ABSTRACT

Bayesian Models for Unmeasured Confounder
in the Analysis of Time-to-Event Data

Wencong Chen, Ph.D.

Chairperson: James D. Stamey, Ph.D.

Observational studies that omit confounders are subject to bias. In this dissertation we consider the specific case of time-to-event data. We also provide both the Bayesian parametric and the semi-parametric "twin regression" approaches with distributional assumptions of an unmeasured confounding variable, and then we compare them with the naive model. This assumes we ignore the effect of the unmeasured confounder.

To explore the ability of bias adjustment from different sources of information, we offer a Bayesian parametric regression with a normal unmeasured confounder. We also develop a Bayesian semi-parametric proportional hazards model accounting for unmeasured confoundings with binary and normal distributions. We can see that the approaches adequately decrease the bias, even with a small validation size. Furthermore, we offer a novel Bayesian bias adjustment model when only summary statistics are available in the external validation data.

Finally, we discuss and obtain several sets of solutions for different sources of validation data, censoring rates and sample sizes through simulation studies.

Bayesian Models for Unmeasured Confounder

in the Analysis of Time-to-Event Data

by

Wencong Chen, B.Eco., M.S.B.A.

A Dissertation

Approved by the Department of Statistical Science

_____

Jack D. Tubbs, Ph.D., Chairperson

Submitted to the Graduate Faculty of
Baylor University in Partial Fulfillment of the
Requirements for the Degree
of
Doctor of Philosophy

Approved by the Dissertation Committee

_____

James D. Stamey, Ph.D., Chairperson

_____

John W. Seaman, Jr., Ph.D.

_____

Dean M. Young, Ph.D.

_____

Jack D. Tubbs, Ph.D.

_____

David J. Ryden, Ph.D.

Accepted by the Graduate School
May 2016

_____

J. Larry Lyon, Ph.D., Dean

Page bearing signatures is kept on file in the Graduate School.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Professor James D. Stamey, for his insightful guidance, generous support, and his constant encouragement. Thank you for believing in me.

I also wish to thank the other members of my dissertation committee. I am grateful to Professor John W. Seaman, Jr. for his insight into many questions. I also appreciate Professors Dean M. Young and Professor Jack D. Tubbs, for their sense of humor and thoughtful suggestions.

My special thanks goes to Mrs. Alissa G. Wiseman. I gratefully acknowledge her valuable technical writing help and professional comments. I would like to thank my Baylor classmates, Wenqi Wu, Michelle Marcovitz, Youjiao Yu, Angel Lu, Qi Zhou, and Jonathon Vallejo. Thanks for making my stay at Baylor University a memorable one and for inspirational discussions.

Drs. Robert L. Steiner and Charlotte C. Gard at New Mexico State University both encouraged and inspired me to pursue the advanced degree at Baylor. Thanksgoes to them for their instrumental guidance and continued support.

I am deeply indebted to my wife, Limei Wang, my son, Gavin M. Chen, my daughter, Zoe J. Chen, and my parents for their everlasting love, care, and encouragement throughout my doctoral program.

DEDICATION

To

James D. Stamey

CHAPTER ONE

Introduction

1.1 Unmeasured Confounder

The gold standard for experimental research is the randomized controlled trial. For a variety of reasons, e.g., the clinical outcomes are rare (as in most genetic diseases), the outcome of interest is far in the future (Black 1996), or ethical objections such as the clinician believes that the specific intervention is a benefit to the participant, the randomized controlled trial (RCT) cannot always be conducted. Instead, well-designed observational studies play an important role in deriving the evidence of clinical intervention, although the problem of confounding can lead to biased estimation (Lin, Psaty and Kronmal 1998).

A confounder is defined as a risk factor for the disease outcome, which is associated with the exposure variable under study, and it is not an intermediate step in the causal path between exposure and the disease outcome (Rothman, Greenland, and Lash 2008). However, the mixed relationship between a potential confounder, an exposure variable, and a disease outcome is not easily detected. For instance, the neighborhood level socioeconomic deprivation was a confounder in McCandless, Richardson, and Best (2012). Weinberg (1993) presented a special case in which the history of spontaneous abortion is not a confounder when the disease outcome is the occurrence of spontaneous abortion. In that case, a woman with a history of spontaneous abortion was already exposed to the risk factor, so there was no confounding to be adjusted for. An additional complication occurs when the confounder is unmeasured.

Schneeweiss (2006) illustrated the concept of confounding and unmeasured confounding with causal graphs. Let T denote the survival time, Z denote the mea-

sured confounder, X_1 be the treatment exposure indicator, and U be the unmeasured confounder. The relationship between X_1, Z, and T is presented in Figure 1.1. Assuming that there is an unmeasured confounder, the relationship is shown in Figure 1.2.

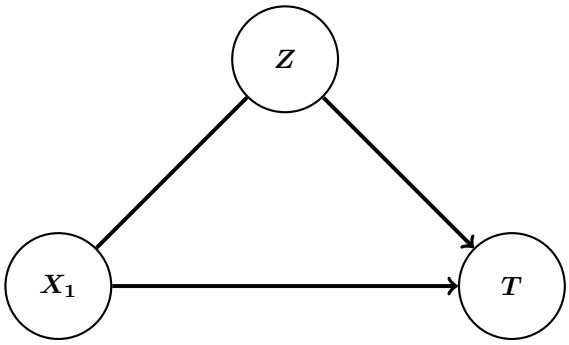

Figure 1.1: A graphical concept of a measured confounder

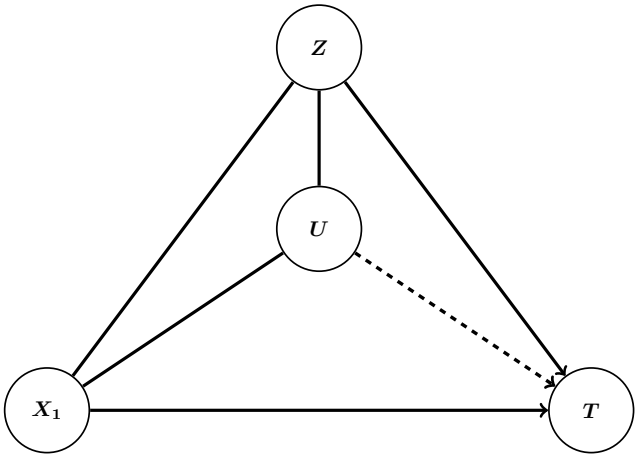

Figure 1.2: A graphical concept of a measured and an unmeasured confounder

Potential confounders can be variables such as family history, social status (Rothman et al. 2008), and household income. When the outcome is a disease, then the confounder may include body mass index (BMI), disease categories, stage of disease, specific diagnostic criteria, disease subtype, or degree of severity (Song and Chung 2010).

In an observational study, it is desirable to collect as much data as possible. Due to technical issues, limited budget, or ethical concerns, this is not always

possible as some confounders cannot be collected. This can lead to unmeasured confounding, like when the patient characteristics have made a doctor choose a specific intervention but is then not recorded (Collet 2003; Jepsen et al. 2004).

1.2 Motivation and Problems

1.2.1 Right-Heart Catheterization

It is a widespread belief that the direct measurement of cardiac function provided by right-heart catheterization (RHC) is necessary to guide therapy for certain critically ill patients and that such management leads to better health outcomes. To examine the association between the use of right heart catheterization during the first 24 hours of care in the intensive care unit, the survival time, length of stay, and cost of care were analyzed in the Study to Understand Prognoses and Preference for Outcomes and Risks of Treatments (SUPPORT) across five medical centers in the United States.

Connors et al. (1996) performed a thorough analysis of these data. All the major risk factors were identified by a panel, which consisted of four intensivists and three cardiologists. Furthermore, a propensity score for RHC was derived to adjust for the selection bias. Finally, they deployed the proportional hazards model that incorporated the propensity score, as well as the other major health risk factors.

Suppose that the primary outcome for SUPPORT is survival time. Let T_i denote the survival time of the i^{th} subject, x_i denote the i^{th} subject's RHC status, $z_{1i}, z_{2i}, \cdots, z_{qi}$ denote the risk factors in the SUPPORT (e.g., disease categories, heart rate, temperature, etc.), and let u_i denote an unmeasured risk factor. For $i = 1, 2, \cdots, 5735$, it follows that

$$P(T_i|x_i, \mathbf{Z}_i, u_i) = f(x_i, \mathbf{Z_i}, u_i), \tag{1.1}$$

where $\mathbf{Z} = (z_{1i}, z_{2i}, \cdots, z_{qi})$, and f is the function for log-normal regression, non-parametric methods, or a Cox proportional hazards model.

Although Connors et al. (1996) adjusted for the treatment bias with a large number of covariates, the possibility of missing an important confounder can never entirely be excluded in observational studies. Suppose the survival time T only depends on the status of RHC, x ($x = 1$ represents the subject with RHC), risk factors $\mathbf{Z}$, and an unmeasured risk factor U. Ignoring the association between X and U, we apply the Cox proportional hazards model for the survival time with the risk factor U. That is,

$$h(t|x, \mathbf{Z}, U) = h_0(t) \exp(\beta_1 x + \eta \mathbf{Z} + \lambda u), \tag{1.2}$$

where $\eta = (\eta_1, \eta 2, \cdots, \eta_q)$, and β_1 and λ are unknown parameters. Since X is a binary exposure indicator, we can decompose the effect for an unmeasured risk factor as follows:

$$\lambda = \begin{cases} \lambda_0 & x = 0 \\ \lambda_1 & x = 1. \end{cases}$$

After re-parameterization of Equation 1.2, we have

$$h(t|x, \mathbf{Z}, U) = h_0(t) \exp[\beta_1 x + \eta \mathbf{Z} + \lambda_0 u + (\lambda_1 - \lambda_0) x u], \tag{1.3}$$

where $x = 0, 1$. In the case where $\lambda_0 \neq \lambda_1$, there is an interaction effect $(\lambda_1 - \lambda_0)xu$. Thus, U is an unmeasured confounder, which relates to the treatment exposure indicator X and the response variable T.

1.2.2 Time-to-Sputum Culture Conversion in MDR

Although epidemiologists are always concerned about the proportion of tuberculosis (TB) cases that are resistant to multiple drugs (Zhao et al. 2012), the status of mycrobacterial cultures is considered the most important interim indicator of the efficacy of treatment for multidrug-resistant (MDR) tuberculosis (Holtz et al. 2006). Besides the ultimate goal of sputum culture conversion, the time required

to achieve sputum culture conversion is also an important interim indicator for the anti-tuberculosis treatment.

Let T denote the time to negative indicator of sputum culture conversion, and let X be the indicator of living in the inner-city. Suppose we are interested in the hazard ratio between the inner-city and rural areas, and the clinical outcome is time-to-sputum culture conversion. Four binary variables (Z_1 ,Z_2, Z_3, and Z_4) are assigned to indicate whether the patients follow the self-administered treatment protocol, previous TB history (Millet et al. 2013), HIV infection, and an alcohol abuse indicator (Holtz et al. 2006). Then, the Cox proportional hazards regression can be written as follows:

$$h(t|D) = h_0(t)\exp(\beta_1 x + \beta_2 z_1 + \beta_3 z_2 + \beta_4 z_3 + \beta_5 z_4), \tag{1.4}$$

where β_i for $i = 1, \cdots, 5$ are the coefficients of regression, and $D = (x, z_1, z_2, z_3, z_4, z_5)$. However, Djibuti et al. (2014) showed that poverty is associated with an increased risk of active TB disease onset and treatment outcomes. Also, there is a still-expanding gap between the averages of urban and rural household incomes in China (Khan and Riskin 2005). Thus, the household income is an unmeasured confounder when there is no income information in the patient report form (PRF). The appropriate model is

$$h(t|D) = h_0(t)\exp(\beta_1 x + \beta_2 z_1 + \beta_3 z_2 + \beta_4 z_3 + \beta_5 z_4 + \lambda u), \tag{1.5}$$

where u is the household income.

1.3 Bias Parameters

Non-sampling biases, such as misclassification, measurement error, and unmeasured confounding, are common problems in observational studies. Both frequentist and Bayesian approaches with various assumptions about available information for the bias parameters have been proposed to address each of these problems. For

instance, non-differential response misclassification in logistic regression adds two additional parameters to the model: sensitivity and specificity. Madger and Hughes (1997) used fixed values for the misclassification parameters to account for the measurement error and to find maximum likelihood estimates for the regression parameters. Authors such as Paulino, Soares, and Neuhaus (2003) and McInTurff et al. (2004) replaced the fixed values with informative prior distributions in a Bayesian context to allow for the uncertainty in the estimation. For unmeasured confounding, Lin et al. (1998) took an approach similar to that of Madger and Hughes (1997) and considered several different models.

Accounting for unmeasured confounding, misclassification, and other non-sampling bias situations requires adding more parameters to the model than are estimable. These additional parameters are referred to as "bias parameters". Gustafson and Gustafson et al. (2005 and 2015) provided a thorough discussion of the issues of non-sampling bias as they relate to the non-identifiability of the model. That is, without restrictions or additional information, the data alone are not able to estimate all the parameters, and standard asymptotic results (for example, convergence of point estimators and normality of MLEs) do not hold.

First, recall the definition of identifiability. A parameter θ for a family of distributions $\{f(x|\theta); \theta \in \Theta\}$ is identifiable if distinct values of θ correspond to distinct distributions (Casella and Berger 2002). Due to the unmeasured confounder, the model is nonidentifiable. In other words, different values of parameters may result in the same distributions of disease outcomes. To account for the lack of identifiability, model expansion and contraction is discussed in Gustafson et al. (2005). Most sensitivity analysis methods for unmeasured confounding (Lin et al. 1998; Steenland and Greenland 2004; Lin, Logan, and Henley 2013) can be categorized as model contraction techniques, since all of the methods assume that the coefficients

for the unmeasured confounder are derived from pre-existing relationships or are in some pre-defined ranges.

Rosenbaum and Rubin (1983) introduced the classical propensity score method, which is a model expansion technique. Most Monte Carlo methods and Bayesian approaches (Steenland and Greenland 2004; McCandless, Gustafson, and Levy 2007; McCandless, Gustafson, and Austin 2009) explore model expansion when the model is identified with an informative prior distribution (Gustafson et al. 2005).

1.4 Overview of Sensitivity Analysis and Bayesian Approaches

Steenland and Greenland (2004) presented a traditional sensitivity analysis with a lung cancer example, which only relies on a conditional probability assumption on the different exposure strata. Lin et al. (1998) assessed the sensitivity of a treatment effect through distributional assumptions of the unmeasured confounder. Furthermore, they showed a simple additive algebra formula for the relationship between the true treatment effect and the original biased estimate for time-to-event data.

VanderWeele (2008) presented a generalized model of Equation 1.2 without the conditional independence between the unmeasured confounder and the treatment exposure. Lin et al. (2013) developed a large sample approximation of bias adjustment with omitted covariates. Due to the different characteristics of the unmeasured confounder, Handorf et al. (2013) assumed that an unmeasured confounder follows the gamma and Poisson distributions. They provided a general formula for the different distributional assumptions. The two-stage calibration approach is discussed by Lin and Chen (2014), which is similar to Lin et al. (2013).

Steenland and Greenland (2004) also presented a Bayesian analysis of an unmeasured confounder, although they did not provide an adjusted estimate of the treatment effect (other than a full range of bias due to the unmeasured confounder).

McCandless et al. (2007) developed Bayesian sensitivity analysis to improve the uncertainty assessments for unmeasured confounding using Bayesian propensity scores. To account for a more complex structure of the unmeasured confounder, McCandless et al. (2012) presented a flexible quadratic form adjustment equation in a Bayesian perspective. Faries et al. (2013) and Stamey et al. (2014) showed that the Bayesian adjustment with validation data efficiently corrects for the bias due to the unmeasured confounder, even with a small validation sample size.

1.5 Objectives and Thesis Organization

There are three primary goals in this thesis. First, we investigate the effect of unmeasured confounding with different distributional assumptions. Second, we develop Bayesian adjustment methods to assess the efficacy of treatment exposure for time-to-event data with unmeasured confounding. Finally, we perform the Markov Chain Monte Carlo (MCMC) simulation to evaluate the performance of parametric and semi-parametric hazard regression models with unmeasured confounding.

Chapter Two investigates a Bayesian parametric survival regression model analysis with a continuous unmeasured confounder and a fixed censoring rate.

In Chapter Three, we consider the Bayesian semi-parametric Cox proportional hazards model and adjust for an unmeasured confounder, assuming that the baseline hazards function is a sequence of fixed values. Furthermore, MCMC simulations are conducted with discrete and continuous unmeasured confounders at three different censoring rates.

Chapter Four provides some important remarks of Bayesian bias adjustment due to unmeasured confounding. We also address some recommendations for future work.

Appendices at the end of the thesis include important mathematical derivations and simulation codes.

CHAPTER TWO

Bayesian Parametric Survival Regression Model with Unmeasured Confounders

2.1 Introduction

Data in pharmacoepidemiology are often subject to unmeasured confounding, for example Schneeweiss (2006) and Sturmer et al. (2005). Ignoring unmeasured confounding is known to yield biased estimates for regression parameters. While considerable work has been done in the frequentist paradigm, Bayesian methods have recently shown great promise in terms of correcting for unmeasured confounding. For instance, McCandless et al. (2007) considered a Bayesian approach to adjust logistic regression for unmeasured confounding, and they used a mildly informative priors in order to estimate the model parameters. The main advantage of their model is that the coverage of interval estimates are above nominal due to the large width. The lack of information leads to only a slight correction in the bias. Faries et al. (2013) extended the approach of McCandless et al. (2007) by calibrating the priors for the regression parameters with internal validation data. That is, for a small sub-sample of the data, the unmeasured confounder is ascertained. This allows for considerably better bias correction than the much less informative priors used by McCandless et al. (2007).

An area of interest in pharmacoepidemiology that has not received much research from the Bayesian perspective is survival models with unmeasured confounding. Lin et al. (1998) considered a simple adjustment plugging in fixed values for the unmeasured confounding parameters in a Cox regression model. The propensity score approach of Sturmer et al. (2005) can also be applied to the Cox model. Klungsoyr et al. (2008) considered algebraic approaches to correcting for discrete unmeasured confounding in a survival model. Here, we extend the work of Faries

et al. (2013) by using validation data to correct a parametric Bayesian approach to survival modeling.

Although the Cox proportional hazards model is the most popular model for survival data in the health sciences, marketing and social sciences, the parametric regression model still plays a vital role in survival analysis. The basic idea is that the outcome (i.e., time to event) follows some specific distribution with unknown parameters (such as exponential, Weibull, log-logistic, etc.). Hosmer, Lemeshow, and May (2008) showed the advantages of parametric regression models, such as the estimation of parameters using the full maximum likelihood and the fitted values from models which can provide estimates of survival time.

In the absence of information about survival data, distributional assumptions can be quite strong. This is the reason why parametric survival regression models are not as popular as semi-parametric or non-parametric models. In disease control and prevention, the primary interest is often the hazard function or the acceleration rate of some specific infectious disease, like tuberculosis. For an example of research that is focused on parametric survival regression techniques, see Carroll (2003).

To accommodate the flexible characteristics of survival data, Mudholkar, Srivastava, and Kollia (1996) developed the beta-Weibull distribution, and Wahed, Luong, and Jeong (2009) generalized the beta-Weibull family. These extensions of the Weibull model are able to capture the real pattern of the data distribution. Also, Carroll (2003) introduced the large sample approximation of survival times, which provides a feasible alternative approach for a large sample cohort study.

In this chapter, exponential and Weibull regression models with unmeasured confounding are discussed. All the survival data are right censored with the assumption that the censoring time is independent of the survival time. Left censored, interval censored, and truncated survival data are not discussed here. The single unmeasured confounder is assumed to be a continuous normally distributed variable.

In Section 2.2, we present Weibull survival models with and without accounting for the normal unmeasured confounder. Three validation sizes were applied in the simulation studies. In Section 2.3, we illustrate the exponential regression for the same simulated data, accounting for the unmeasured confounder. In Section 2.4, we provide a discussion of Bayesian parametric regression for time-to-event data and whether or not it performs accurately in terms of bias correction. The coverage of 95% credible intervals are also considered in both sections.

2.2 Weibull Regression Model with Unmeasured Confounders

In the start of this section, we overview the distributions and transformations needed for generating the time-to-event data with a continuous unmeasured confounder.

Let $y_1, y_2, \cdots, y_n$ denote independent and identically distributed survival times with possible right censoring. Let δ denote the indicator function of censoring. That is, $\delta_i = 1$ if the survival time of the i^{th} subject was observed, as in death or disease progression. $\delta_i = 0$ if the survival time of the i^{th} subject was not observed (i.e., the i^{th} subject was still alive or missing at time y_i). In this case, we do not know the exact time of the clinical outcome (death or disease progression).

Let $x_1, z_1, \cdots, z_q, u$ denote regression covariates: x_1 is the exposure variable of primary interest, the $z_1, \cdots, z_q$ are other covariates, and u is the unmeasured confounder. The covariate vector $\Psi = (x_1, z_1, z_2, \cdots, z_q, u)$ could be discrete, continuous, or mixed. We assume that all of the covariates are fully measured except u, and we denote the i^{th} unit as

$$(y_i, \delta_i, \Psi_i). \tag{2.1}$$

Let T denote the true survival time with the covariate vector Ψ. By definition of the survival function, we have

$$S_X(y) = P_X(T > y) = \int_y^{\infty} f(t)dt, \tag{2.2}$$

where $f(t)$ is the density function of the survival time.

Since the domain for the survival time is $t \in [0, \infty)$, we can map it to the whole real line with the log transformation, $\log(y) \in (-\infty, \infty)$. Furthermore, it is applicable to introduce location and scale parameters

$$Y = \log(T) = \alpha + \sigma W, \tag{2.3}$$

where W has support in $(-\infty, \infty)$.

Survival regression models are sometimes called accelerated failure time models. The survival time can be modeled as the product of the effect of independent covariates Ψ and the time scale. Suppose $\sigma = 1$, and let t denote the observed time Ψ is the covariate vector of all units. From Equation 2.3, the baseline survival function is defined as

$$S_0(t) = Pr(T > t|\Psi = \mathbf{0}) = Pr[e^w > t]. \tag{2.4}$$

Then, we explore the survival time model as follows:

$$\begin{aligned} S(t|\Psi) &= P_r(T > t|\Psi) \\ &= P_r[Y > \log(t)|\Psi] \\ &= P_r[Y - \Psi\beta > \log(t) - \Psi\beta] \\ &= P_r[e^w > te^{(-\Psi\beta)}]. \end{aligned}$$

Therefore, t is accelerated life by $\exp(-x\beta)$.

