes the baseline hazard $h_0(t)$. The relevant partial likelihood function (see Tibshirani, 1997) is

$$L(\boldsymbol{\beta}) = \prod_{i\in D} \frac{h_0(t)e^{\boldsymbol{x_i\beta}}}{\sum_{l\in R_i} h_0(t)e^{\boldsymbol{x_l\beta}}} = \prod_{i\in D} \frac{e^{\boldsymbol{x_i\beta}}}{\sum_{l\in R_i} e^{\boldsymbol{x_l\beta}}} \tag{12}$$

which leads to the log partial likelihood function

$$l(\boldsymbol{\beta}) = \sum_{i\in D} \ln\left(\frac{e^{\boldsymbol{x_i\beta}}}{\sum_{l\in R_i} e^{\boldsymbol{x_l\beta}}}\right) = \sum_{i\in D}\left\{\boldsymbol{x_i\beta} - \ln\left(\sum_{l\in R_i} e^{\boldsymbol{x_l\beta}}\right)\right\} \tag{13}$$

As shown, though the parameter of a base hazard function, $h_0(t)$, is technically needed to describe the hazard function $h_i(t|\boldsymbol{x})$, it ends up factoring out of the likelihood function for every individual and therefore is removed to achieve the partial likelihood function.

1.5. Applying the LASSO to Cox Proportional Hazard Regression Model

When applying the LASSO procedure to a Cox Proportional Hazards Regression Model (CoxPH model), the x_{ij} must be standardized, according to Tibshirani (1997). The standardization results in the following equations being satisfied for each feature (j) in $\boldsymbol{X}$

$$\frac{\sum_{i=1}^{N} x_{ij}}{N} = 0 \; ; \; \frac{\sum_{i=1}^{N} x_{ij}^2}{N-1} = 1 \tag{14}$$

To achieve this standardization, each column is independently recentered by subtracting the mean of the column from all observations, and then scaled by dividing by the standard deviation of the column, as shown below

$$x_{ij(Std)} = \frac{x_{ij} - \bar{x}_{.j}}{s_{.j}} \tag{15}$$

where $\bar{x}_{.j} = \frac{\sum_{i=1}^{N} x_{ij}}{N}$ and $s_{.j}^2 = \frac{\sum_{i=1}^{N} (x_{ij} - \bar{x}_{.j})^2}{N-1}$.

This process allows β from various types of variables to be comparable in the penalty term. More precisely, by standardizing, the varying units across different variables are consolidated, thus making it possible to add β from variables with differing units together in a comprehensible manner. Once the selection procedure is complete and estimations are found, the $\widehat{\boldsymbol{\beta}}$ are converted back to the appropriate units by dividing the $\widehat{\boldsymbol{\beta}}$ by the corresponding standard deviation originally used on the column.

As discussed above, in the general linear case the objective function minimizes the sum of squared residuals; however, in the CoxPH model, the objective function aims to minimize the log partial likelihood with respect to $\boldsymbol{\beta}$; in order to do model selection we also impose a penalty (Taylor & Tibshirani, 2015). Taylor and Tibshirani (2018) describe a process of combining the Newton-Raphson update step with an iterative reweighted least squares (IRLS) procedure to create a constrained weighted least squares procedure that minimizes $l(\boldsymbol{\beta})$ with a constraint to compute estimates for $\boldsymbol{\beta}$.

To simplify further algebraic computations, define $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} = [\eta_1 \quad \cdots \quad \eta_N]'$ to obtain the log partial likelihood function for the CoxPH model with respect to $\boldsymbol{\eta}$

$$l(\boldsymbol{\eta}) = \sum_{i \in D} \left\{ \eta_i - \ln\left(\sum_{l \epsilon R_i} e^{\eta_l} \right) \right\}$$

(16)

All the components necessary to form the Newton Raphson step for minimizing $l(\boldsymbol{\eta})$ are derived from this log partial likelihood. Taking the first derivative with respect to η_k leads to the Score function, as shown below.

$$U_k(\boldsymbol{\eta}) = \frac{\partial l(\boldsymbol{\eta})}{\partial \eta_k} = \sum_{i \in D} \left\{ 1(i = k) - \frac{1}{\left(\sum_{l \epsilon R_i} e^{\eta_l} \right)} e^{\eta_k} 1(k \in R_i) \right\}; k = 1 \ldots N$$

(17)

$$\boldsymbol{U}(\boldsymbol{\eta}) = \begin{bmatrix} U_1(\boldsymbol{\eta}) \\ \vdots \\ U_N(\boldsymbol{\eta}) \end{bmatrix}$$

(18)

In addition to the Score function, the Hessian will be needed for future calculations. By taking the derivative again, the negative Hessian matrix can be computed as follows.

$$H_{km}(\boldsymbol{\eta}) = -\frac{\partial U_k(\boldsymbol{\eta})}{\partial \eta_m} = -\frac{\partial^2 l(\boldsymbol{\eta})}{\partial \eta_k \partial \eta_m}$$

$$= \sum_{i \in D} \left\{ \frac{e^{\eta_m} 1(k \in R_i) 1(k = m)}{\sum_{l \epsilon R_i} e^{\eta_l}} - \frac{e^{\eta_m} 1(m \in R_i) e^{\eta_k} 1(k \in R_i)}{\left(\sum_{l \epsilon R_i} e^{\eta_l} \right)^2} \right\}$$

(19)

$$\boldsymbol{H} = \begin{bmatrix} H_{11}(\boldsymbol{\eta}) & \cdots & H_{1N}(\boldsymbol{\eta}) \\ \vdots & \ddots & \vdots \\ H_{N1}(\boldsymbol{\eta}) & \cdots & H_{NN}(\boldsymbol{\eta}) \end{bmatrix}$$

(20)

As explained by Tibshirani (1997) and further described by Taylor and Tibshirani (2018), these formulas are then combined to form the objective function, derived from a Newton Raphson step for minimizing $l(\boldsymbol{\eta})$ (subject to the constraint that $\boldsymbol{\eta}$ is in the column space of the design matrix $\mathbf{X}$) as

$$\frac{1}{2}(\boldsymbol{z} - \boldsymbol{\eta})^T \boldsymbol{W}(\boldsymbol{z} - \boldsymbol{\eta}) = \frac{1}{2}(\boldsymbol{z} - \mathbf{X}\boldsymbol{\beta})^T \boldsymbol{W}(\boldsymbol{z} - \mathbf{X}\boldsymbol{\beta}) \tag{21}$$

where

$$\boldsymbol{W} = \boldsymbol{H} = -\frac{\partial^2 l(\boldsymbol{\eta})}{\partial \eta_k \partial \eta_m} \tag{22}$$

and

$$\boldsymbol{z} = \boldsymbol{\eta} + \boldsymbol{W}^{-1}\boldsymbol{U}(\boldsymbol{\eta}) \tag{23}$$

In a Newton Raphson step, to compute a new value of $\widehat{\boldsymbol{\beta}}$ from a current value of $\widehat{\boldsymbol{\beta}}$, we compute the first and second derivatives of the log likelihood at the current value of $\widehat{\boldsymbol{\beta}}$. Then we approximate the log likelihood by a quadratic function whose maximum will be at the new value of $\widehat{\boldsymbol{\beta}}$. This maximum is found, in our case, by minimizing the right-hand side of (21).

In our problem, we are seeking to minimize a penalized version of the log likelihood, so at each step we minimize a penalized version of our quadratic approximation; see (24) below. Taylor and Tibshirani (2018) start the algorithm with a pre-specified λ and an initially specified $\widehat{\boldsymbol{\beta}} = \mathbf{0}$. The next step is to compute $\boldsymbol{W}$ and $\boldsymbol{z}$ using $\widehat{\boldsymbol{\beta}}$ and apply them to a penalized version of (21). That is, solve the constrained weighted least squares problem

$$argmin_{\boldsymbol{\beta}} G(\boldsymbol{\beta}) = argmin_{\boldsymbol{\beta}} \frac{1}{2}(\boldsymbol{z} - \mathbf{X}\boldsymbol{\beta})^T \boldsymbol{W}(\boldsymbol{z} - \mathbf{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j|$$

(24)

to find the next $\widehat{\boldsymbol{\beta}}$ value.

The minimizer is found as follows. Choose an initial $\boldsymbol{\beta}$ and compute $\boldsymbol{W}$ and $\boldsymbol{z}$ using (22) and (23). Then minimize (21) with respect to $\boldsymbol{\beta}$, holding $\boldsymbol{W}$ and $\boldsymbol{z}$ fixed. Now iterate between updating $\boldsymbol{W}$ and $\boldsymbol{z}$ using the $\widehat{\boldsymbol{\beta}}$ just found and minimizing the objective function to find an improved estimate of $\widehat{\boldsymbol{\beta}}$ until the change in estimation is less than a pre-specified threshold (Taylor & Tibshirani, 2018). The resulting $\widehat{\boldsymbol{\beta}}$ from the final iteration will be the desired LASSO estimate, denoted $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$. This process will be referred to as the Iterative Reweighted Least Squares (IRLS) approach for solving the LASSO problem.

Chapter 2.

With foundational concepts understood, the post-selection inference method from Taylor and Tibshirani (2018) can be explored. In this chapter, the details of implementing Taylor and Tibshirani's (2018) method are explained for the context of the Cox Proportional Hazards Regression model setting.

2.1 Taylor and Tibshirani's Process for Obtaining the Adjusted Estimate for β

A second estimator for $\boldsymbol{\beta}$ will be needed in order to adapt the selective inference ideas described above to the proportional hazards problem. The process for obtaining this new adjusted estimate for $\boldsymbol{\beta}$, as described by Taylor and Tibshirani (2018), begins by applying the LASSO procedure to a CoxPH model as explained above. The LASSO procedure, with a pre-specified λ, will select a set of 'active' variables, denoted by $\widehat{M}$, and a set of 'inactive' variables, denoted by $-\widehat{M}$; a variable is active if its estimated coefficient in $\widehat{\boldsymbol{\beta}}$ is not zero. Taylor and Tibshirani (2018) define the adjusted estimator $\overline{\boldsymbol{\beta}}_{\widehat{M}}$ as

$$\overline{\boldsymbol{\beta}}_{\widehat{M}} = \widehat{\boldsymbol{\beta}}_{\widehat{M}} + \boldsymbol{I}_{\widehat{M}}\left(\widehat{\boldsymbol{\beta}}_{\widehat{M}}\right)^{-1}\lambda \boldsymbol{s}_{\widehat{M}} = \widehat{\boldsymbol{\beta}}_{\widehat{M}} + \boldsymbol{I}_{\widehat{M}}\left(\widehat{\boldsymbol{\beta}}_{\widehat{M}}\right)^{-1}\frac{\partial}{\partial \boldsymbol{\beta}_{\widehat{M}}} l_{\widehat{M}}\left(\widehat{\boldsymbol{\beta}}_{\widehat{M}}\right). \quad (25)$$

where the inverse Fisher Information matrix, evaluated at $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$, is

$$\boldsymbol{I}_{\widehat{M}}\left(\widehat{\boldsymbol{\beta}}_{\widehat{M}}\right)^{-1} = \left(\boldsymbol{X}_{\widehat{M}}^{T}\boldsymbol{W}\boldsymbol{X}_{\widehat{M}}\right)^{-1} \quad (26)$$

and $\boldsymbol{s}_{\widehat{M}} = sign\left(\widehat{\boldsymbol{\beta}}_{\widehat{M}}\right)$. Here, $sign(x)$ is 1 if x is positive and -1 if x is negative. The function is applied to each component of $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$.

In the case of general linear regression, $\boldsymbol{W}$ is the identity matrix, but in the CoxPH setting, $\boldsymbol{W}$ is a much more complex matrix. In this case, as well, the estimator corresponding to $\overline{\boldsymbol{\beta}}_{\widehat{M}}$ is the OLS estimator when the response is regressed on $\boldsymbol{X}_{\widehat{M}}$. This formula for $\overline{\boldsymbol{\beta}}_{\widehat{M}}$ can be rewritten as

$$\overline{\boldsymbol{\beta}}_{\widehat{M}} = \left(\boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W} \boldsymbol{X}_{\widehat{M}}\right)^{-1} \boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W} \boldsymbol{z} = \boldsymbol{\gamma}^T \boldsymbol{z} \tag{27}$$

In general, $\boldsymbol{z}$ will take on values as specified in equation (23), but in the general linear regression case, these values simplify to $\boldsymbol{z} = \boldsymbol{y}$. Taylor and Tibshirani (2018) explain that $\overline{\boldsymbol{\beta}}_{\widehat{M}}$ will be asymptotically normally distributed; $\overline{\boldsymbol{\beta}}_{\widehat{M}} \approx N\left(\boldsymbol{\beta}^*_{\widehat{M}}, \left(\boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W} \boldsymbol{X}_{\widehat{M}}\right)^{-1}\right)$. Now that the estimator $\overline{\boldsymbol{\beta}}_{\widehat{M}}$ is obtained, as well as the corresponding $\boldsymbol{\gamma}^T$, Taylor and Tibshirani (2018) use the Polyhedral Lemma, as discussed in Chapter 1, and this normal approximation to establish a basis for inference which is intended to allow for the effects of model selection.

2.2 Polyhedral Lemma and Truncation Limits for Post-Selection Cox Proportional Hazard Inference

In Chapter 1 we described Lee et al.'s (2016) approach to post-selection inference. The goal is to get confidence intervals and hypothesis tests for a coefficient $\bar{\beta}_j$ of a variable in the estimated 'active' set $\widehat{M}$, which combine to form the vector $\overline{\boldsymbol{\beta}}_{\widehat{M}} = \boldsymbol{\gamma}^T \boldsymbol{z}$ where $\boldsymbol{\gamma}^T = \left(\boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W} \boldsymbol{X}_{\widehat{M}}\right)^{-1} \boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W}$. The least squares estimator of $\bar{\beta}_j$ takes the form $\boldsymbol{\gamma}_j^T \boldsymbol{z}$ where the vector $\boldsymbol{\gamma}_j^T$ gives the row of $\left(\boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W} \boldsymbol{X}_{\widehat{M}}\right)^{-1} \boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W}$ corresponding to variable j. To do so, Lee et al. (2016) find the conditional distribution of the $\boldsymbol{\gamma}^T \boldsymbol{z}$ given the selected model, the signs of the estimates of the 'active' variables and the vector $\left(I - \frac{\boldsymbol{\gamma}\boldsymbol{\gamma}^T}{\boldsymbol{\gamma}^T\boldsymbol{\gamma}}\right)\boldsymbol{z}$. The Polyhedral Lemma, in Lee et al. (2016), describes, in a simple way, the event that $\boldsymbol{\gamma}^T \boldsymbol{z}$ is in a certain range and all the conditions mentioned occur. Taylor and Tibshirani (2018) adapt these ideas to the IRLS method for LASSO for more general likelihoods.

Utilizing the results from the LASSO method on the CoxPH model, as described above, Taylor and Tibshirani propose treating the "final iteration [of the Newton Raphson

algorithm described in Chapter 1] as a weighted least squares regression", and approximating the distribution of $\boldsymbol{z}$ by

$$\boldsymbol{z} \sim N(\mu, \boldsymbol{W}^{-1})$$

(28)

and then following the Polyhedral Lemma ideas. They argue somewhat heuristically that this should lead to asymptotically correct inferences (Taylor & Tibshirani, 2018), at least if the estimated 'active' set, $\widehat{M}$, contains the true active set, M, with probability close to 1.

We now describe the results of this strategy. Using the Karush-Kuhn Tucker (KKT) conditions described by Taylor and Tibshirani (2018), the 'active' variables will satisfy

$$\boldsymbol{X}_{\widehat{M}}^{T}\boldsymbol{W}\left(\boldsymbol{z} - \boldsymbol{X}_{\widehat{M}}\widehat{\boldsymbol{\beta}}_{\widehat{M}}\right) = \lambda \boldsymbol{s}_{\widehat{M}}$$

(29)

where, as in (25),

$$\boldsymbol{s}_{\widehat{M}} = sign\left(\widehat{\boldsymbol{\beta}}_{\widehat{M}}\right).$$

(30)

The 'inactive' variables also satisfy the KKT conditions, but through the following equation instead.

$$\boldsymbol{X}_{-\widehat{M}}^{T}\boldsymbol{W}\left(\boldsymbol{z} - \boldsymbol{X}_{\widehat{M}}\widehat{\boldsymbol{\beta}}_{\widehat{M}}\right) = \lambda \boldsymbol{s}_{-\widehat{M}}$$

(31)

In this case, the vector $\boldsymbol{s}_{-\widehat{M}}$ must contain entries between -1 and 1; this is equivalent to saying that every entry on the left-hand side of (31) has absolute value less than or equal to the penalty parameter λ. These equations are then used below in the Polyhedral Lemma. Taylor and Tibshirani (2018) describe the application of the

Polyhedral Lemma in the Gaussian Inference case, which can be expanded to determine the corresponding application for the Cox Proportional Hazards case.

To present the results of Taylor and Tibshirani's (2018) method, we need to define two matrices and two vectors to describe the event that the KKT conditions are satisfied with the observed 'active' set $\widehat{M}$ and the observed signs. We define the 'active' components, A_1 and b_1, in the Cox Proportional Hazards setting, as shown below, to describe the event that the variables in $\widehat{M}$ are 'active' and their estimates have the observed signs we need.

$$A_1 = -diag(\boldsymbol{s}_{\widehat{M}})\left(\boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W} \boldsymbol{X}_{\widehat{M}}\right)^{-1} \boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W} \tag{32}$$

$$b_1 = -diag(\boldsymbol{s}_{\widehat{M}})\left(\boldsymbol{X}_{\widehat{M}}^T \boldsymbol{W} \boldsymbol{X}_{\widehat{M}}\right)^{-1} \lambda \boldsymbol{s}_{\widehat{M}} \tag{33}$$

To describe the event that the variables not in $\widehat{M}$ are 'inactive' we will need the components

$$A_0 = \frac{1}{\lambda}\begin{bmatrix} \boldsymbol{X}_{-\widehat{M}}^T \boldsymbol{W} \\ -\boldsymbol{X}_{-\widehat{M}}^T \boldsymbol{W} \end{bmatrix} \tag{34}$$

$$b_0 = \begin{bmatrix} 1 + \boldsymbol{X}_{-\widehat{M}}^T \boldsymbol{W} \boldsymbol{X}_{\widehat{M}} \dfrac{\widehat{\boldsymbol{\beta}}_{\widehat{M}}}{\lambda} \\ 1 - \boldsymbol{X}_{-\widehat{M}}^T \boldsymbol{W} \boldsymbol{X}_{\widehat{M}} \dfrac{\widehat{\boldsymbol{\beta}}_{\widehat{M}}}{\lambda} \end{bmatrix} \tag{35}$$

These four components are then combined to form

$$\boldsymbol{A} = \begin{bmatrix} A_1 \\ A_0 \end{bmatrix}$$

(36)

and

$$\boldsymbol{b} = \begin{bmatrix} b_1 \\ b_0 \end{bmatrix}$$

(37)

as described by Taylor and Tibshirani (2018). Following Lee et al. (2016), Taylor and Tibshirani (2018) define

$$\mathcal{V}^-(r) = \max_{j:\,(\boldsymbol{Ac})_j<0} \frac{\boldsymbol{b}_j - (\boldsymbol{Ar})_j}{(\boldsymbol{Ac})_j}$$

(38)

$$\mathcal{V}^+(r) = \max_{j:\,(\boldsymbol{Ac})_j>0} \frac{\boldsymbol{b}_j - (\boldsymbol{Ar})_j}{(\boldsymbol{Ac})_j}$$

(39)

$$\mathcal{V}^0(r) = \max_{j:\,(\boldsymbol{Ac})_j=0} \boldsymbol{b}_j - (\boldsymbol{Ar})_j$$

(40)

where

$$c \equiv \boldsymbol{W}^{-1}\boldsymbol{\gamma}(\boldsymbol{\gamma}^T\boldsymbol{W}^{-1}\boldsymbol{\gamma})^{-\mathbf{1}}$$

(41)

and

$$r \equiv (I_N - c\boldsymbol{\gamma}^T)\boldsymbol{z}$$

(42)

These are the key components in the identity

$$\{\boldsymbol{Az} \leq \boldsymbol{b}\} = \{\mathcal{V}^-(r) \leq \boldsymbol{\gamma}^T\boldsymbol{z} \leq \mathcal{V}^+(r), \mathcal{V}^0(r) \geq 0\} \tag{43}$$

which is the main result of the Polyhedral Lemma; it expresses the event that the KKT conditions hold with the given 'active' set, the given signs, and that the orthogonal complement r in (42) is as observed. Taylor and Tibshirani (2018) clarify the interpretation that the event that a certain range, dependent on $\boldsymbol{A}$ and $\boldsymbol{b}$, contains $\boldsymbol{\gamma}^T\boldsymbol{z}$ is equivalent to the event of selecting $\{\boldsymbol{Az} \leq \boldsymbol{b}\}$, and thus this is equivalent to the event that $\overline{\boldsymbol{\beta}}_{\widehat{M}}$ is within this certain range as well. A more detailed explanation is found by Lee et al. (2016) explaining that the three versions of $\mathcal{V}$, specifically $\mathcal{V}^-(r)$, $\mathcal{V}^+(r)$, $\mathcal{V}^0(r)$, are independent of $\boldsymbol{\gamma}^T\boldsymbol{z}$, and therefore $\boldsymbol{\gamma}^T\boldsymbol{z}$ "is conditionally like a random normal variable, truncated to be between $\mathcal{V}^-(r)$ and $\mathcal{V}^+(r)$". More accurately, by conditioning on the selection event and r, the conditional law $\boldsymbol{\gamma}^T\boldsymbol{z}|\{\boldsymbol{Az} \leq \boldsymbol{b}, r = r_0\}$ follows a truncated normal distribution (Lee et al., 2016). Performing a probability integral transform using this distribution will give $F^{\mathcal{V}^-(r),\mathcal{V}^+(r)}_{\boldsymbol{\beta}^*_{\widehat{M}},(X^T_{\widehat{M}}WX_{\widehat{M}})^{-1}}(\overline{\boldsymbol{\beta}}_{\widehat{M}})|\{\boldsymbol{Az} \leq \boldsymbol{b}\}$, a statistic which can be used to make conditional inferences on $\overline{\boldsymbol{\beta}}_{\widehat{M}}$ (Taylor & Tibshirani, 2018). Lee et al. (2016) describes how to obtain this statistic through the following formula,

$$F^{a,b}_{\boldsymbol{\mu},\sigma^2}(x) = F^{\mathcal{V}^-(r),\mathcal{V}^+(r)}_{\boldsymbol{\gamma}^T\boldsymbol{\mu},\boldsymbol{\gamma}^T\sigma^2\boldsymbol{\gamma}^T}(\boldsymbol{\gamma}^T\boldsymbol{z})|\{\boldsymbol{Az} \leq \boldsymbol{b}\} = F^{\mathcal{V}^-(r),\mathcal{V}^+(r)}_{\boldsymbol{\beta}^*_{\widehat{M}},(X^T_{\widehat{M}}WX_{\widehat{M}})^{-1}}(\overline{\boldsymbol{\beta}}_{\widehat{M}})|\{\boldsymbol{Az} \leq \boldsymbol{b}\} \sim Unif(0,1) \tag{44}$$

Using this pivot makes it possible to obtain conditional inferences such as hypothesis tests and in principal post-selection confidence intervals.