To simplify the simulation and follow the assumption of conditional independence of the unmeasured confounder U and other confounders $\mathbf{Z}$, given treatment exposure indicator X_1 (Rosenbaum and Rubin 1983; Lin et al. 1998 and Handorf et al. 2013), we have

$$P(U|X_1, Z) = P(U|X_1). \tag{2.5}$$

Although VanderWeele (2008) showed that this independence is not always true, we assume the unmeasured confounder u_i is a continuous variable such that it is linearly related to the treatment exposure indicator x_1 and is independent of the other covariate $\mathbf{Z}$. Also, it is not the case that there already exists a causal inference from x_1 to u, such as the spontaneous abortion history (Weinberg 1993). The mean of the unmeasured confounder is assumed as follows:

$$\mu_i = \eta_{0i} + \eta_1 x_1. \tag{2.6}$$

This can be generalized to multiple regression or other types of generalized linear models such that

$$u_i = f(\eta_0 + \eta_1 x_{1i} + \Theta \mathbf{Z_i}) + \epsilon_i, \tag{2.7}$$

where f could be any link function, such as log, logit, etc., and ϵ_i is the random error under different assumptions.

The survival time y is assumed to have a parametric distribution and be conditional on the covariates. Besides the unmeasured confounder, we assume that there is no missing data from all of the subjects across the different treatment exposure groups. We also assume that the censoring time is independent of the survival time, which is a common assumption in survival analysis. In other words, there is no informative censoring.

Without accounting for the unmeasured confounder, the general likelihood function can be written as follows:

$$L(\Theta|D) = \left[\prod_{\delta_i=0} P_{\Psi_i}(Y > y_i)\right] \left[\prod_{\delta_i=1} P_{\Psi_i}(Y = y_i)\right], \tag{2.8}$$

where Θ is the vector of parameters and $D = (x_1, z_1, \cdots, z_n)$.

2.2.1 Survival Data with Weibull Distributions

The Weibull regression model is widely used in survival analysis, and it is sometimes described as the "bathtub curve." From the previous definition, we have

$$w = \frac{\log(T) - \alpha}{\sigma}, \qquad \text{and} \qquad \frac{\partial w}{\partial T} = \frac{1}{\sigma T}. \tag{2.9}$$

By the probability density function transformation, we have

$$g(T) = \exp\left\{\frac{\log(T) - \alpha}{\sigma} - \exp\left[\frac{\log(T) - \alpha}{\sigma}\right]\right\}.\frac{1}{\sigma T} \tag{2.10}$$

Let $\sigma = \frac{1}{\alpha}$ and $\alpha = -\frac{\log(\omega)}{\alpha}$. It follows that

$$g(T) = \omega T^{\nu} \exp(-\omega T^{\nu}) \frac{\nu}{T} \tag{2.11}$$

$$= \nu\omega T^{\nu-1} \exp(-\omega T^{\nu}). \tag{2.12}$$

This means that

$$T \sim \text{Weibull}(\nu, \omega), \tag{2.13}$$

where ν is the shape parameter and ω is the scale parameter. Suppose $\alpha = 1$, we then have

$$\omega = \exp(-\alpha) = \exp(-\Psi'\beta). \tag{2.14}$$

Let $y_1, y_2, \cdots, y_n$ denote identically and independently distributed Weibull observed survival times with the same shape parameter ν and a different scale parameter ω. That is,

$$y_i|\nu, \omega_i \sim \text{Weibull}(\nu, \omega_i), \tag{2.15}$$

where ω_i is derived as follows:

$$\omega_i = \beta_1 x_{1i} + \beta_2 z_{1i} + \beta_3 z_{2i} + \lambda u_i. \tag{2.16}$$

Suppose the conditional independence holds. Given Equations 2.5 and 2.6, the i^{th} unmeasured confounder can be expressed as

$$u_i|\eta_0, \eta_1 \sim \text{Normal}(\mu_{u_i}, \sigma^2_{u_i}), \tag{2.17}$$

where μ_{u_i} is defined in Equation 2.6.

In this section, we assume that we are able to obtain the unmeasured confounder for a small random subsample of the data. This is an example of internal validation. Let δ_i, x_{1i}, and $\mathbf{Z_{1i}}$ be defined as above, assuming m subjects have an unmeasured confounder available. The likelihood function for the Weibull regression survival model with an unmeasured confounder is expressed as follows:

$$\begin{aligned} L(\Theta|D) \propto & \prod_{i=1}^{n} \left[\nu\omega y_i^{\nu-1} \exp(-\omega y_i^{\nu})\right]^{\delta_i} \left[\exp(-\omega y_i^{\nu})\right]^{1-\delta_i} \exp\left[-\frac{(u_i-\mu_i)^2}{2\sigma_u^2}\right] \\ & \times \prod_{j=1}^{m} \left[\nu\omega \tilde{y_j}^{\nu-1} \exp(-\omega \tilde{y_j}^{\nu})\right]^{\delta_j} \left[\exp(-\omega \tilde{y_j}^{\nu})\right]^{1-\delta_j} \exp\left[-\frac{(\tilde{u_j}-\tilde{\mu_j})^2}{2\sigma_u^2}\right]. \end{aligned}$$

The parameters and observed data are defined as follows:

$$\begin{aligned} \Theta &= (\beta, \nu, \eta, \sigma_u^2) \\ D &= (Y, \delta, \tilde{U}, \Psi, \tilde{X_1}) \\ \omega &= \exp\left[-(\beta_1 x_1 + \Theta \mathbf{Z}' + \lambda U)\right] \\ \mu &= \eta_0 + \eta_1 X_1 \\ \tilde{\mu} &= \eta_0 + \eta_1 \tilde{X_1}. \end{aligned}$$

For a fully Bayesian analysis, we need prior distributions for the parameters. We assume that there is no knowledge of model parameters, especially for the regression parameters. Non-informative priors are applied to all of the regression parameters. Since the hazard function has an exponential base for both the exponential and Weibull distributions, a prior with a standard deviation of 10 leads to diffuse priors.

Generally, non-informative priors are used except for the bias parameters. In many instances, we may have partial information about the effect of the unmeasured confounder, such as negative or positive direction. However, diffuse priors for the bias parameters are applied in MCMC analysis since we have validation data. In general the priors are

$$\beta_i \sim \text{Normal}(\mu_{\beta_i}, \sigma^2_{\beta_i}) \tag{2.18}$$

$$\eta_j \sim \text{Normal}(\mu_{\eta_j}, \sigma^2_{\eta_j}). \tag{2.19}$$

For our simulation studies, we set $\sigma^2_{\beta_i} = \sigma^2_{\eta_j} = 10$ and $\mu_{\beta_i} = \mu_{\eta_j} = 0$, and we had

$$\beta_i \sim \text{Normal}(0, 10) \quad \text{for } i = 1, 2, 3$$

$$\eta_j \sim \text{Normal}(0, 10) \quad \text{for } j = 1, 2.$$

A diffuse normal prior for the unmeasured confounder coefficient was also assumed

$$\lambda \sim\sim \text{Normal}(0, 10).$$

Without the internal validation, we would need informative priors for λ, η, or β. Since $\nu > 0$ and $\sigma^2_u > 0$, we used gamma priors

$$\nu \sim \text{Gamma}(0.01, 0.01)$$

$$\tau \sim \text{Gamma}(0.01, 0.01),$$

where τ was the precision of the normal distribution such that $\tau = 1/\sigma^2_u$.

2.2.2 Simulation Studies

In this section, we present three simulation studies to explore the usefulness of internal validation data to correct the bias due to unmeasured confounding. The case of external validation will be discussed in Chapter 4. Both full and reduced models were fitted, where the reduced model was the survival regression model without

accounting for the unmeasured confounder. We evaluated the modeling performance by the amount of bias and the coverage probability of the 95% posterior intervals.

The survival times were generated by a Weibull distribution process as we discussed above (Equations 2.13 and 2.14). The treatment exposure indicator and other regression covariates were generated as follows:

$$X_1 \sim \text{Bernoulli}(0.6)$$

$$Z_1 \sim \text{Bernoulli}(0.3)$$

$$Z_2 \sim \text{Normal}(0, 1).$$

After generating the observed covariate $(X_1, \mathbf{Z})$, the unmeasured confounder was drawn from a normal distribution (Equations 2.3 and 2.21). That is,

$$U \sim \text{Normal}(\mu_u, \sigma_u^2),$$

where $\mu_u = 0.1 - 0.4X_1$ and $\sigma_u^2 = 3$. The internal validation data were generated from the same process with a smaller sample size. The sample size for the main study was $n = 1000$, and the sample sizes for the validation data were $m = (50, 100, 200)$. Therefore, the validation fractions were $(0.05, 0.1, 0.2)$, and the validation data included the unmeasured confounding information.

Differing from regular data generation processes, we also had to generate the censoring times. The censoring rate was fixed at 60%, which was tuned with the different values of the scale parameter (Lin et al. 1998).

For the regression coefficients, we assume that the following model represented the real relationship between the covariate and the outcome:

$$\omega_i = -0.3x_{1i} + 0.2z_{1i} + 0.3z_{2i} - 0.8u_i. \tag{2.20}$$

Since there is no closed-form posterior distribution, MCMC was employed to sample the target posterior distribution. The STAN package (Carpenter et el. ND) was

used to simulate draws from the posteriors of the parameters via the `rstan` package in R. The code is available in the appendix A.

2.2.3 Results

Our primary interest was to estimate the real hazard ratio between two groups. We performed 40 iterations with a sample size of $n = 1000$ for the main study, and we used $m = 100$ for the validation data since it took almost 10 hours for each simulation. To evaluate the ability of bias correction, the same simulation without any additional information was conducted, assuming that the unmeasured confounder was ignored. This second model is referred to as the naive model.

The `stan` procedure with a chain length of 20000 and a burn-in of 7000 iterations with a thinning of 10 was performed, so that $\frac{20000-7000}{10} = 1300$. Therefore, 1300 sample draws were kept in our analysis. The initial values were randomly generated, and the random seed was set at 100.

To assess the performance of the MCMC sampling, the Gelman-Rubin statistics and plots were checked. Also, the sampling autocorrelation was examined using the `coda` package. In Figure 2.1, we can see that there was no evidence against convergence in the simulation. In Figure 2.2, we see there were no issues with autocorrelation.

The parameter estimations of the Weibull regression models with and without accounting for the unmeasured confounder are presented in Table 2.1. When the sample size for the main study was $n = 1000$ and the validation size was $m = 100$, we see that the posterior mean for the naive model was substantially biased, while the model with internal validation had considerably less bias. Also, the coverage of the 95% intervals increased from 77.5% to 97.5%. This illustrates the corrected model has approximately nominal coverage while the naive model has poor coverage. All of the other parameter estimates accounting for the unmeasured confounder are

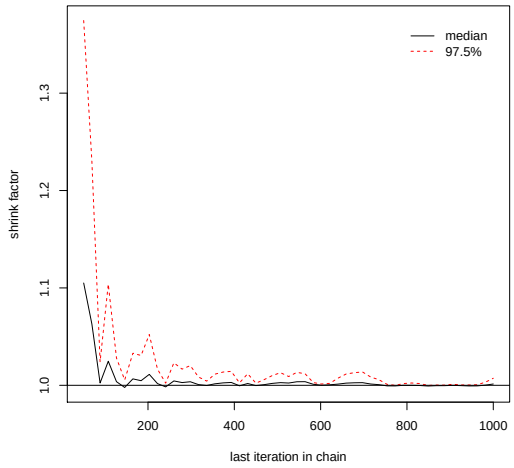

Figure 2.1: Gelman-Rubin Statistics diagnostic plot of β_1

significantly improved, both in terms of coverage probabilities and bias. However, this comes at the price of increased interval widths. In Table 2.2, we see that even a

Table 2.1: Summary statistics of the posterior for the model that account for the unmeasured confounder (top) and the naive model (bottom)

Parameter	True value	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
β_1	-0.3	-0.37	0.975	0.819
		-0.13	0.775	0.535
β_2	0.2	0.22	0.95	0.71
		0.07	0.875	0.71
β_3	0.3	0.3	1	0.34
		0.25	0.9	0.354
λ	-0.8	-0.79	1	0.184
		NA[a]	NA	NA
η_0	0.1	0.07	0.95	0.832
		NA	NA	NA
η_1	-0.4	-0.41	0.925	1.271
		NA	NA	NA

[a] NA means not applicable

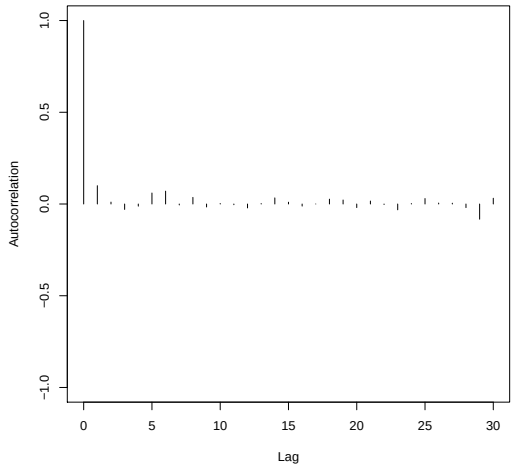

Figure 2.2: Autocorrelation diagnostic plot of β_1

validation sample size of $m = 50$ improved the posterior mean from -0.13 to -0.27, which significantly corrects for the unmeasured confounder. When the validation sample size was $m = 200$, the fraction of validation was 20%, the coverage of the 95% interval was 97.5%, and the posterior mean was close to the true value $\beta_1 = -0.3$. Also, the average 95% interval width decreased sharply from 1.09 to 0.608 when the validation sample size increased from 50 to 200.

Table 2.2: The estimation of $\beta_1 = -0.3$ under simulations with different sample size of validation data

sample size	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
Naive	-0.13	0.775	0.535
$m = 50$	-0.27	0.95	1.09
$m = 100$	-0.31	0.975	0.819
$m = 200$	-0.31	0.975	0.608

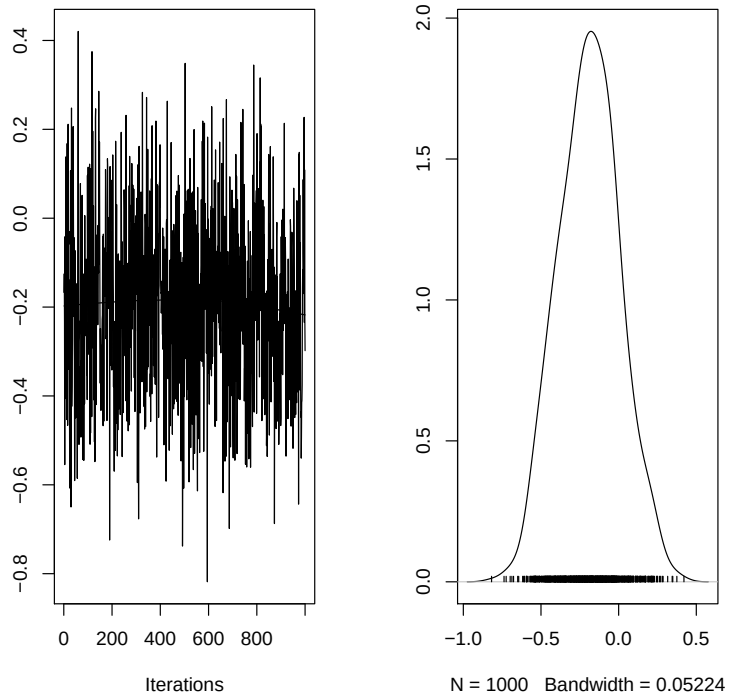

Figure 2.3: Posterior density and traceplot of β_1

2.3 Exponential Regression with Unmeasured Confounders

Due to the inverse relationship between work experience and the risk of injury, the hazard rate in many occupations often decreases when the employee gets more experience over time. For some occupational exposures or environmental contaminations, the hazard rate for individuals stays at the same level over time if there are no other outside factors. In cases like this, the exponential regression survival model would be considered as a primary analysis method. Also, the large sample approximation for the PFS (Progression Free Survival) uses the exponential distribution (Carroll 2007).

Hosmer et al. (2008) and Ibrahim, Chen, and Sinha (2013) discussed the exponential regression model in survival analysis. The mathematical properties were illustrated in the texts by Lawless (2002) and Collett (2003). In this section, we first

review the regression components. After that, we illustrate the Bayesian exponential regression survival model with the same simulation configurations as in the previous section.

2.3.1 Survival Data with Exponential Regression

To derive the linear expression of the exponential distribution, we still assume that w follows an extreme value distribution. The density function for w is

$$f(w) = \exp(w - e^w). \tag{2.21}$$

Suppose the survival time T has an exponential distribution with a single parameter γ. Based on Equations 2.3 and 2.21, we have $w = \log(T) - \alpha$. By a one-to-one random variable transformation, it follows that

$$\begin{aligned} g_(t) &= \exp\left\{[\log(t) - \alpha] - \exp[\log(t) - \alpha]\right\} \frac{1}{t} \\ &= \exp(-\alpha) \exp[-\exp(-\alpha)t]. \end{aligned}$$

It is apparent that the survival time T follows an exponential distribution with $\gamma = \exp(-\alpha) = \exp(-X\beta)$.

Again, let $y_1, y_2, \cdots, y_n$ denote identically, independently, and exponentially distributed survival times, δ_i denote the censoring indicator, U_i denote the unmeasured confounder, X_{1i} denote the treatment exposure indicator, and $\mathbf{Z_i}$ denote the characteristic covariate vectors or environmental risk factor vectors. We then have

$$y_i|\gamma_i \sim \text{Exponential}(\gamma_i), \tag{2.22}$$

where $\gamma_i = \beta_1 x_{1i} + \beta_2 z_{1i} + \beta_3 z_{2i} + \lambda u_i$, and u_i follows a normal distribution such that

$$u_i|\mu_{u_i}, \sigma^2_{u_i} \sim \text{Normal}(\mu_{u_i}, \sigma^2_{u_i}). \tag{2.23}$$

Suppose there exists m continuous internal validation subjects with means as in Equation 2.23, and $\tilde{\mu}_{u_i} = \eta_{0i} + \eta_{1i}\tilde{x}_{1i}$. Then, the likelihood function is composed

of two parts:

$$
\begin{aligned}
L(\beta, \lambda, \eta, \sigma_u, U|D, \tilde{D}) \propto & \prod_{i=1}^{n} \exp[-\delta_i(\beta x_{1i} + \lambda u_i + \Theta \mathbf{Z_i}')] \exp\{-\exp[-(\beta x_{1i} + \lambda u_i + \Theta \mathbf{Z_i}')]\} \\
& \times \exp\left\{-\frac{[u_i - (\eta_0 + \eta_1 x_{1i})]^2}{2\sigma_u^2}\right\} \\
& \times \prod_{j=1}^{m} \exp\left\{-\frac{[\tilde{u_j} - (\eta_0 + \eta_1 \tilde{x_{1j}})]^2}{2\sigma_{\tilde{u}}^2}\right\} \exp[-\delta_j(\beta \tilde{x}_{1j} + \lambda \tilde{u}_j + \Theta \tilde{\mathbf{Z}}_\mathbf{j}')] \\
& \times \exp\{-\exp[-(\beta \tilde{x}_{1j} + \lambda \tilde{u}_j + \Theta \tilde{\mathbf{Z}}_\mathbf{j}')]\},
\end{aligned}
$$

where $\tilde{D} = (\tilde{x}_1, \tilde{\mathbf{Z}}, \tilde{u})$ is the internal validation data set and $D = (x_1, \mathbf{Z})$ is defined as in the previous section.

2.3.2 Simulation Studies

The data generation configurations and regression parameters were the same as with the Weibull distribution case. The sample size for main study was $n = 1000$. The validation sizes were $m = (50, 100, 200)$. Ibrahim et al. (2013) showed that the posterior of the exponential regression with a conjugate gamma prior has a closed form distribution. Due to the non-informative priors for the regression coefficients and the unmeasured confounder, there was no closed form marginal posterior available. MCMC analysis was employed with the `stan` package in `R`.

In each simulation, the length of the posterior chain was 15000 draws with a 5000 burn-in. To tune the autocorrelation problem, the thinning value was chosen at 10. For each configuration, we generated 40 data sets. Convergence was checked by the Gelman-Rubin statistics and trace plots (Figures 2.4 and 2.6). The plots indicate good mixing.

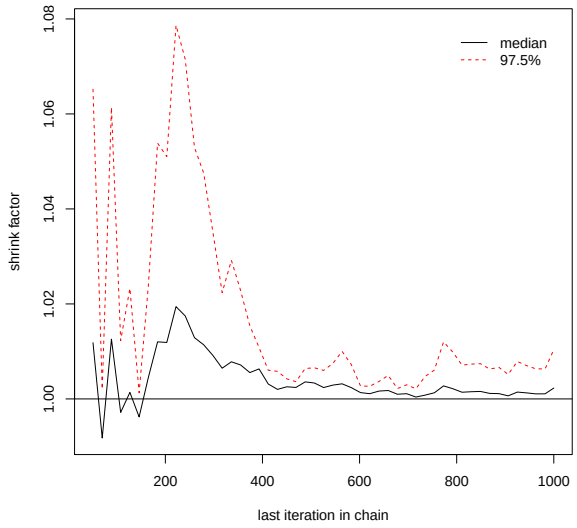

Figure 2.4: Gelman-Rubin Statistics diagnostic plot of β_1 (exponential distribution)

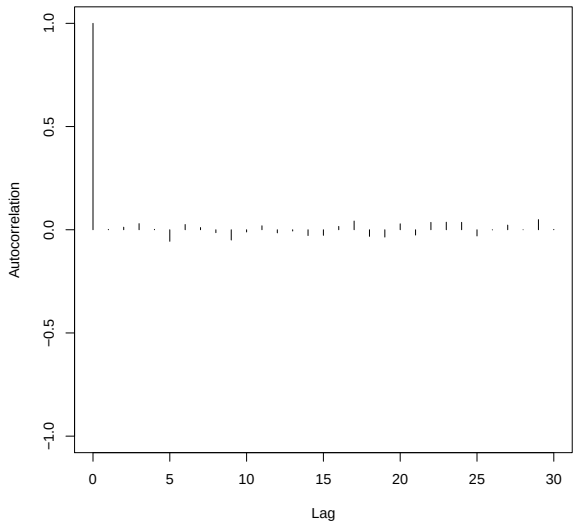

Figure 2.5: Autocorrelation diagnostic plot of β_1 (exponential distribution)

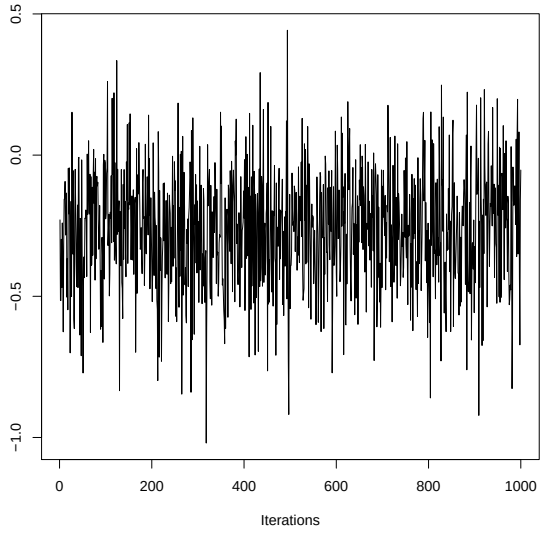

Figure 2.6: Trace plot of β_1 (exponential distribution)

In Figure 2.7, we can see that the density was unimodal and smooth. The mode of the posterior distribution was close to the true value $\beta_1 = 0.30$.