Chapter 3.

With the new theory clearly laid out and equations defined, we designed a Monte Carlo study to explore the behaviour of Taylor and Tibshirani's (2018) new method. In this chapter, we start by highlighting details of the original data set and explaining steps taken to ensure proper formatting of the data for further analysis. This is followed with details of the Monte Carlo study design, including simulation specifics, formula decisions, and methods for tracking results.

3.1 Understanding the Dataset

Taylor and Tibshirani (2018) use a data set provided by D. Harrington and T. Fleming (2013) to illustrate examples. In order to explore and analyze the behaviour of the adjusted estimator and inference methods, the same data set will be used in this Monte Carlo study. The data set contains 424 individuals originally, however, only 312 individuals were truly part of the clinical trial while 112 simply provided additional measurements for some of the covariates. For this Monte Carlo study, only the 312 individuals truly involved in the clinical trial are kept; any missing values in their covariates were imputed through mean values (Mean Imputation). The data was originally collected to study the effect of D-penicillamine (DPCA) on a rare disease known as Primary Biliary Cirrhosis (PBC), a fatal and chronic liver disease (Taylor & Tibshirani, 2018).

The variables in this data set include a follow-up time (*futime*), measured in days; this variable is the time between registration and either death, transplant, or end of study (whichever occurred first); status at the end of a study (*status*), which is either death (2), transplant (1), or survival (0); and the following predictor variables:

X_1 (*drug.Y*): Treatment code , 1 = D-penicillamine, 2 = placebo

X_2 (*age*): Age of patient measured in days

X_3 (*sex.M*): Sex of patient , 0 = male, 1 = female

X_4 (*ascites.Y*): Presence of ascites , 0 = no, 1 = yes

X_5 (*hepato.Y*): Presence of Hepatomegaly , 0 = no, 1 = yes

X_6 (*spiders.Y*): Presence of Spiders , 0 = no, 1 = yes

X_7 (*edema.S, edema.Y*): Presence of Edema, 0 = no, 0.5 = yes but respond to diuretic therapy, 1 = yes, not responsive to diuretic therapy

X_8 (*bili*): Serum Bilirubin in mg/dl

X_9 (*chol*): Serum Cholesterol in mg/dl

X_{10} (*albumin*): Albumin in gm/dl

X_{11} (*copper*): Urine Copper in ug/day

X_{12} (*alk_phos*): Alkaline Phosphatase in U/liter

X_{13} (*sgot*): SGOT in U/ml

X_{14} (*trig*): Triglicerides in mg/dl

X_{15} (*platelet*): Platelets per cubic ml/1000

X_{16} (*protime*): Prothrombin time in seconds

X_{17} (*stage.2, stage.3, stage.4*): Histologic stage of disease, 1 = stage 1, 2 = stage 2, 3 = stage 3, and 4 = stage 4

The categorical variables (X_1, X_3, X_4, X_5, X_6, X_7, and X_{17}) were recoded into indicator variables to ensure compatibility with LASSO coding packages in the RStudio environment. In particular, X_7 became two binary variables and X_{17} became 3 binary variables. All the variables in $\boldsymbol{X}$ are then standardized to mean 0 and standard deviation 1, as in Tibshirani (1997). Following the example by Taylor and Tibshirani (2018), the *status* variable was recoded to be 1 if the patient died, and 0 if the patient is still alive (whether they had a transplant or not). Once the dataset was correctly formatted, 'true' parameter values for the simulations were obtained by fitting a proportional hazards model with variable selection as described below.

3.2 A Monte Carlo Study

In order to explore and analyze the behaviour of the method suggested by Taylor and Tibshirani (2018), a Monte Carlo study will be performed. A Monte Carlo study is composed of creating many simulations, applying the method of interest, collecting results, and then analyzing these results before a conclusion is drawn.

Before any simulations can be created, a known 'true' set of data must be established. In preparation for the Monte Carlo simulations, the liver data (Fleming & Harrington, 1991) is correctly formatted as described above and will serve as a basis of known data. The LASSO method with cross validation is applied to the liver data. The fit gives an estimated active set of chosen variables ($\widehat{M}$), estimates of the corresponding parameters $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$, and a value $\hat{\lambda}$ for the penalty parameter. This value of $\hat{\lambda}$ is one standard deviation (as measured by cross validation) above the minimum $\hat{\lambda}_{min}$, which minimizes the cross validated LASSO objective function (21). The estimates $\widehat{M}$ and $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$ are then taken to be the 'truth' when generating new data sets. Thus, for this study, $\widehat{\boldsymbol{\beta}}_{\widehat{M}} = \boldsymbol{\beta}$, $\widehat{M} = M$, $\hat{\lambda} = \lambda$, and $\boldsymbol{\mu_Y} = \boldsymbol{X\beta}$. While these values are not likely to be the exact true values of the original liver data set, they will be used as the established known truth in the simulations. They should be credible parameter values for a real setting, therefore should give simulated data sets with known parameters that are of an appropriate form for a CoxPH model. To explore the effect of $\boldsymbol{\beta}$ values and λ values on the behaviour of the suggested method (Taylor & Tibshirani, 2018), lower and higher values of each were also used to create separate simulations. The complete set of 'true' values is summarized in the tables below (Table 3.1, Table 3.2, Table 3.3, and Table 3.4).

Table 3.1: True β Setting Used to Create Simulations

Alteration	Age	bili	chol	albumin	copper	alk_phos	sgot
Small $(\mathbf{0.1}\boldsymbol{\beta})$	0	0.037639	0	-0.019172	0.015844	0	0
Regular $(\mathbf{1}\boldsymbol{\beta})$	0	0.37639	0	-0.19172	0.15844	0	0
Large$(\mathbf{10}\boldsymbol{\beta})$	0	3.7639	0	-1.9172	1.54844	0	0

Table 3.2: True β Setting Used to Create Simulations

Alteration	trig	platelet	protime	drug.Y	sex.M	ascites.Y	hepato.Y
Small $(\mathbf{0.1}\boldsymbol{\beta})$	0	0	0.009720	0	0	0.006740	0
Regular $(\mathbf{1}\boldsymbol{\beta})$	0	0	0.09720	0	0	0.067401	0
Large$(\mathbf{10}\boldsymbol{\beta})$	0	0	0.9720	0	0	0.67401	0

Table 3.3: True β Setting Used to Create Simulations

Alteration	spiders.Y	edema.S	edema.Y	stage.2	stage.3	stage.4
Small($0.1\boldsymbol{\beta}$)	0	0	0.008671	0	0	0.006770
Regular($1\boldsymbol{\beta}$)	0	0	0.086714	0	0	0.067704
Large($10\boldsymbol{\beta}$)	0	0	0.86714	0	0	0.67704

Notice that the pre-determined λ below are all created by inflating the original λ by a factor. The original λ, valued at approximately 0.136, is close to 0 and therefore inflicts very little penalty in the LASSO portion of the method. When the penalty is too small, more variables than necessary are typically selected. To ensure that adequate variable selection is performed, the original λ is inflated to three levels. These values provide enough variation in the penalties for different selection scenarios to be observed in the Monte Carlo study results.

Table 3.4: Pre-Determined λ Used During Simulations

Name	Symbol	Value
Low Lambda (10 λ)	λ_{Low}	1.361389
Mid Lambda (50 λ)	λ_{Mid}	6.806945
High Lambda (100 λ)	λ_{High}	13.61389

Using the estimates of the original data set ($\boldsymbol{\beta}$, M, λ) to create pre-determined 'truths', simulations can be created. To preserve the correlation structure of the covariates, the original $\boldsymbol{X}$ from the liver data will be used with new follow-up times and death/censoring occurrences. The new follow-up times are randomly generated from an exponential distribution with the rate specified as $e^{\boldsymbol{\mu_Y}}$, ensuring that the hazard rate behaviour for the CoxPH model is appropriate. This allows the correlation structures already present in the liver data to be preserved while still generating new and reasonable follow-up times for the simulations. Notice that the base hazard rate for the model does not affect the partial likelihood so it does not impact the behaviour of the model. Following the example set by Taylor and Tibshirani (2018), censoring is applied randomly from a binomial distribution with probability of (right-)censoring set to 50%. The reasonableness of this censoring rate is supported by the original data, which had a censored proportion of 60%. There are technically three variations of simulations in terms of censoring setting which could be created at this point: the scenario with no censoring, the scenario with censoring, and the scenario with all censoring. It is

important to note that the scenario with all censoring provides no results because there is no likelihood in that case, regardless of the follow-up time generation. In both the no censoring and censoring scenarios, the data set is sorted from smallest follow-up time to largest follow-up time in order to simplify the organization of our mathematics and our computing. A table summarizing the number of simulations created for each $\boldsymbol{\beta}$ and λ setting is shown below (Table 3.5).

Table 3.5: Number of Simulations to be Done at Each Setting

		Predetermined λ Values for Use in Methods					
		λ_{Low} = 1.361389		**λ_{Mid} = 6.806945**		**λ_{High}= 13.61389**	
		Censor	**No Censor**	**Censor**	**No Censor**	**Censor**	**No Censor**
$\boldsymbol{\beta}$ Values Used to Create	**Small ($0.1\boldsymbol{\beta}$)**	NA	NA	10000	10000	NA	NA
	Regular ($1\boldsymbol{\beta}$)	10000	10000	10000	10000	10000	10000
	Large ($10\boldsymbol{\beta}$)	NA	NA	10000	10000	NA	NA

These settings allowed the behaviour of $\boldsymbol{\beta}$ and λ to be explored by holding one fixed while the other is varied. The extremes of both $\boldsymbol{\beta}$ and λ, shown as the corners of the table containing 'NA', were not tested due to the numerical difficulties that become present in these circumstances in addition to time constraints. When both $\boldsymbol{\beta}$ and λ are low (top left corner of Table 3.5), little to no variable selection is performed due to the lack of penalty weight and hard to detect effects. While this setting is possible, the anticipated results are not likely to provide any additional understanding to that already gained from the other settings. In the cases were $\boldsymbol{\beta}$ is large (bottom left and bottom right of the Table 3.5), troublesome numerical difficulties cause the algorithm to run much longer than in the other settings, and occasionally fail altogether due to the complexity of the numerical work. These numerical difficulties are mainly caused by overflow. Since the equations being optimized contain exponential functions as the main components, numerical difficulties are encountered even at moderate values for $\boldsymbol{\beta}$. To mitigate this challenge, we used the log sum exponential trick, which is discussed in further detail in the next paragraph. Unfortunately, even with this effort to handle overflow, $\boldsymbol{\beta}$ is still originally large enough for this to be a time-consuming process. Therefore, we only

varied the size of $\boldsymbol{\beta}$ in the scenario with λ_{Mid}, allowing the results with a large $\boldsymbol{\beta}$ to be available compared to those with moderate or small $\boldsymbol{\beta}$. We also chose to examine the impact of changing the prespecified λ in the case of the moderate (original) $\boldsymbol{\beta}$. In general, the four extreme corners of the grid seemed unlikely to increase our understanding very much and we omitted them.

For each simulated dataset we implemented the method described by Taylor and Tibshirani (2018), as well as some traditional methods, collecting resulting details. The first step is to implement the LASSO variable selection on the simulated dataset and retrieve the corresponding penalized estimates for those variables ($\widehat{\boldsymbol{\beta}}_{\widehat{M}}$). We wrote our own code to do this task. Our overall algorithm has an outer loop (implemented in an R function called IRLS()) and an inner loop (implemented in an R function called Algorithm() which in turn calls a special purpose co-ordinate descent function, CD()). The outer loop was described above in Chapter 1; in it we do a sequence of penalized reweighted least squares problems, recomputing the objects $\boldsymbol{z}$ and $\boldsymbol{W}$ at each step. This IRLS() function also contains the log sum exponential trick to handle overflow. The specifics of this trick are as follows. Keeping in mind that the data have been sorted chronologically by follow-up time, for each $i \in D$, find the maximum value of η from the list of values from η_i to η_N and denote it as $\eta_{max,i}$. The resulting list contains the maximum η corresponding with each term in the log partial likelihood. In each term of the log partial likelihood we then subtract $\eta_{max,i}$ from every η, as shown below in equation (45), effectively resulting in all $\eta \leq 0$ without changing the ratio.

$$l(\boldsymbol{\eta}) = \sum_{i \in D} \ln\left(\frac{e^{\eta_i - \eta_{max,i}}}{\sum_{l \in R_i} e^{\eta_l - \eta_{max,i}}}\right) = \sum_{i \in D} \ln\left(\frac{\frac{1}{e^{\eta_{max,i}}} e^{\eta_i}}{\frac{1}{e^{\eta_{max,i}}} \sum_{l \in R_i} e^{\eta_l}}\right) \tag{45}$$

This simplifies to (46), where the argument in the logarithm is a sum of numbers, one of which is 1, and the rest of which are ≤ 1. Thus, by computing $\eta_i - \eta_{max,i}$ in the first half directly, we avoid underflow from the logarithm and exponential functions, and the second half with the log sum exponential prevents overflow.

$$l(\boldsymbol{\eta}) = \sum_{i \in D} \left(\eta_i - \eta_{max,i} - \ln \left(\sum_{l \in R_i} e^{\eta_l - \eta_{max,i}} \right) \right)$$

(46)

In the inner loop we use an iterative procedure to solve the penalized reweighted least squares step; we now describe the algorithm we settled on for this step.

In the RStudio environment, it is possible to utilize the optim() function from base R to optimize a custom user-defined function. Initially, this seemed to be an ideal way of manually implementing the LASSO procedure with the custom goal equation (24). The problem is that optim() never produces an estimated $\boldsymbol{\beta}$, denoted $\widehat{\boldsymbol{\beta}}$, with any coefficients which are exactly 0; without exact zeros it is hard to decide what the fitted active set is. We therefore replaced optim() with a custom built co-ordinate descent function, CD(), which is embedded in Algorithm() and in turn IRLS(), which we now describe.

In the custom IRLS() function, the steps outlined in Chapter 1 for performing the Iterated Reweighted Least Squares approach to solving the LASSO method are managed. When the process calls for minimization, IRLS() calls upon Algorithm() and passes the relevant $\boldsymbol{z}, \boldsymbol{W}, \widehat{\boldsymbol{\beta}}$ values and the constant $\boldsymbol{X}$ and λ values which are then passed on to the CD() function. When Algorithm() completes a run, the resulting vector $\widehat{\boldsymbol{\beta}}$ is passed back to IRLS() which proceeds to update $\boldsymbol{z}$ and $\boldsymbol{W}$, and checks if the value of $\widehat{\boldsymbol{\beta}}$ before Algorithm() was called and the $\widehat{\boldsymbol{\beta}}$ resulting from Algorithm() are similar enough to declare convergence or not. If the difference between the values is below a pre-determined threshold, then the IRLS() process is considered complete and the final vector $\widehat{\boldsymbol{\beta}}$ is to give the LASSO estimate, $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$. If the convergence criterion is not met, then IRLS() calls Algorithm() again with the updated $\boldsymbol{z}$ and $\boldsymbol{W}$. The result from IRLS(), the vector $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$, serves as the starting point for Taylor and Tibshirani's (2018) new method, as the process is described in Chapter 2.

Algorithm(), manages the tracking of the convergence during the minimization process of the goal formula (24). Algorithm() calls CD() repeatedly, making note each time of the $\widehat{\boldsymbol{\beta}}$ being passed into CD() and the new $\widehat{\boldsymbol{\beta}}$ being returned by it. Before calling CD() again, Algorithm() calculates the absolute differences between the old and new $\widehat{\boldsymbol{\beta}}$ values, and only calls CD() again if the values have a greater difference than a pre-

determined threshold. When the difference for all the $\widehat{\boldsymbol{\beta}}$ is below the threshold, the function is considered converged and the final $\widehat{\boldsymbol{\beta}}$ are passed on to the IRLS() function.

The custom coordinate descent function (CD()) is given the current $\boldsymbol{z}, \boldsymbol{W}, \widehat{\boldsymbol{\beta}}$ values (and the unchanging $\boldsymbol{X}$ and λ values). The value of the LASSO objective function (24) with these values is computed and saved for future comparisons. The function then updates each of the β_j values one at a time. The goal function of the LASSO (24) is non-negative and strictly convex. It is possible to find the value of β_j which minimizes (24) with all the other co-ordinates of $\boldsymbol{\beta}$ fixed. To do so we compute the derivative with respect to β_j of (24). At any non-zero value of β_j this derivative exists and is given by

$$\frac{\partial G(\beta_j)}{\partial \beta_j} = \frac{\partial \left(\frac{1}{2}(\boldsymbol{z} - \mathbf{X}\boldsymbol{\beta})^T \boldsymbol{W}(\boldsymbol{z} - \mathbf{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j|\right)}{\partial \beta_j} = \begin{cases} \boldsymbol{X_j}'\boldsymbol{W}\boldsymbol{X_j}\beta_j - \boldsymbol{z}'\boldsymbol{W}\boldsymbol{X_j} - \lambda & \text{if } \beta_j < 0 \\ \boldsymbol{X_j}'\boldsymbol{W}\boldsymbol{X_j}\beta_j - \boldsymbol{z}'\boldsymbol{W}\boldsymbol{X_j} + \lambda & \text{if } \beta_j > 0 \end{cases}$$

(47)

where $\boldsymbol{X_j}$ is the j-th column of the $\boldsymbol{X}$. When $\beta_j = 0$, this strictly convex function $G(\beta_j)$ is not differentiable, but it does have left and right derivatives at 0 given by

$$\frac{\partial G(0-)}{\partial \beta_j} = -\boldsymbol{z}'\boldsymbol{W}\boldsymbol{X_j} - \lambda$$

(48)

and

$$\frac{\partial G(0+)}{\partial \beta_j} = -\boldsymbol{z}'\boldsymbol{W}\boldsymbol{X_j} + \lambda$$

(49)

For a strictly convex function, any place where the function is differentiable and the derivative is 0 must be the global minimum. If the global minimum is at a place where the function is not differentiable but has left and right derivatives, then the left derivative at that point must be non-positive and the right derivative must be non-negative. Thus, to minimize $G(\beta_j)$ our algorithm first computes the left and right derivatives at 0. If they are opposite in sign, as shown in Figure 3.1, then the function sets β_j to 0 and moves on.

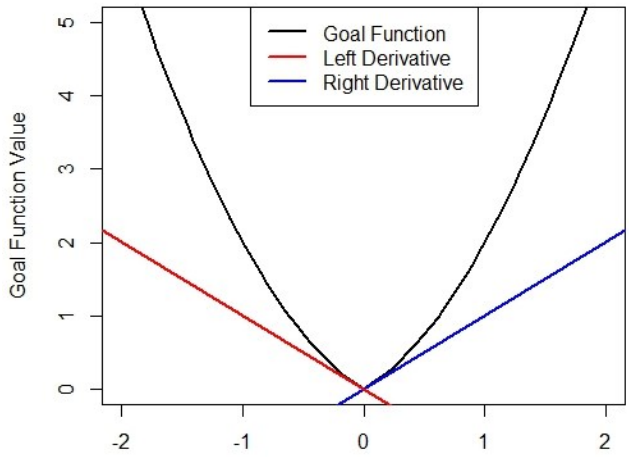


Figure 3.1: Visual of Coordinate Descent Logic for Minimization Achieved at 0

If both derivatives are positive, then the minimum occurs at a negative β_j, as shown in Figure 3.2.

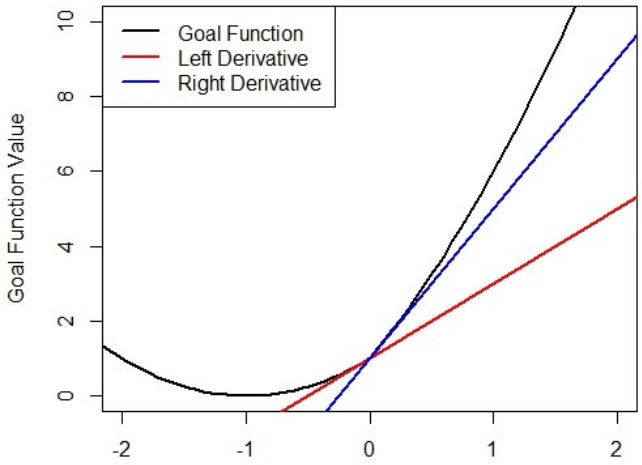


Figure 3.2: Visual of Coordinate Descent Logic for Minimum to the Left of 0

We find this estimate by setting the top formula on the right-hand side of (47) equal to 0 and solving for β_j, resulting in equation (50).

$$\hat{\beta}_j = \left(\boldsymbol{X}_j'\boldsymbol{W}\boldsymbol{X}_j\right)'\left(\boldsymbol{z}'\boldsymbol{W}\boldsymbol{X}_j + \lambda\right). \tag{50}$$

If both derivatives are negative, then the minimum occurs at a positive β_j, as shown in Figure 3.3.

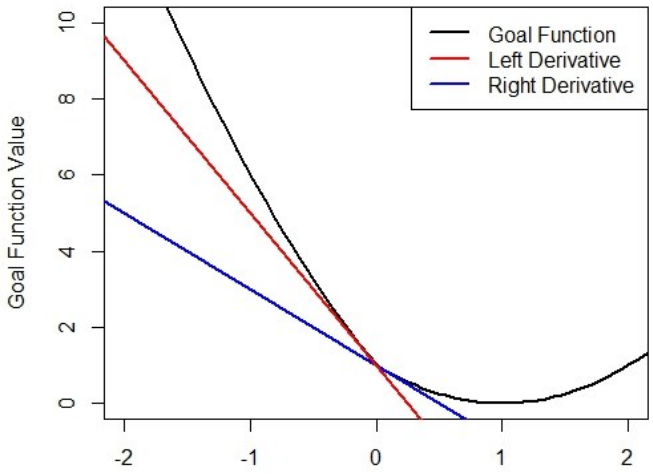


Figure 3.3: Visual of Coordinate Descent Logic for Minimum to the Right of 0

This is found by setting the bottom formula on the right-hand side of (47) equal to 0 and solving for β_j, resulting in equation (51).

$$\hat{\beta}_j = \left(\boldsymbol{X}_j'\boldsymbol{W}\boldsymbol{X}_j\right)'\left(\boldsymbol{z}'\boldsymbol{W}\boldsymbol{X}_j - \lambda\right). \tag{51}$$

After updating an individual $\hat{\beta}_j$, CD() updates the value of the objective function and then proceeds to update the rest of the $\hat{\beta}_j$. A single call of CD() will update each $\hat{\beta}_j$ once in a single pass and finish by returning the final values of the updated $\widehat{\boldsymbol{\beta}}$ to Algorithm().

In addition to implementing Taylor and Tibshirani's method (2018), we fitted the corresponding unpenalized CoxPH model fit for the specific simulated data set and chosen variables. By using only the variables selected by the LASSO procedure in the CoxPH model, the estimates ($\hat{\beta}_{CoxPH_{\widehat{M}}}$) are made comparable to those found in the new method ($\overline{\boldsymbol{\beta}}_{\widehat{M}}$), since the datasets are standardized. Tracking these values over numerous simulations in various settings will provide a way of visualizing the differences, and potential improvements, of one method over the other.

In Chapter 1, in our discussion of linear regression and model selection, we introduced the notation $\boldsymbol{\beta}^*_{\widehat{M}}$, which minimizes $(\boldsymbol{\mu_Y} - \boldsymbol{X}_{\widehat{M}}\boldsymbol{\beta})'(\boldsymbol{\mu_Y} - \boldsymbol{X}_{\widehat{M}}\boldsymbol{\beta})$ over $\boldsymbol{\beta}$, in linear models. When a potential 'active' set $\widehat{M}$ does not contain every active variable, the quantity $\boldsymbol{\beta}^*_{\widehat{M}}$ is not equal to the subvector $\boldsymbol{\beta}_{\widehat{M}}$ of the true $\boldsymbol{\beta}$ vector. Instead, $\boldsymbol{\beta}^*_{\widehat{M}}$ is the

expected value of the ordinary least squares estimate $\widehat{\boldsymbol{\beta}}_{\widehat{M}} = (\boldsymbol{X}_{\widehat{M}}{}'\boldsymbol{X}_{\widehat{M}})^{-1}\boldsymbol{X}_{\widehat{M}}{}'\boldsymbol{y}$ (analogous to equation (4)). That is, $\boldsymbol{\beta}^*_{\widehat{M}} = (\boldsymbol{X}_{\widehat{M}}{}'\boldsymbol{X}_{\widehat{M}})^{-1}\boldsymbol{X}_{\widehat{M}}{}'\boldsymbol{\mu}_{\boldsymbol{Y}}$. Here, $\boldsymbol{\mu}_{\boldsymbol{Y}} = E(\boldsymbol{y})$ and it is not generally true that $\boldsymbol{\mu}_{\boldsymbol{Y}} = \boldsymbol{X}_{\widehat{M}}\boldsymbol{\beta}^*_{\widehat{M}}$. The estimates from these models ($\boldsymbol{\beta}^*_{\widehat{M}}$) represent the corresponding true $\boldsymbol{\beta}$ values for the model containing the specified variables $\widehat{M}$. If all the truly active variables are selected in $\widehat{M}$, then $\boldsymbol{\beta}^*_{\widehat{M}}$ should be the same as $\boldsymbol{\beta}$, since any additional variables would be set to their true value of 0. However, if at least one truly active variable is not selected, then there will be differences between $\boldsymbol{\beta}^*_{\widehat{M}}$ and $\boldsymbol{\beta}$, since other variables now need to account for the effect of the dropped active variable. Tracking these values will help provide insight on how various estimation methods compare to each other and which methods maintain the most accuracy over various scenarios.

To quantify coverage probability in a meaningful and time-efficient manner, hypothesis tests were done to determine whether or not the true values of the model, $\boldsymbol{\beta}^*_{\widehat{M}}$, would be captured by 95% confidence intervals. Using $\boldsymbol{\beta}^*_{\widehat{M}}$ in equation (44) and then determining the p-value for a two-sided hypothesis test makes this possible with the following hypotheses.

$$H_0: \overline{\boldsymbol{\beta}}_{\widehat{M}} = \boldsymbol{\beta}^*_{\widehat{M}}$$

$$H_1: \overline{\boldsymbol{\beta}}_{\widehat{M}} \neq \boldsymbol{\beta}^*_{\widehat{M}}$$

If the resulting p-value is greater than 0.05, then the corresponding new 95% confidence interval is successful in capturing the true model values $\boldsymbol{\beta}^*_{\widehat{M}}$. It is theoretically possible to create confidence intervals with the cumulative distribution function $F^{\mathcal{V}^-(r),\mathcal{V}^+(r)}_{\boldsymbol{\beta}^*_M,(\boldsymbol{X}_M^T\boldsymbol{W}\boldsymbol{X}_M)^{-1}}(\overline{\boldsymbol{\beta}}_M)|\{\boldsymbol{Az} \leq \boldsymbol{b}\}$, however, since this distribution does not have easy algorithms for quantiles such as are available in the Normal distribution case, this would involve heavy computation and careful numerical work in determining limits on these intervals. By using a direct hypothesis test, we are able to obtain a measure of the coverage probability without going through the additional mathematical challenges in creating the actual confidence intervals for every simulation performed.

In summary, for each simulation the following details are collected and saved to be used in further analysis:

- 'True' $\boldsymbol{\beta}$ used to create simulation
- Pre-determined λ used
- Which variables are chosen $\widehat{M}$
- The observed P-Values from the adjusted method
- The lower limit of the polyhedral lemma $\mathcal{V}^{-}(r)$
- The upper limit of the polyhedral lemma $\mathcal{V}^{+}(r)$
- The $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$ from the LASSO procedure
- The $\boldsymbol{z} = \eta + \boldsymbol{W}^{-1}\boldsymbol{U}(\eta)$
- The $\overline{\boldsymbol{\beta}}_{\widehat{M}}$ resulting from the adjusted method
- The test statistic using $F^{\mathcal{V}^{-}(r),\mathcal{V}^{+}(r)}_{\boldsymbol{\beta}^{*}_{\widehat{M}},(\boldsymbol{X}^{T}_{\widehat{M}}\boldsymbol{W}\boldsymbol{X}_{\widehat{M}})^{-1}}\left(\overline{\boldsymbol{\beta}}_{\widehat{M}}\right)|\{\boldsymbol{A}\boldsymbol{z} \leq \boldsymbol{b}\}$ for null hypothesis $H_0: \overline{\boldsymbol{\beta}}_{M} = \boldsymbol{\beta}^{*}_{\widehat{M}} = 0$
- The observed traditional $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$ from the CoxPH model after LASSO selection
- The observed traditional P-Values from the CoxPH model after LASSO selection
- The observed traditional Z-statistics from the CoxPH model after LASSO selection
- The corresponding 'true' $\boldsymbol{\beta}^{*}_{\widehat{M}}$, which results from linear regression of the known $\boldsymbol{\mu}_{\boldsymbol{Y}}$ on the covariates selected ($\widehat{M}$)
- The observed traditional P-Values associated with $\boldsymbol{\beta}^{*}_{\widehat{M}}$
- The observed traditional t-statistic associated with $\boldsymbol{\beta}^{*}_{\widehat{M}}$
- The test statistic using $F^{\mathcal{V}^{-}(r),\mathcal{V}^{+}(r)}_{\boldsymbol{\beta}^{*}_{\widehat{M}},(\boldsymbol{X}^{T}_{\widehat{M}}\boldsymbol{W}\boldsymbol{X}_{\widehat{M}})^{-1}}\left(\overline{\boldsymbol{\beta}}_{\widehat{M}}\right)|\{\boldsymbol{A}\boldsymbol{z} \leq \boldsymbol{b}\}$ for null hypothesis $H_0: \overline{\boldsymbol{\beta}}_{M} = \boldsymbol{\beta}^{*}_{\widehat{M}}$
- How many variables are chosen
- How many of the chosen variables are from the 'true' active variables M
- How many of the chosen variables are from the 'true' inactive variables $-M$
- Whether the simulation was successful or not (convergence and boundary problems will cause a simulation to terminate with error)

Chapter 4.

Once all the simulations are complete and the results collected, we are ready to summarize and analyze these results. In this chapter, we go through the results of the simulations in detail. Interesting results are summarized in visuals and problems encountered are noted. We finish this chapter with a conclusion to highlight key takeaways from the study and a discussion on possible further work and additional exploration for future studies.

4.1. Results

In this Monte Carlo study, 10000 simulations were run at each of the pre-determined settings (Table 3.5). However, some simulated datasets came to no conclusion due to errors encountered with numerical difficulties. The speculated causes of these errors, principally overflow, were discussed earlier in Chapter 3. These errors were tracked in two main groups: solutions occurring at boundaries of the goal function and no solutions because of no convergence. The table below (Table 4.1) summarizes the counts of these failed simulations in each setting. It is clearly shown in the table that some settings have more failed simulations than others. In particular, the setting with the large $\boldsymbol{\beta}$ values ($10\boldsymbol{\beta}$) contains the most, having only 8032 successfully completed simulations. Although this may appear as alarming initially, it is important to notice that these inflated $\boldsymbol{\beta}$ are exaggerating the effect of the active covariates to an unrealistic extent but are being utilized to enforce the base assumptions for the theory. In the settings were $\boldsymbol{\beta}$ are closer to the coefficients in original data set, the rate of simulation failure is typically less than 1%, and therefore these are not terribly concerning for this study.

Table 4.1: Counts of Failed Simulations at Each Setting

		Pre-Determined λ Values for Use in Methods					
		λ = 1.361389		λ = 6.806945		λ = 13.61389	
		Censor	No Censor	Censor	No Censor	Censor	No Censor
β Values Used to Create	Small (0.1β)	NA	NA	B – 0 C – 0 BC – 0 NV – 0	B – 0 C – 0 BC – 0 NV – 0	NA	NA
	Regular (1β)	B – 87 C – 75 BC – 311 NV – 1 *	B – 24 C – 19 BC – 51 NV – 0	B – 15 C – 17 BC – 22 NV – 0	B – 3 C – 2 BC – 6 NV – 0	B – 0 C – 1 BC – 0 NV – 0	B – 1 C – 0 BC – 0 NV – 0
	Large (10β)	NA	NA	B – 635 C – 854 BC – 468 NV – 2 **	B – 69 C – 94 BC –47 NV – 1	NA	NA

Legend:
B – solutions on boundary (cause overflow)
C – failure to converge
BC – failure to converge and last result at boundary
NV – no variable selected for model, thus no theory can be applied
* 5 additional failures occurred within the CD() function, likely due to overflow
** 9 additional failures occurred within the CD() function, likely due to overflow

The key assumption in Taylor and Tibshirani (2018) is that the active set of variables is chosen correctly, more commonly referred to as being 'correctly screened'. If at least all the active variables are selected, then the theory claims to create confidence intervals with appropriate coverage probability. However, if even one active variable is missed, then the assumption is not met, regardless of how many other variables are selected in its place. There are three ways that this selection process can be affected: the size of the penalty used, the pre-determined λ; and the intensity of the variable effects, either in terms of the size of $\boldsymbol{\beta}$; or in terms of the amount of censoring. Shown below are two tables (Table 4.2, Table 4.3) displaying the proportion of models which contained at least all the active variables, thus satisfying the above assumption, at each setting. The first table (Table 4.2) displays the proportion of total correctly screened out of the total number of simulations attempted. The second table (Table 4.3) displays the

total correctly screened out of the total number of successfully completed simulations, thus removing any simulations considered as failures.

Table 4.2: Proportion of Correctly Screened Models Out of All 10000 Simulations

		Pre-Determined λ Values for Use in Methods					
		λ = 1.361389		λ = 6.806945		λ = 13.61389	
		Censor	No Censor	Censor	No Censor	Censor	No Censor
β Values Used to Create	Small (0.1β)	NA	NA	0.0045	0.0262	NA	NA
	Regular (1β)	0.5510	0.7478	0.1401	0.3847	0.0614	0.2997
	Large (10β)	NA	NA	0.8011	0.9789	NA	NA

The most noticeable difference occurs in the last row, where the large β vector is used. This difference is due to the large number of failed simulations being removed from the total simulations considered. Thus, for large β, when the simulation is successfully completed, the assumption of correct screening is essentially guaranteed to be met. However, for the other settings, it is evident from the above table that the probability of meeting the assumption of variables being correctly screened ranges from low to high. Correct screening is therefore not necessarily a credible assumption to make. Keeping this in mind, we continue to evaluate and analyze the results from the study.

Table 4.3: Proportion of Correctly Screened Models Out of Successfully Completed Simulations

		Pre-Determined λ Values for Use in Methods					
		λ = 1.361389		λ = 6.806945		λ = 13.61389	
		Censor	No Censor	Censor	No Censor	Censor	No Censor
β Values Used to Create	Small (0.1β)	NA	NA	0.0045	0.0262	NA	NA
	Regular (1β)	0.5787	0.7549	0.1409	0.3851	0.0614	0.2997
	Large (10β)	NA	NA	0.9974	1.0000	NA	NA

It is known that there are seven variables in the active set M from the established truth, though it is also seen that these seven active variables are not always chosen together for most models. Most models include more than seven variables in the model by including non-active variables. The sizes of the models typically selected at each setting is displayed in the histogram grid below (Figure 4.1). These histograms only consider the cases where simulations were successfully completed, allowing the total areas under the curves to sum to one. As is the nature of LASSO, the selection process attempts to keep enough variables to adequately model results without including too many. As the LASSO penalty increases (towards the right side of the grid), the size of the models selected decreases compared to the setting with a lower LASSO penalty (on the left side of the grid). A moderate LASSO penalty (in the center columns of the grid) appears to typically select models of roughly the same size regardless of the $\boldsymbol{\beta}$ size with some slight variation along the $\boldsymbol{\beta}$ values. Overall, the models tend to contain more than seven variables, though not necessarily all the active variables. This is shown clearly in the next histogram grid below (Figure 4.2).

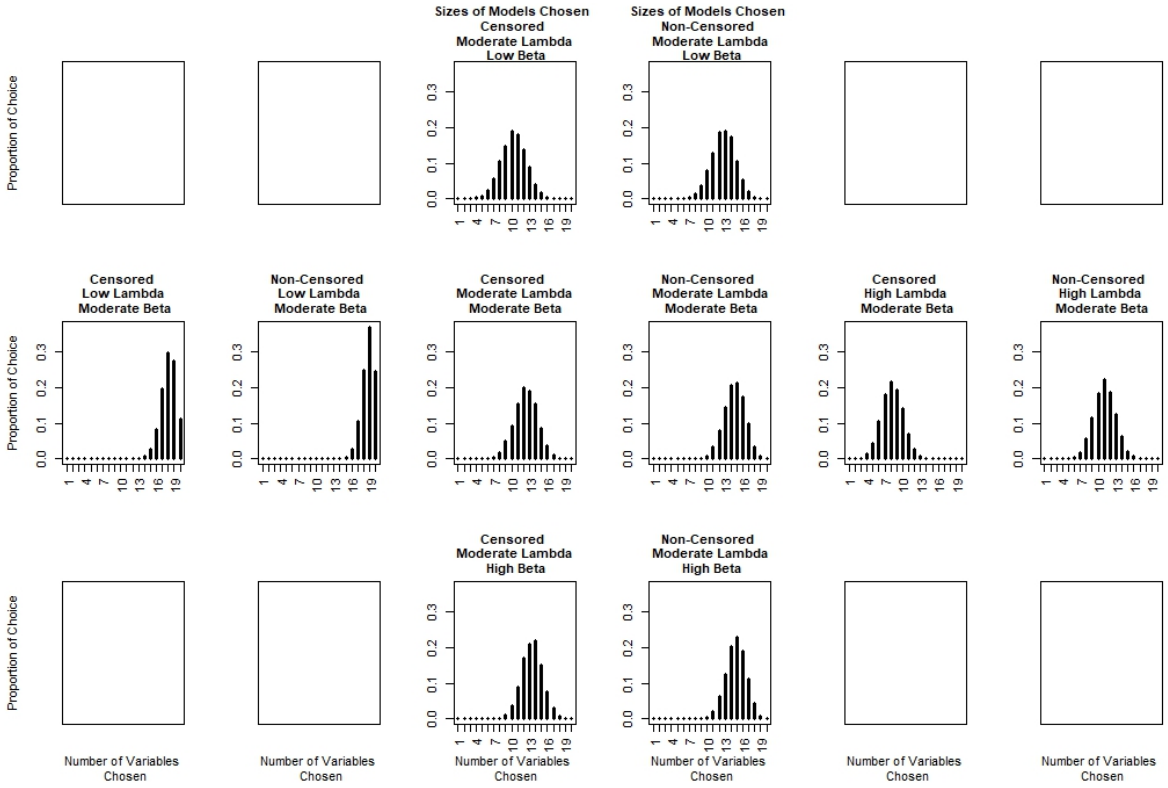


Figure 4.1: Histogram Grid of Number of Variables Selected in Models

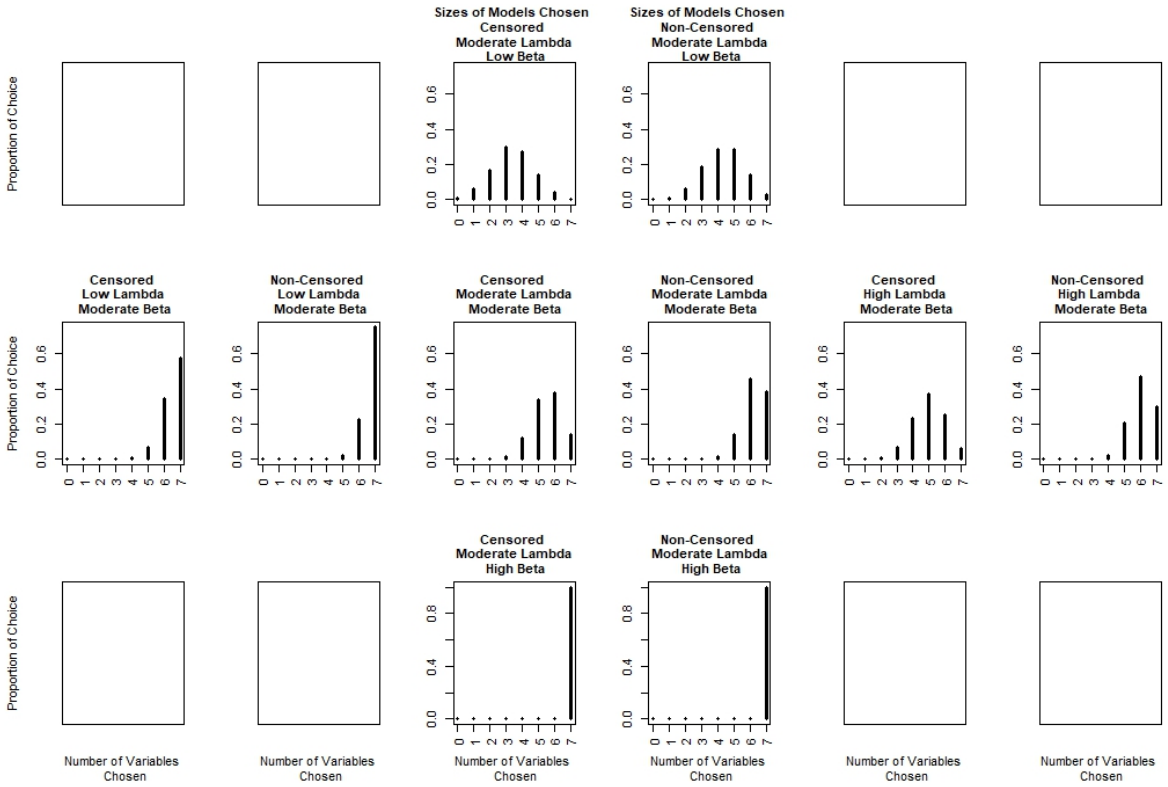


Figure 4.2: Histogram Grid of Number of Active Variables Selected in Models

Most notably, the scenario aimed at guaranteeing the correct screening assumption (bottom row with large $\boldsymbol{\beta}$) does indeed select all the active variables, M, in every model where the simulation was successfully completed. In the scenario with the moderate, and realistic, $\boldsymbol{\beta}$, some of the active variables tend to be missed in the model selection. This tendency gets worse as the LASSO penalty increases; larger penalties produce fewer correctly screened models in Table 4.3. Understanding that the assumption of correct variable screening is usually not credible, and in the case where it is credible there are other numerical problems that arise, we proceed to investigate the behaviours of the various methods and the coverage probability of the confidence intervals.

Taylor and Tibshirani (2018) apply their new method to the original liver data. Though there are some minor differences in set up, such as dropping cases with missing data instead of mean imputation, the theory applied in their paper behaves in the same manner as the theory applied in this study. This is most notably seen in the visual representations comparing the distribution of p-values between the new method and the traditional CoxPH method. The graphs obtained using our own results are shown below (Figure 4.3). The graph on the left corresponds with the graph for the liver data shown in

Taylor and Tibshirani (2018), but with the axes flipped. The p-values displayed in these graphs (Figure 4.3) are concerned with testing for significance of the variable of interest at a 95% confidence level, therefore the hypotheses are

$$H_0: \beta_j = 0$$

$$H_1: \beta_j \neq 0.$$

If the null hypothesis (H_0) is true, then the resulting p-values are expected to have a uniform distribution, which is marked as a red line on the graphs. For the truly inactive variables (the variables for which the null hypothesis is true), if p-values are below the red line then this indicates Type 1 error rates going above the allowable Type 1 error rate, also known as an anti-conservative test (Taylor & Tibshirani, 2018). In contrast, when the p-value is below the red line for the truly active variables, this indicates power of the test (Taylor & Tibshirani, 2018). Keeping these explanations in mind, an ideal test would have p-values along the red line for truly inactive variables while still having p-values below the red line for truly active variables, creating a balance for Type 1 error and power. This behaviour is observed for the adjusted p-values in the graphs below (Figure 4.3). The traditional p-values from the CoxPH model, however, tend to have lower observed values than expected in both graphs, an indication that the p-values are not performing as they should.

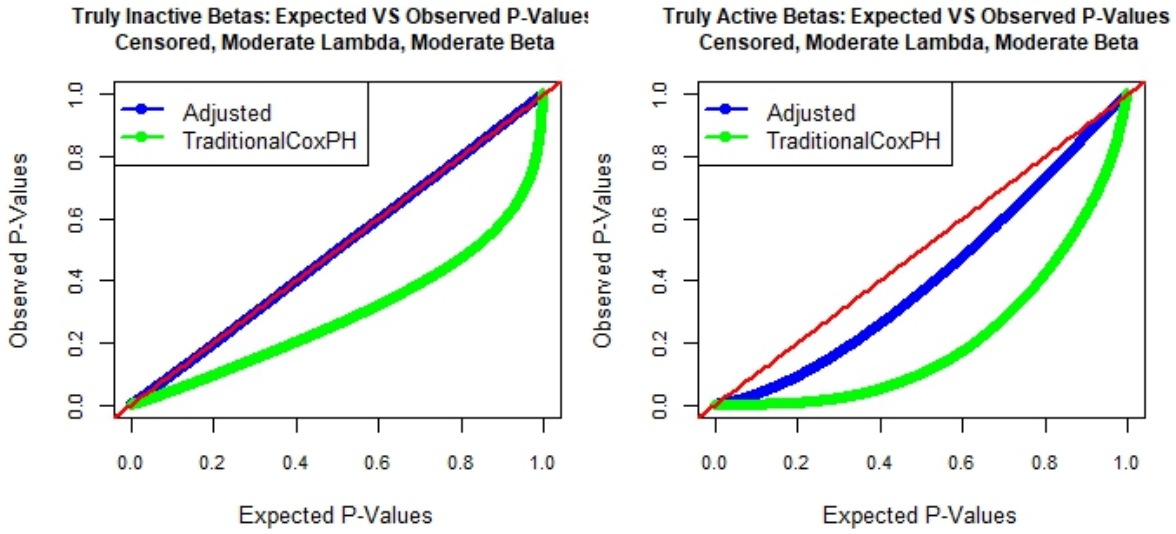


Figure 4.3: Expected VS Observed P-Values for All Inactive and All Active Variables Across All Simulations, Moderate λ and Moderate β

Even though the setting of censored data with a moderate λ and moderate $\boldsymbol{\beta}$ only correctly screens the variables 14.09% of the time, the p-value distributions of the new method appear to outperform those of the traditional method, according to the graphs above (Figure 4.3). The new method continues to outperform the traditional method in terms of coverage probability, as shown in Figure 4.4. In these graphs we utilize the probability that $\boldsymbol{\beta}^*_{\widehat{M}}$, the true $\boldsymbol{\beta}$ corresponding with the selected model, is included in the 95% confidence intervals (the method for calculating these probabilities for the new method was discussed in Chapter 3). Manipulating the Z-statistic with $\hat{\beta}_{CoxPH_{\widehat{M}}}$ from the traditional CoxPH model, we are able to obtain the corresponding standard errors for these estimates. These are taken and used to determine traditional p-values for the following hypotheses

$$H_0: \boldsymbol{\beta}_{CoxPH_M} = \boldsymbol{\beta}^*_{\widehat{M}}$$

$$H_1: \boldsymbol{\beta}_{CoxPH_M} \neq \boldsymbol{\beta}^*_{\widehat{M}}$$

If the resulting p-value is greater than 0.05, this indicates that $\boldsymbol{\beta}^*_{\widehat{M}}$ is included in the traditional 95% confidence interval.