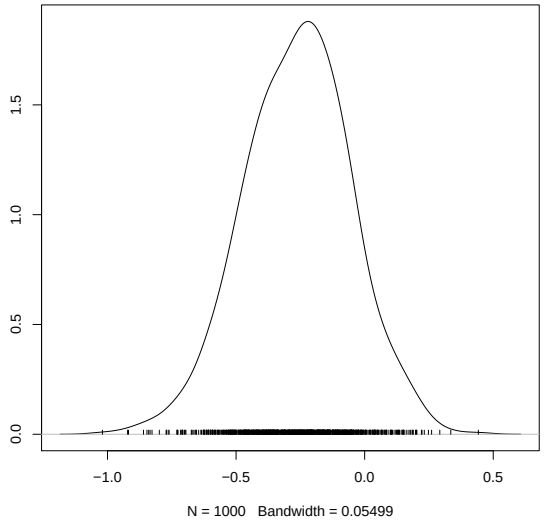

Figure 2.7: Density plot of β_1 (exponential distribution)

2.3.3 Results

To explore the ability of bias adjustment for the exponential regression model, we first considered the sample validation size $m = 100$. In Table 2.3, we display the posterior means, 95% widths, and coverages respectively. Compared with the naive model, all the posterior means for the model accounting for the unmeasured confounder were much closer to the true means. It is interesting that the coverage of a 95% interval for β_1 improved significantly from 77.5% to 97.5%, but it was at the price of a wider interval. Although the original data were generated from the Weibull distribution, the exponential regression fit the model adequately.

Table 2.3: Summary statistics of the posterior for the exponential model that account for the unmeasured confounder (top) and the naive model (bottom)

Parameter	True value	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
β_1	-0.3	-0.37	0.975	0.819
		-0.13	0.775	0.535
β_2	0.2	0.22	0.95	0.71
		0.07	0.875	0.71
β_3	0.3	0.3	1	0.34
		0.25	0.9	0.354
λ	-0.8	-0.79	1	0.184
		NA [a]	NA	NA
η_0	0.1	0.07	0.95	0.832
		NA	NA	NA
η_1	-0.4	-0.41	0.925	1.271
		NA	NA	NA

[a] NA means not applicable

In Table 2.4, we can see that the Weibull regression models performed better than the exponential ones across the $m = (50, 100, 200)$ validation sizes, regarding the posterior means and the average 95% intervals. Also, the exponential regression estimates appeared unstable for the estimation of β_1 when the sample size increased.

Unlike the Weibull distribution, when the validation size increased from 50 to 100, both the performance of the 95% intervals and posterior means worsened.

Table 2.4: The estimation of $\beta_1 = -0.3$ under simulations with different sample sizes of validation data and distributions

Sample size	Naive	Weibull	Exponential
$m = 50$	-0.13(78%, 0.54)	-0.27(95%, 1.09)	-0.33(95%, 1.12)[a]
$m = 100$	-0.13(78%, 0.54)	-0.31(90%, 0.81)	-0.37(90%, 0.82)
$m = 200$	-0.13(78%, 0.54)	-0.31(97.5%, 0.61)	-0.36(100%, 0.60)

[a] the posterior mean (coverage of 95% interval, 95% interval length)

2.4 Discussion

Although the parameter estimation methods for time-to-event data with an unmeasured confounder and internal validation data are developed in this chapter, we only discussed the normal unmeasured confounder. However, it can be generalized to any existing distributions. This applies not only to binary data, but also to gamma or log-normal data. Using notation consistent with the above, we have

$$f(\beta, \lambda, \eta, u|y, x, \tilde{y}, \tilde{u}, \tilde{x}) = f(\beta, u, \lambda|y, x) f(\eta|x, \tilde{y}, \tilde{u}, \tilde{x}). \tag{2.24}$$

The survival times and unmeasured confounder are modeled with the "twin regressions":

$$f(t|x_1, \mathbf{Z}) = \beta_1 x_1 + \Theta \mathbf{Z}' + \lambda U \tag{2.25}$$

$$g(u|x_1, \mathbf{Z}) = \eta_0 + \eta_1 \tilde{X}_1 + \eta_2 \mathbf{Z}'. \tag{2.26}$$

Without validation data to account for the unmeasured confounder, we can either put informative priors on the bias parameter, or we can take the non-parametric function for the unmeasured confounder U to capture the universe of the unmeasured confounder. This is discussed in McCandless et al. (2012).

In summary, validation sample fractions as small as 5% yielded corrected estimates considerably better than the naive model. However, there did seem to be some sensitivity to the choice of parametric distribution.

There were several limitations to the work in this chapter. We made the strong assumption that the unmeasured confounder only depends on the treatment exposure indicator, which is an untestable assumption that is unlikely to hold exactly in applications (Handorf et al. 2013). We also did not expand on the details of the sample size issues, since there is no "golden" rule for the validation fraction. Further sensitivity analysis is an area of future work.

CHAPTER THREE

Bayesian Cox Proportional Hazards Model with Unmeasured Confounding

3.1 Introduction

There are several methods to handle confounders in the design and analysis of data, such as restriction (study participation is restricted to individuals who fall within a specified category or categories of the confounder), propensity score, stratified analysis, and matching techniques. All of these techniques require full or at least partially measured confounders. If missing or unmeasurable confounders exist, it is impossible to implement the matching, restriction, and stratified methods. In this chapter, we investigate the Bayesian semi-parametric proportional hazards model, adjusting for binary and normal unmeasured confounders.

In observational studies, sensitivity analysis is necessary due to the effect of unmeasured confounding (Lin, Psaty and Kronmal 1998; VanderWeele 2008; Handorf et al. 2013). Regular sensitivity analysis dominates bias assessment, since it yields identifiable models. However, most sensitivity analyses can underestimate overall uncertainty (Mitra and Heitjan 2007).

Most bias adjustment techniques for unmeasured confoundings are frequentist. There is not as much Bayesian literature. Steenland and Greenland (2004) considered Bayesian sensitivity analysis for smoking status as the unmeasured confounder. McCandless et al. (2007, 2009, 2012) developed full Bayesian approaches for the adjustment of missing confounders with propensity scores and binary outcomes. Faries et al. (2013) and Stamey et al. (2014) extended the Bayesian approach for unmeasured confounding to cost-effective analysis.

Considering time-to-event data with missing covariates, Bradshaw, Ibrahim, and Gammon (2010) provided a straightforward Bayesian approach to model the

survival time with non-ignorably missing time-varying covariates. Chen et al. (2014) introduced a more sophisticated Bayesian proportional hazards model for irregular data structure.

For time-to-event data, Cox proportional hazards regression is widely used, since it is free of distributional assumptions. There are several texts that discuss Bayesian proportional hazards model, such as Ibrahim et al. (2013) and Christensen et al. (2010). Ibrahim et al. (2013) put more attention on the theoretical aspects of Bayesian proportional hazards model. Conversely, Christensen et al. (2010) were more concerned about the practical implementation, and they provided a number of examples.

To simplify the modeling complexity, a single unmeasured confounder is assumed in this chapter. In Section 3.1, we overview the literature related to Bayesian analysis with an unmeasured confounder and survival data with missing covariates. Next, we overview the basic survival analysis formula in Section 3.2, which is needed for data generation. Then in Section 3.3, the Bayesian semi-parametric Cox proportional hazards model with a normal unmeasured confounder is presented. In Section 3.4, we consider a binary confounder, and a different simulation design is employed. Lastly, we provide concluding comments in Section 3.5.

3.2 Materials and Methods

We denote T_i as the event time and C_i as the censoring time for the i^{th} subject. For each of N subjects, the observed event time is $y_i = \min\{T_i, C_i\}$ and $\delta_i = I_{T_i<C_i}$, where $I_{T_i<C_i}$ is the indicator function. This is defined as follows:

$$\delta_i = \begin{cases} 1 & \text{if } T_i < C_i \\ 0 & \text{Otherwise.} \end{cases}$$

Let X_1 denote the treatment vector, $Z_1, \cdots, Z_p$ be a p dimensional vector of measured covariates, and U be the unmeasured confounder.

We assume $Z_1, Z_2, \cdots, Z_p$ are fully observed. We also assume there is no missing data in the treatment exposure indicator group X_1. Finally, we assume either internal or external validation data is available.

3.2.1 Cox Proportional Hazards Model

The Cox proportional hazards regression model is probably the most popular technique for regression analysis of survival data. The equation for a basic Cox regression model is usually written as follows:

$$h_i(t) = h_0(t) \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i), \tag{3.1}$$

where $h_0(t)$ is the baseline hazard function that only depends on time t. The values $X_1, \mathbf{Z}$, and U are regression covariates. X_1 and $\mathbf{Z}$ were defined in the previous section, and U is an unmeasured confounder vector. $\Theta = (\beta_2, \beta_3, \beta_4)$ is a vector of regression coefficients for the proportional hazards model. Since $h_0(t)$ varies with time (Kasza et al. 2014), it is obvious that the baseline hazard is not fixed. This is true in most studies of epidemiology and oncology. Due to the lack of information, it is not appropriate to specify a statistical distribution on the baseline function, although statistical inference of the $h_0(t)$ function is not the primary goal in survival analysis. An alternative approach is a semi-parametric model, which divides all time t into finite equal groups with the assumption of homogeneity across the groups. Then, the subjects in the same time slot will share the same hazard rate.

Suppose the survival times are divided into J intervals for the proportional hazards model. Let λ_j be the hazard rates for the subjects who fall in the interval (s_j, s_{j+1}) for $j = 1, 2, \cdots, J$. To avoid the problem of some intervals not covering the data points, we let s_j and s_{j+1} depend on real data. This so-called data driven interval was implemented by Christensen et al. (2010). This method guarantees that each interval contains data points, ensuring that the Bayesian model is not updated without any data information.

The event time points can be grouped by quantiles, real values, or some other specific interests. In this chapter, we are only concerned with the equal quantiles approach. Suppose $J = 10$ and $N = 2000$. We had 10 intervals and 2000 event time points. To guarantee that each interval had the same number of data points, the quantile splitting process was implemented in our analysis. In other words, we had a boundary sequence $(s_1, s_2, \cdots, s_{J+1})$. By the previous definition, each interval had $2000/10 = 200$ event time points.

The equal quantile approach has two obvious advantages in our study. First, it guarantees that each interval has the same weight of information, which will reduce the bias that some intervals have considerably more data points than others. This is especially true when the event time is a skewed distribution. Second, the number of data points in each interval is sufficient. When the sample size is large enough, the likelihood would dominate inference.

3.2.2 Basic Formulas for Survival Analysis

In this section, we overview the survival model with unmeasured confoundings. Let y be a continuous variable, with CDF $F(y)$ on the interval $[0, +\infty)$. Then, the survival function is defined as

$$S(y) = P(Y > y) = \int_y^{\infty} f(t)dt.$$

The hazard function (unconditional failure rate) is defined as

$$h(y) = \frac{f(y)}{S(y)} = -\frac{d}{dy}\ln[S(y)]. \tag{3.2}$$

Let $S(t)$ denote the survival function and $h(t)$ denote the hazard function. Incorporating the unmeasured confounder, the survival function can be expressed as

$$\begin{aligned} S(t|x_{1i}, \mathbf{Z_i}, u_i) &= \exp[-H_0(t)\exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)] \\ &= \{\exp[-H_0(t)]\}^{\exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)} \\ &= [S_0(t)]^{\exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)}, \end{aligned}$$

where $H_0(t)$ is the baseline cumulative hazard function, $\mathbf{Z} = (z_1, z_2, z_3)$ is a three dimensional covariates vector, and $\Theta = (\beta_2, \beta_3, \beta_4)$ is a coefficients vector.

3.2.3 Likelihood Function and Poisson Approximation

Let $X_1, \mathbf{Z}$, and U have the same definition as in the previous section. Let $\psi_1, \psi_2, \cdots, \psi_J$ denote the baseline hazard rates. We assume the data is right censored (which is common in the clinical trials or observational studies). The i^{th} term of the core kernel of the likelihood can be written as follows (Christensen et al. 2010):

$$\begin{aligned}
L_i(\beta_1, \lambda, \Theta | X_i) &= [f(t_i)^{\delta_i}][S(t_i)]^{1-\delta_i} \\
&= [h(t_i)]^{\delta_i} S(t_i) \\
&= [\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)]^{\delta_i} \exp[-H(t_i)] \\
&= [\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)]^{\delta_i} \\
&\times \exp\left\{ - \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i) \left[\psi_{i*}(t_i - s_j) + \sum_{g=1}^{i*-1} \psi_g (s_{j+1} - s_j) \right] \right\} \\
&= [\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)]^{\delta_i} \\
&\times \exp\left\{ -\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i) \left[\sum_{j=1}^{i*} \omega(i,j) \right] \right\},
\end{aligned}$$

where $\Theta = (\beta_2, \beta_3, \beta_4)$, $i*$ is the largest integer such that $s_{i*} < t_i$, t_i is the i^{th} event time, u_i is an unmeasured confounder (main study data), δ_i is the indicator function for whether the event is observed or not, and s_j is the j^{th} lower bound of the interval. $\omega(i,j)$ is defined as follows:

$$\omega_{i,j} = \begin{cases} s_{j+1} - s_j & \text{if } t_i \geq s_{j+1} \\ t_i - s_j & \text{if } t_i \in [s_j, s_{j+1}) \\ 0 & \text{if } t_i < s_j \ . \end{cases} \tag{3.3}$$

There is no available distributional form for the survival part of likelihood, but there is a common method to handle this irregular distribution (Christensen et al. 2010).

The core kernel likelihood function can be reconstructed as follows:

$$d_{i,j} = \begin{cases} 1 & \text{when } \delta_i = 1 \text{ and } t_i \in [s_j, s_{j+1}) \\ 0 & \text{Otherwise,} \end{cases} \tag{3.4}$$

where $d_{i,j}$ is an indicator function for when the subject's event time was observed, and t_i falls in the interval $[s_j, s_{j+1})$. Let $\varphi(i,j) = \psi_j \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)\omega(i,j)$. Then,

$$L_i(\beta_1, \Theta, \psi) \propto \prod_{j=1}^{i*} \left[\psi_j \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)\right]^{d(i,j)} \tag{3.5}$$

$$\times \exp\left[-\psi_j \omega(i,j) \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)\right]. \tag{3.6}$$

Since $\omega(i,j)$ is not a function of the parameters $(\beta_1, \Theta, \psi, \lambda)$, if $\omega(i,j) > 0$, $\prod_{j=1}^{i*}[\psi_j \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)]^{d(i,j)}$ and $\exp[\varphi(i,j)]^{d(i,j)}$ are proportional. The difficult part is when $\omega(i,j) = 0$ (this is $\exp[\omega(i,j)] = 1$), we have to make sure that $[\psi_j \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)]^{d(i,j)}$ also equals 1. That is the reason we create the $d(i,j)$ indicator, which guarantees that $\omega(i,j) = 0$. This implies that $d(i,j) = 0$ and $0^0 = 1$. Therefore, the likelihood could be rewritten as

$$L_i(\beta_1, \Theta, \psi) = [\varphi(i,j)]^{d(i,j)} \exp\left[-\varphi(i,j)\right], \tag{3.7}$$

which is a Poisson distribution with an outcome of 0 and 1. That is,

$$d(i,j) \sim \text{Poisson}[\varphi(i,j)]. \tag{3.8}$$

3.3 Bayesian Proportional Hazards Model with Normal Unmeasured Confounders

External validation data sets are often available. Most healthcare data sources are readily available and have high external validity (Black 1996). For example, health care providers have all the patient level data. McCandless et al. (2012) performed Bayesian adjustment with missing confounders using external validation data, specifically, a cohort study with information on the unmeasured confounder.

Sturmer (2005) used national survey data to improve inferences in a large healthcare database study.

There are several different adjustment methods using external validation data, such as two-stage calibration (Lin and Chen 2014), sensitivity additive formulas (Lin et al. 1998), and Bayesian propensity scores (McClandless et al. 2007, McClandless et al. 2012). Suppose we have partial information without full measurements; then, not all of the approaches discussed above are applicable. Also, it is a strong assumption that the validation data falls in the "same cluster" or the "same population" with the main study data. In other words, the information is potentially biased. More often than not, we only get partial information about the unmeasured confounder, like summary statistics (mean or standard deviation) or the correlations between the unmeasured confounder and the other measured confounders in our analysis.

3.3.1 External Validation with Continuous Summary Statistics

For the continuous unmeasured confounder, suppose there is only external summary statistics, such as the sample means and standard deviations across the treatment exposure variables X_1. Assuming that there is a linear relationship between U and X_1, the data could be summarized as in Table 3.1.

Table 3.1: Summary statistics of the unmeasured confounder across a binary treatment exposure indicator

Statistics	$X_1 = 0$	$X_1 = 1$
Mean	$\bar{U}_{X_1=0}$	$\bar{U}_{X_1=1}$
Standard Deviation	$S_{X_1=0}$	$S_{X_1=1}$
Sample size	$n_{X_1=0}$	$n_{X_1=1}$

Since the unmeasured confounder U is continuous, a normal distribution was assumed here. For the i^{th} unmeasured confounder, we have

$$u_i \sim \text{Normal}(\mu_i, \tau_U), \tag{3.9}$$

where the mean of the unmeasured confounder u_i is defined as $\mu_i = \eta_0 + \eta_1 x_i$, and $\tau_u = 1/\sigma_U^2$ is the precision. The summary statistics in Table 3.1 can be incorporated into the Bayesian analysis as informative normal priors for (η_0, η_1). To simplify the simulation, independence is assumed between η_0 and η_1. Since X_1 is binary, note that $\bar{x}_1 = \frac{n_{X_1=1}}{n_{X_1=1}+n_{X_1=0}}$. That is, the number of subjects who get the treatment of interest is divided by the total number of subjects in the study. We can then determine the informative priors for parameters η_0 and η_1.

The means for the parameters are $\bar{u}_{x_1=0}$ and $\bar{u}_{x_1=1} - \bar{u}_{x_1=0}$ for η_0 and η_1 respectively. Here $\bar{u}_{x_1=0}$ is the sample mean of the unmeasured confounder at the $x_1 = 0$ group, and $\bar{u}_{x=1}$ is the sample mean for the unmeasured confounder at the $x_1 = 1$ group. The variance of η_0 can be derived as

$$\sigma_{\eta_0}^2 = \sigma_u^2 \left(\frac{1}{N} + \frac{\bar{x}}{SSx} \right), \tag{3.10}$$

where $N = n_{X_1=0} + n_{X_1=1}$ and $SSx = \sum_{i=1}^{N} (x_i - \bar{x})^2$. The variance for the other parameter η_1 is

$$\sigma_{\eta_1}^2 = \frac{\sigma_u^2}{SSx}. \tag{3.11}$$

3.3.2 Bayesian Inference

Suppose the validation sample size is m. Let $X_1, \mathbf{Z}$, and δ be defined as above. Together with Equations 3.7 and 3.9, the likelihood function with a normal unmeasured confounder can be expressed as follows:

$$L_i(\beta_1, \Theta, \psi, \eta | D, E) \propto \prod_{j=1}^{J} \left[\varphi(i,j)\right]^{d(i,j)} \exp\left[-\varphi(i,j)\right] \tag{3.12}$$

$$\times \exp\left[-\frac{(u_i - \eta_0 - \eta_1 x_{1i})^2}{2\sigma_u^2} \right], \tag{3.13}$$

where $\eta = (\eta_0, \eta_1)$, and (β_1, Θ, ψ) are defined above. D is the observed data $D = (X_1, \mathbf{Z})$, and $E = (\bar{\tilde{U}}_{X_1=0}, \bar{\tilde{U}}_{X_1=1}, \sigma^2_{U:X_1=0}, \sigma^2_{U:X_1=1})$.

To implement a Bayesian analysis, we construct relatively informative priors for η_0 and η_1 based on the external validation data. Suppose η_0 and η_1 follow normal distributions. The mean for η_0 is $u_{x_1=0}$, and the mean for η_1 is $u_{x_1=1} - u_{x_1=0}$. The variance for η_0 and η_1 are derived in Equations 3.10 and 3.11. That is,

$$\eta_0 \sim \text{Normal}\left[\tilde{u}_{x_1=0}, \sigma_u^2\left(\frac{1}{N} + \frac{\bar{x}_1}{SSx}\right)\right]$$
$$\eta_1 \sim \text{Normal}\left[\tilde{u}_{x_1=1} - \tilde{u}_{x_1=0}, \frac{\sigma_u^2}{SSx}\right],$$

where σ_u^2 is the standard deviation for the unmeasured confounder. To speed up the simulation, a conjugate gamma prior for τ_u is derived as follows:

$$\tau_u \sim \text{Gamma}\left[\frac{n_{x_1=0} + n_{x_1=1} - 2}{2}, \frac{(n_{x_1=0} - 1)S^2_{x_1=0} + (n_{x_1=1} - 1)S^2_{x_1=1}}{2}\right], \quad (3.14)$$

where τ_u is the precision such that $\tau = 1/\sigma_u^2$. Since we do not have any knowledge about the regression coefficients, we assume that the priors for the regression coefficients all follow normal distributions such that

$$\beta_i \sim \text{Normal}(\mu_{\beta_i}, \sigma^2_{\beta_i}) \quad (3.15)$$

$$\lambda \sim \text{Normal}(\mu_\lambda, \sigma^2_\lambda). \quad (3.16)$$

We take $\sigma^2_\lambda = \sigma^2_{\beta_i} = 10$ and $\mu_{\beta_i} = \mu_\lambda = 0$.