For each variable, the number of confidence intervals which successfully captured $\boldsymbol{\beta}^*_{\widehat{M}}$ over the total number of confidence intervals created for that variable (how often it was chosen) provides the estimate of the coverage probability. In the left graph, all the models are considered for these calculations, while the right graph considers only models which met the correct screening assumption. Since the nature of these coverage probabilities is binomial, the corresponding 95% Wald Confidence Intervals for the Binomial random variable are displayed for each variable as well.

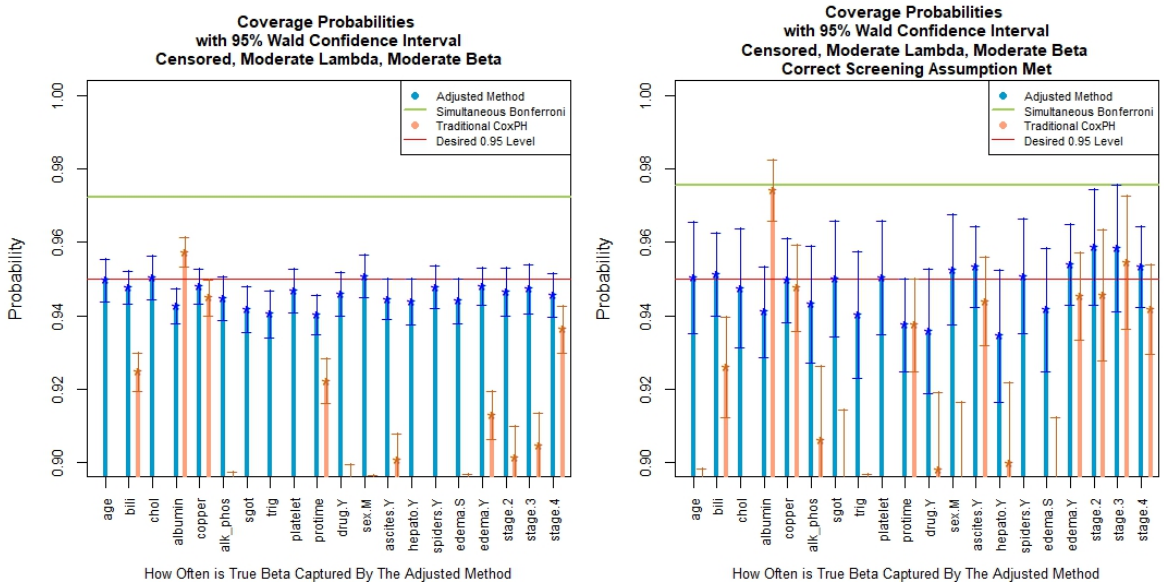


Figure 4.4: Comparison of Coverage Probabilities for All Models (left) and Models Correctly Screened (right), Moderate λ and Moderate β

An interesting observation from the left graph in Figure 4.4 is that even with most models failing to be correctly screened, Taylor and Tibshirani's (2018) method appears to be more stable across the variables and has better coverage probability than the traditional CoxPH model. This characteristic of the coverage probability for Taylor and Tibshirani's (2018) method is present in all the settings explored (for those who are curious, graphs for each setting can be found in the Appendix). A surprising result, however, was found in the setting for censored data with moderate λ and high $\boldsymbol{\beta}$, as shown in Figure 4.5. While Taylor and Tibshirani's (2018) method still outperforms the traditional method, it is surprising to see that in the setting where the assumption is guaranteed to be met, the methods perform worse than those in the other settings. This is most noticeable for the variables bili and albumin, two variables known to be most significantly active. This particular setting also has surprising results when examining and comparing estimations from various methods in Figure 4.6.

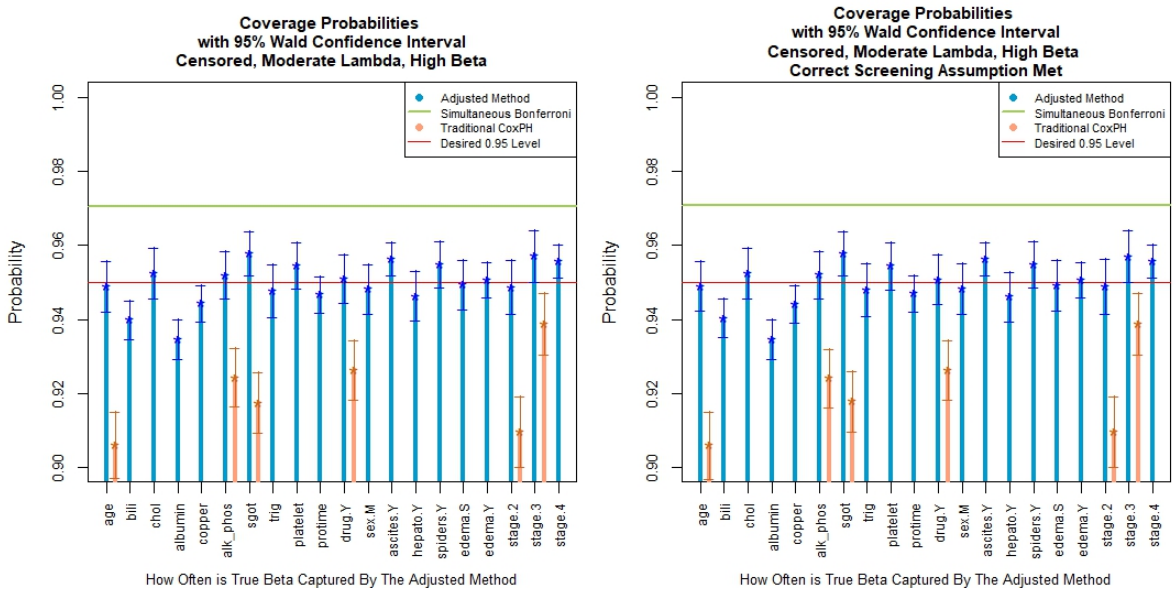


Figure 4.5: Comparison of Coverage Probabilities for All Models (left) and Models Correctly Screened (right), Moderate λ and High β

Values in Figure 4.6 are calculated as mean estimates for each type of estimation method. More specifically, for each variable and each estimation method, the sum of all estimates is divided by the number of estimates made; each variable has a different denominator. Labels in the legend correspond to the tracked estimates as follows: TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; and B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$. In the graphs below (Figure 4.6), we expect to see the $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$ to be closer to $\boldsymbol{\beta}$ than $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$ since $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$ includes penalties. However, the relationship of these is completely opposite in this setting. We explored this strange result slightly by testing our own version of the methods with $\lambda = 0$. This effectively makes our code run the same equations as the theoretical CoxPH model and should result in $\widehat{\boldsymbol{\beta}}_{\widehat{M}} = \widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$, which was indeed the result. Thus, the post-selection inference techniques being used here appear to work well in terms of coverage probability and level of hypothesis tests, but our results highlight a potential problem with how CoxPH models fitted using coxph() in R handle large $\boldsymbol{\beta}$ values. Since this phenomenon is not the primary focus of the study, no further exploration was carried out on this particular issue.

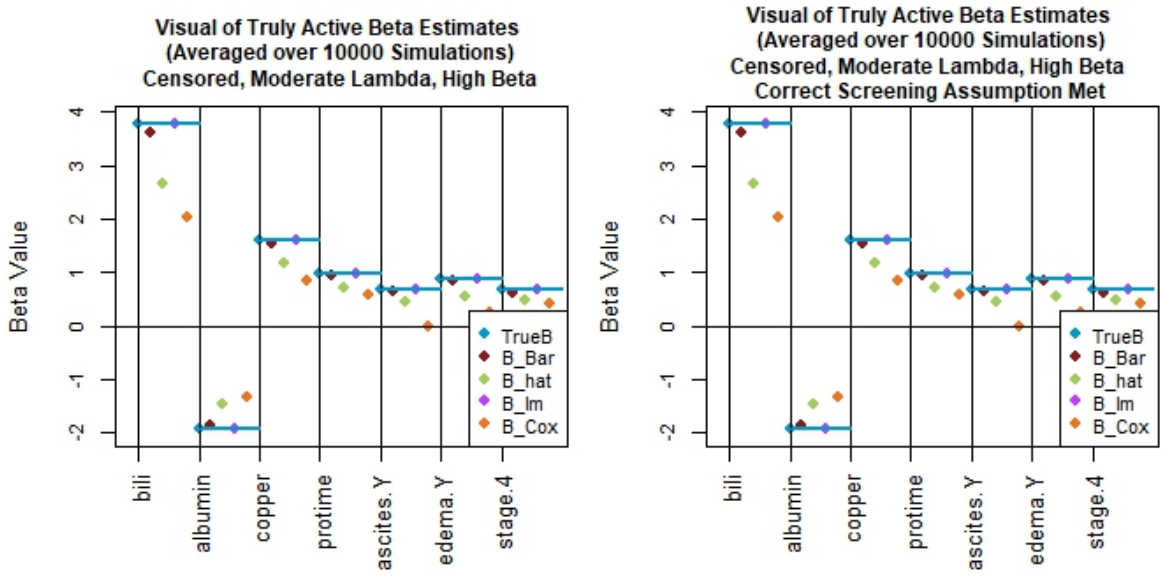


Figure 4.6: Comparison of Estimations from Various Methods, Censored Data with Moderate λ and High β

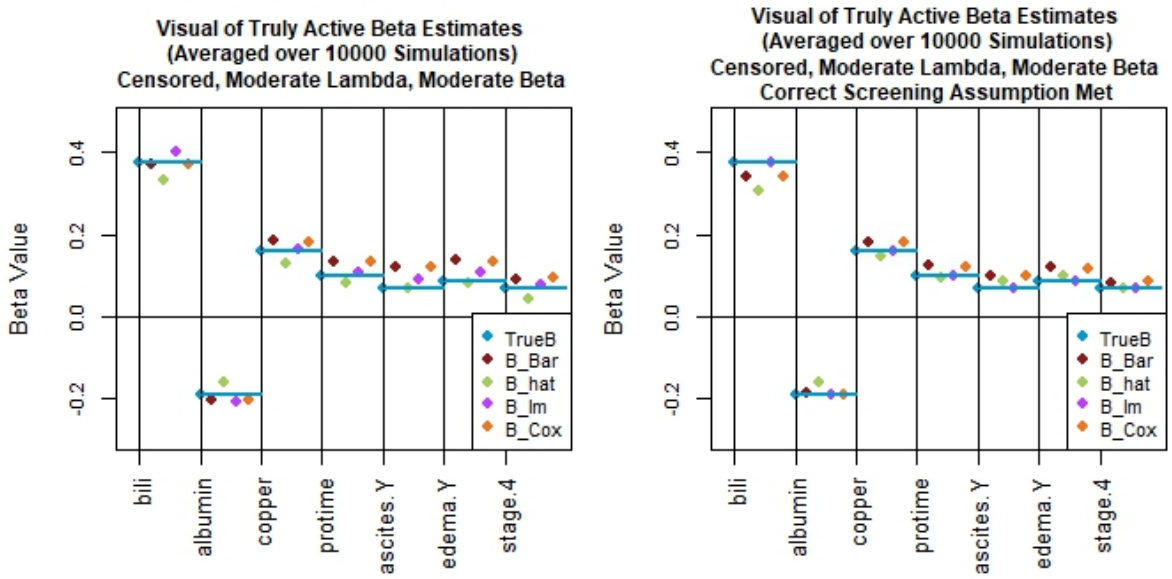


Figure 4.7: Comparison of Estimations from Various Methods, Censored Data with Moderate λ and Moderate β

The graphs above (Figure 4.7), provide an example of the estimation behaviours in the other settings where, as a general rule, the estimates from all the methods are relatively close to the true $\boldsymbol{\beta}$. In all the settings, except for the previously mentioned high $\boldsymbol{\beta}$, the estimate from the new method, $\overline{\boldsymbol{\beta}}_{\hat{M}}$, and the traditional CoxPH model estimates,

$\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$, are almost identical. This is reassuring as it indicates that the new method is establishing values that are appropriate for the survival analysis setting and are in agreement with the traditional CoxPH values. However, the new method has the advantage of more stable and credible coverage probability compared to the traditional CoxPH method, as noted in the discussion about Figure 4.4.

4.2. Conclusion

Overall, Taylor and Tibshirani's (2018) new method appears to be a better method for achieving relatively reliable estimators with post-selection inference capabilities. In all the settings tested in this study, the new method certainly appears to outperform other methods in terms of coverage probability. Even in extreme circumstances, such as when the assumption of correct screening of the variables is usually not met, this method still produces coverage probabilities that outperform those of other traditional methods. The estimates of the $\boldsymbol{\beta}$, however, appear to be slightly further from the true values than those in the cases where the assumption is met more regularly, though they are not further than any of the other methods considered. Though it is beneficial to gain post-selection inference capabilities without sacrificing reliability of the estimators too much, this method can present very difficult numerical challenges. In this study, some of these problems, such as overflow, could be handled by careful numerical work, but other problems arising from arithmetic difficulties, such as solutions occurring on boundaries, are still in need of a solution. In our case, since the solutions to these were not the focus of the study, we noted the problems and reported them (in Table 4.1). After considering and evaluating the results from all the various simulations in this study, we can conclude that Taylor and Tibshirani's (2018) new method is definitely worth consideration as a way of finding relatively reliable estimators with post-selection inferences capable of achieving credible coverage probabilities.

4.3. Discussion

There are evidently benefits to Taylor and Tibshirani's (2018) new method, though there are still some additional challenges and alternative situations to be explored. In this study we saw how most of the settings tested, even with a decent size data set, resulted in the assumption of correct variable screening not being met, and in

the one setting where it was met there were numerous other arithmetic problems that arose. Further study could perhaps explore if the problem of meeting the assumption can be solved with larger datasets. In addition, it would be interesting to see how this theory behaves on different datasets, such as a dataset containing more variables than observations. This may even yield a preferred method for estimation and inference on these types of datasets where traditional methods, such as CoxPH models, are not possible. However, before this method could truly be accessible, solutions are needed for the numerical difficulties that are present in the equations. This would be another area that could be explored further; if better methods for optimization of the goal function can be determined, then potentially the problems of solutions occurring at boundaries or lack of convergence could be solved.

References

Cox, D. R. (1972). Regression Models and Life-Tables. Journal of the Royal Statistical Society: Series B (Methodological), 34(2), 187–202. https://doi.org/10.1111/j.2517-6161.1972.tb00899.x

Devore, J. L. (2016). *Probability and Statistics for Engineering and the Sciences*, 9e, International Metric Edition. Brooks/Cole.

Fleming, T. R., & Harrington, D. P. (2013). *Counting Processes and Survival Analysis*. Wiley.

Lee, J. D., Sun, D. L., Sun, Y., & Taylor, J. E. (2016). Exact post-selection inference, with application to the lasso. The Annals of Statistics, 44(3). https://doi.org/10.1214/15-aos1371

Lockhart, R., Taylor, J., Tibshirani, R. J., & Tibshirani, R. (2014). A significance test for the lasso. The Annals of Statistics, 42(2). https://doi.org/10.1214/13-aos1175

Taylor, J., & Tibshirani, R. J. (2015). Statistical learning and selective inference. Proceedings of the National Academy of Sciences of the United States of America, 112(25), 7629–7634. https://doi.org/10.1073/pnas.1507583112

Taylor, J., & Tibshirani, R. (2018). Post-Selection Inference for ℓ1-Penalized Likelihood Models. The Canadian journal of statistics = Revue canadienne de statistique, 46(1), 41–61. https://doi.org/10.1002/cjs.11313

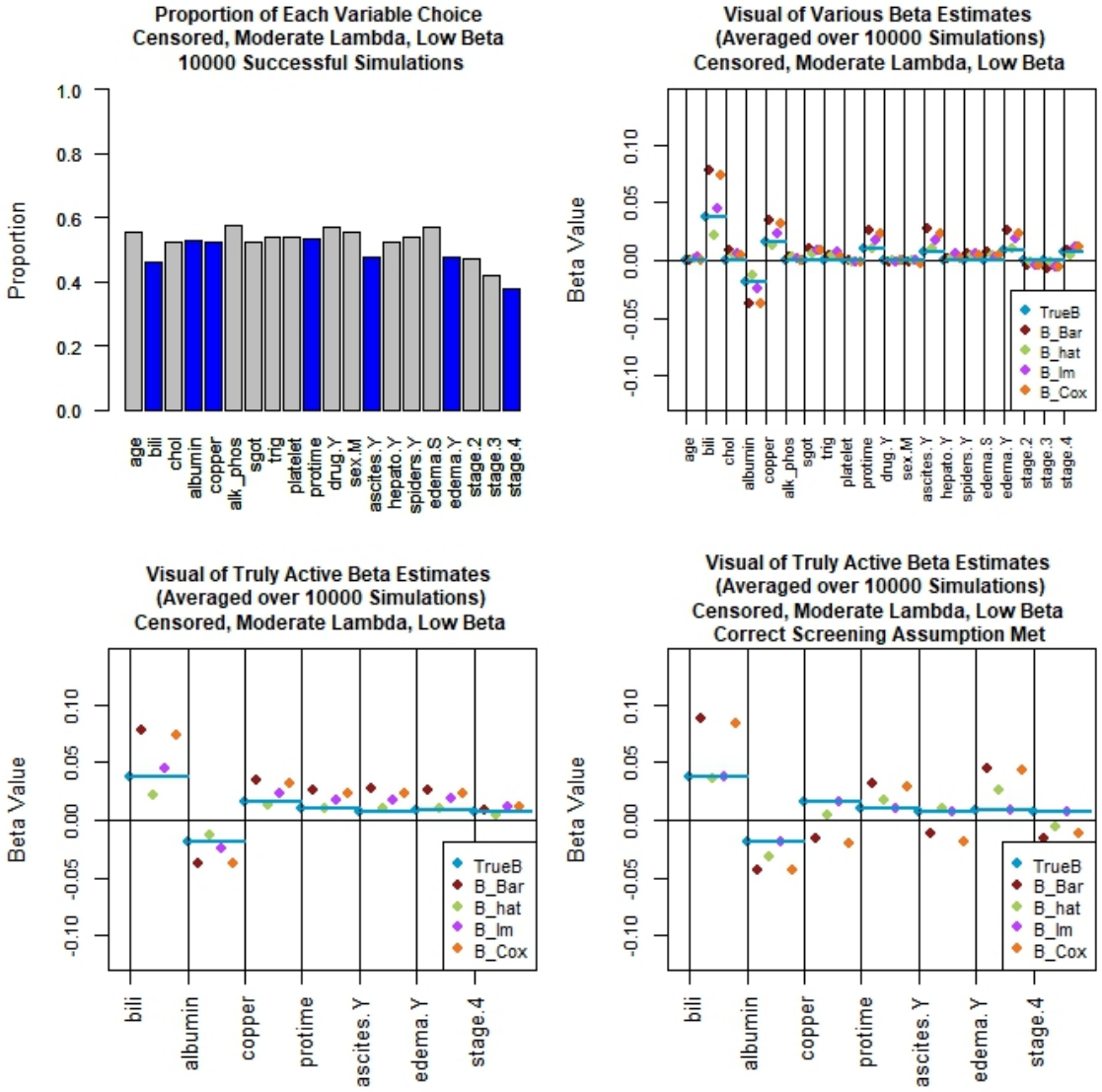
Tibshirani R. (1996) Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society, Series B, 58(1), 267-288.

Tibshirani R. (1997). The lasso method for variable selection in the Cox model. Statistics in medicine, 16(4), 385–395. https://doi.org/10.1002/(sici)1097-0258(19970228)16:4<385::aid-sim380>3.0.co;2-3

Tibshirani, R. J., Taylor, J., Lockhart, R., & Tibshirani, R. (2016). Exact Post-Selection Inference for Sequential Regression Procedures. Journal of the American Statistical Association, 111(514), 600–620. https://doi.org/10.1080/01621459.2015.1108848

Appendix A.

Visualize Results for Censored Data, Moderate λ, Low β:

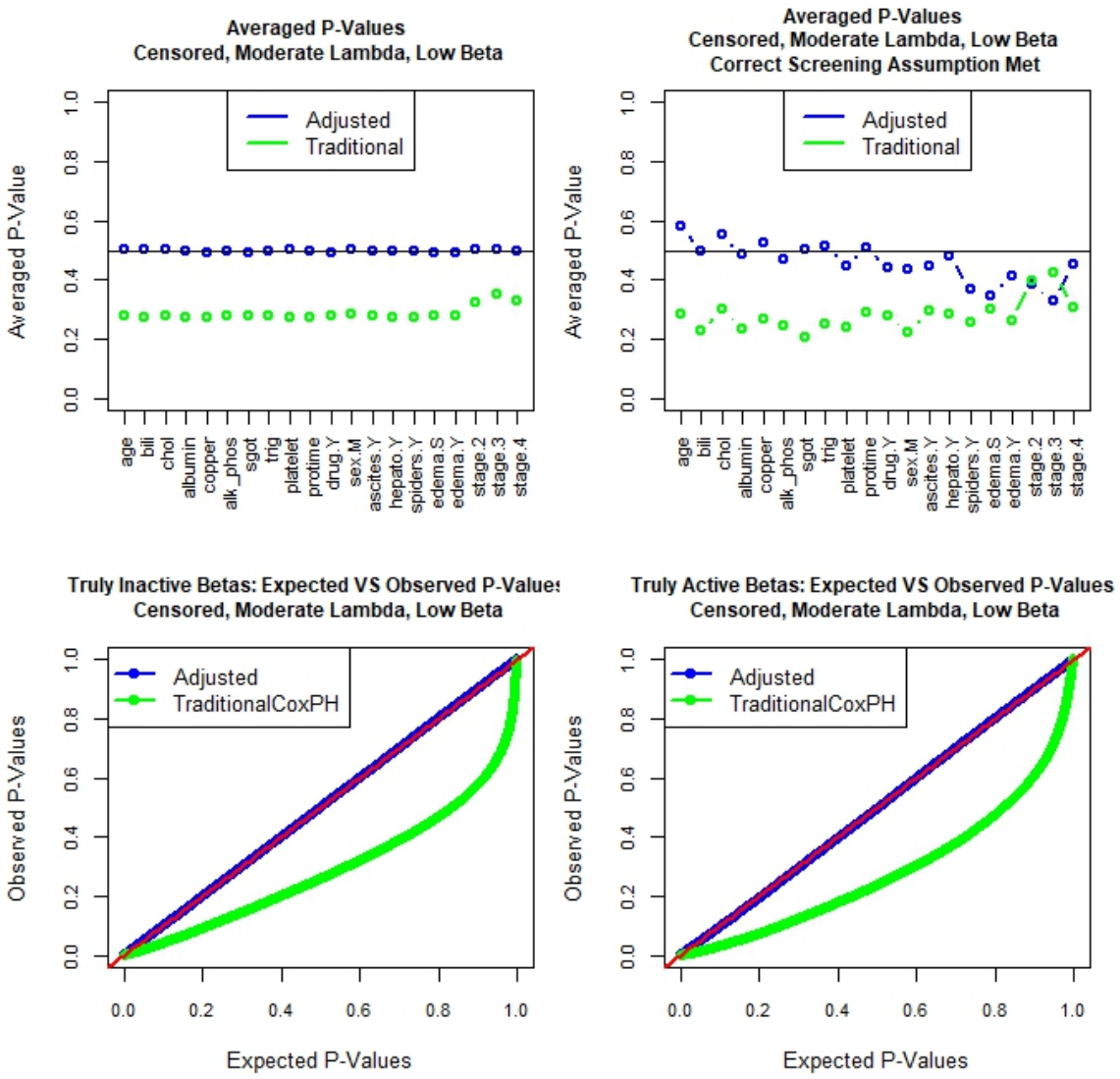


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

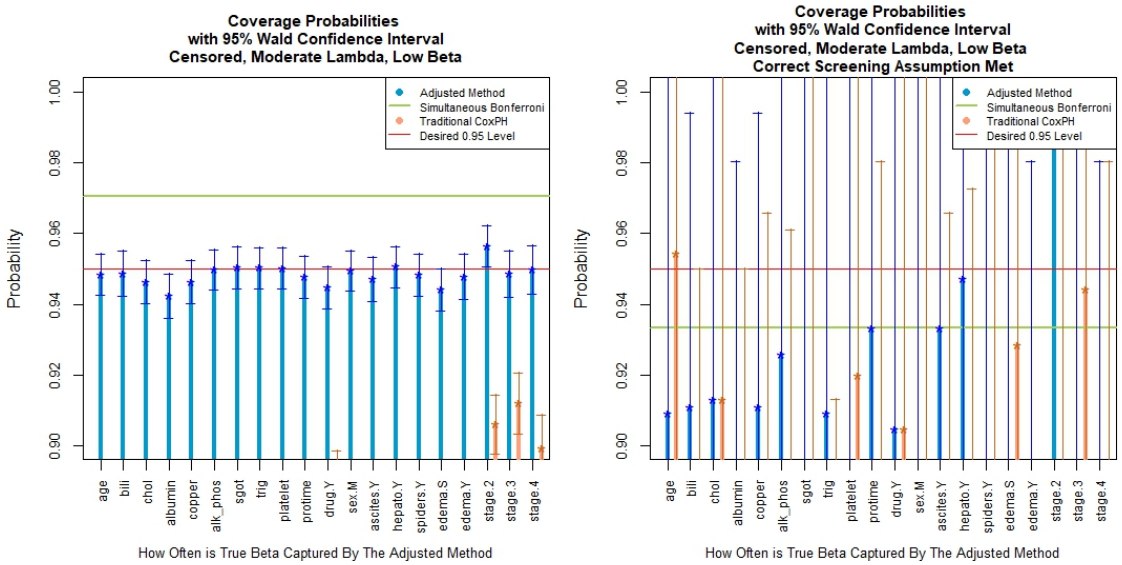
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

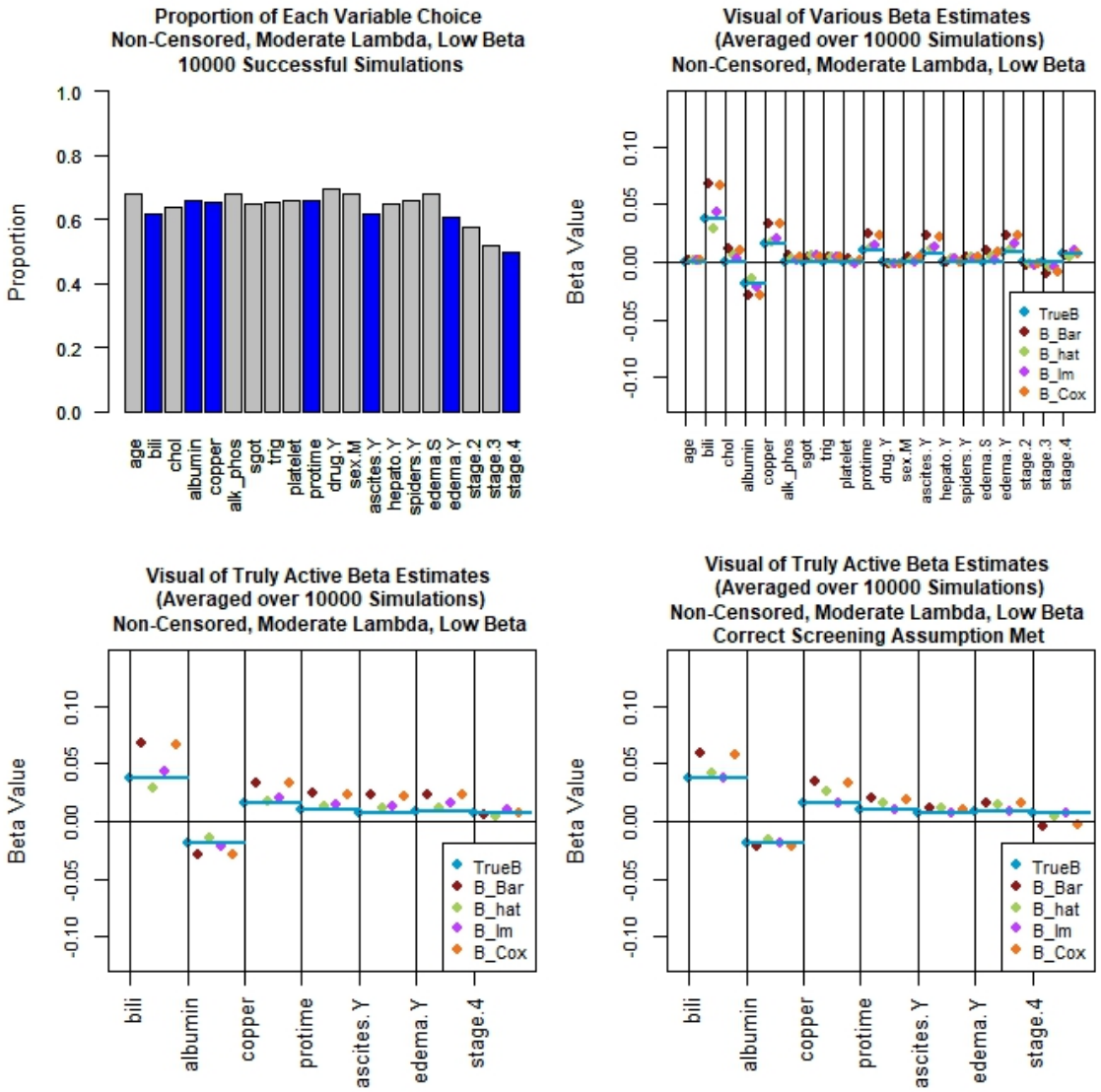


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered). Note that in this particular graph, the Bonferroni Simultaneous Corrected level is lower than anticipated, but this is due to the Bonferroni missing the value 3 times out of the 45 models where the assumption was met.

Visualize Results for Non-Censored Data, Moderate λ, Low $\boldsymbol{\beta}$:

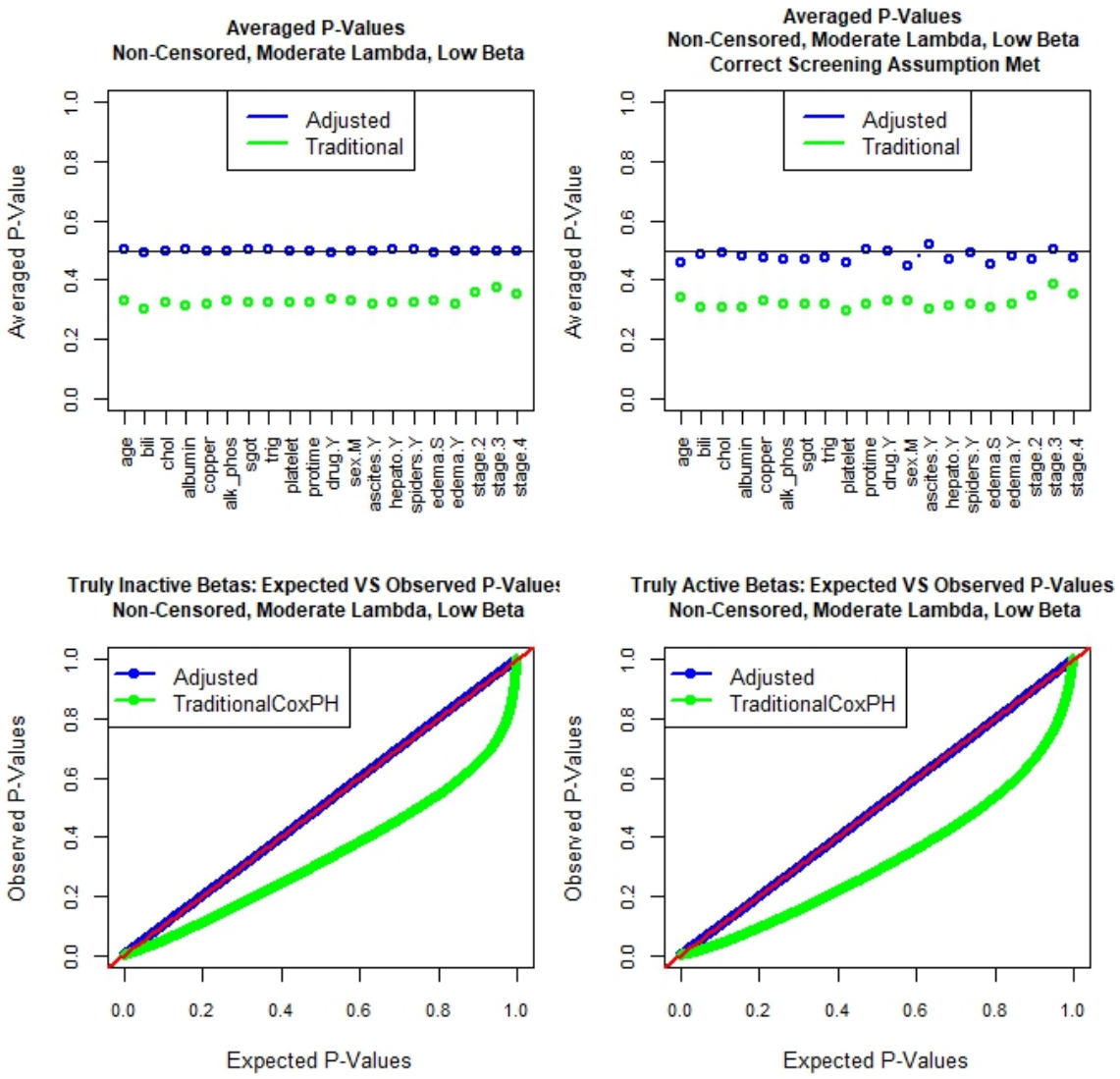


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

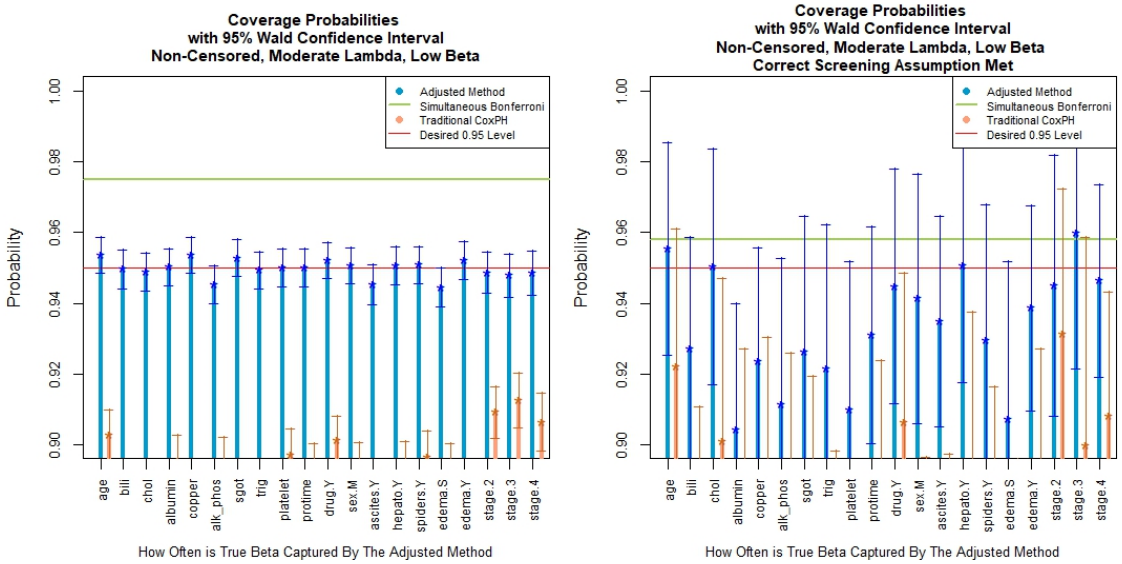
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

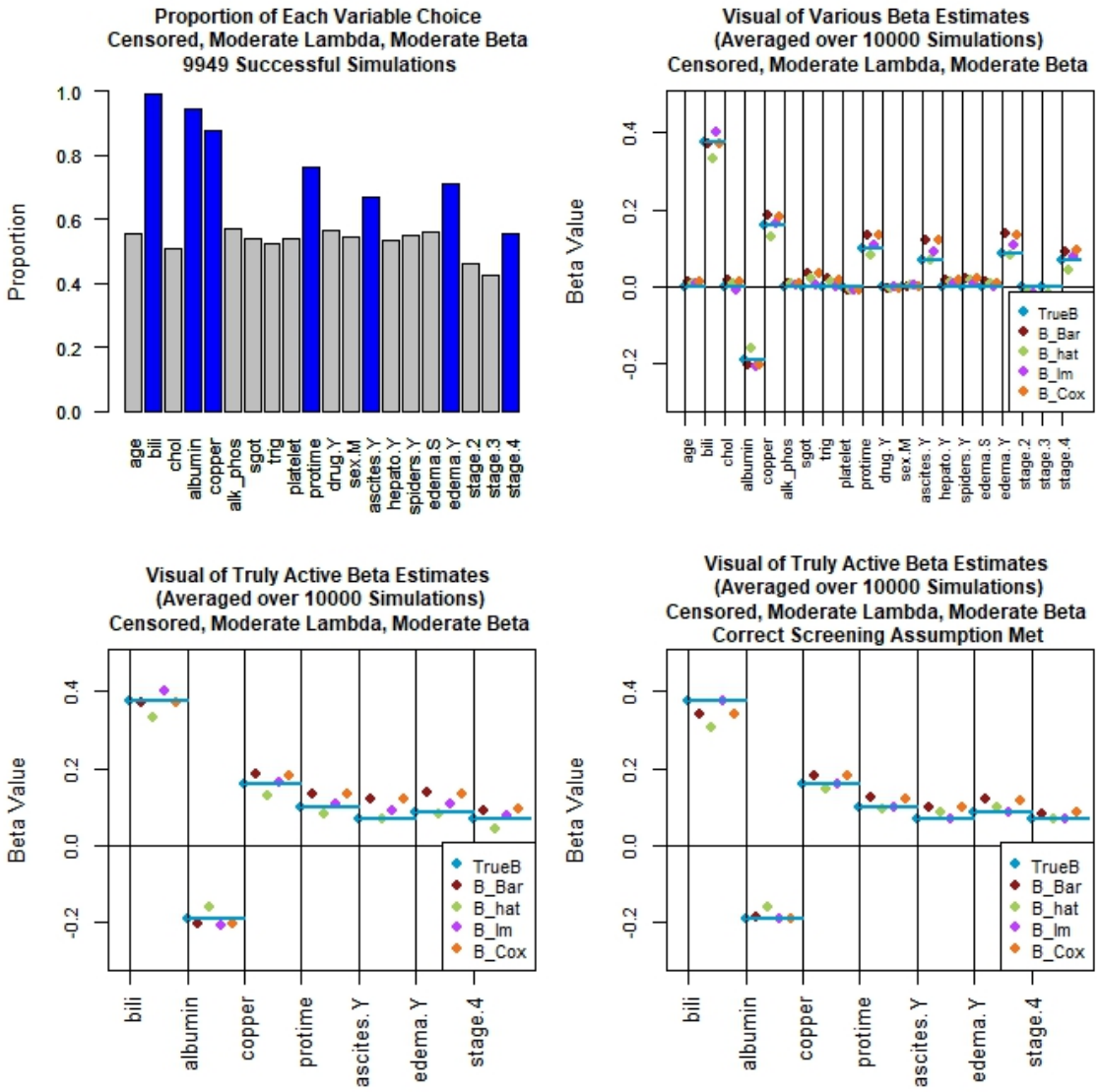


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).

Visualize Results for Censored Data, Moderate λ, Moderate $\boldsymbol{\beta}$:

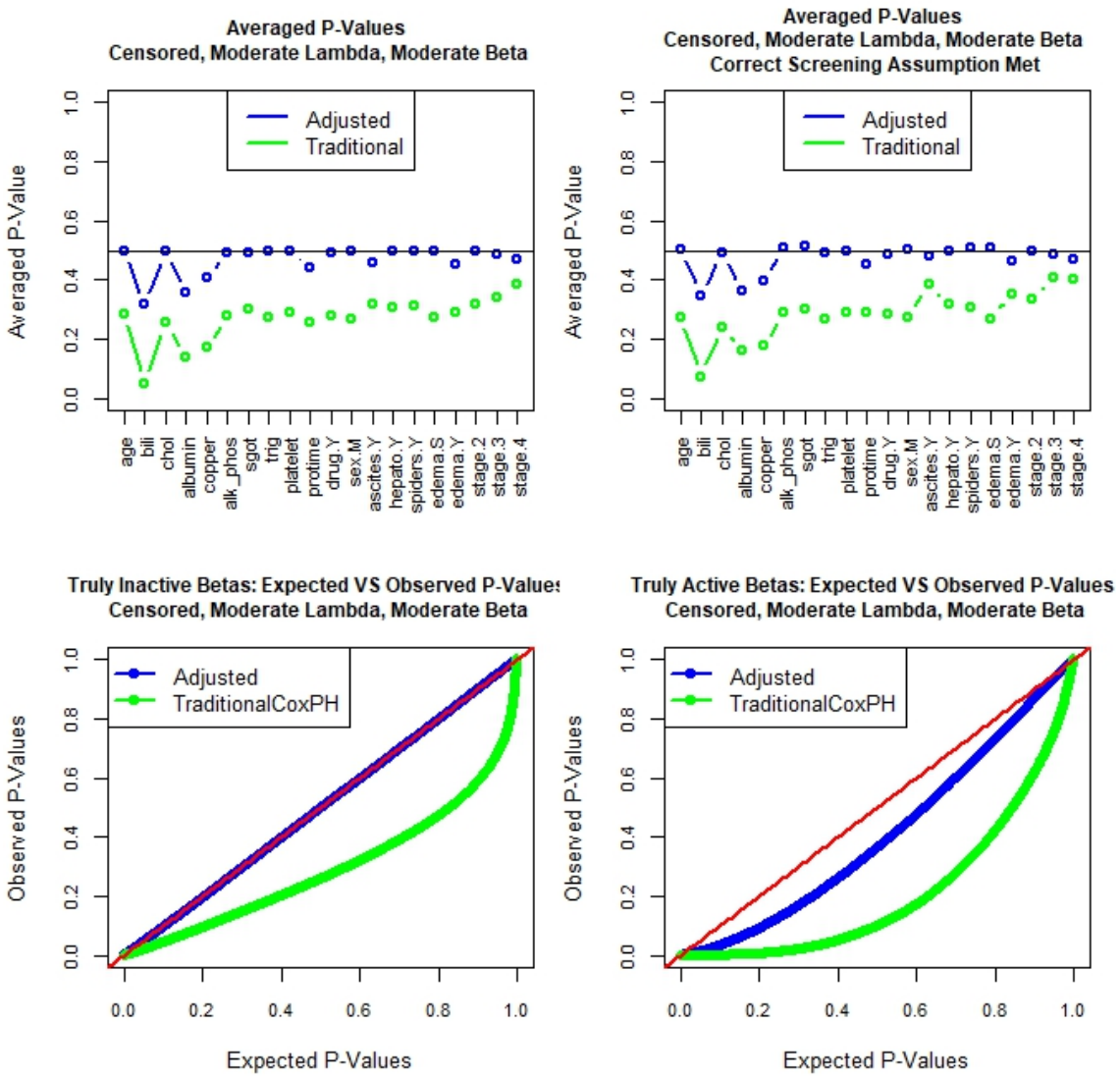


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

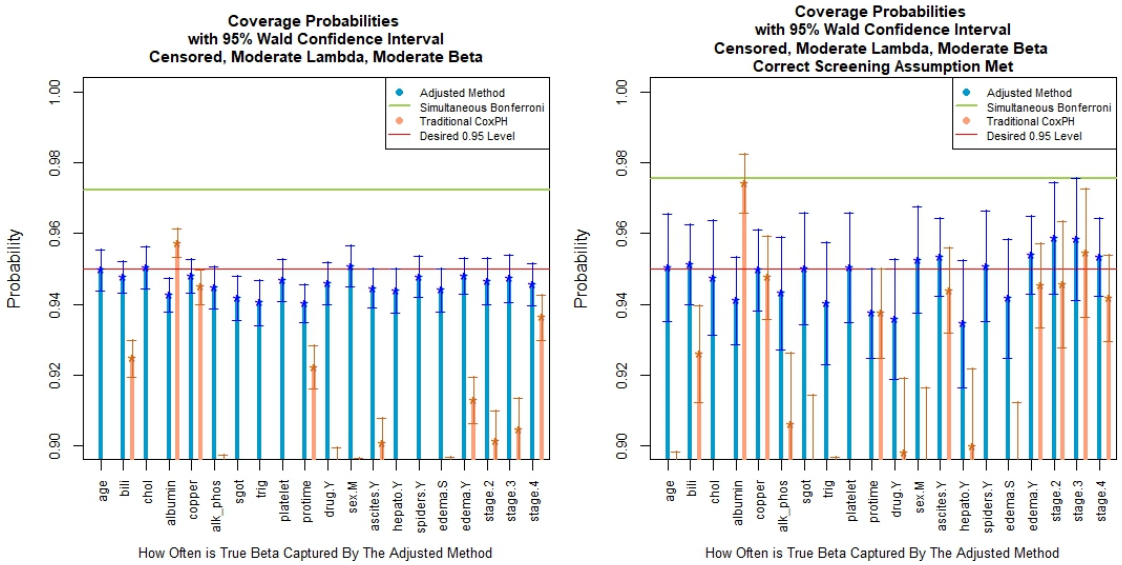
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