Since the baseline hazard function is positive over time, we develop the priors for baseline hazard functions as a product of independent gamma distributions. The mean of the gamma distribution is chosen as α_j/κ_j, and the variance is α_j/κ_j^2. Then, we have

$$\pi(\psi|\alpha, \kappa) \propto \prod_{j=1}^{J} \psi_j^{\alpha_j - 1} \exp(-\psi_j \kappa_j) \quad \text{for} \quad j = 1, \cdots, J. \quad (3.17)$$

This is a practical approach for constructing a gamma process prior for the positive baseline hazard function. We still use diffuse priors for these discrete baseline

functions. That is, $\alpha_j = \kappa_j = 0.01$ for all $j \in 1, 2, \cdots, J$, and the variance for ψ_j is $\alpha_j/\kappa_j^2 = 100$.

3.3.3 Simulation Studies

To study the effect of the unmeasured confounder on the posterior distribution, we performed several simulation studies. The parameters used are similar to the RHC example (Connors et al. 1996).

Let T denote the survival time and C be the random censoring time, which is independent of T. Let X_1, $\mathbf{Z}$, and U be defined as earlier. The Cox proportional hazards model with an unmeasured confounder is expressed as

$$h(t) = h_0(t) \exp(\beta_1 x_1 + \Theta \mathbf{z}' + \lambda u). \tag{3.18}$$

Bender, Augustin, and Blettner (2005) and Austin (2012) provided a survival time generation procedure based on the Cox proportional hazards model. Together with Equations 3.18 and 3.2, we have

$$F(t|x_1, \mathbf{z}, u) = 1 - \exp[-H_0(t) \exp(\beta_1 x_1 + \Theta \mathbf{z}' + \lambda u)], \tag{3.19}$$

where $F(t|x_1, \mathbf{z}, u)$ is the cumulative density function that falls between 0 and 1. Suppose the survival time follows a Weibull distribution. That is,

$$T \sim \text{Weibull}(w, \nu), \tag{3.20}$$

where w is the scale parameter and ν is the shape parameter. Then, we have

$$H_0(t) = wt^{\nu}, \tag{3.21}$$

and the inverse cumulative hazard function is

$$H_0^{-1}(t) = (w^{-1}t)^{1/\nu}. \tag{3.22}$$

Recalling Equation 3.19, we drew ξ from a standard uniform distribution. Then, the survival time t could be generated as follows:

$$t = \left[-\frac{\log(\xi)}{w \exp(\beta_1 x_1 + \Theta \mathbf{z}' + \lambda u)} \right]^{1/\nu}. \tag{3.23}$$

The censoring time was drawn from a uniform distribution,

$$c \sim \text{Uniform}(0, \gamma), \tag{3.24}$$

where the value of γ is dependent on the censoring rate.

To make the generated data match a real example, we set the scale parameter $w = 0.0035$ and the shape parameter $\nu = 0.13$. Thus, the Cox proportional hazards model with a continuous unmeasured confounder was assumed as

$$h(t|x_1, \mathbf{z}, u) = h_0(t) \exp(0.3x_1 - 1.2z_1 + 0.4z_2 + z_3 - 3u), \tag{3.25}$$

where u follows a normal distribution such that

$$u \sim \text{Normal}(0.6 - 0.04x_1, 0.2). \tag{3.26}$$

We performed 40 replications in each simulation. The sample size for the main study was $n = 1000$, while the sample size for the validation was $m = 100$. $30\%, 60\%$, and 90% censoring rates were considered in our analysis. There was no closed form marginal posterior available, thus we employed MCMC analysis using the `rjags` package in `R`. The chain length of the simulation was 20000, keeping every tenth draw for Bayesian inference with a burn-in of 7000 iterations. The convergence was checked with the Gelman-Rubin statistics (Figure 3.1). There did not appear to be any convergence problems. After a thinning of 10, we did not detect any strong autocorrelation.

To summarize the simulation results, we plotted the trace and density of β_1. In Figure 3.2, we can see it was unimodal and smooth. The average of the posterior means, empirical coverage probabilities, and interval widths for the 95% intervals

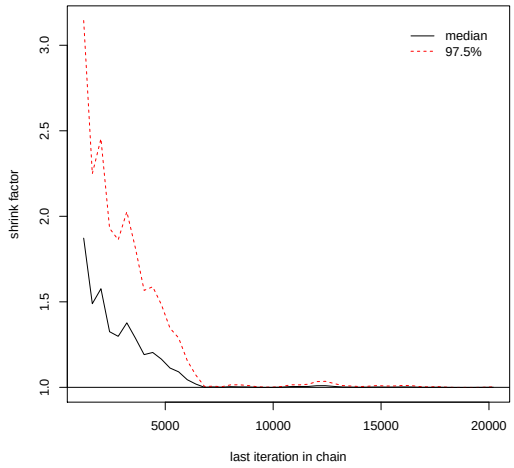

Figure 3.1: Gelman-Rubin Statistics diagnostic plot of β_1 (external validaion with normal data)

are provided in Table 3.2 for the regression coefficients β_1, β_2, β_3, β_4, λ, η_0, and η_1. It is interesting that this model did not capture the effect of the unmeasured confounder. That is, the coverage of the 95% interval for λ was 0 in the external validation scenario, but all the Bayesian estimates with different censoring rates still performed much better than the naive model. We can see that β_1 decreased from 0.47 to 0.31, which is close to the nominal value 0.30. Also, the coverage of the 95% interval was 0.95, compared to the naive model at only 0.825, without inflating the interval width.

In Table 3.3, the interval widths became wider when the censoring rates increased from 30% to 90%, which indicates that the uncertainty increased. Especially when the censoring rate was 90%, the interval width was three times wider than the 30% intervals. At the same time, the posterior mean of β_1 was far from the true value 0.30, about a 39% increase from the 30% to the 90% the censoring rate.

In Figure 3.3, the posterior densities for both the naive model and the external validation model were plotted in one graph. Again, we see that when we did not account for the unmeasured confounder, the distribution was centered away from

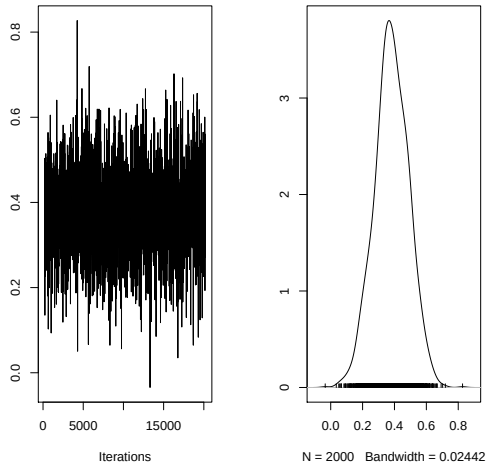

Figure 3.2: Density and trace plots of β_1 (external validaion with normal data)

the nominal value, compared to the external validation case. Also, the naive model was less variable, which is consistent with the results from Table 3.2.

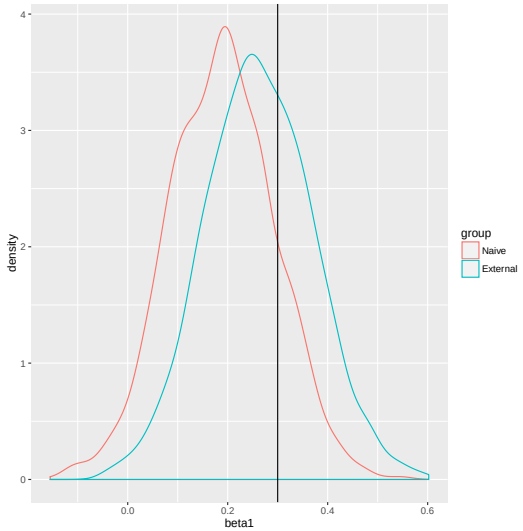

Figure 3.3: Density plots of the naive and external validation models for β_1

3.3.4 Internal Validation with Continuous Summary Statistics

In the retrospective observational and case controlled studies, the studies often start after the disease outcomes are observed. It is impossible to control the quality of data, especially when the data are from different healthcare providers. Also,

Table 3.2: External: summary statistics of the posterior for the model that account for the unmeasured confounder (top) and the naive model (bottom)

Parameter	True value	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
β_1	0.3	0.31	0.95	0.34
		0.47	0.925	0.31
β_2	-1.2	-0.74	0.725	0.61
		-0.67	0.425	0.58
β_3	0.4	0.29	0.775	0.35
		0.29	0.725	0.35
β_4	1	0.81	0.625	0.48
		0.83	0.70	0.47
λ	-3	-0.02	0	2.17
		NA [a]	NA	NA
η_0	0.6	0.60	1.00	0.65
		NA	NA	NA
η_1	-0.04	-0.03	0.80	0.1
		NA	NA	NA

[a] NA means not applicable

Table 3.3: The estimation of $\beta_1 = 0.3$ under different censoring rates with external validation

Censoring rate	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
30%	0.31	0.950	0.34
60%	0.35	0.900	0.45
90%	0.43	0.925	1.05

the researcher may lose some confounder information, since the clinical practitioners dominate the process before the studies. However, there may exist a small fraction of data from the same study. Chen and Chen (2000) developed a two-stage sampling design method using the internal validation data. Stamey et al. (2014) presented a Bayesian simulation for the cost-effectiveness studies, including the internal validation scenario. Furthermore, Lin and Chen (2014) provided both Bayesian and large sample approximation approaches with internal validation using the Chen and Chen (2000) approach.

Let T denote the survival times and δ be the censoring status. Let X_1, $\mathbf{Z}$, and U be defined as in the previous section. Also, let $\tilde{X}_1$, $\tilde{\mathbf{Z}}$, and $\tilde{U}$ denote the validation data. Let $\psi = (\psi_1, \psi_2, \cdots, \psi_J)$ be the baseline hazard functions. Suppose the validation size is m, and the continuous unmeasured confounder follows a normal distribution such that

$$u_i \sim \text{Normal}(\mu_i, \sigma^2_{u_i}), \tag{3.27}$$

where the mean of the unmeasured confounder is

$$\mu_i = \eta_0 + \eta_1 x_{1i}. \tag{3.28}$$

Then, the likelihood function of $(\beta, \Theta, \lambda, \eta)$ can be written as

$$\begin{aligned} L(\beta_1, \Theta, \psi, \eta | D, E^*) \propto & \prod_{i=1}^{n} \left\{ \prod_{j=1}^{J} \left[\varphi(i,j)\right]^{d(i,j)} \exp\left[-\varphi(i,j)\right] \exp\left[-\frac{(u_i - \eta_0 - \eta_1 x_{1i})^2}{2\sigma_u^2} \right] \right\} \\ & \times \prod_{k=1}^{m} \left\{ \prod_{j=1}^{J} \left[\varphi^*(k,j)\right]^{d^*(k,j)} \exp\left[-\varphi^*(k,j)\right] \exp\left[-\frac{(\tilde{u}_k - \eta_0 - \eta_1 \tilde{x}_{1k})^2}{2\sigma_{\tilde{u}}^2} \right] \right\}, \end{aligned}$$

where D is defined the same as in the previous section and $E^* = (\tilde{X}_1, \tilde{\mathbf{Z}}, \tilde{U})$.

Again, we still used diffuse normal priors for the regression coefficients β_1, η, Θ, and λ. That is,

$$\beta_1, \eta, \Theta, \lambda \sim \text{Normal}(0, 10). \tag{3.29}$$

The gamma process priors were assigned for the baseline hazard function as in Equation 3.15.

The data generation process was similar to the external validation case, which is given in Equations 3.7 and 3.15. The validation data sets were generated with the same coefficients of the main study, and the only difference was the sample size. Again, the simulation iteration was 40 with $n = 1000$ subjects in the main study data, while the validation size was 100. The censoring rates for validation data were

also the same as in the main study, in which the upper bound of uniform distribution was dependent on the rates at 30%, 60%, and 90%.

The time-to-event model is given in Equations 3.7 and 3.15, and the naive model assumes that $\lambda = 0$ in Equation 3.7. In Table 3.4, we can see that the posterior mean of β_1 was equal to the nominal value, compared with 0.47 in the navie model. All of the other posterior means of parameters performed much better than the naive model without inflating the variability. It is interesting that the posterior means of η_0 and η_1 also got close to the simulation set-ups.

Table 3.4: Internal: summary statistics of the posterior for the model that account for the unmeasured confounder (top) and the naive model (bottom)

Parameter	True value	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
β_1	0.30	0.30	0.975	0.34
		0.47	0.925	0.31
β_2	-1.2	-0.82	0.725	0.60
		-0.67	0.425	0.58
β_3	0.40	0.31	0.825	0.35
		0.29	0.725	0.35
β_4	1	0.81	0.625	0.48
		0.83	0.70	0.47
λ	-3	-0.25	0	2.37
		NA [a]	NA	NA
η_0	0.60	0.60	0.95	0.11
		NA	NA	NA
η_1	-0.04	-0.04	1.00	0.16
		NA	NA	NA

[a] NA means not applicable

Tables 3.5 and 3.6 show the simulation results for β_1 under 30% and 90% censoring rates. Consistent with the external validation scenario, the performance of Bayesian posterior means are negatively associated with the censoring rates. That is, when the censoring rate is increasing, we lose information from the observed data, which increases the bias. Furthermore, we can see that the model with internal

validation provided a more stable posterior mean estimation, compared with the external validation and naive model, even when the censoring rate increased from 30% to 90%.

Table 3.5: The posterior estimation of $\beta_1 = 0.3$ when the censoring rate is 30%

Model	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
Naive	0.47	0.925	0.31
External	0.31	0.95	0.34
Internal	0.30	0.975	0.34

Although the posterior means of β_1 in the external and internal validation models were very close at a 30% censoring rate, the gap of posterior means between these two models was enlarged when the censoring rate was 90%. Furthermore, the posterior mean for the naive model was deteriorated from the nominal value when the censoring rate increased, e.g., β_1 changed from 0.30 to 0.56.

Table 3.6: The posterior estimation of $\beta_1 = 0.3$ when the censoring rate is 90%

Model	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
Naive	0.56	0.95	0.825
External	0.43	0.925	1.05
Internal	0.37	0.975	1.11

The posterior densities of internal and external validation models are plotted in Figure 3.4. We can see that the internal validation model had slightly less variation, compared to the external model in our normal unmeasured confounder case. Also, the external validation model showed similar performance in terms of parameter estimates.

In Figure 3.5, the internal validation model performed better than the others. However, all three models do not show a sharp variation between each other.

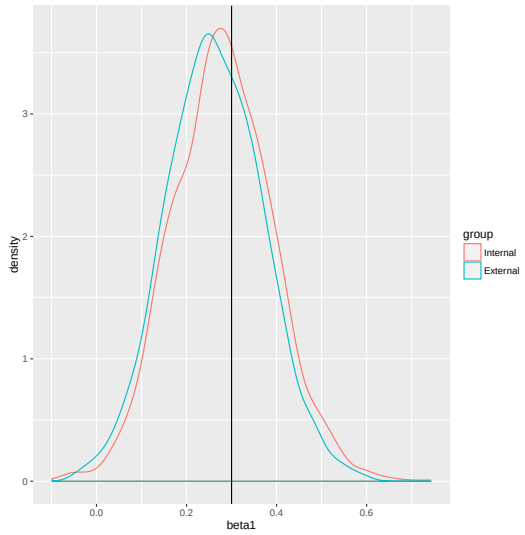

Figure 3.4: Density plots of internal and external validation models for β_1

3.3.5 Discussion

A continuous unmeasured confounder is common in observational studies, such as the blood pressure in the RHC example. Most of these confounders can be assumed to follow a normal distribution. However, the other continuous distributions are also applicable for the unmeasured confounder, such as the gamma, beta, and log-normal distributions. Unlike the regular sensitivity analysis (Lin, Psaty, and Kronmal 1998; Schneeweiss 2006), our approach provides not only the variability of the treatment effect due to the unmeasured confounder, but also the full posterior means of coefficients for the exposure effect and other covariates.

When the censoring rate is low, say 30% as in our study, we only have partial information of the unmeasured confounder, such as summary statistics across the exposure indicator. The bias adjustment is performed well, compared with the unadjusted model. Also, we can extend our method to any other missing covariates or complex measurement errors with the appropriate assumptions of the relationships between unmeasured and measured variables or the summary statistics from meta-analysis.

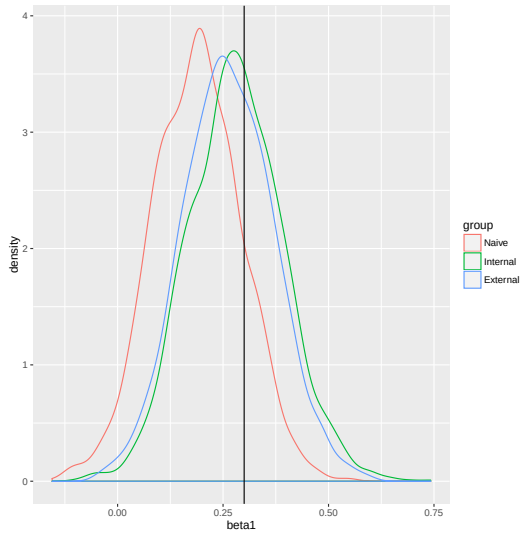

Figure 3.5: Density plots of naive, internal, and external validation model for β_1

3.4 Bayesian Proportional Hazards Model with Binary Unmeasured Confounders

The frequentist sensitivity analysis approaches for binary unmeasured confoundings were discussed by Lin et al. (1998), Schneeweiss (2006), and Lin and Chen (2014). In addition, Bayesian approach for bias correction and sensitivity analysis were discussed by McCandless et al. (2007, 2012), Faries et al. (2013), and Stamey et al. (2014).

In this section, we explore a Bayesian semi-parametric Cox proportional hazards model with a binary unmeasured confounder. Both external and internal validations are discussed. For the external binary unmeasured confounder, we assume that there are only summary statistics in a contingency table (Table 3.7). For the internal validation, all the data are observed, including the unmeasured confounder. There is no missing data issue here. Again, we assume a relationship exists between the unmeasured confounder and the observed covariates.

Let U denote the binary unmeasured confounder, and let X_1 and $\mathbf{Z}$ be defined as in the previous section. $\omega(i, j)$ is defined at the start of this chapter. Suppose U follows a Bernoulli distribution such that

$$u_i \sim \text{Bernoulli}(P_{u_i}), \tag{3.30}$$

where $P_{u_i} = Pr(u_i = 1|x_{1i})$. Since u_i is binary data, logistic regression is employed. Thus, we have

$$\text{logit}(P_{u_i}) = \eta_0 + \eta_1 x_{1i}. \tag{3.31}$$

Together with Equation 3.7, the likelihood function can be expressed as

$$\begin{aligned} L_i(\beta_1, \Theta, \lambda, \eta|D) = & [\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)]^{\delta_i} \\ & \times \exp\left\{ -\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i) \left[\sum_{j=1}^{i*} \omega(i,j) \right] \right\} \\ & \times P_{u_i}^{u_i} \left(1 - P_{u_i}\right)^{1-u_i}, \end{aligned}$$

where $i*$ is the largest integer such that $s_{i*} < t_i$ and $\psi_1, \psi_2, \cdots, \psi_J$ are the baseline hazard rates.

To implement Bayesian analysis, diffuse priors were assigned on the unknown regression coefficient parameters. Independent normal priors are derived as follows:

$$\pi(\beta_1, \beta_2, \cdots, \beta_i) \propto \prod_{j=1}^{i} \exp\left\{ -\frac{(\beta_j - \mu_{\beta_j})^2}{2\sigma_{\beta_j}^2} \right\}. \tag{3.32}$$

Suppose there is belief that the odds ratio for the treatment exposure group ($x_1 = 1$) against the placebo ($x_1 = 0$) is between $\frac{1}{k}$ and k with a probability of 95%. These normal prior distributions can be rewritten as

$$\beta_j \sim N\left\{ \mu_{\beta_j}, \left[\frac{\log(k)}{1.96} \right]^2 \right\} \quad \text{for } i = 1, \cdots, 4 \text{ and } j = 1, 2, 3. \tag{3.33}$$

Continuing with the non-informative priors assumption, let $\mu_{\beta_j} = 0$ and $\sigma_{\beta_j}^2 = 10$. After a basic algebra inverse transformation, we have

$$k = \exp[1.96\sqrt{10}] = 491.7961. \tag{3.34}$$

In other words, the prior distribution can capture the odds ratio range from 1/491.7961 to 491.7961, which is large enough to capture the magnitude of typical observational studies or clinical trials.

The last unknown parameters are the baseline hazard rates $\psi_1, \psi_2, \cdots, \psi_J$, which are non-negative distributions. To guarantee the non-negative property, gamma process priors are applied. That is,

$$\pi(\psi_1, \cdots, \psi_J) \propto \prod_{j=1}^{J} \psi_j^{\alpha_j - 1} \exp(-\psi_j \kappa_j) \quad \text{for} \quad j = 1, \cdots, J, \tag{3.35}$$

where J is the number of intervals set up for the semi-parametric Cox proportional hazards models. To offer a sequence of diffuse priors for ψ, we let the values α_j and κ_j be the same as in Equation 3.17. That is,

$$\psi_j \sim \text{Gamma}(0.01, 0.01) \quad \text{for} \quad j = 1, 2, \cdots, J. \tag{3.36}$$

3.4.1 External Validation with Binary Unmeasured Confounders

We assume that the binary unmeasured confounder U is independent with other covariates, except for the exposure indicator X_1. The relationship between the binary unmeasured confounder U and the treatment exposure indicator X_1 can be expressed as follows:

$$\text{logit}(p_U) = \eta_0 + \eta_1 X_1, \tag{3.37}$$

where p_U is the probability of $U = 1$, given X_1. Initially, we assume the data is in the form of a 2×2 table, such as displayed in Table 3.7,

Table 3.7: Contingency table between an unmeasured confounder and the treatment exposure indicator

U	$X_1 = 0$	$X_1 = 1$
$U = 0$	n_{00}	n_{01}
$U = 1$	n_{10}	n_{11}

where n_{10} represents the number of subjects with $U = 1$ when $X_1 = 0$ and n_{11} represents the number of subjects with $U = 1$ when $X_1 = 1$.