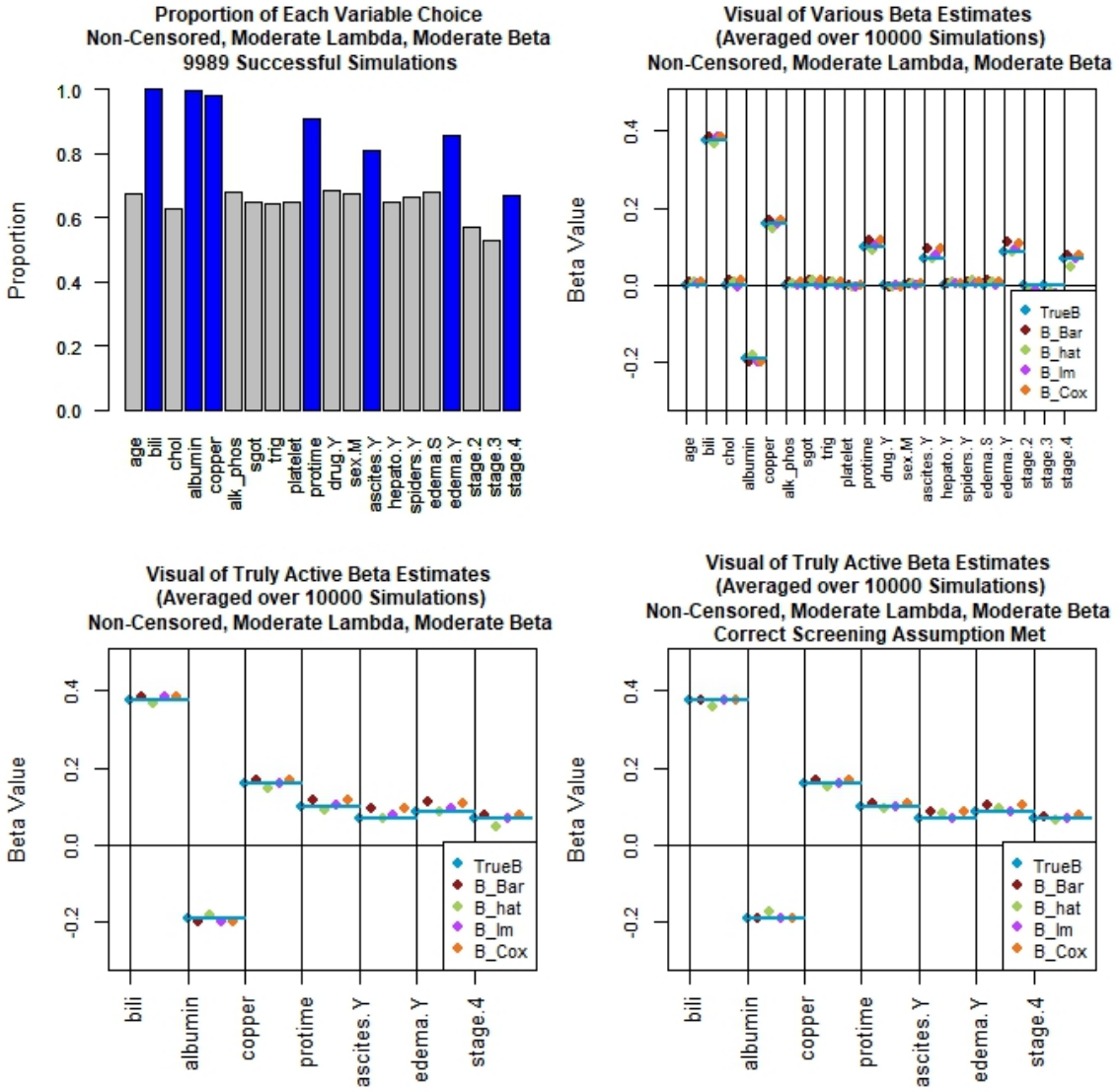


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).

Visualize Results for Non-Censored Data, Moderate λ, Moderate β:

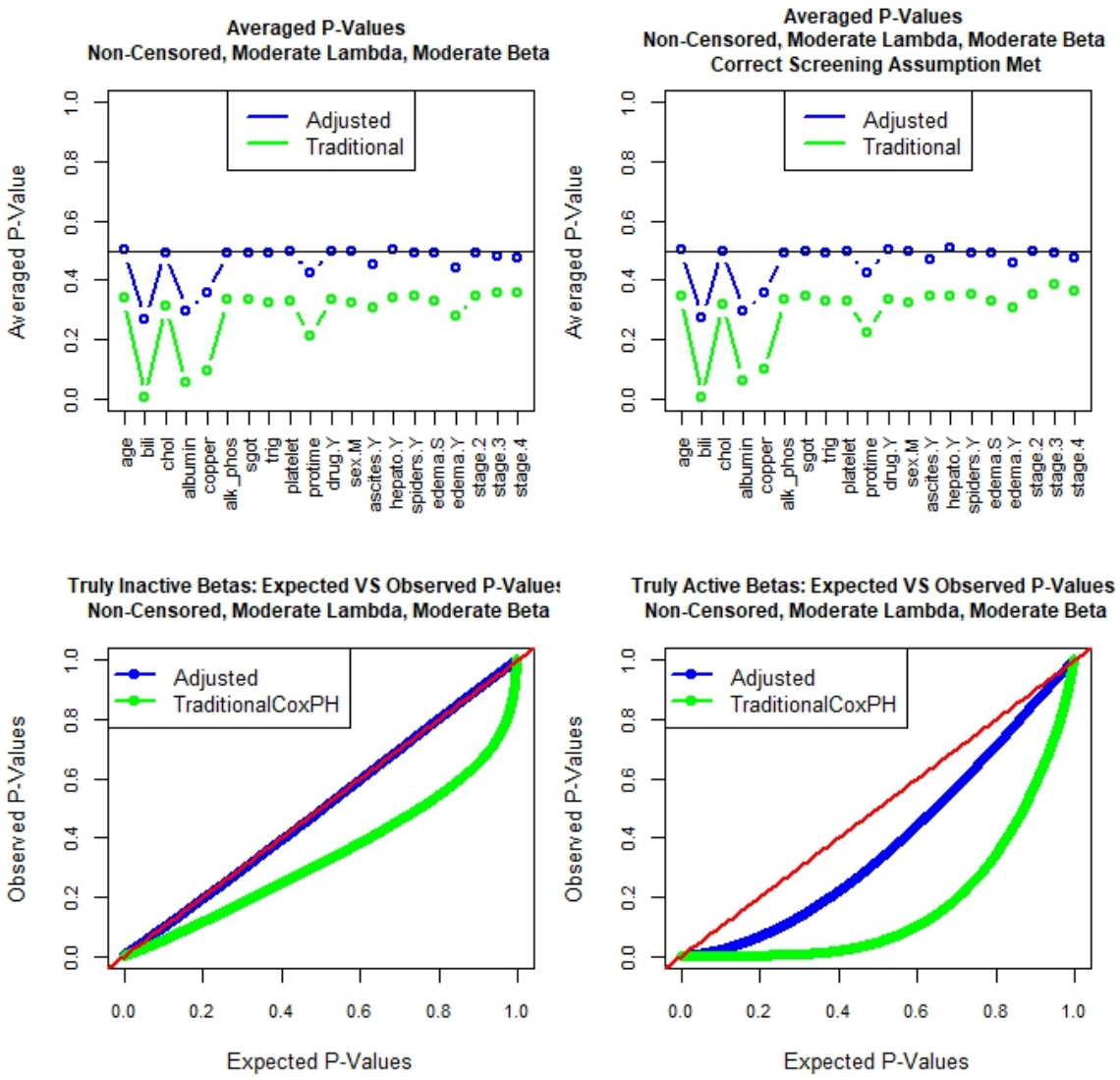


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

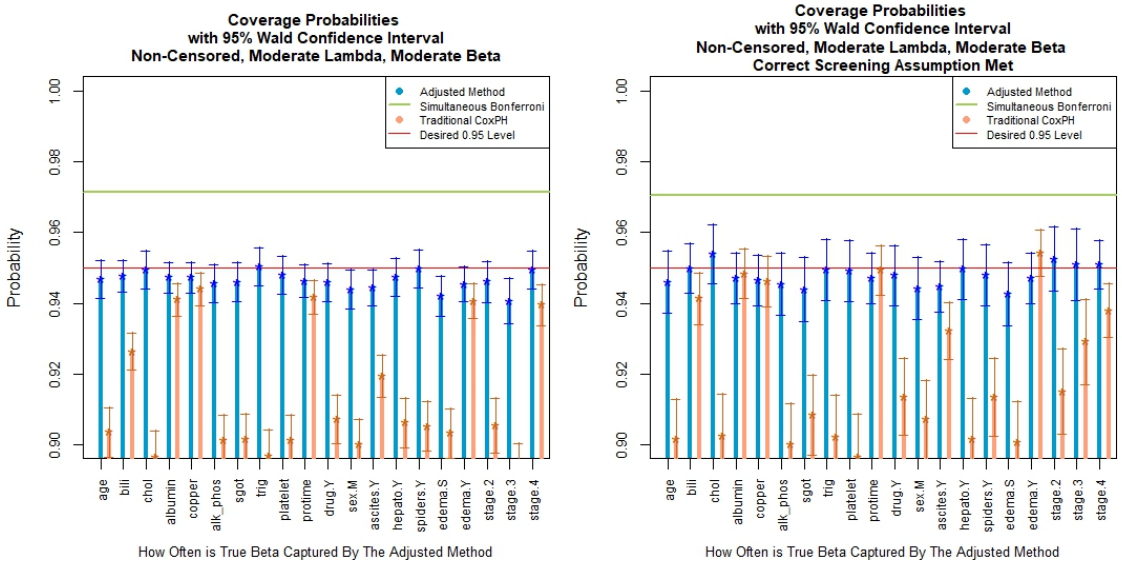
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

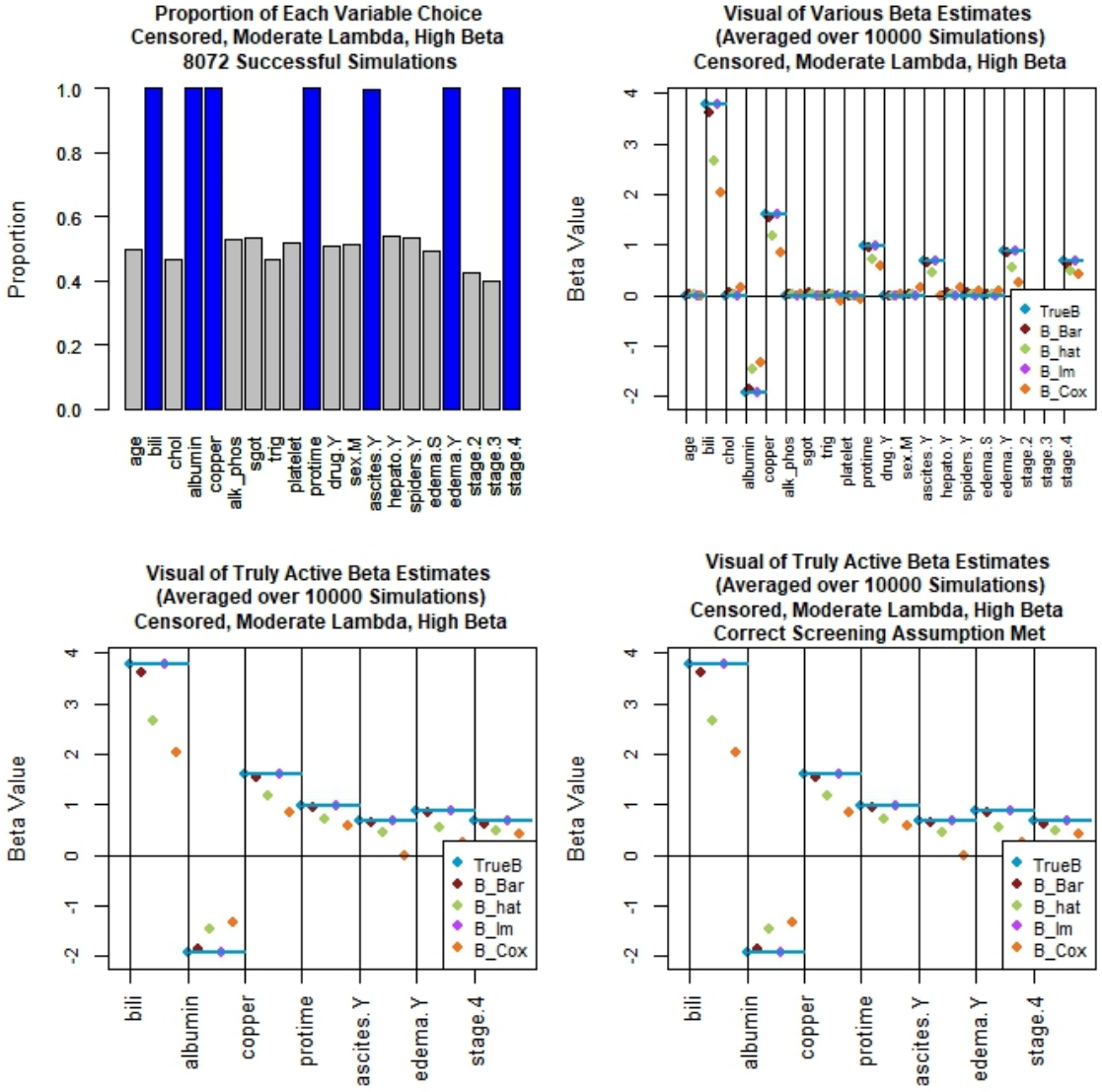


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).

Visualize Results for Censored Data, Moderate λ, High β:

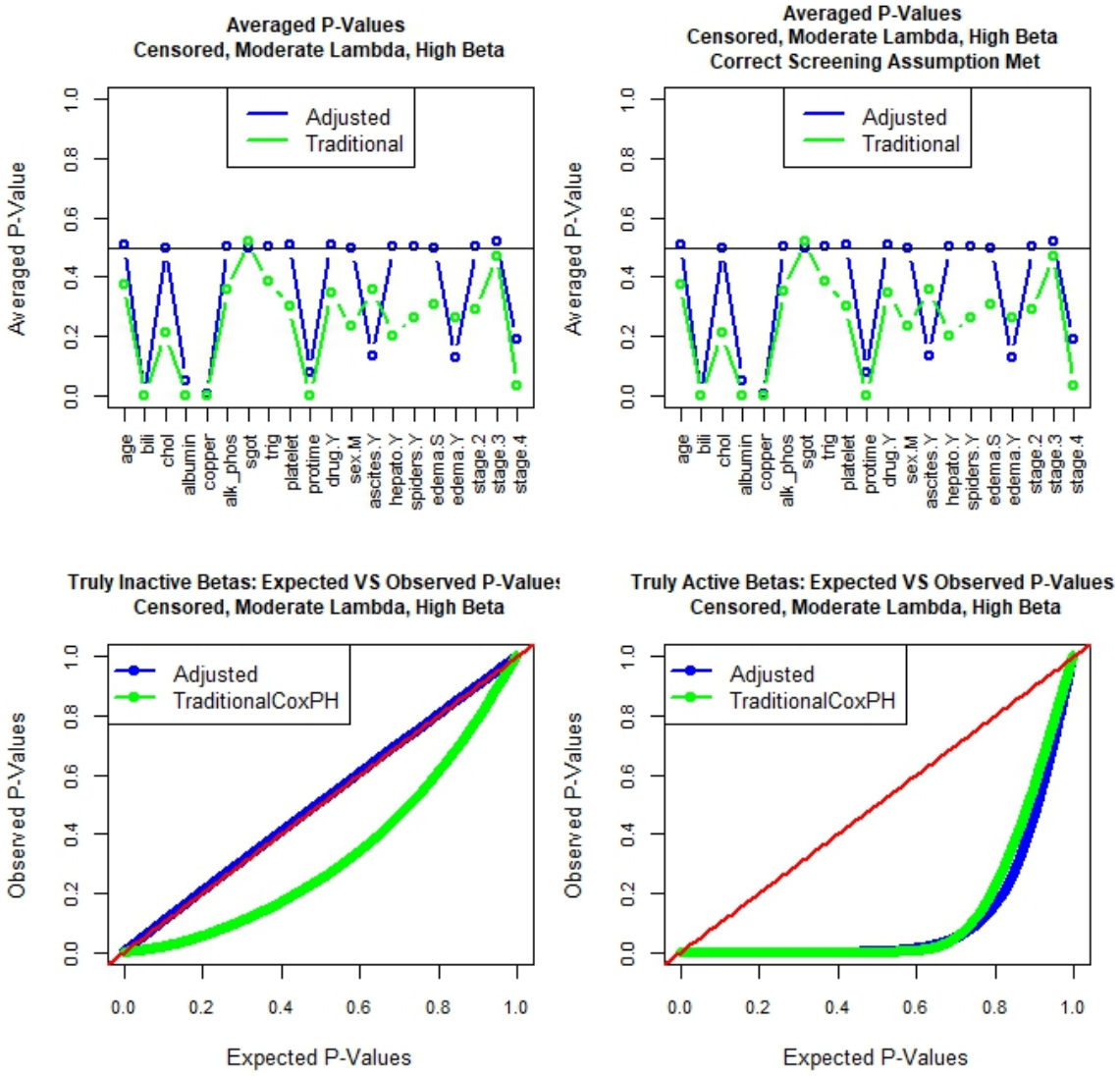


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

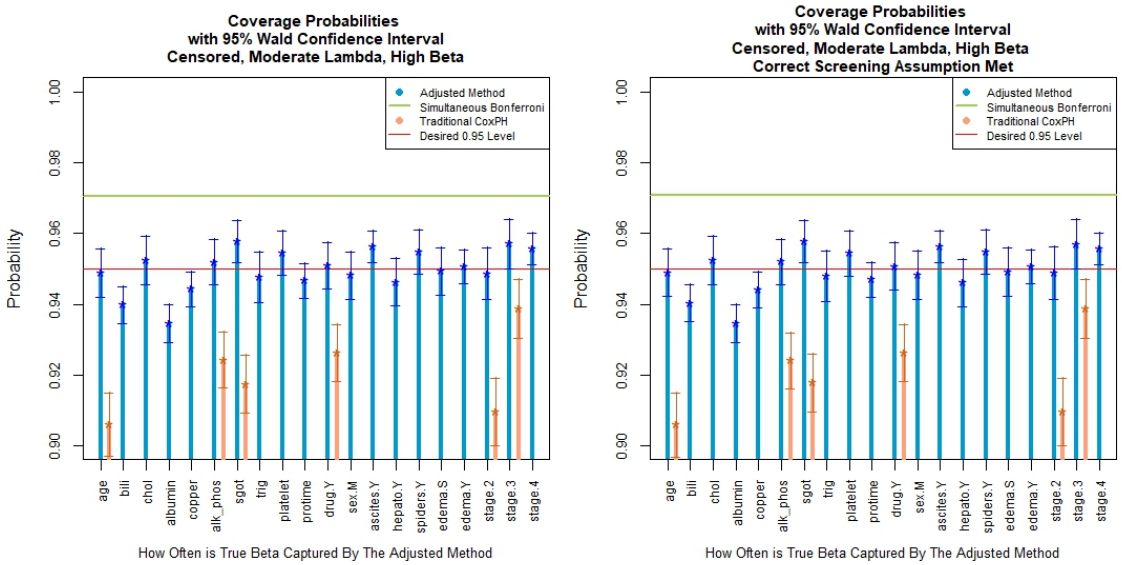
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

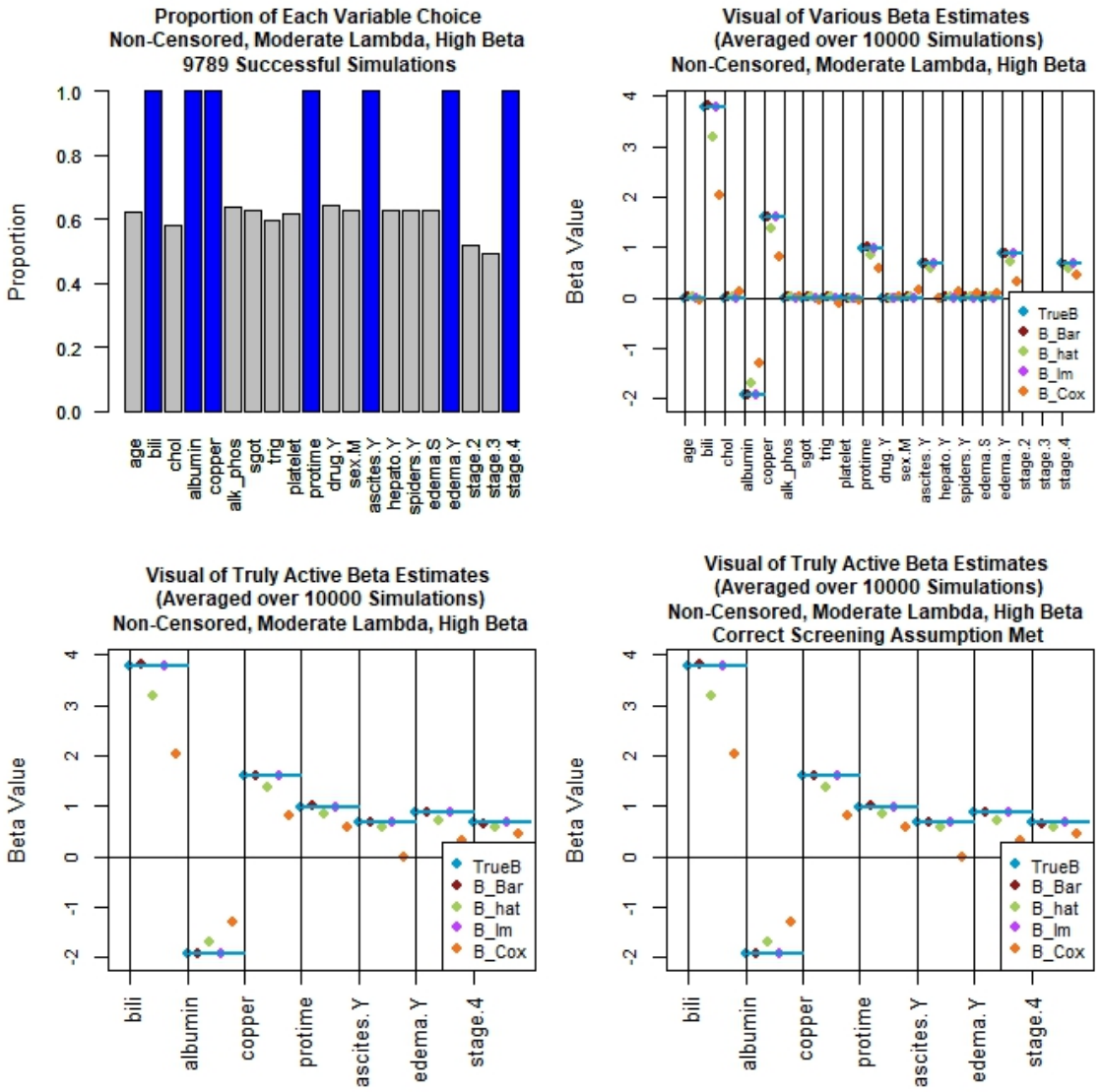


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).