Suppose the marginal distribution of X_1 is fixed. That is, the columns of Table 3.7 are fixed. Then n_{10} and n_{11} follow a binomial distribution

$$n_{10} \sim \text{Binomial}(n_{00} + n_{10}, p_{n_{10}})$$
$$n_{11} \sim \text{Binomial}(n_{01} + n_{11}, p_{n_{11}}),$$

where $p_{n_{10}}$ denotes the probability of $U = 0$ when the total sample size is $n_{00} + n_{10}$ and $X_1 = 0$, and $p_{n_{11}}$ denotes the probability of $U = 1$ when the total sample size is $n_{01} + n_{11}$ and $X_1 = 1$. Since the marginal distribution is fixed, we can define $p_{n_{10}}$ and $p_{n_{11}}$ as

$$p_{n_{10}} = Pr(U = 1|X_1 = 0)$$
$$p_{n_{11}} = Pr(U = 1|X_1 = 1).$$

We employ logistic regression to the binary response variable U, where the link function is logit and the predictor variable is X_1. That is,

$$g(\mu_i) = \eta_0 + \eta_1 x_{1i} \quad \text{for} \quad i = 1, 2, \tag{3.38}$$

where $g(\mu_i)$ is the logit function. This leads to the following equations:

$$\text{logit}(p_{n_{10}}) = \eta_0$$
$$\text{logit}(p_{n_{11}}) = \eta_0 + \eta_1.$$

3.4.2 Bayesian Inference

This form of external validation provides information about η_0 and η_1 in the model. Suppose i^* is the largest integer such that $a[i] \leq t < a[i+1]$, where $a[i]$ was defined at the start of this chapter. Incorporating the validation data information,

the likelihood function for the i^{th} subject is expressed as follows:

$$\begin{aligned} L_i(\beta_1, \Theta, \lambda, \eta | D) \propto & [\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)]^{\delta_i} \\ & \times \exp\left\{ -\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i) \left[\sum_{j=1}^{i*} \omega(i,j) \right] \right\} \\ & \times P_{u_i}^{u_i} \left(1 - P_{u_i}\right)^{1-u_i} \left(P_{n_{10}}\right)^{n_{10}} (1 - P_{n_{10}})^{n_{00}} \\ & \times (P_{n_{11}})^{n_{11}} (1 - P_{n_{11}})^{n_{01}}, \end{aligned}$$

where $D = (X_1, \mathbf{Z}, \delta, n_{00}, n_{01}, n_{10}, n_{11})$. $P_{n_{10}}$ and $P_{n_{11}}$ are defined as follows:

$$\begin{aligned} P_{n_{10}} &= \exp(\eta_0)/[1 + \exp(\eta_0)] \\ P_{n_{11}} &= \exp(\eta_0 + \eta_1)/[1 + \exp(\eta_0 + \eta_1)]. \end{aligned}$$

Together with Equations 3.2, 3.33, and 3.34, we assign the following priors for the parameters:

$$\begin{aligned} \beta_i &\sim \text{Normal}(0, 10) \quad \text{for } i = 1, 2, 3, 4 \\ \lambda &\sim \text{Normal}(0, 10) \\ \eta_j &\sim \text{Normal}(0, 10) \quad \text{for } j = 1, 2. \end{aligned}$$

The prior for the baseline hazards function ψ is

$$\psi_j \sim \text{Gamma}(0.01, 0.01) \quad \text{for} \quad j = 1, 2, \cdots, J.$$

3.4.3 Simulation Studies

Following Equation 3.19, we assumed the sample size for the main study was $n = 1000$, while the validation size was $m = 100$. The data generation process with a binary unmeasured confounder is discussed in this section. First, we assumed that x_1, z_1, and z_2 follow independent Bernoulli distributions such that

$$\begin{aligned} x_1 &\sim \text{Bernoulli}(0.4) \\ z_1 &\sim \text{Bernoulli}(0.08) \\ z_2 &\sim \text{Bernoulli}(0.24). \end{aligned}$$

Second, we assumed z_3 follows a normal distribution, which is also independent of the other covariates. That is,

$$z_3 \sim \text{Normal}(61, 16).$$

Then, the binary unmeasured confounder u was generated as

$$u|x_1 \sim \text{Bernoulli}\left[\frac{\exp(\eta_0 + \eta_1 x_1)}{1 + \exp(\eta_0 + \eta_1 x_1)}\right],$$

where $\eta_0 = -1.3$ and $\eta_1 = -1.6$.

The coefficients for the Cox proportional hazards model were assigned as follows:

$$h(t|x_1, \mathbf{z}, u) = h_0(t) \exp(-0.3x_1 - 1z_1 + 0.3z_2 + 0.005z_3 - 1.4u). \tag{3.39}$$

The validation summary statistics were generated with the same configurations as above, except for the sample size ($m = 100$). Then, we followed the method in Bender et al. (2005) to generate the survival data with a binary unmeasured confounder. The censoring time was drawn from a uniform distribution, which is the same as in Equation 3.24. The censoring rates were 30%, 60%, and 90%, respectively.

Since there is no closed form of marginal posterior distribution (the product of the likelihood function and prior distributions), the MCMC method was deployed to sample the target posterior distribution. The JAGS package was used to simulate drawing the samples of parameters via the `rjags` package in R with 7000 burn-in and 20000 iterations. Convergence was diagnosed by Gelman-Rubin's potential scale reduction factor. Also, we examined the autocorrelation plot. There was no convergence issue.

Table 3.8 displays the posterior means coverage of 95% intervals and the interval widths for the regression coefficients of two models when the censoring rate was at 30%. We can see that the external validation summary statistics had an effect on

our bias adjustment. All the posterior means were close to the nominal value -0.30 for β_1, and the coverage of 95% intervals also increased from 77.5% to 95%.

Aslo, the Bayesian semi-parametric Cox proportional hazards model provided decent estimates for the other parameters in Equation 3.1 without increasing the interval widths dramatically. Equation 3.1 accurately captured the effect of β_4, even though the coefficient was very small. It is interesting that the average 95% credible interval did not enlarge too much compared to the original value in the naive model. We can draw the conclusion that the Bayesian semi-parametric Cox proportional hazards model with a binary unmeasured confounder can be adjusted by the external summary statistics if the information is "informative" enough. In other words, the model is essentially identified (Gustafson et al. 2005).

Table 3.8: External: summary statistics of the posteriors for the model accounting for a binary unmeasured confounder (top) and the naive model (bottom) at a 30% censoring rate

Parameter	True value	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
β_1	-0.30	-0.37	0.95	0.43
		-0.42	0.775	0.31
β_2	-1	-0.78	0.8	0.61
		-0.79	0.85	0.60
β_3	0.30	0.23	0.875	0.36
		0.27	0.95	0.35
β_4	0.005	0.004	1.00	0.01
		0.004	0.925	0.01
λ	1.4	2.02	0.70	6.18
		NA [a]	NA	NA
η_0	-1.3	-1.09	0.925	1.72
		NA	NA	NA
η_1	-1.6	-2.01	0.90	3.63
		NA	NA	NA

[a] NA means not applicable

To investigate the relationship between different censoring rates and the ability of bias correction under a binary unmeasured confounder scenario, we applied 30%,

60%, and 90% censoring rates for the external validation. In Table 3.9, it is apparent that the interval widths increased simultaneously with the increasing of the censoring rate. When the censoring rate was at 90%, the width of the 95% interval was five times larger than the 30% censoring rate, although the posterior mean did not shrink sharply.

Table 3.9: The posterior estimation of $\beta_1 = -0.3$ with an external validation at different censoring rates

Model	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
30%	-0.36	0.95	0.43
60%	-0.44	0.925	0.64
90%	-0.43	0.975	2.09

In Figure 3.6, we can see that the external validation model performed better than the naive model, which did not account for the unmeasured confounder. Also, the naive model showed less variation at the same simulation set-ups.

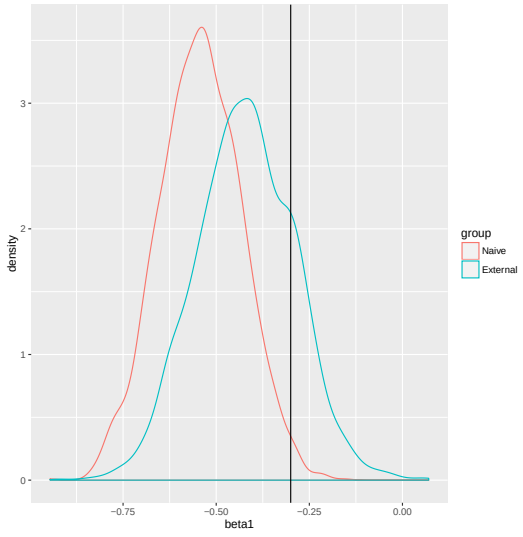

Figure 3.6: Density plots of naive and external validation models for β_1

3.4.4 Internal Validation with Binary Unmeasured Confounders

In the retrospective cohort study, we may not be able to collect all the variables that we are interested in, especially when there are ethical and privacy dilemmas. Also, it is common to have large amounts of missing data in the open label clinical trials, and this missing data is important in the analysis. For example, we are interested in the pain medication information when the key endpoint is time-to-pain medication. However, the patient report forms have messy data due to the different brands of pain-killer drugs. If the patients did not follow the study protocol, then the pain medication will be assigned as a missing record on the self-reported form.

Now, we present a semi-parametric Bayesian Cox proportional hazards model with a binary unmeasured confounder using internal validation data. The data generation process was the same as the previous simulation, except for the validation size. Let the main study sample size be $n = 1000$ and the validation size be $m = 100$.

Let T denote the survival times and X_1 be the treatment exposure indicator. $\mathbf{Z}$, δ, and U were defined in previous sections. Let $\tilde{X}_1$, $\tilde{T}$, $\tilde{\mathbf{Z}}$, $\tilde{\delta}$, and $\tilde{U}$ be the observed variables in the internal validation data set. The binary unmeasred confounder can be determind by the exposure indicator. That is,

$$u_i \sim \text{Bernoulli}(P_{u_i}) \quad \text{for } i = 1, 2, \cdots, n \tag{3.40}$$

$$\tilde{u}_j \sim \text{Bernoulli}(P_{\tilde{u}_j}) \quad \text{for } j = 1, 2, \cdots, m, \tag{3.41}$$

where P_{u_i} and $P_{\tilde{u}_j}$ are derived as follows:

$$\text{logit}(P_{u_i}) = \eta_0 + \eta_1 x_{1i} \tag{3.42}$$

$$\text{logit}(P_{\tilde{u}_j}) = \eta_0 + \eta_1 \tilde{x}_{1j}. \tag{3.43}$$

We assumed that m subjects were fully observed, including the binary unmeasured confounder. Suppose i^* is the largest integer such that $s[i] \leq t$, where $s[i]$ was

defined at the start of this chapter. Then, the likelihood function becomes

$$
\begin{aligned}
L_(\beta_1, \Theta, \lambda, \eta | D) \propto & \prod_{i=1}^{n} [\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i)]^{\delta_i} \\
& \times \exp\left\{ -\psi_{i*} \exp(\beta_1 x_{1i} + \Theta \mathbf{Z_i}' + \lambda u_i) \left[\sum_{j=1}^{i*} \omega(i,j) \right] \right\} \\
& \times P_{u_i}^{u_i} (1 - P_{u_i})^{1-u_i} \\
& \times \prod_{k=1}^{m} [\psi_{k*} \exp(\beta_1 \tilde{x}_{1k} + \Theta \tilde{\mathbf{Z}}_k' + \lambda \tilde{u}_k)]^{\tilde{\delta}_k} \\
& \times \exp\left\{ -\psi_{k*} \exp(\beta_1 \tilde{x}_{1k} + \Theta \tilde{\mathbf{Z}}_k' + \lambda \tilde{u}_k) \left[\sum_{m=1}^{k*} \tilde{\omega}(k,m) \right] \right\} \\
& \times P_{\tilde{u}_k}^{\tilde{u}_k} (1 - P_{\tilde{u}_k})^{1-\tilde{u}_k} ,
\end{aligned}
$$

where K^* is the largest integer such that $s[k] \leq \tilde{t}$. P_{u_i} and $P_{\tilde{u}_i}$ were defined above, and $\omega(i,j)$ and $\tilde{\omega}(k,m)$ were defined at the start of this chapter.

To analyze this simulated data, diffuse priors were assigned to regression coefficients. That is,

$$\beta_1, \Theta, \lambda | \mu_{\phi_i}, \sigma^2_{\phi_i} \sim \text{Normal}(\mu_{\phi_i}, \sigma^2_{\phi_i}),$$

where $\phi = (\beta_1, \Theta, \lambda)$, $\mu_{\phi_i} = 0$, and $\sigma^2_{\phi_i} = 10$.

Again, we relied on gamma process priors for the baseline hazard rates. Let $\alpha = \kappa = 0.01$. The variance for this gamma distribution was $\alpha/\kappa^2 = 100$, which provided a sequence of relatively diffuse priors. That is,

$$\pi(\psi_1, \cdots, \psi_J) \propto \prod_{j=1}^{J} \psi_j^{\alpha_j - 1} \exp(-\psi_j \kappa_j) \quad \text{for} \quad j = 1, \cdots, J,$$

where $J = 5$ in this section.

To perform the Bayesian simulation, the MCMC method was employed via the `rjags` package. We ran 20000 updates with 7000 burn-in. Considering the autocorrelation, we kept every 10th draw in rjags. The convergence and autocorrelation plots were checked, and there was no sign of divergence or autocorrelation.

The simulation results for this model are summarized in Tables 3.10 and 3.11. The naive model was the one in which we did not account for the binary unmeasured confounder. Table 3.10 shows the posterior results of two models with internal

Table 3.10: Internal: summary statistics of the posterior for the model that accounts for a binary unmeasured confounder (top) and the naive model (bottom) at a 30% censoring rate

Parameter	True value	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
β_1	-0.30	-0.30	0.975	0.35
		-0.42	0.775	0.31
β_2	-1	-0.84	0.825	0.61
		-0.79	0.85	0.60
β_3	0.30	0.27	0.95	0.35
		0.27	0.95	0.35
β_4	0.005	0.004	1.00	0.01
		0.004	0.925	0.01
λ	1.4	1.14	0.875	1.22
		NA [a]	NA	NA
η_0	-1.3	-1.40	0.925	1.16
		NA	NA	NA
η_1	-1.6	-2.27	0.90	3.61
		NA	NA	NA

[a] NA means not applicable

validation and a naive model. We can see that the average posterior mean of β_1 was almost equal to the nominal value. Also, it indicates that the credible interval was more narrow, compared to the models with external validation. This was especially true when the censoring rate was high. In other words, the internal validation method provided a more stable bias adjustment compared to the external validation and naive models.

The density plots for external and internal validation models in the binary unmeasured confounder scenario were plotted in Figure 3.7. Consistent with the

Table 3.11: The posterior estimation of $\beta_1 = -0.3$ when the censoring rate is 90%

Model	Average posterior mean	Coverage of 95% intervals	Average 95% interval width
Naive	-0.65	0.65	0.95
External	-0.43	0.975	2.09
Internal	-0.29	0.925	1.37

results of Table 3.11, the parameter estimate of the internal validation model was close to the nominal value at the price of slightly larger variation, compared to the external validation model.

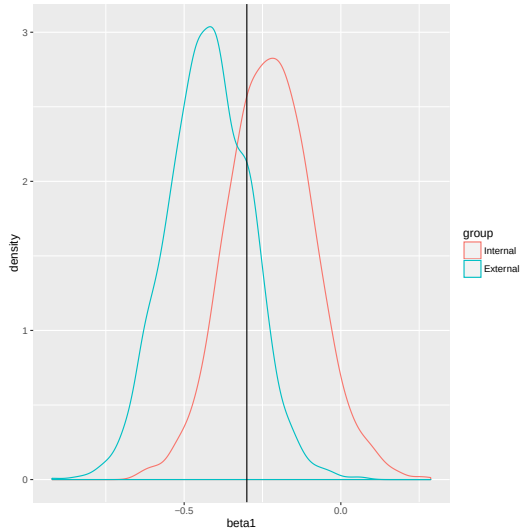

Figure 3.7: Density plots of internal and external validation models for β_1

In Figure 3.8, we can see that the models with a binary unmeasured confounder showed significant discrepancy compared to the normal unmeasured confounder scenario. Both the external and the internal validation models performed much better than the naive model, which did not account for the binary unmeasured confounder.

3.5 Discussion

In this chapter, we developed the semi-parametric Bayesian Cox proportional hazards model with a binary unmeasured confounder in the light of McCandless et al.

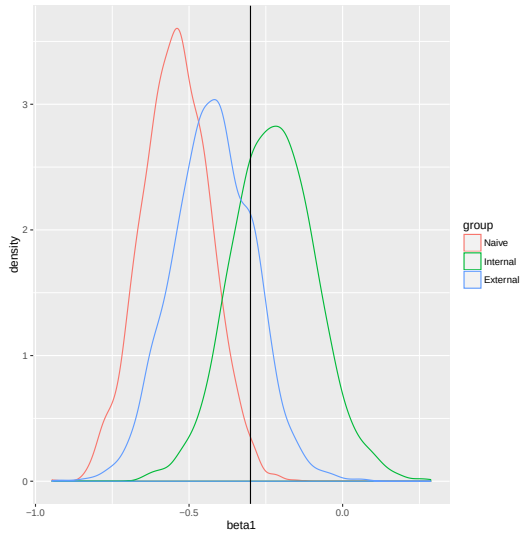

Figure 3.8: Density plots of naive, internal, and external validation models for β_1

(2012) and Stamey et al. (2014). One model accounted for the continuous normal unmeasured confounder, and the other one adjusted for the binary unmeasured confounder. Multiple simulations were conducted to study the performance of the validation data from different sources.

The frequentist sensitivity analyses of the binary unmeasured confounder were implemented by Lin et al. (1998) and Sturmer et al. (2005). McCandless et al. (2007) and Stamey et al. (2014) proposed Bayesian approaches with the distributional assumption of the unmeasured confounder using external validation data. However, none of these methods can account for the binary unmeasured confounder, unlike the external summary statistics from the other studies.

The simulation results confirmed that external summary statistics play a role on the estimation of parameters, as well as the coverage of an interval without inflating the variability of interval widths. Furthermore, the censoring rate has influence on the parameter estimations and the variation of 95% interval widths. When the censoring rate is higher than 90%, we may need to consider other techniques, such as the non-parametric adjustment in McCandless et al. (2012), or we may need to look for the internal validation data.

Although we only offer five baseline hazard rates ($J = 5$) for the analysis, the semi-parametric Bayesian Cox proportional hazards model with normal and binary unmeasured confounders shows a good ability to adjust for bias. The number of baseline hazard rates is discussed in Christensen et al. (2010). In the continuous unmeasured confounder scenario, the constructed informative priors performed much better than in the binary case, considering the bias correction and the coverage of 95% credible intervals. Furthermore, it is wise to look for the availability of internal validation data, which provides much more stable and accurate estimations of regression coefficients, even when the censoring rate is high.

CHAPTER FOUR

Conclusion

In this dissertation, we considered a Bayesian approach to model time-to-event data with an unmeasured confounder. We also developed a Bayesian parametric survival regression model to account for the effect of a continuous unmeasured confounder. The linear regression models proposed for the unmeasured confounder assume there is an association between the treatment exposure and the unmeasured confounder. We introduced a Bayesian semi-parametric Cox proportional hazards model, that accounts for unmeasured confounders with binary and normal distributions.

In Chapter Two, we presented the Bayesian parametric survival regression models with the normal unmeasured confounder, assuming that the survival time data followed Weibull and exponential distributions. The censoring rates were set at 60%. Next, made a comparison between Bayesian parametric survival regression models and the naive model, which did not account for the unmeasured confounder. Besides the parameter estimation, a novel survival data generation process was introduced in this chapter. We found that having even a small amount of internal validation data can dramatically improve the estimation of the parameters. However, it is important to choose the correct distribution assumption for the survival data, especially when the censoring rate is high. Also, more parametric techniques for survival data will be of interest, due to the EMA (European Medicines Agency) regulations (Carroll 2007) and the rising of new generalized distributions, such as the beta-Weibull distribution family.

In Chapter Three, we developed the Bayesian semi-parametric Cox proportional hazards model for survival data, adjusting for the unmeasured confounder.

Instead of putting a distribution on the baseline hazard function, we assumed that the baseline hazard function at a specific time window was a fixed value, which we did not know. Mixing with the internal and external validations and the unmeasured confounders with binary and normal distributions, we conducted the Bayesian analysis with six different models. Through the simulations, we found that even relatively small amounts of prior information on the unmeasured confounder will correct the bias. Consistent with the Bayesian parametric survival regression model, the internal validation performed much better than the external validation. Also, the censoring rate is a big concern when there exists only external validation. For instance, assuming that the censoring rate is high, say 90%, the external validation data is incorporated into the Bayesian semi-parametric Cox proportional hazards models, as informative priors may introduce new bias into the analysis.

In the future, we are interested in the effect of informative priors on the unmeasured confounder. Furthermore, the large sample approximation power priors for the Cox proportional hazards regression coefficients will be another area for future research.

APPENDICES

APPENDIX A

R and Stan Programs for Parametric Models

The programs presented here were used for Bayesian parametric (Weibull and exponential) regression for survival times with Weibull regression presented in Chapter 2 . The data were generated by accelerated failure time models, and the simulations were performed with different validation sizes.

A.1 Weibull Regression

```
#####################################################
## Parametric model with Weibull distribution
## Simulations were performed by rstan package
## Survival times were generated with AFT model
#####################################################

library(rstan)

parastan<-function(nmain, nstar, m, b1, b2, b3, lambda, a1,  a2 ,
shape, sdu, seed){

## storage of output
## ave=average  cov=coverage  len=length
## beta
ave.b1<-rep(NA,m)
cov.b1<-rep(NA,m)
len.b1<-rep(NA,m)
```

```
lower.b1<-rep(NA,m)
upper.b1<-rep(NA,m)

ave.b2<-rep(NA,m)
cov.b2<-rep(NA,m)
len.b2<-rep(NA,m)

ave.b3<-rep(NA,m)
cov.b3<-rep(NA,m)
len.b3<-rep(NA,m)

ave.a1<-rep(NA,m)
cov.a1<-rep(NA,m)
len.a1<-rep(NA,m)

ave.a2<-rep(NA,m)
cov.a2<-rep(NA,m)
len.a2<-rep(NA,m)

## lambda
ave.lam<-rep(NA,m)
cov.lam<-rep(NA,m)
len.lam<-rep(NA,m)
lower.lam<-rep(NA,m)
upper.lam<-rep(NA,m)

## shape parameter of Weibull distribution
```

```
ave.shp<-rep(NA,m)
cov.shp<-rep(NA,m)
len.shp<-rep(NA,m)

##The survival time and censoring
##rule depends on specific study
cens.rate<-rep(NA,m)

for (t in 1:m){
# baseline hazard of survival time
lambdaT <- 1
# baseline hazard of censoring
lambdaC <- 0.5

## x1 is treatment, 1 is treatment, 0 is placebo
## the coefficient of x1 is our primary interest
x1<-rbinom(nmain,1,0.6)
z1<-rbinom(nmain,1,0.3)
z2<-rnorm(nmain, 0, 1)

# Generate unmeasured confoundings
u <- rnorm(nmain, cbind(1,x1) %*% c(a1, a2), sdu)

# Generate the survival time
T <- rweibull(nmain, shape, scale=lambdaT*
exp(-b1*x1-b2*z1-b3*z2-lambda*u))
C <- rweibull(nmain, shape, scale=lambdaC)
```

```
dur<-pmin(T,C)
event <- (dur==T)*1   # set to 1 if event is

cens.rate[t]<-1-sum(event)/nmain

## Validation data generation
x1.star<-rbinom(nstar,1,0.6)
z1.star<-rbinom(nstar,1,0.3)
z2.star<-rnorm(nstar, 0, 1)
## continuous confounding
## R and Stan has different expression
u.star <- rnorm(nstar, cbind(1,x1.star) %*% c(a1, a2), sdu)

time.star <- rweibull(nstar, shape, scale=lambdaT*
exp(-b1*x1.star-b2*z1.star-b3*z2.star-lambda*u.star))
cens.star <- rweibull(nstar, shape, scale=lambdaC)
iscens.star<-(time.star>cens.star)
event.star <-1-iscens.star*1
dur.star<-pmin(time.star, cens.star)
## begin stan code
code="data{
int <lower=1> nmain;
int <lower=1> nstar;

int <lower=0,upper=1> event[nmain];
int <lower=0,upper=1> eventstar[nstar];
```

```
vector<lower=0.00,upper=1.00>[nmain]x1;
vector<lower=0.00,upper=1.00>[nmain]z1;
vector[nmain]z2;

vector<lower=0.00,upper=1.00>[nstar]x1star;
vector<lower=0.00,upper=1.00>[nstar]z1star;
vector [nstar] z2star;
vector [nstar] ustar;

vector [nstar] durstar;
vector [nmain] dur;
}

// parameters
parameters{
vector[3] beta;
vector[2] alpha;
real lambda;
vector[nmain] u;
real <lower=0> b;
real <lower=0> sigu;
}

transformed parameters{
vector [nstar] b1;
vector [nmain] b2;
b1<-beta[1]*x1star+beta[2]*z1star+beta[3]*z2star+lambda*ustar;
```

```
b2<-beta[1]*x1+beta[2]*z1+beta[3]*z2+lambda*u;
}
// models

model{
// prior information
lambda~normal(0,10);
beta~normal(0,10);
alpha~normal(0,10);
b~gamma(0.01, 0.01);
sigu~gamma(0.01, 0.01);

// validation data set
ustar~normal(alpha[1]+alpha[2]*x1star, sigu);

for(i in 1:nstar){
increment_log_prob(eventstar[i]*(log(b)-b*log(exp(-b1[i]))+
(b-1)*log(durstar[i]))-pow((durstar[i]/exp(-b1[i])),b));
}

// main study
u~normal(alpha[1]+alpha[2]*x1,sigu);
for(j in 1:nmain){
increment_log_prob(event[j]*(log(b)-b*log(exp(-b2[j]))+
(b-1)*log(dur[j]))-pow((dur[j]/exp(-b2[j])),b));
}
}
```

```
"
data=list(nmain=nmain,nstar=nstar,x1=x1,z1=z1,z2=z2,
x1star=x1.star, z1star=z1.star, z2star=z2.star, ustar=u.star,
dur=dur,durstar=dur.star, event=event, eventstar=event.star)
fit=stan(model_code=code, data=data, iter=20000, thin=10, chains=1,
warmup=7000, pars=c("beta","alpha","lambda","b"))
ss<-extract(fit)
## output the results

ave.b1[t]<-mean(ss$beta[,1])
cov.b1[t]<-(b1>=quantile(ss$beta[,1],probs=c(0.025),
na.rm=TRUE))&(b1<=quantile(ss$beta[,1],probs=c(0.975),na.rm=TRUE))
len.b1[t]<-quantile(ss$beta[,1],probs=c(0.975),na.rm=TRUE)-
quantile(ss$beta[,1],probs=c(0.025),na.rm=TRUE)
lower.b1[t]<-quantile(ss$beta[,1],probs=c(0.025),na.rm=TRUE)
upper.b1[t]<-quantile(ss$beta[,1],probs=c(0.975),na.rm=TRUE)

ave.b2[t]<-mean(ss$beta[,2])
cov.b2[t]<-(b2>=quantile(ss$beta[,2],probs=c(0.025),na.rm=TRUE))&
(b2<=quantile(ss$beta[,2],probs=c(0.975),na.rm=TRUE))
len.b2[t]<-quantile(ss$beta[,2],probs=c(0.975),na.rm=TRUE)-
quantile(ss$beta[,2],probs=c(0.025),na.rm=TRUE)

ave.b3[t]<-mean(ss$beta[,3])
cov.b3[t]<-(b3>=quantile(ss$beta[,3],probs=c(0.025),na.rm=TRUE))&
(b3<=quantile(ss$beta[,3],probs=c(0.975),na.rm=TRUE))
len.b3[t]<-quantile(ss$beta[,3],probs=c(0.975),na.rm=TRUE)-
```

```
quantile(ss$beta[,3],probs=c(0.025),na.rm=TRUE)

ave.a1[t]<-mean(ss$alpha[,1])
cov.a1[t]<-(a1>=quantile(ss$alpha[,1],probs=c(0.025),na.rm=TRUE))&
(a1<=quantile(ss$alpha[,1],probs=c(0.975),na.rm=TRUE))
len.a1[t]<-quantile(ss$alpha[,1],probs=c(0.975),na.rm=TRUE)-
quantile(ss$alpha[,1],probs=c(0.025),na.rm=TRUE)

ave.a2[t]<-mean(ss$alpha[,2])
cov.a2[t]<-(a2>=quantile(ss$alpha[,2],probs=c(0.025),na.rm=TRUE))&
(a2<=quantile(ss$alpha[,2],probs=c(0.975),na.rm=TRUE))
len.a2[t]<-quantile(ss$alpha[,2],probs=c(0.975),na.rm=TRUE)-
quantile(ss$alpha[,2],probs=c(0.025),na.rm=TRUE)

## lambda
ave.lam[t]<-mean(ss$lambda)
cov.lam[t]<-(lambda>=quantile(ss$lambda, probs=c(0.025),na.rm=TRUE))&
(lambda<=quantile(ss$lambda, probs=c(0.975),na.rm=TRUE))
len.lam[t]<-(quantile(ss$lambda, probs=c(0.975),na.rm=TRUE))-
(quantile(ss$lambda, probs=c(0.025),na.rm=TRUE))
lower.lam[t]<-quantile(ss$lambda, probs=c(0.025),na.rm=TRUE)
upper.lam[t]<-quantile(ss$lambda, probs=c(0.975),na.rm=TRUE)
## shape
ave.shp[t]<-mean(ss$b)
cov.shp[t]<-(shape>=quantile(ss$b, probs=c(0.025),na.rm=TRUE))&
(shape<=quantile(ss$b, probs=c(0.975),na.rm=TRUE))
len.shp[t]<-(quantile(ss$b, probs=c(0.975),na.rm=TRUE))-
```

```
(quantile(ss$b, probs=c(0.025),na.rm=TRUE))

}# end of first loop

ave_b1<-mean(ave.b1)
cov_b1<-mean(cov.b1)
len_b1<-mean(len.b1)
lower_b1<-mean(lower.b1)
upper_b1<-mean(upper.b1)

ave_b2<-mean(ave.b2)
cov_b2<-mean(cov.b2)
len_b2<-mean(len.b2)

ave_b3<-mean(ave.b3)
cov_b3<-mean(cov.b3)
len_b3<-mean(len.b3)

ave_a1<-mean(ave.a1)
cov_a1<-mean(cov.a1)
len_a1<-mean(len.a1)

ave_a2<-mean(ave.a2)
cov_a2<-mean(cov.a2)
len_a2<-mean(len.a2)

## lambda
```

```
ave_lam<-mean(ave.lam)
cov_lam<-mean(cov.lam)
len_lam<-mean(len.lam)
lower_lam<-mean(lower.lam)
upper_lam<-mean(upper.lam)

## shape
ave_shp<-mean(ave.shp)
cov_shp<-mean(cov.shp)
len_shp<-mean(len.shp)

csra<-mean(cens.rate)
# pass the return value of function
return(c(ave_b1,cov_b1,len_b1,lower_b1, upper_b1,ave_b2,cov_b2,
len_b2,ave_b3,cov_b3,len_b3,ave_a1,cov_a1,len_a1,
ave_a2,cov_a2,len_a2, ave_lam,cov_lam,len_lam, lower_lam,
upper_lam,ave_shp,cov_shp,len_shp, csra))
}#end of function
ptm<-proc.time()
parastan(1000, 100, 40, -0.3, 0.2, 0.3, -0.8, 0.1, -0.4, 1 ,3, 100)
proc.time()-ptm
```

A.2 Exponential Regression

```
#####################################################################
## Parametric Survival Regression with Exponential Distribution
## The data were generated by Weibull distribution
#####################################################################
library(rstan)
```

```
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
## Begin the self-defined function
parastan<-function(nmain, nstar, m, b1, b2, b3, lambda, a1,  a2 ,
shape, sdu, seed){
## storage of output
## ave=average  cov=coverage  len=length
## beta
ave.b1<-rep(NA,m)
cov.b1<-rep(NA,m)
len.b1<-rep(NA,m)

ave.b2<-rep(NA,m)
cov.b2<-rep(NA,m)
len.b2<-rep(NA,m)

ave.b3<-rep(NA,m)
cov.b3<-rep(NA,m)
len.b3<-rep(NA,m)

ave.a1<-rep(NA,m)
cov.a1<-rep(NA,m)
len.a1<-rep(NA,m)

ave.a2<-rep(NA,m)
cov.a2<-rep(NA,m)
len.a2<-rep(NA,m)
```

```
ave.lam<-rep(NA,m) ## lambda
cov.lam<-rep(NA,m)
len.lam<-rep(NA,m)
## ## the precision of unmeasued confounder 25FEB2015 Wiki
#ave.sigu<-rep(NA,m)
#cov.sigu<-rep(NA,m)
#len.sigu<-rep(NA,m)

cens.rate<-rep(NA,m)
# The survival time and censoring rule depends on specific study

for (t in 1:m){
lambdaT <- 1 # baseline hazard
lambdaC <- 0.5  # hazard of censoring

x1<-rbinom(nmain,1,0.6) # x is treatment, 1 is treatment, 0 is placebo
z1<-rbinom(nmain,1,0.3)# Characteristic 1
z2<-rnorm(nmain, 0, 1)# Characteristic 2
# Generate unmeasured confoundings
u <- rnorm(nmain, cbind(1,x1) %*% c(a1, a2), sdu)

# Generate the survival time
T <- rweibull(nmain, shape, scale=lambdaT*exp(-b1*x1-b2*z1-b3*z2-lambda*u))
C <- rweibull(nmain, shape, scale=lambdaC)   #censoring time
dur<-pmin(T,C)  #observed time is min of censored time and survival time
event <- (dur==T)*1   # set to 1 if event is
```

```
cens.rate[t]<-1-sum(event)/nmain

x1.star<-rbinom(nstar,1,0.6)
z1.star<-rbinom(nstar,1,0.3)
z2.star<-rnorm(nstar, 0, 1)
## continuous confounding
## R and Stan has different expression
u.star <- rnorm(nstar, cbind(1,x1.star) %*% c(a1, a2), sdu)

time.star <- rweibull(nstar, shape, scale=lambdaT*
exp(-b1*x1.star-b2*z1.star-b3*z2.star-lambda*u.star))
cens.star <- rweibull(nstar, shape, scale=lambdaC)   #censoring time
iscens.star<-(time.star>cens.star)
event.star <-1-iscens.star*1
dur.star<-pmin(time.star, cens.star)
## begin stan code
code="data{
int <lower=1> nmain;
int <lower=1> nstar;

int <lower=0,upper=1> event[nmain];
int <lower=0,upper=1> eventstar[nstar];

vector<lower=0.00,upper=1.00>[nmain]x1;
vector<lower=0.00,upper=1.00>[nmain]z1;
vector[nmain]z2;
```

```
vector<lower=0.00,upper=1.00>[nstar]x1star;
vector<lower=0.00,upper=1.00>[nstar]z1star;
vector[nstar]z2star;
vector[nstar]ustar;

vector [nstar] durstar;
vector [nmain] dur;
}

// parameters
parameters{
vector[3] beta;
vector[2] alpha;
real lambda;
vector[nmain] u;
real <lower=0> sigu;
}

transformed parameters{
vector [nstar] b1;
vector [nmain] b2;
b1<-beta[1]*x1star+beta[2]*z1star+beta[3]*z2star+lambda*ustar;
b2<-beta[1]*x1+beta[2]*z1+beta[3]*z2+lambda*u;
}
// models
```

```
model{
// prior information
lambda~normal(0,10);
beta~normal(0, 10);
alpha~normal(0,10);
sigu~gamma(0.01, 0.01);

// validation data set

ustar~normal(alpha[1]+alpha[2]*x1star, sigu);

for(i in 1:nstar){
increment_log_prob(eventstar[i]*(b1[i])-exp(b1[i])*durstar[i]);
}

// main study
u~normal(alpha[1]+alpha[2]*x1,sigu);
for(j in 1:nmain){
increment_log_prob(event[j]*(b2[j])-exp(b2[j])*dur[j]);
}
}
"

data=list(nmain=nmain,nstar=nstar,x1=x1,z1=z1,z2=z2,
x1star=x1.star, z1star=z1.star, z2star=z2.star, ustar=u.star,
dur=dur,durstar=dur.star, event=event, eventstar=event.star)
fit=stan(model_code=code, data=data, iter=15000,chains=1, thin=10,
```

```
warmup=5000, pars=c("beta","alpha","lambda"))
ss<-extract(fit)
## output the results

ave.b1[t]<-mean(ss$beta[,1])
cov.b1[t]<-(b1>=quantile(ss$beta[,1],probs=c(0.025),na.rm=TRUE))&
(b1<=quantile(ss$beta[,1],probs=c(0.975),na.rm=TRUE))
len.b1[t]<-quantile(ss$beta[,1],probs=c(0.975),na.rm=TRUE)-
quantile(ss$beta[,1],probs=c(0.025),na.rm=TRUE)

ave.b2[t]<-mean(ss$beta[,2])
cov.b2[t]<-(b2>=quantile(ss$beta[,2],probs=c(0.025),na.rm=TRUE))&
(b2<=quantile(ss$beta[,2],probs=c(0.975),na.rm=TRUE))
len.b2[t]<-quantile(ss$beta[,2],probs=c(0.975),na.rm=TRUE)-
quantile(ss$beta[,2],probs=c(0.025),na.rm=TRUE)

ave.b3[t]<-mean(ss$beta[,3])
cov.b3[t]<-(b3>=quantile(ss$beta[,3],probs=c(0.025),na.rm=TRUE))&
(b3<=quantile(ss$beta[,3],probs=c(0.975),na.rm=TRUE))
len.b3[t]<-quantile(ss$beta[,3],probs=c(0.975),na.rm=TRUE)-
quantile(ss$beta[,3],probs=c(0.025),na.rm=TRUE)

ave.a1[t]<-mean(ss$alpha[,1])
cov.a1[t]<-(a1>=quantile(ss$alpha[,1],probs=c(0.025),na.rm=TRUE))&
(a1<=quantile(ss$alpha[,1],probs=c(0.975),na.rm=TRUE))
len.a1[t]<-quantile(ss$alpha[,1],probs=c(0.975),na.rm=TRUE)-
quantile(ss$alpha[,1],probs=c(0.025),na.rm=TRUE)
```

```
ave.a2[t]<-mean(ss$alpha[,2])
cov.a2[t]<-(a2>=quantile(ss$alpha[,2],probs=c(0.025),na.rm=TRUE))&
(a2<=quantile(ss$alpha[,2],probs=c(0.975),na.rm=TRUE))
len.a2[t]<-quantile(ss$alpha[,2],probs=c(0.975),na.rm=TRUE)-
quantile(ss$alpha[,2],probs=c(0.025),na.rm=TRUE)

## lambda
ave.lam[t]<-mean(ss$lambda)
cov.lam[t]<-(lambda>=quantile(ss$lambda, probs=c(0.025),na.rm=TRUE))&
(lambda<=quantile(ss$lambda, probs=c(0.975),na.rm=TRUE))
len.lam[t]<-(quantile(ss$lambda, probs=c(0.975),na.rm=TRUE))-
(quantile(ss$lambda, probs=c(0.025),na.rm=TRUE))
}#loop

ave_b1<-mean(ave.b1)
cov_b1<-mean(cov.b1)
len_b1<-mean(len.b1)

ave_b2<-mean(ave.b2)
cov_b2<-mean(cov.b2)
len_b2<-mean(len.b2)

ave_b3<-mean(ave.b3)
cov_b3<-mean(cov.b3)
len_b3<-mean(len.b3)
```

```
ave_a1<-mean(ave.a1)
cov_a1<-mean(cov.a1)
len_a1<-mean(len.a1)

ave_a2<-mean(ave.a2)
cov_a2<-mean(cov.a2)
len_a2<-mean(len.a2)

## lambda
ave_lam<-mean(ave.lam)
cov_lam<-mean(cov.lam)
len_lam<-mean(len.lam)
## shape

csra<-mean(cens.rate)
# pass the return value of function
return(c(ave_b1,cov_b1,len_b1,ave_b2,cov_b2,len_b2,
ave_b3,cov_b3,len_b3,ave_a1,cov_a1,len_a1,ave_a2,cov_a2,
len_a2, ave_lam,cov_lam,len_lam,csra))
}

## The validation size can be 50, 100 or 200
## 40 iterations
## Simulation seed is at 100
## The size for main study is 1000

ptm<-proc.time()
```

```
parastan(1000, 100, 40, -0.3, 0.2, 0.3, -0.8, 0.1, -0.4, 1 ,3, 100)
proc.time()-ptm
```

A.3 Naive model

```
#####################################################
## Naive Model
## Ignoring the unmeasured confounder
#####################################################
library(rstan)
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())

naivstan<-function(nmain, nstar, m, b1, b2, b3, lambda,
a1,  a2 , shape, sdu, seed){
## storage of output
## ave=average  cov=coverage  len=length
## beta
ave.b1<-rep(NA,m)
cov.b1<-rep(NA,m)
len.b1<-rep(NA,m)
lower.b1<-rep(NA,m)
upper.b1<-rep(NA,m)

ave.b2<-rep(NA,m)
cov.b2<-rep(NA,m)
len.b2<-rep(NA,m)

ave.b3<-rep(NA,m)
```

```
cov.b3<-rep(NA,m)
len.b3<-rep(NA,m)

## shape parameter of weibull distribution
ave.shp<-rep(NA,m)
cov.shp<-rep(NA,m)
len.shp<-rep(NA,m)
for (t in 1:m){
lambdaT <- 1 # baseline hazard
lambdaC <- 0.5  # hazard of censoring
# x1 is treatment, 1 is treatment, 0 is placebo
x1<-rbinom(nmain,1,0.6)
z1<-rbinom(nmain,1,0.3)
z2<-rnorm(nmain, 0, 1)
# Generate unmeasured confounding
u <- rnorm(nmain, cbind(1,x1) %*% c(a1, a2), sdu)

# Generate the survival time
T <- rweibull(nmain, shape, scale=lambdaT*
exp(-b1*x1-b2*z1-b3*z2-lambda*u))
C <- rweibull(nmain, shape, scale=lambdaC)
dur<-pmin(T,C)
event <- (dur==T)*1

## begin stan code
code="data{
int <lower=1> nmain;
```

```
int <lower=0,upper=1> event[nmain];
vector<lower=0.00,upper=1.00>[nmain]x1;
vector<lower=0.00,upper=1.00>[nmain]z1;
vector[nmain]z2;
vector [nmain] dur;
}
// parameters
parameters{
vector[3] beta;
real <lower=0> b;
real <lower=0> sigu;
}

transformed parameters{
vector [nmain] b2;
b2<-beta[1]*x1+beta[2]*z1+beta[3]*z2;
}
// models

model{
// prior information
beta~normal(0,10);
b~gamma(1,0.5);

for(j in 1:nmain){
increment_log_prob(event[j]*(log(b)-b*log(exp(-b2[j]))+(b-1)*
log(dur[j]))-pow((dur[j]/exp(-b2[j])),b));
```

```
}
}
"
data=list(nmain=nmain,x1=x1,z1=z1,z2=z2, dur=dur, event=event)
fit=stan(model_code=code, data=data, iter=45000,chains=1, thin=10,
warmup=10000, pars=c("beta","b"))
ss<-extract(fit)
## output the results

ave.b1[t]<-mean(ss$beta[,1])
cov.b1[t]<-(b1>=quantile(ss$beta[,1],probs=c(0.025),na.rm=TRUE))&
(b1<=quantile(ss$beta[,1],probs=c(0.975),na.rm=TRUE))
len.b1[t]<-quantile(ss$beta[,1],probs=c(0.975),na.rm=TRUE)-
quantile(ss$beta[,1],probs=c(0.025),na.rm=TRUE)
lower.b1[t]<-quantile(ss$beta[,1],probs=c(0.025),na.rm=TRUE)
upper.b1[t]<-quantile(ss$beta[,1],probs=c(0.975),na.rm=TRUE)

ave.b2[t]<-mean(ss$beta[,2])
cov.b2[t]<-(b2>=quantile(ss$beta[,2],probs=c(0.025),na.rm=TRUE))&
(b2<=quantile(ss$beta[,2],probs=c(0.975),na.rm=TRUE))
len.b2[t]<-quantile(ss$beta[,2],probs=c(0.975),na.rm=TRUE)-
quantile(ss$beta[,2],probs=c(0.025),na.rm=TRUE)

ave.b3[t]<-mean(ss$beta[,3])
cov.b3[t]<-(b3>=quantile(ss$beta[,3],probs=c(0.025),na.rm=TRUE))&
(b3<=quantile(ss$beta[,3],probs=c(0.975),na.rm=TRUE))
len.b3[t]<-quantile(ss$beta[,3],probs=c(0.975),na.rm=TRUE)-
```

```
quantile(ss$beta[,3],probs=c(0.025),na.rm=TRUE)
## shape
ave.shp[t]<-mean(ss$b)
cov.shp[t]<-(shape>=quantile(ss$b, probs=c(0.025),na.rm=TRUE))&
(shape<=quantile(ss$b, probs=c(0.975),na.rm=TRUE))
len.shp[t]<-(quantile(ss$b, probs=c(0.975),na.rm=TRUE))-
(quantile(ss$b, probs=c(0.025),na.rm=TRUE))
}#loop
## beta
ave_b1<-mean(ave.b1)
cov_b1<-mean(cov.b1)
len_b1<-mean(len.b1)
lower_b1<-mean(lower.b1)
upper_b1<-mean(upper.b1)

ave_b2<-mean(ave.b2)
cov_b2<-mean(cov.b2)
len_b2<-mean(len.b2)

ave_b3<-mean(ave.b3)
cov_b3<-mean(cov.b3)
len_b3<-mean(len.b3)
## shape
ave_shp<-mean(ave.shp)
cov_shp<-mean(cov.shp)
len_shp<-mean(len.shp)
# pass the return value of function
```

```
return(c(ave_b1,cov_b1,len_b1,lower_b1, upper_b1,ave_b2,
cov_b2,len_b2,ave_b3,cov_b3,len_b3, ave_shp,cov_shp,len_shp))
}
ptm<-proc.time()
## The validation size can be 50, 100 or 200
## 40 iterations
## Simulation seed is at 100
## The size for main study is 1000
naivstan(1000, 100, 40, -0.3, 0.2, 0.3, -0.8, 0.1, -0.4, 1 ,3, 100)
proc.time()-ptm
```

APPENDIX B

R and Jags Programs for Semi-Parametric Models

These codes were presented to model the time-to-event data with an normally or binary distributed unmeasured confounder. We also performed the external and internal validation for the Cox proportional hazards models.