Visualize Results for Non-Censored Data, Moderate λ, High $\boldsymbol{\beta}$:

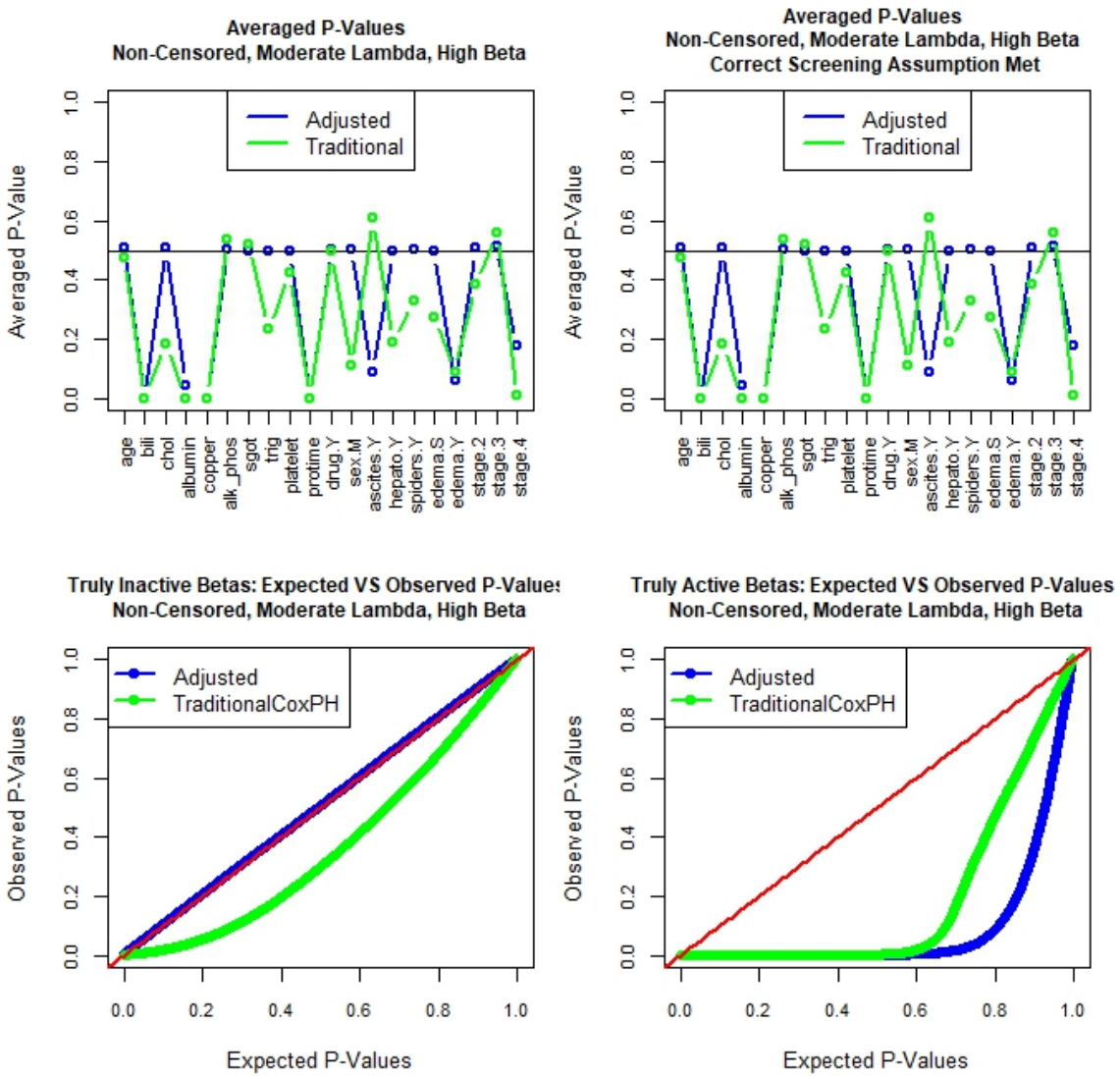


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

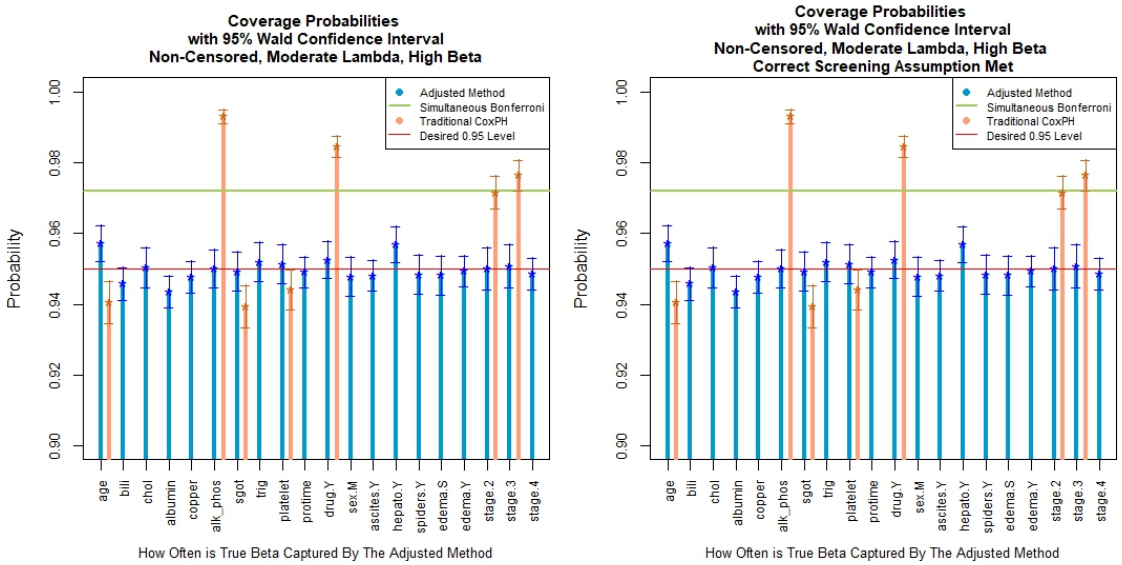
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

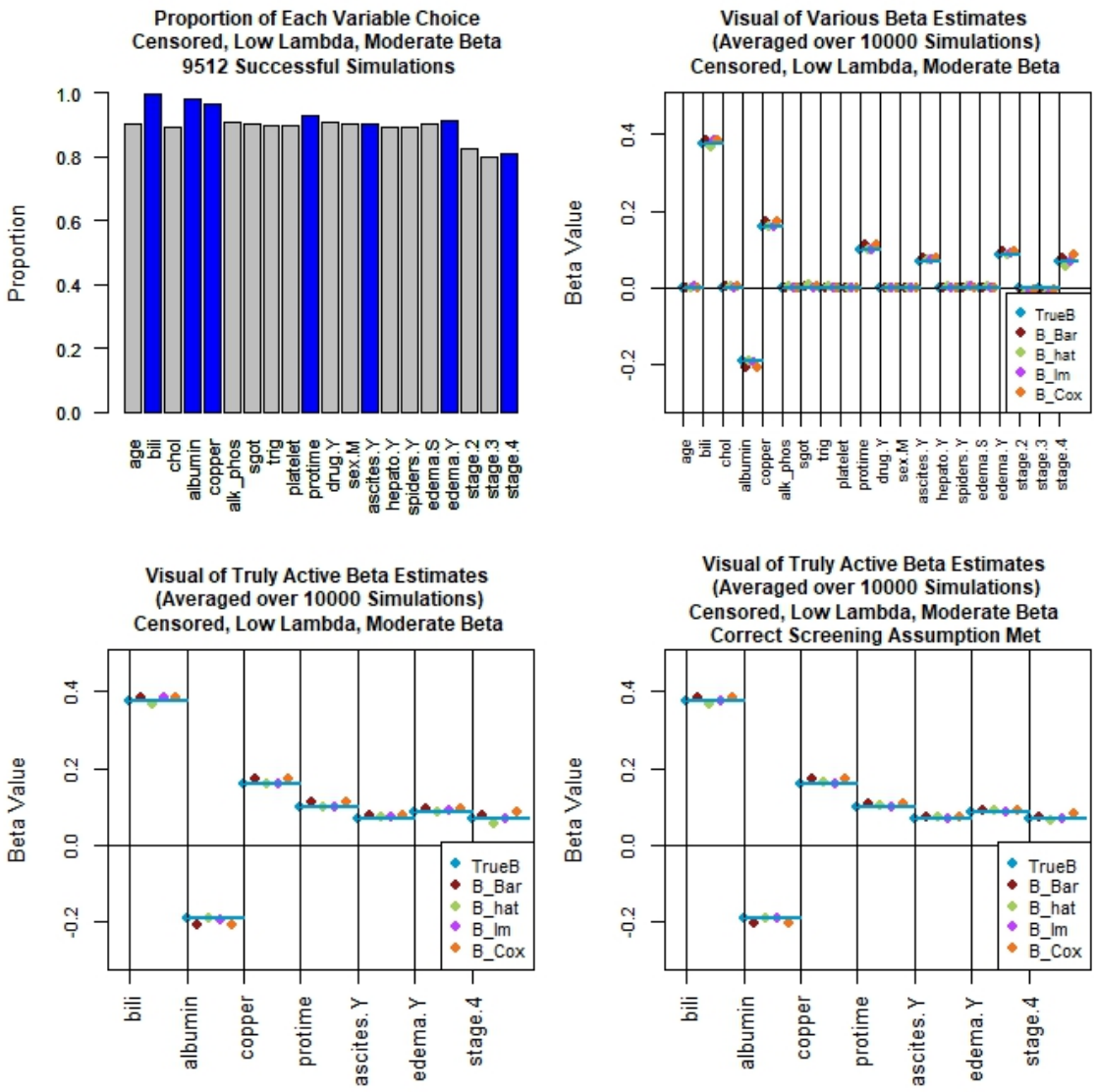


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).

Visualize Results for Censored Low λ, Moderate β:

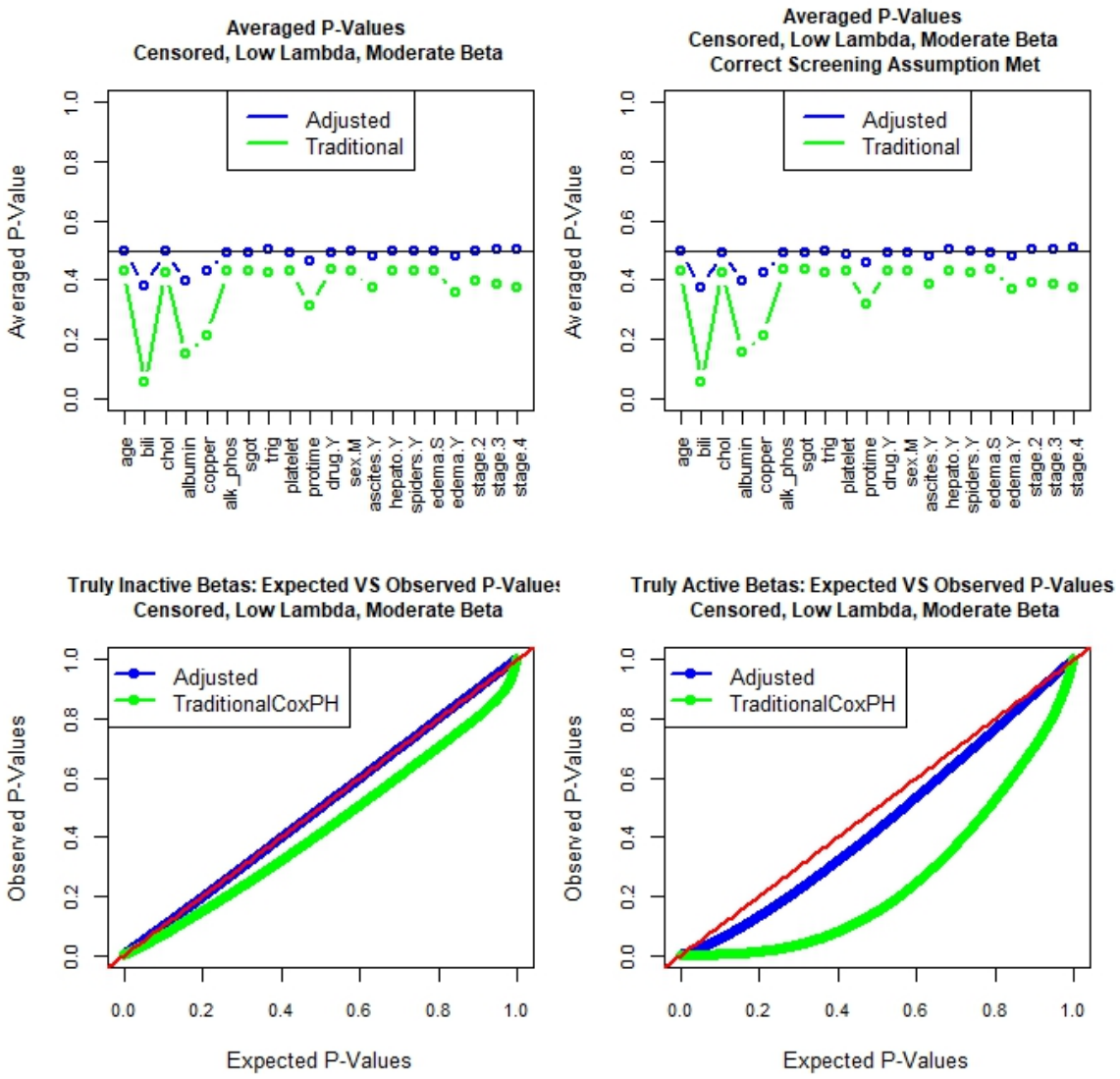


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

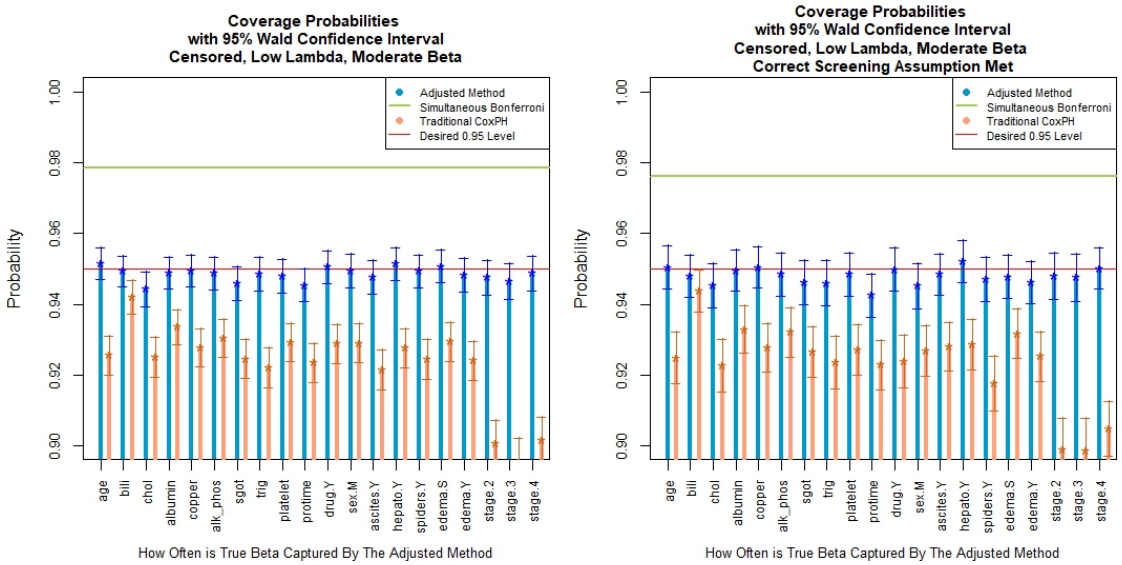
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^*_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

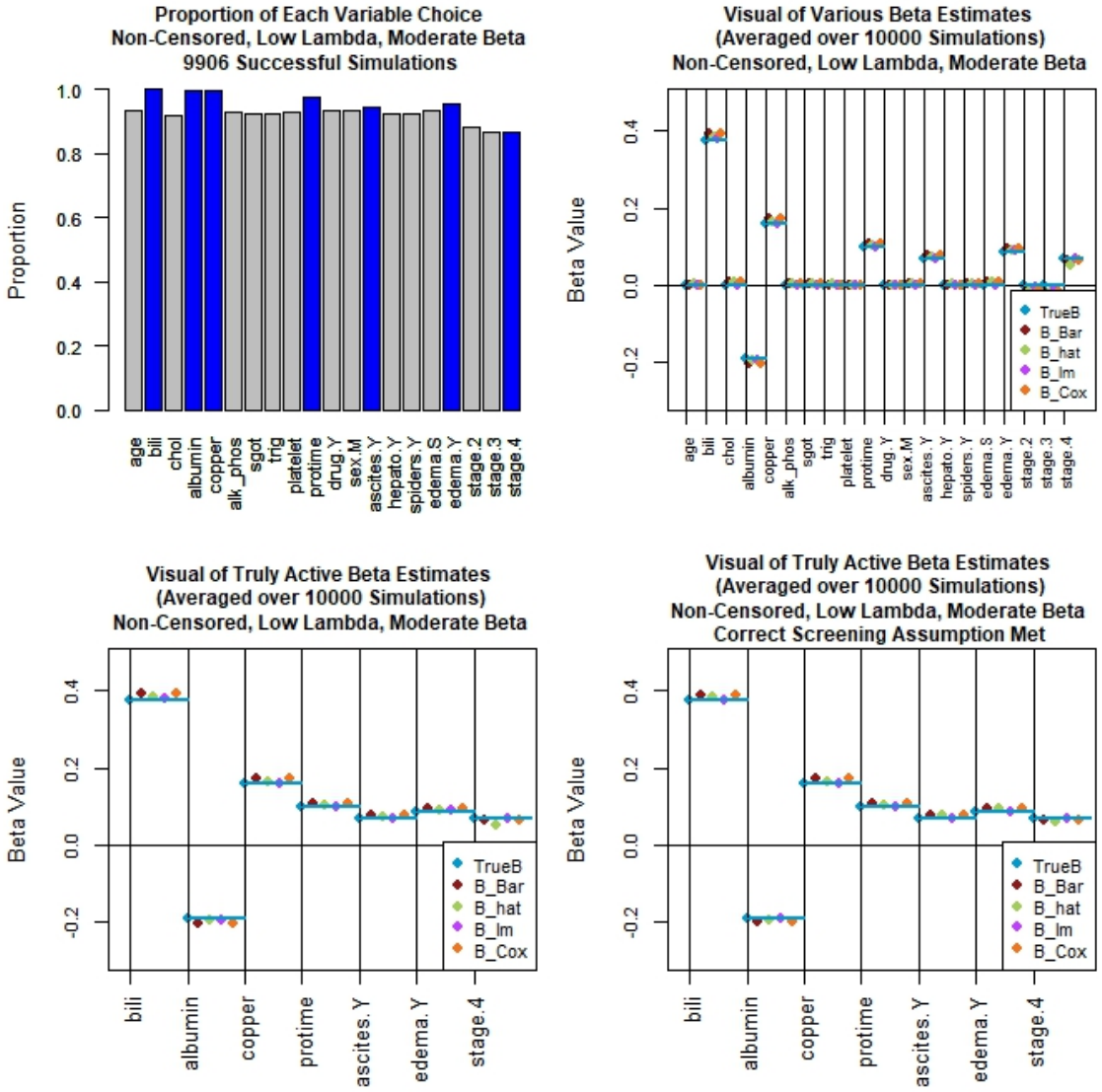


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).

Visualize Results for Non-Censored Data, Low λ, Moderate $\boldsymbol{\beta}$:

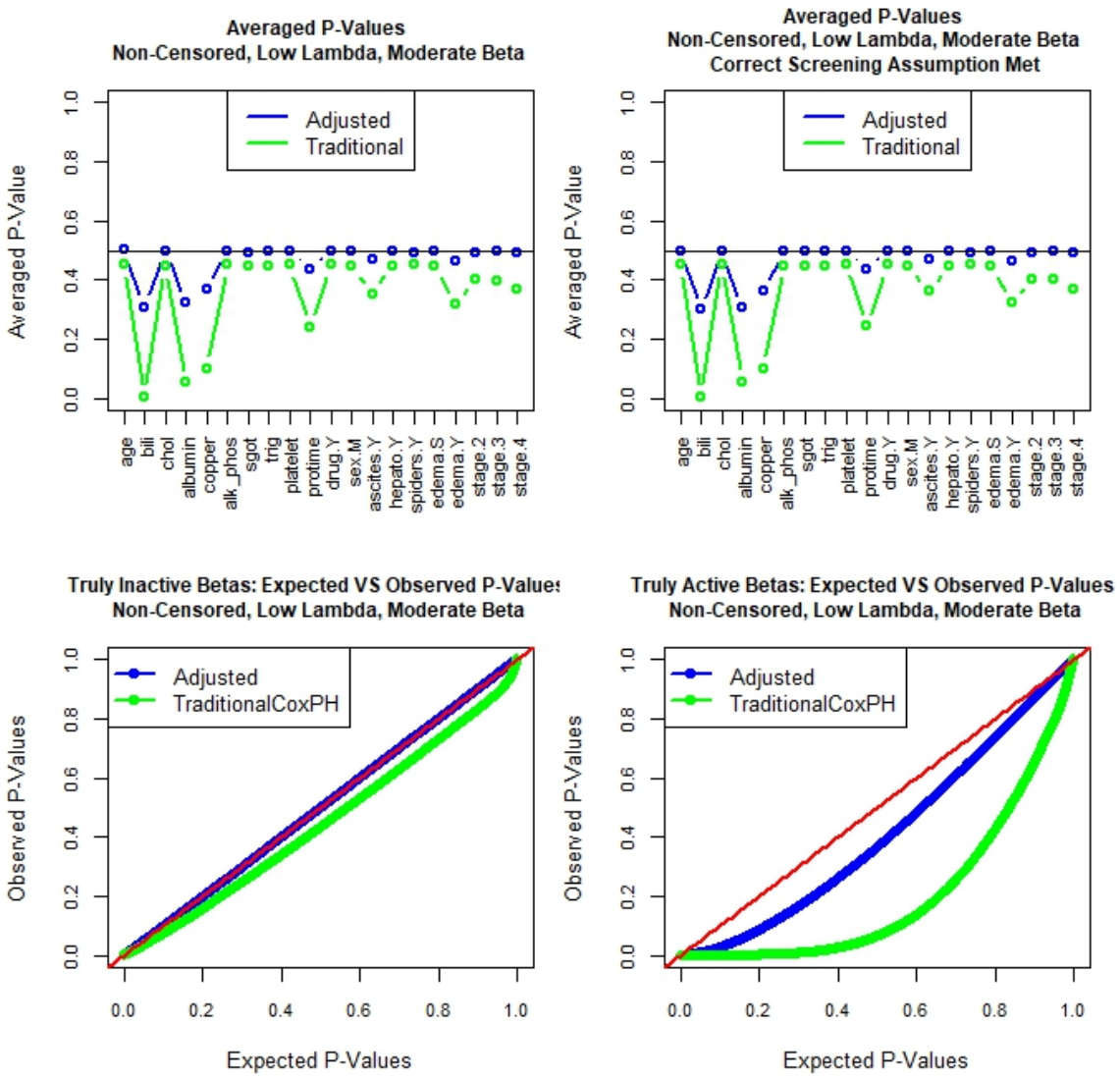


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

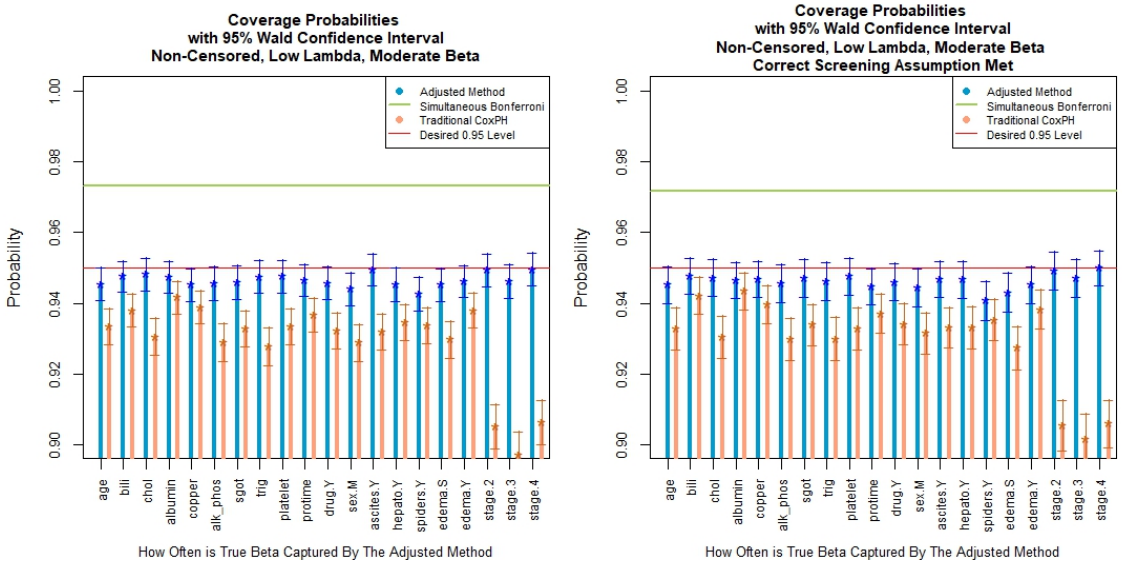
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