B.1 External Validation with Normal Unmeasured Confounders

```
#####################################################################
## External validation data with continuous unmeasured confounder
## m=100 validation size
## n=1000 main study
## Censoring rates 30%, 60%, and 90%
## #####################################################################

## Load the JAGS package
library(rjags)

excs<-function(m, csr, n, n.tilde, J, b1, b2,b3,b4,
lambda,eta1,eta2,sigu,seed){

set.seed(seed)
ave.beta1 = rep(NA, m)
coverage.beta1 = rep(NA,m)
length.beta1 = rep(NA,m)
```

```
## Keep the 95% of credible interval
b1lb<-rep(NA, m) # Low bound of beta1
b1ub<-rep(NA, m) # Upper bound of beta1

ave.beta2 = rep(NA, m)
coverage.beta2 = rep(NA,m)
length.beta2 = rep(NA,m)

ave.beta3 = rep(NA, m)
coverage.beta3 = rep(NA,m)
length.beta3 = rep(NA,m)

ave.beta4 = rep(NA, m)
coverage.beta4 = rep(NA,m)
length.beta4 = rep(NA,m)
## Bias parameter
ave.lambda = rep(NA, m)
coverage.lambda = rep(NA,m)
length.lambda = rep(NA,m)

ave.eta1 = rep(NA, m)
coverage.eta1 = rep(NA,m)
length.eta1 = rep(NA,m)
ave.eta2 = rep(NA, m)
coverage.eta2 = rep(NA,m)
length.eta2 = rep(NA,m)
## censoring rate
```

```
cens.rate<-rep(NA, m)
## Begin the simulation loop
for (i in 1:m){
## All binary data except the unmeasured confounder
x1<-rbinom(n, 1, 0.4)
z1<-rbinom(n, 1, 0.08)
z2<-rbinom(n, 1, 0.24)
z3<-rbinom(n, 1, 0.11)
## Assuming the unmeasured confounder is normal distributed
u<-rnorm(n, (cbind(1,x1))%*%c(eta1,eta2), sigu)
# Generate the survival time
uni<-runif(n, 0, 1)
## Data sets are similar to RHC example (Connors, 1996)
## Suppose the survival time follow Weibull distribution
time<-1/(0.13)*log(1-(0.13*log(uni))/(0.0035*exp(b1*x1+
b2*z1+b3*z2+b4*z3+lambda*u)))

## Different censor rates
limit<-0
repeat{
cens.t<- runif(n,0, limit) #censoring time
dur<- pmin(time,cens.t)
event <- (dur==time)*1 # set to 1 if it is event
limit<-limit+0.01
if (1-sum(event)/n<=csr){
break
}
```

```
}
## Create external validation summary statistics
x1.tilde<-rbinom(n.tilde, 1, 0.4)
z1.tilde<-rbinom(n.tilde, 1, 0.08)
z2.tilde<-rbinom(n.tilde, 1, 0.24)
z3.tilde<-rbinom(n.tilde, 1, 0.11)
u.tilde<-rnorm(n.tilde, (cbind(1,x1.tilde))%*%c(eta1,eta2), sigu)

## Produce the continuous summary statistics
sdx0<-by(u.tilde, x1.tilde, sd)[1]
sdx1<-by(u.tilde, x1.tilde, sd)[2]
ux0<-by(u.tilde, x1.tilde, mean)[1]
ux1<-by(u.tilde, x1.tilde, mean)[2]
nx0<-n.tilde-sum(x1.tilde)
nx1<-sum(x1.tilde)

ubar<-(nx0*ux0+nx1*ux0)/(nx0+nx1)
xbar<-nx1/n.tilde
ssx<-nx1-nx1^2/n.tilde
sigmau<-1/(((nx0-1)*sdx0^2+(nx1-1)*sdx1^2)/(n.tilde))
gam1<-ux0
gam2<-ux1-ux0
tau1<-(nx0+nx1)/2-1
tau2<-((nx0-1)*sdx0^2+(nx1-1)*sdx1^2)/2
## Data driven intervals
quan<-quantile(dur,seq(0,1,length=J+1))
a<-rep(0,J+1)
```

```
for(t in 2:J){
a[t]<-quan[t]
}
a[J+1]<-100*quan[J+1]

## Observed rersults on the grid
d<-array(0, dim=c(n,J))
for(h in 1:n){
for (k in 1:J){
d[h,k]<-event[h]*((dur[h]-a[k])>=0)*((a[k+1]-dur[h])>0)
}
}

data<-list(n=n,J=J,d=d, dur=dur,a=a,x1=x1, z1=z1,z2=z2,
z3=z3, xbar=xbar, ubar=ubar, ssx=ssx, sigmau=sigmau,
gam2=gam2, gam1=gam1, tau1=tau1, tau2=tau2, nx0=nx0, nx1=nx1)
parameters<-c("eta1","eta2","beta", "lambda")
inits<-list(beta=c(rnorm(4,0,1)),eta1=rnorm(1,0,1),eta2=rnorm(1,0,1),
lam=c(rgamma(J,1,2)), lambda=.2,u=rnorm(n,0,1), sig=rgamma(1,1,3))
inits<-list(inits)
jag.sim<-jags.model(file="~/diss/graph/final/excs_in.txt",
data=data, inits=inits,n.chains =1, n.adapt = 200)

burn.in<-7000
samps <- coda.samples(jag.sim, parameters,thin=10, n.iter = 20000)
ss.sim<-summary(window(samps, start = burn.in))
```

```
ave.beta1[i] = ss.sim$statistics[1, 1]
coverage.beta1[i]<-(b1 >= ss.sim$quantiles[1,1])&
(b1 <= ss.sim$quantiles[1,5])
length.beta1[i] = ss.sim$quantiles[1,5]-ss.sim$quantiles[1,1]
b1ub[i]<-ss.sim$quantiles[1,5]
b1lb[i]<-ss.sim$quantiles[1,1]

ave.beta2[i] = ss.sim$statistics[2, 1]
coverage.beta2[i]<-(b2 >= ss.sim$quantiles[2,1])&
(b2 <= ss.sim$quantiles[2,5])
length.beta2[i] = ss.sim$quantiles[2,5]-ss.sim$quantiles[2,1]

ave.beta3[i] = ss.sim$statistics[3, 1]
coverage.beta3[i]<-(b3 >= ss.sim$quantiles[3,1])&
(b3 <= ss.sim$quantiles[3,5])
length.beta3[i] = ss.sim$quantiles[3,5]-ss.sim$quantiles[3,1]

ave.beta4[i] = ss.sim$statistics[4, 1]
coverage.beta4[i]<-(b4 >= ss.sim$quantiles[4,1])&
(b4 <= ss.sim$quantiles[4,5])
length.beta4[i] = ss.sim$quantiles[4,5]-ss.sim$quantiles[4,1]

ave.eta1[i] = ss.sim$statistics[5, 1]
coverage.eta1[i]<-(eta1 >= ss.sim$quantiles[5,1])&
(eta1 <= ss.sim$quantiles[5,5])
length.eta1[i] = ss.sim$quantiles[5,5]-ss.sim$quantiles[5,1]
```

```
ave.eta2[i] = ss.sim$statistics[6, 1]
coverage.eta2[i]<-(eta2 >= ss.sim$quantiles[6,1])&
(eta2 <= ss.sim$quantiles[6,5])
length.eta2[i] <-ss.sim$quantiles[6,5]-ss.sim$quantiles[6,1]

ave.lambda[i]<-ss.sim$statistics[7, 1]
coverage.lambda[i]<-(lambda >= ss.sim$quantiles[7,1])&
(lambda <= ss.sim$quantiles[7,5])
length.lambda[i]<-ss.sim$quantiles[7,5]-ss.sim$quantiles[7,1]
## Censoring rates
cens.rate[i]<-round(1-sum(event)/n, 2)

}

return(cbind(ave.beta1,coverage.beta1,length.beta1,ave.beta2,
coverage.beta2,length.beta2,ave.beta3,coverage.beta3,length.beta3,
ave.beta4,coverage.beta4,length.beta4, ave.eta1,coverage.eta1,
length.eta1,ave.eta2,coverage.eta2,length.eta2,ave.lambda,
coverage.lambda,length.lambda, b1lb, b1ub, cens.rate))
}
## The end of function

csr<-c(0.31, 0.61, 0.91)
res<-array(0, dim=c(length(csr), 24))
ptm <- proc.time()
for (m in 1:length(csr)){
## Call the excs function
```

```
res[m,]<-colMeans(excs(40,csr[m],1000,100,5, 0.3,-1,0.4,1,-3,
0.6,-0.04, 0.2, 104))
}
proc.time() - ptm
#################################################################
## External validation with normally distributed
## unmeasured confounder
## This is jags code

model{
for(i in 1:n){
for (k in 1:J){
d[i,k]~dpois(delta[i,k])
delta[i,k]<-((min(dur[i],a[k+1])-a[k])*step(dur[i]-a[k]))*lam[k]*
exp(beta[1]*x1[i]+beta[2]*z1[i]+beta[3]*z2[i]+beta[4]*z3[i]
+lambda*u[i])
}
u[i]~dnorm(eta1+eta2*x1[i], sig)
}
ubar~dnorm(eta1+eta2*xbar, sig)
# prior information
for(k in 1:J){lam[k]~dgamma(0.01,0.01)}
for(l in 1:4){beta[l]~dnorm(0,0.1)}
sig~dgamma(tau1, tau2)
eta1~dnorm(gam1, 1/(sigmau*ssx))
eta2~dnorm(gam2, 1/(1/sigmau*(1/(nx0+nx1)+xbar^2/ssx)))
lambda~dnorm(0,0.1)
```

```
}
```

B.2 Internal Validation with Normal Unmeasured Confounders

```
#####################################################################
## Internal validation data with continuous unmeasured confounder
## n=1000
## m=100
#####################################################################
library(rjags)

incs<-function(m, cs, n, n.tilde, J, b1, b2,b3,b4, lambda,eta1,eta2,
sigu,seed){

set.seed(seed)

# vectors to store results
ave.beta1 = rep(NA, m)
coverage.beta1 = rep(NA,m)
length.beta1 = rep(NA,m)

## Keep the 95% credible interval of beta1
b1lb<-rep(NA, m) # Low bound of beta1
b1ub<-rep(NA, m) # Upper bound of beta1

ave.beta2 = rep(NA, m)
coverage.beta2 = rep(NA,m)
length.beta2 = rep(NA,m)
```

```
ave.beta3 = rep(NA, m)
coverage.beta3 = rep(NA,m)
length.beta3 = rep(NA,m)

ave.beta4 = rep(NA, m)
coverage.beta4 = rep(NA,m)
length.beta4 = rep(NA,m)

ave.lambda = rep(NA, m)
coverage.lambda = rep(NA,m)
length.lambda = rep(NA,m)



ave.eta1 = rep(NA, m)
coverage.eta1 = rep(NA,m)
length.eta1 = rep(NA,m)

ave.eta2 = rep(NA, m)
coverage.eta2 = rep(NA,m)
length.eta2 = rep(NA,m)
## Different censoring rates
cens.rate<-rep(NA, m)

for (i in 1:m){

## Data generation is similar RHC example (Connors, 1996)
x1<-rbinom(n, 1, 0.4)
```

```
z1<-rbinom(n, 1, 0.08)
z2<-rbinom(n, 1, 0.24)
z3<-rbinom(n, 1, 0.11)
u<-rnorm(n, (cbind(1,x1))%*%c(eta1,eta2), sigu)
# Generate the survival time
uni<-runif(n, 0, 1)

## Suppose the survival time follow Weibull distribution
time<-1/(0.13)*log(1-(0.13*log(uni))/(0.0035*
exp(b1*x1+b2*z1+b3*z2+b4*z3+lambda*u)))

## Different censor rates
limit<-0
repeat{
cens.t<- runif(n,0, limit) #censoring time
dur<- pmin(time,cens.t)  #observed time
event <- (dur==time)*1 # set to 1 if it is event
limit<-limit+0.01
if (1-sum(event)/n<=cs){
break
}
}

## A small fraction of internval validation data
x1.tilde<-rbinom(n.tilde, 1, 0.4)
z1.tilde<-rbinom(n.tilde, 1, 0.08)
z2.tilde<-rbinom(n.tilde, 1, 0.24)
```

```
z3.tilde<-rbinom(n.tilde, 1, 0.11)
## Normal unmeasured confounder
u.tilde<-rnorm(n.tilde, (cbind(1,x1.tilde))%*%c(eta1,eta2), sigu)

uni.tilde<-runif(n.tilde, 0, 1)
## Suppose the survival time follow Weibull distribution
time.tilde<-1/(0.13)*log(1-(0.13*log(uni.tilde))/(0.0035*
exp(b1*x1.tilde+b2*z1.tilde+b3*z2.tilde+b4*z3.tilde+lambda*u.tilde)))

limit<-0
repeat{
cens.tilde<- runif(n.tilde,0, limit)
dur.tilde<- pmin(time.tilde,cens.tilde)
event.tilde <- (dur.tilde==time.tilde)*1
limit<-limit+0.01
if (1-sum(event.tilde)/n.tilde<=cs){
break
}
}

## Data driven baseline hazard rates
quan<-quantile(dur,seq(0,1,length=J+1))
a<-rep(0,J+1)
for(t in 2:J){
a[t]<-quan[t]
}
a[J+1]<-100*quan[J+1]
```

```
## Observed results on the grid
d<-array(0, dim=c(n,J))
for(h in 1:n){
for (k in 1:J){
d[h,k]<-event[h]*((dur[h]-a[k])>=0)*((a[k+1]-dur[h])>0)
}
}

d.tilde<-array(0, dim=c(n.tilde,J))
for(h in 1:n.tilde){
for (k in 1:J){
d.tilde[h,k]<-event.tilde[h]*((dur.tilde[h]-a[k])>=0)*
((a[k+1]-dur.tilde[h])>0)
}
}

data<-list(n=n,J=J,d=d, dur=dur,a=a,x1=x1, z1=z1,z2=z2,z3=z3,
n.tilde=n.tilde, x1.tilde=x1.tilde,z1.tilde=z1.tilde,
z2.tilde=z2.tilde,z3.tilde=z3.tilde, dur.tilde=dur.tilde,
 d.tilde=d.tilde, u.tilde=u.tilde)

parameters<-c("beta","eta", "lambda")
inits3<-list(beta=c(rnorm(4,0,1)),eta=c(rnorm(2,0,1)),
lam=c(rgamma(J,1,2)), lambda=.2,u=rnorm(n,0,1), sigu=rgamma(1,1,3))

inits<-list( inits3)
```

```
jag.sim<-jags.model(file="~/diss/graph/normal/incs_in.txt",
data=data, inits=inits,n.chains =1, n.adapt = 200)
samps <- coda.samples(jag.sim, parameters,thin=10, n.iter = 20000)
burn.in<-7000
ss.sim<-summary(window(samps, start = burn.in))

ave.beta1[i] = ss.sim$statistics[1, 1]
coverage.beta1[i]<-(b1 >= ss.sim$quantiles[1,1])&
(b1 <= ss.sim$quantiles[1,5])
length.beta1[i] = ss.sim$quantiles[1,5]-ss.sim$quantiles[1,1]
b1ub[i]<-ss.sim$quantiles[1,5]
b1lb[i]<-ss.sim$quantiles[1,1]

ave.beta2[i] = ss.sim$statistics[2, 1]
coverage.beta2[i]<-(b2 >= ss.sim$quantiles[2,1])&
(b2 <= ss.sim$quantiles[2,5])
length.beta2[i] = ss.sim$quantiles[2,5]-ss.sim$quantiles[2,1]

ave.beta3[i] = ss.sim$statistics[3, 1]
coverage.beta3[i]<-(b3 >= ss.sim$quantiles[3,1])&
(b3 <= ss.sim$quantiles[3,5])
length.beta3[i] = ss.sim$quantiles[3,5]-ss.sim$quantiles[3,1]

ave.beta4[i] = ss.sim$statistics[4, 1]
coverage.beta4[i]<-(b4 >= ss.sim$quantiles[4,1])&
(b4 <= ss.sim$quantiles[4,5])
length.beta4[i] = ss.sim$quantiles[4,5]-ss.sim$quantiles[4,1]
```

```
ave.eta1[i] = ss.sim$statistics[5, 1]
coverage.eta1[i]<-(eta1 >= ss.sim$quantiles[5,1])&
(eta1 <= ss.sim$quantiles[5,5])
length.eta1[i] = ss.sim$quantiles[5,5]-ss.sim$quantiles[5,1]

ave.eta2[i] = ss.sim$statistics[6, 1]
coverage.eta2[i]<-(eta2 >= ss.sim$quantiles[6,1])&
(eta2 <= ss.sim$quantiles[6,5])
length.eta2[i] <-ss.sim$quantiles[6,5]-ss.sim$quantiles[6,1]

ave.lambda[i]<-ss.sim$statistics[7, 1]
coverage.lambda[i]<-(lambda >= ss.sim$quantiles[7,1])&
(lambda <= ss.sim$quantiles[7,5])
length.lambda[i]<-ss.sim$quantiles[7,5]-ss.sim$quantiles[7,1]
## Censored rate
cens.rate[i]<-round(1-sum(event)/n, 2)

}

return(cbind(ave.beta1,coverage.beta1,length.beta1,
ave.beta2,coverage.beta2,length.beta2,
ave.beta3,coverage.beta3,length.beta3,
ave.beta4,coverage.beta4,length.beta4,
ave.eta1,coverage.eta1,length.eta1,
ave.eta2,coverage.eta2,length.eta2,
ave.lambda,coverage.lambda,length.lambda,
```

```
b1lb, b1ub, cens.rate))
}
## The end of function
csr<-c(0.31, 0.61, 0.91)
res<-array(0, dim=c(length(csr), 24))
ptm <- proc.time()
for (m in 1:length(csr)){
res[m,]<-colMeans(incs(40,csr[m],1000,100,5,0.3,-1.2,0.4,1,-3,
0.6,-0.04, 0.2, 104))
}
proc.time() - ptm
#####################################################################
## Internal validation with normally distributed
## unmeasured confounder
## This is jags model

## Model the main study data
model
{
for(i in 1:n){
for (k in 1:J){
d[i,k]~dpois(mu[i,k])
mu[i,k]<-(min(dur[i],a[k+1])-a[k])*step(dur[i]-a[k])*
lam[k]*exp(beta[1]*x1[i]+beta[2]*z1[i]+beta[3]*z2[i]+beta[4]*z3[i]
+lambda*u[i])
}
u[i]~dnorm(eta[1]+eta[2]*x1[i], sigu)
```

```
}

## Modeling validation data

for(j in 1:n.tilde){
for (m in 1:J){
d.tilde[j,m]~dpois(mu.tilde[j,m])
mu.tilde[j,m]<-(min(dur.tilde[j],a[m+1])-a[m])*step(dur.tilde[j]-a[m])*
lam[m]*exp(beta[1]*x1.tilde[j]+beta[2]*z1.tilde[j]+beta[3]*z2.tilde[j]+
beta[4]*z3.tilde[j]+lambda*u.tilde[j])
}
u.tilde[j]~dnorm(eta[1]+eta[2]*x1.tilde[j], sigu)
}

# prior information
for(k in 1:J){lam[k]~dgamma(0.01,0.01)}
for(l in 1:4){beta[l]~dnorm(0,0.1)}
for(l in 1:2){eta[l]~dnorm(0,0.1)}
sigu~dgamma(0.01, 0.01)
lambda~dnorm(0,0.1)
}
```