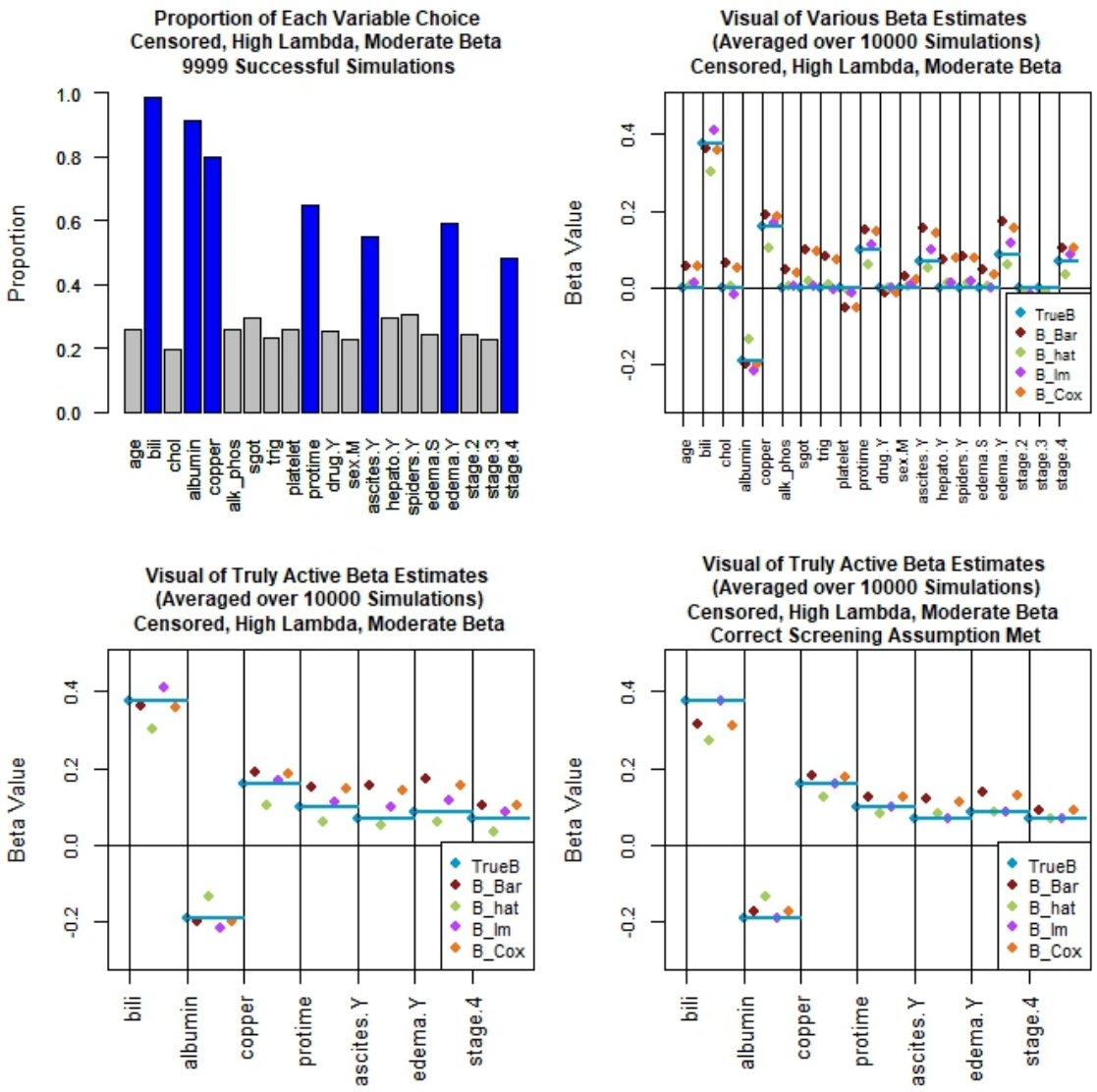


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).

Visualize Results for Censored Data, High λ, Moderate β:

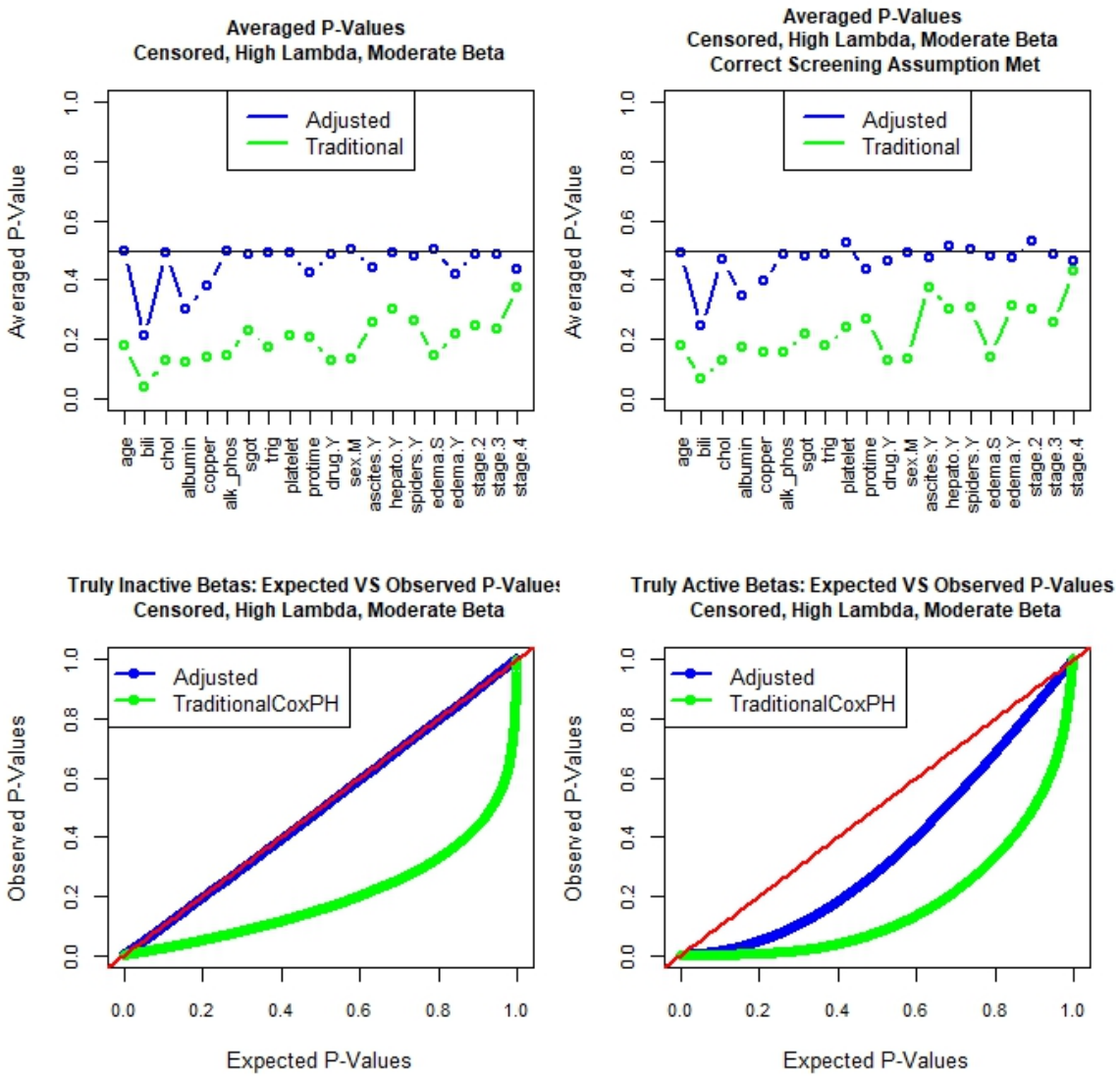


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

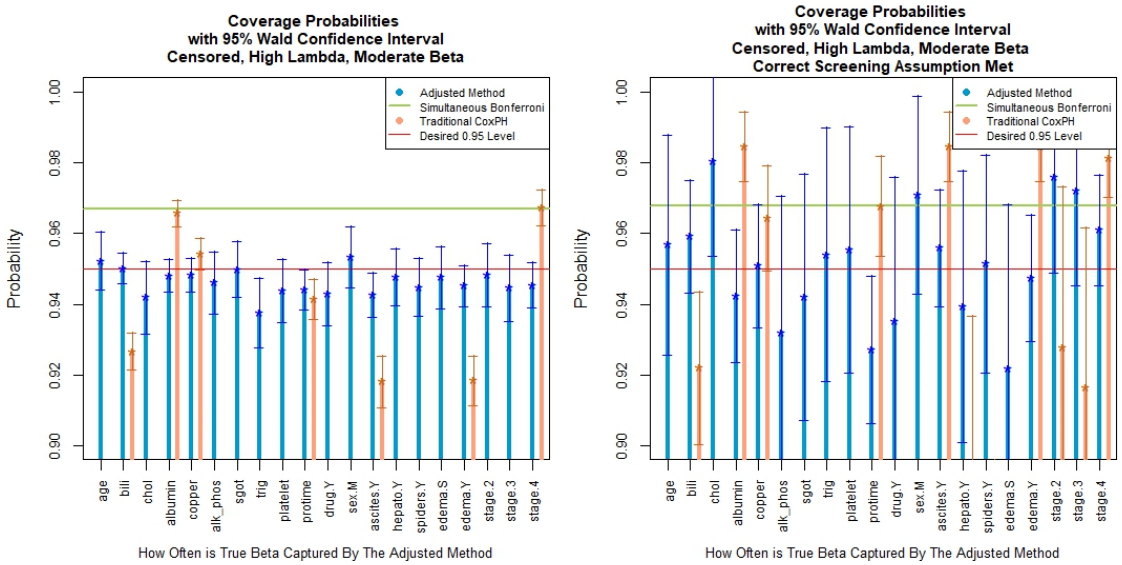
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations

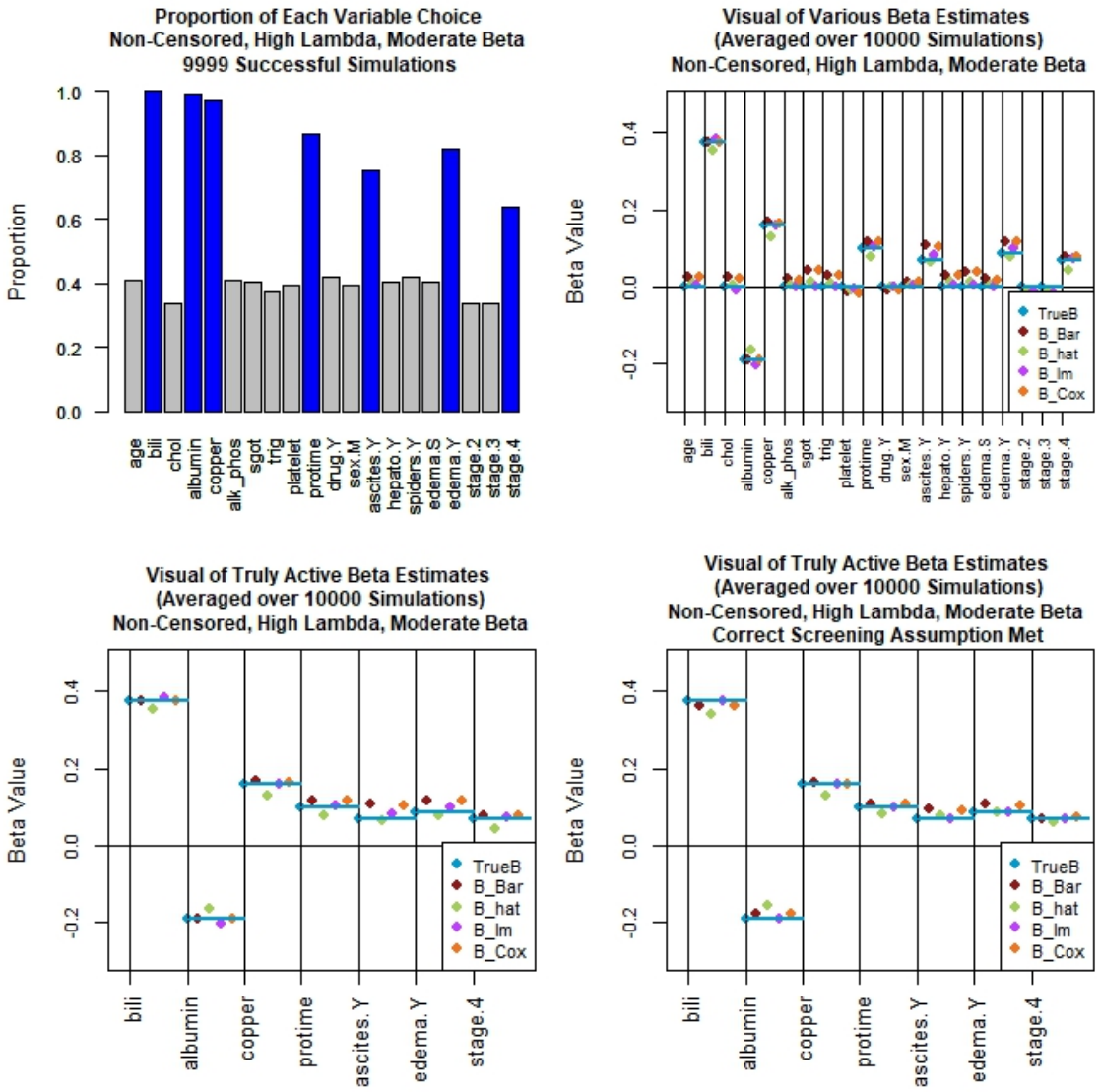


Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).

Visualize Results for Non-Censored Data, High λ, Moderate $\boldsymbol{\beta}$:

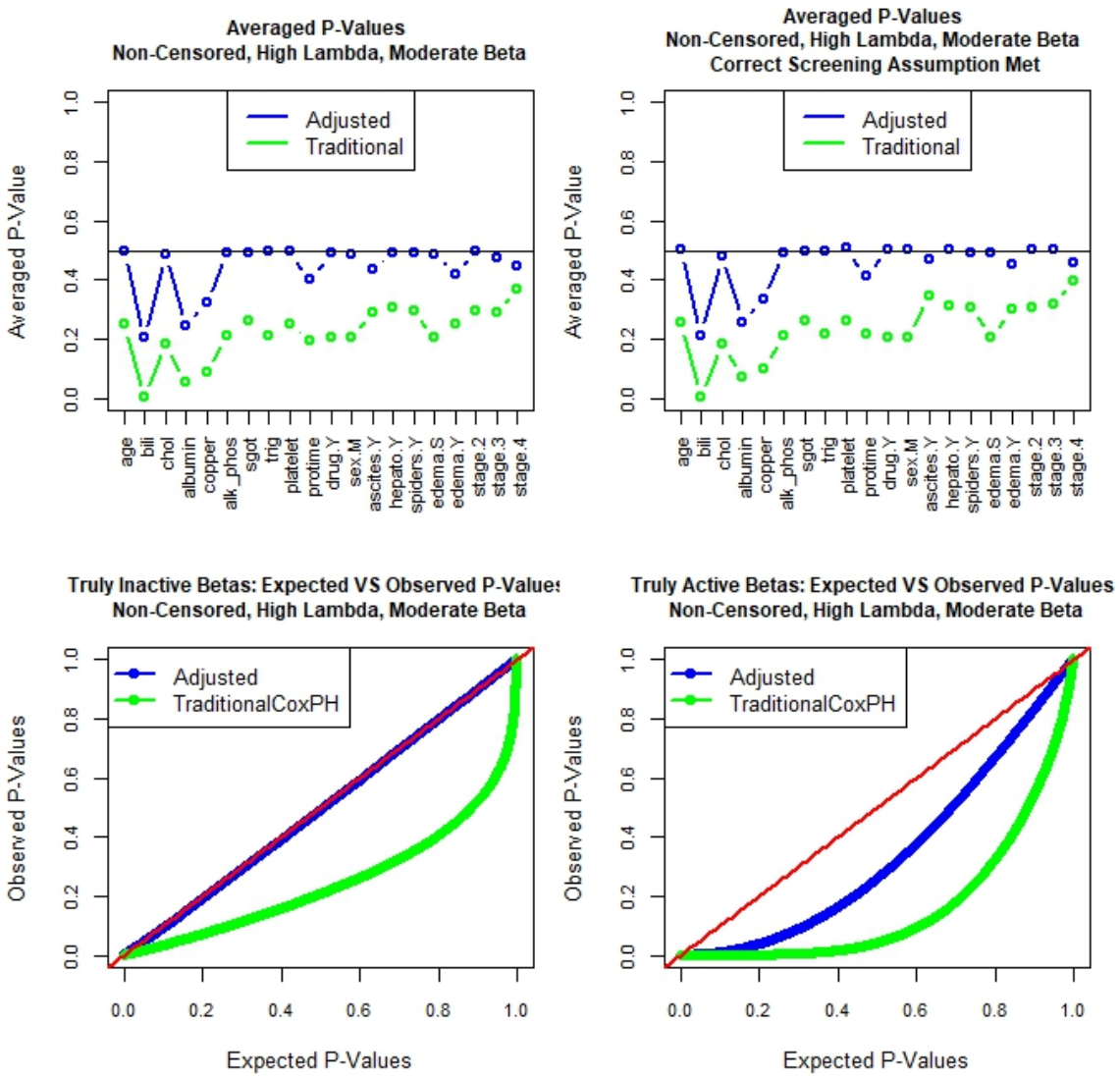


Top Left: Histogram to visualize the probability each variable is chosen in the current setting; the proportion is calculated as number of times chosen/number of successful simulations. Note that LASSO struggles to correctly screen variables with smaller true parameter values (they appear insignificant)

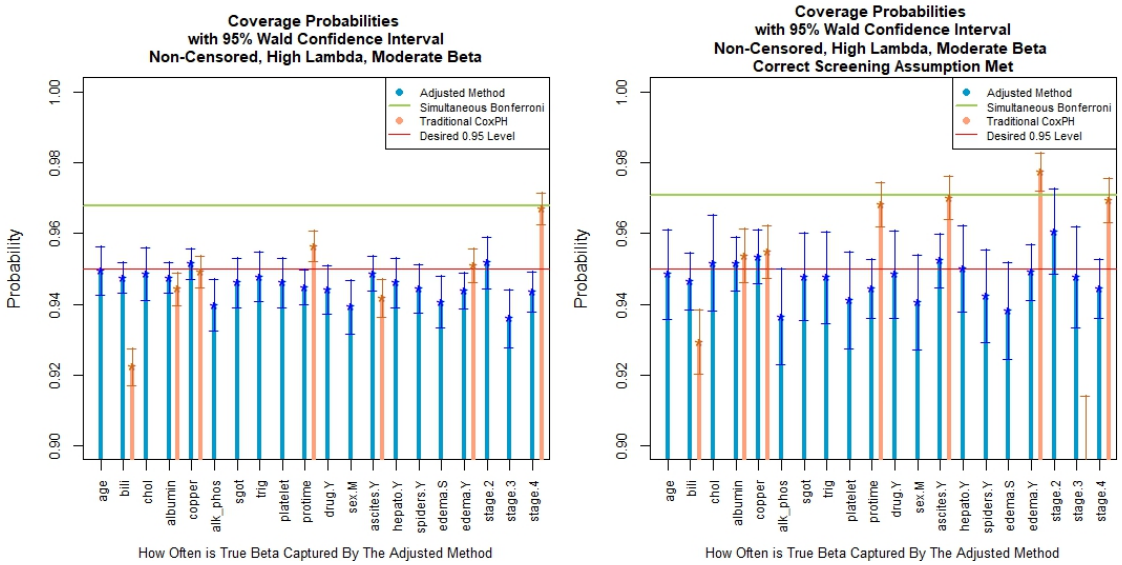
Top Right: Visual of various estimates from different methods on each variable. Values are calculated as sum of all estimates (of one type)/number of estimates made (of same type). TrueB = $\boldsymbol{\beta}$; B_Bar = $\overline{\boldsymbol{\beta}}_{\widehat{M}}$; B_hat = $\widehat{\boldsymbol{\beta}}_{\widehat{M}}$; B_lm = $\boldsymbol{\beta}^{*}_{\widehat{M}}$; B_Cox = $\widehat{\boldsymbol{\beta}}_{CoxPH_{\widehat{M}}}$

Bottom Left: Similar to Top Right, all simulations

Bottom Right: Similar to Top Right, but only correctly screened simulations



Top Left: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; all models were considered for the calculations.
Top Right: Average p-value for each variable, calculated as the sum of the p-values/number of p-values calculated; only models with correct variable screening were considered for the calculations.
Bottom Left: Expected VS Observed P-Values for all truly inactive $\boldsymbol{\beta}$ across all models; values along red line indicates proper Type 1 Error rate control, while values below red line indicate higher Type 1 Error rate than allowable
Bottom Right: Expected VS Observed P-Values for all truly active $\boldsymbol{\beta}$ across all models; values below red line indicates power of test while values along or above red line indicate no sensitivity and thus poor power



Left: Display of coverage probability for each variable across all simulations. Coverage probability estimate is determined using methods discussed in Chapter 4. Note that these coverage probability estimates are binomial (either cover truth or do not), and thus the Confidence Interval included on the graph is a 95% Wald Confidence Interval based on a Binomial Random Variable. Right: Same calculations as the left graph, but only simulations where models were correctly screened are included (assumption must be met to be considered).