B.3 Naive Model with Normal Unmeasured Confounders

```
## Naive model, ignoring unmeasured confounder
## n=1000; m=100;
## Censoring rates: 30%, 60% and 90%
#####################################################################
```

```
## Load the JAGS package
library(rjags)
naiv<-function(m, cs, n, J, b1, b2, b3, b4, lambda,eta1,eta2,
sigu, seed){
set.seed(seed)
# vectors to store results
ave.beta1 <- rep(NA, m)
coverage.beta1 <- rep(NA,m)
length.beta1 <- rep(NA,m)

## Keep the 95% of credible interval
b1lb<-rep(NA, m) # Low bound of beta1
b1ub<-rep(NA, m) # Upper bound of beta1

ave.beta2 <- rep(NA, m)
coverage.beta2 <- rep(NA,m)
length.beta2 <- rep(NA,m)

ave.beta3 <- rep(NA, m)
coverage.beta3 <- rep(NA,m)
length.beta3 <- rep(NA,m)

ave.beta4 <- rep(NA, m)
coverage.beta4 <- rep(NA,m)
length.beta4 <- rep(NA,m)
## Censor Rate
cens.rate<-rep(NA, m)
```

```
for(i in 1:m){
x1<-rbinom(n, 1, 0.4)
z1<-rbinom(n, 1, 0.08)
z2<-rbinom(n, 1, 0.24)
z3<-rbinom(n, 1, 0.11)
u<-rnorm(n, (cbind(1,x1))%*%c(eta1,eta2), sigu)
# Generate the survival time
uni<-runif(n, 0, 1)
time<-1/(0.13)*log(1-(0.13*log(uni))/(0.0035*
exp(b1*x1+b2*z1+b3*z2+b4*z3+lambda*u)))

## show different censor rate
limit<-0
repeat{
cens.t<- runif(n,0, limit)
dur<- pmin(time,cens.t)
event <- (dur==time)*1
limit<-limit+0.01
if (1-sum(event)/n<=cs){
break
}
}
## creating the boundaries of
quan<-quantile(dur,seq(0,1,length=J+1))
a<-rep(0,J+1)
for(t in 2:J){
```

```
a[t]<-quan[t]
}
a[J+1]<-100*quan[J+1] ## The last interval should be infinite

## get the observed results
d<-array(0, dim=c(n,J))
for(h in 1:n){
for (k in 1:J){
d[h,k]<-event[h]*((dur[h]-a[k])>=0)*((a[k+1]-dur[h])>0)
}
}
data<-list(n=n,J=J,d=d, dur=dur,a=a,x1=x1, z1=z1,z2=z2, z3=z3)

parameters<-c("beta")
inits<-list(beta=c(rnorm(4,0,1)),lam=c(rgamma(J,1,2)))

inits<-list(inits)
jag.sim<-jags.model(file="~/diss/graph/normal/ncs_in.txt",
data=data, inits=inits,n.chains =1, n.adapt = 200)
burn.in<-7000
samps<-coda.samples(jag.sim, parameters,thin=10, n.iter = 25000)
ss.sim<-summary(window(samps, start = burn.in))
ave.beta1[i] = ss.sim$statistics[1, 1]
coverage.beta1[i]<-(b1 >= ss.sim$quantiles[1,1])&
(b1 <= ss.sim$quantiles[1,5])
length.beta1[i] = ss.sim$quantiles[1,5]-ss.sim$quantiles[1,1]
```

```
b1ub[i]<-ss.sim$quantiles[1,5]
b1lb[i]<-ss.sim$quantiles[1,1]

ave.beta2[i] = ss.sim$statistics[2, 1]
coverage.beta2[i]<-(b2 >= ss.sim$quantiles[2,1])&
(b2 <= ss.sim$quantiles[2,5])
length.beta2[i] = ss.sim$quantiles[2,5]-ss.sim$quantiles[2,1]

ave.beta3[i] = ss.sim$statistics[3, 1]
coverage.beta3[i]<-(b3 >= ss.sim$quantiles[3,1])&
(b3 <= ss.sim$quantiles[3,5])
length.beta3[i] = ss.sim$quantiles[3,5]-ss.sim$quantiles[3,1]

ave.beta4[i] = ss.sim$statistics[4, 1]
coverage.beta4[i]<-(b4 >= ss.sim$quantiles[4,1])&
(b4 <= ss.sim$quantiles[4,5])
length.beta4[i] = ss.sim$quantiles[4,5]-ss.sim$quantiles[4,1]

cens.rate[i]<-1-sum(event)/n
}
return(cbind(ave.beta1,coverage.beta1,length.beta1,
ave.beta2,coverage.beta2,length.beta2,
ave.beta3,coverage.beta3,length.beta3,
ave.beta4,coverage.beta4,length.beta4,
b1ub, b1lb,cens.rate))
}
## End of self-defined function
```

```
csr<-c(0.31, 0.61, 0.91)
res<-array(0, dim=c(length(csr), 15))
ptm <- proc.time()
for (i in 1:length(csr)){
res[i,]<-colMeans(naiv(40,csr[i],1000,5, 0.3,-1,0.4,1,-3,
0.6,-0.04, 0.2, 104))
}
proc.time() - ptm

## Naive model, ignoring unmeasured confounder
## This is jags code
model
{
for(i in 1:n){
for (k in 1:J){
d[i,k]~dpois(mu[i,k])
mu[i,k]<-(min(dur[i],a[k+1])-a[k])*step(dur[i]-a[k])*
lam[k]*exp(beta[1]*x1[i]+beta[2]*z1[i]+beta[3]*z2[i]+beta[4]*z3[i])
}

}
# prior information
for(k in 1:J){lam[k]~dgamma(0.01,0.01)}
for(l in 1:4){beta[l]~dnorm(0,0.1)}
}
```

B.4 External Validation with Binary Unmeasured Confounders

```
#####################################################################
```

```
# ## External validation with binary unmeasured confounder
# 1) 3 different censored rates (30%, 60%, 90%)
# 2) Unmeasured confounder is binary data
# 3) Including continuous covariate
# 4) Only has external contingency table (u vs x1)
#####################################################################

library(rjags)

exds<-function(m, n, n.tilde, J, b1, b2,b3, b4, lambda,a1,a2, csr,
seed){

set.seed(seed)
ave.beta1 = rep(NA, m)
coverage.beta1 = rep(NA,m)
length.beta1 = rep(NA,m)

## Keep the 95% credible interval of beta1
b1lb<-rep(NA, m) # Low bound of beta1
b1ub<-rep(NA, m) # Upper bound of beta1

ave.beta2 = rep(NA, m)
coverage.beta2 = rep(NA,m)
length.beta2 = rep(NA,m)

ave.beta3 = rep(NA, m)
coverage.beta3 = rep(NA,m)
```

```
length.beta3 = rep(NA,m)


ave.beta4 = rep(NA, m)
coverage.beta4 = rep(NA,m)
length.beta4 = rep(NA,m)
ave.lambda = rep(NA, m)
coverage.lambda = rep(NA,m)
length.lambda = rep(NA,m)
ave.eta1 = rep(NA, m)
coverage.eta1 = rep(NA,m)
length.eta1 = rep(NA,m)

ave.eta2 = rep(NA, m)
coverage.eta2 = rep(NA,m)
length.eta2 = rep(NA,m)

cens.rate<-rep(NA,m)
for(i in 1:m){
x1<-rbinom(n, 1, 0.4)
z1<-rbinom(n, 1, 0.08)
z2<-rbinom(n, 1, 0.24)
z3<-rnorm(n, mean=61, sd=4)
## The data is similar to Connors(1996)
u<-rbinom(n,1,1/(1+exp(-(cbind(1,x1))%*%c(a1,a2))))
## Generate the survival time
uni<-runif(n, 0, 1)
```

```
## Suppose the survival time follow Weibull distribution
time<-1/(0.13)*log(1-(0.13*log(uni))/(0.0035*
exp(b1*x1+b2*z1+b3*z2+b4*z3+lambda*u)))

## Defined the  different censor rate
limit<-0
repeat{
cens.t<- runif(n,0, limit)
dur<- pmin(time,cens.t)
event <- (dur==time)*1
limit<-limit+0.01
if (1-sum(event)/n<=csr){
break
}
}
x1.tilde<-rbinom(n.tilde, 1, 0.4)
z1.tilde<-rbinom(n.tilde, 1, 0.08)
z2.tilde<-rbinom(n.tilde, 1, 0.24)
#z3<-rbinom(n, 1, 0.11)
z3.tilde<-rnorm(n.tilde, mean=61, sd=4)
## connor(1996)   dnr  -1.81  and -0.76
u.tilde<-rbinom(n.tilde,1,1/(1+exp(-(cbind(1,x1.tilde))%*%c(a1,a2))))


x0<-c( table(u.tilde, x1.tilde)[1,1]+table(u.tilde, x1.tilde)[2,1],
table(u.tilde, x1.tilde)[1,2]+table(u.tilde, x1.tilde)[2,2])
```

```
w10<-c(table(u.tilde, x1.tilde)[2,1], table(u.tilde, x1.tilde)[2,2])
x10<-c(0,1)

quan<-quantile(dur,seq(0,1,length=J+1))
a<-rep(0,J+1)
for(t in 2:J){
a[t]<-quan[t]
}
a[J+1]<-100*quan[J+1] ## The last interval should be infinite

d<-array(0, dim=c(n,J))
for(h in 1:n){
for (k in 1:J){
d[h,k]<-event[h]*((dur[h]-a[k])>=0)*((a[k+1]-dur[h])>0)
}
}

data<-list(n=n,J=J,d=d, dur=dur, a=a, z1=z1,z2=z2,x1=x1, z3=z3,
x0=x0, w10=w10, x10=x10)
parameters<-c("beta", "eta", "lambda")
inits1<-list(beta=c(rnorm(4,0,1)),lam=c(rgamma(J,1,2)),
eta=c(rnorm(2,0,1)), lambda=rnorm(1))
inits<-list(inits1)
ss.sim<-jags.model(file="~/diss/graph/final/exds_in.txt",
data=data, inits=inits, n.chains =1, n.adapt = 200)

burn.in<-7000
```

```
samps <- coda.samples(ss.sim, parameters, thin=10, n.iter = 20000)
ss.sim<-summary(window(samps,start = burn.in))


ave.beta1[i] = ss.sim$statistics[1, 1]
coverage.beta1[i]<-(b1 >= ss.sim$quantiles[1,1])&
(b1 <= ss.sim$quantiles[1,5])
length.beta1[i] = ss.sim$quantiles[1,5]-ss.sim$quantiles[1,1]
b1ub[i]<-ss.sim$quantiles[1,5]
b1lb[i]<-ss.sim$quantiles[1,1]

ave.beta2[i] = ss.sim$statistics[2, 1]
coverage.beta2[i]<-(b2 >= ss.sim$quantiles[2,1])&
(b2 <= ss.sim$quantiles[2,5])
length.beta2[i] = ss.sim$quantiles[2,5]-ss.sim$quantiles[2,1]

ave.beta3[i] = ss.sim$statistics[3, 1]
coverage.beta3[i]<-(b3 >= ss.sim$quantiles[3,1])&
(b3 <= ss.sim$quantiles[3,5])
length.beta3[i] = ss.sim$quantiles[3,5]-ss.sim$quantiles[3,1]

ave.beta4[i] = ss.sim$statistics[4, 1]
coverage.beta4[i]<-(b4 >= ss.sim$quantiles[4,1])&
(b4 <= ss.sim$quantiles[4,5])
length.beta4[i] = ss.sim$quantiles[4,5]-ss.sim$quantiles[4,1]

ave.eta1[i] = ss.sim$statistics[5, 1]
```

```
coverage.eta1[i]<-(a1 >= ss.sim$quantiles[5,1])&
(a1 <= ss.sim$quantiles[5,5])
length.eta1[i] = ss.sim$quantiles[5,5]-ss.sim$quantiles[5,1]

ave.eta2[i] = ss.sim$statistics[6, 1]
coverage.eta2[i]<-(a2 >= ss.sim$quantiles[6,1])&
(a2 <= ss.sim$quantiles[6,5])
length.eta2[i] <-ss.sim$quantiles[6,5]-ss.sim$quantiles[6,1]

ave.lambda[i]<-ss.sim$statistics[7, 1]
coverage.lambda[i]<-(lambda >= ss.sim$quantiles[7,1])&
(lambda <= ss.sim$quantiles[7,5])
length.lambda[i]<-ss.sim$quantiles[7,5]-ss.sim$quantiles[7,1]
## Censored rate
cens.rate[i]<-round(1-sum(event)/n, 2)

}
return(cbind(ave.beta1,coverage.beta1,length.beta1,
ave.beta2,coverage.beta2,length.beta2,
ave.beta3,coverage.beta3,length.beta3,
ave.beta4,coverage.beta4,length.beta4,
ave.eta1, coverage.eta1, length.eta1,
ave.eta2, coverage.eta2, length.eta2,
ave.lambda, coverage.lambda, length.lambda,
b1lb, b1ub, cens.rate))
}
cs<-c(0.31, 0.61, 0.91)
```

```
res<-array(0, dim=c(length(cs),24))
ptm <- proc.time()
for (i in 1:length(cs)){
## Call the self-defined function
res[i,]<-colMeans(exds(40,1000,100,5,-0.3,-1,0.3,0.005, 1.4 ,
-1.3, -1.6, cs[i],104))

}
proc.time() - ptm
## External validation with binary unmeasured confounder
## This is jags code
model{
for(i in 1:n){
for (k in 1:J){
d[i,k]~dpois(delta[i,k])
delta[i,k]<-((min(dur[i],a[k+1])-a[k])*step(dur[i]-a[k]))*lam[k]*
exp(beta[1]*x1[i]+beta[2]*z1[i]+beta[3]*z2[i]+beta[4]*z3[i]
+lambda*u[i])
}
u[i]~dbern(pu[i])
logit(pu[i])<-eta[1]+eta[2]*x1[i]
}



for(j in 1:2){
w10[j]~dbin(pu.tilde[j], x0[j])
logit(pu.tilde[j])<-eta[1]+eta[2]*x10[j]
```

```
}

# informative prior
for(k in 1:J){lam[k]~dgamma(0.01,0.01)}
for(l in 1:4){beta[l]~dnorm(0,0.1)}
for(l in 1:2){eta[l]~dnorm(0,0.1)}
lambda~dnorm(0, 0.1)
}
```

B.5 Internal Validation with Binary Unmeasured Confounders

```
#################################################################
## Internal validation with binary unmeasured confounder
## n=1000; m=100;
## Censoring rates: 30%, 60% and 90%
#################################################################
library("rjags")
## Interal and binary unmeasured confounder
inbin<-function(m, n, n.tilde, J, b1, b2,b3, b4, lambda,a1,a2,
csr, seed){
set.seed(seed)
# vectors to store results
ave.beta1 = rep(NA, m)
coverage.beta1 = rep(NA,m)
length.beta1 = rep(NA,m)

## Keep the 95% of beta1, which is treatment exposure
b1lb<-rep(NA, m) # Low bound of beta1
b1ub<-rep(NA, m) # Upper bound of beta1
```

```
ave.beta2 = rep(NA, m)
coverage.beta2 = rep(NA,m)
length.beta2 = rep(NA,m)

ave.beta3 = rep(NA, m)
coverage.beta3 = rep(NA,m)
length.beta3 = rep(NA,m)

ave.beta4 = rep(NA, m)
coverage.beta4 = rep(NA,m)
length.beta4 = rep(NA,m)

ave.lambda = rep(NA, m)
coverage.lambda = rep(NA,m)
length.lambda = rep(NA,m)


ave.eta1 = rep(NA, m)
coverage.eta1 = rep(NA,m)
length.eta1 = rep(NA,m)

ave.eta2 = rep(NA, m)
coverage.eta2 = rep(NA,m)
length.eta2 = rep(NA,m)
cens.rate=rep(NA, m)
```

```
# hazard of censoring

for(i in 1:m){

x1<-rbinom(n, 1, 0.4)
z1<-rbinom(n, 1, 0.08)
z2<-rbinom(n, 1, 0.24)

z3<-rnorm(n, mean=61, sd=4)
## Data is similar to Connors(1996)
u<-rbinom(n,1,1/(1+exp(-(cbind(1,x1))%*%c(a1,a2))))
# Generate the survival time
uni<-runif(n, 0, 1)
## Suppose the survival time follow Weibull distribution
time<-1/(0.13)*log(1-(0.13*log(uni))/(0.0035*
exp(b1*x1+b2*z1+b3*z2+b4*z3+lambda*u)))

## Defined the  different censor rate
limit<-0
repeat{
cens.t<- runif(n,0, limit)
dur<- pmin(time,cens.t)
event <- (dur==time)*1
limit<-limit+0.01
if (1-sum(event)/n<=csr){
break
}
```

```
}
x1.tilde<-rbinom(n.tilde, 1, 0.4)
z1.tilde<-rbinom(n.tilde, 1, 0.08)
z2.tilde<-rbinom(n.tilde, 1, 0.24)
z3.tilde<-rnorm(n.tilde, mean=61, sd=4)
u.tilde<-rbinom(n.tilde,1,1/(1+exp(-(cbind(1,x1.tilde))%*%c(a1,a2))))

uni.tilde<-runif(n.tilde, 0, 1)

## Suppose the survival time follow Weibull distribution
time.tilde<-1/(0.13)*log(1-(0.13*log(uni.tilde))/(0.0035*
exp(b1*x1.tilde+b2*z1.tilde+b3*z2.tilde+b4*z3.tilde+lambda*u.tilde)))

limit<-0
repeat{
cens.tilde<- runif(n.tilde,0, limit)
dur.tilde<- pmin(time.tilde,cens.tilde)
event.tilde <- (dur.tilde==time.tilde)*1
limit<-limit+0.01
if (1-sum(event.tilde)/n.tilde<=csr){
break
}
}

## creating the boundaries of
quan<-quantile(dur,seq(0,1,length=J+1))
```

```
a<-rep(0,J+1)
for(t in 2:J){
a[t]<-quan[t]
}
a[1]<-0
a[J+1]<-100*quan[J+1]
## The event in different intervals
d<-array(0, dim=c(n,J))
for(h in 1:n){
for (k in 1:J){
d[h,k]<-event[h]*((dur[h]-a[k])>=0)*((a[k+1]-dur[h])>0)
}
}

d.tilde<-array(0, dim=c(n.tilde,J))
for(h in 1:n.tilde){
for (k in 1:J){
d.tilde[h,k]<-event.tilde[h]*((dur.tilde[h]-
a[k])>=0)*((a[k+1]-dur.tilde[h])>0)
}
}
## The last interval should be infinite

parameters<-c( "beta", "eta", "lambda")
data<-list(    n=n,J=J,d=d, dur=dur, a=a, z1=z1,z2=z2,x1=x1, z3=z3,
u.tilde=u.tilde, n.tilde=n.tilde, x1.tilde=x1.tilde,
z1.tilde=z1.tilde,z2.tilde=z2.tilde, z3.tilde=z3.tilde,
```

```
dur.tilde=dur.tilde, d.tilde=d.tilde)
inits<-list(beta=c(rnorm(4,0,1)),lam=c(rgamma(J,1,2)), eta=c(rnorm(2,0,1)),
lambda=rnorm(1))
inits<-list(inits)
ss.sim<-jags.model(file="~/diss/graph/final/inds_in.txt",data=data,
inits=inits, n.chains =1, n.adapt = 200)

burn.in<-7000
samps <- coda.samples(ss.sim, parameters, n.iter = 20000)
ss.sim<-summary(window(samps, thin=10,start = burn.in))

ave.beta1[i] = ss.sim$statistics[1, 1]
coverage.beta1[i]<-(b1 >= ss.sim$quantiles[1,1])&
(b1 <= ss.sim$quantiles[1,5])
length.beta1[i] = ss.sim$quantiles[1,5]-ss.sim$quantiles[1,1]
b1ub[i]<-ss.sim$quantiles[1,5]
b1lb[i]<-ss.sim$quantiles[1,1]

ave.beta2[i] = ss.sim$statistics[2, 1]
coverage.beta2[i]<-(b2 >= ss.sim$quantiles[2,1])&
(b2 <= ss.sim$quantiles[2,5])
length.beta2[i] = ss.sim$quantiles[2,5]-ss.sim$quantiles[2,1]

ave.beta3[i] = ss.sim$statistics[3, 1]
coverage.beta3[i]<-(b3 >= ss.sim$quantiles[3,1])&
(b3 <= ss.sim$quantiles[3,5])
length.beta3[i] = ss.sim$quantiles[3,5]-ss.sim$quantiles[3,1]
```

```
ave.beta4[i] = ss.sim$statistics[4, 1]
coverage.beta4[i]<-(b4 >= ss.sim$quantiles[4,1])&
(b4 <= ss.sim$quantiles[4,5])
length.beta4[i] = ss.sim$quantiles[4,5]-ss.sim$quantiles[4,1]

ave.eta1[i] = ss.sim$statistics[5, 1]
coverage.eta1[i]<-(a1 >= ss.sim$quantiles[5,1])&
(a1 <= ss.sim$quantiles[5,5])
length.eta1[i] = ss.sim$quantiles[5,5]-ss.sim$quantiles[5,1]

ave.eta2[i] = ss.sim$statistics[6, 1]
coverage.eta2[i]<-(a2 >= ss.sim$quantiles[6,1])&
(a2 <= ss.sim$quantiles[6,5])
length.eta2[i] <-ss.sim$quantiles[6,5]-ss.sim$quantiles[6,1]

ave.lambda[i]<-ss.sim$statistics[7, 1]
coverage.lambda[i]<-(lambda >= ss.sim$quantiles[7,1])&
(lambda <= ss.sim$quantiles[7,5])
length.lambda[i]<-ss.sim$quantiles[7,5]-ss.sim$quantiles[7,1]
cens.rate[i]<-round(1-sum(event)/n, 2)

}
return(cbind(ave.beta1,coverage.beta1,length.beta1,
ave.beta2,coverage.beta2,length.beta2,
ave.beta3,coverage.beta3,length.beta3,
ave.beta4,coverage.beta4,length.beta4,
```

```
ave.eta1, coverage.eta1, length.eta1,
ave.eta2, coverage.eta2, length.eta2,
ave.lambda, coverage.lambda, length.lambda,
b1lb, b1ub, cens.rate))
}
cs<-c(0.31, 0.61, 0.91)
res<-array(0, dim=c(length(cs),24))
ptm <- proc.time()
for (i in 1:length(cs)){
res[i,]<-colMeans(inbin(40,1000,100,5,-0.3,-1,0.3,0.005, 1.4 ,
-1.3, -1.6, cs[i],104))
}
proc.time() - ptm
#################################################################
## Binary unmeasured confounder
## This is jags code
#################################################################
## Model the main study data
model
{
for(i in 1:n){
for (k in 1:J){
d[i,k]~dpois(mu[i,k])
mu[i,k]<-(min(dur[i],a[k+1])-a[k])*step(dur[i]-a[k])*
lam[k]*exp(beta[1]*x1[i]+beta[2]*z1[i]+beta[3]*z2[i]+beta[4]*z3[i]
+lambda*u[i])
}
```

```
u[i]~dbern(pu[i])
logit(pu[i])<-eta[1]+eta[2]*x1[i]
}

## Modeling validation data

for(j in 1:n.tilde){
for (m in 1:J){
d.tilde[j,m]~dpois(mu.tilde[j,m])
mu.tilde[j,m]<-(min(dur.tilde[j],a[m+1])-a[m])*
step(dur.tilde[j]-a[m])*lam[m]*
exp(beta[1]*x1.tilde[j]+beta[2]*z1.tilde[j]+beta[3]*
z2.tilde[j]+beta[4]*z3.tilde[j]+lambda*u.tilde[j])
}
u.tilde[j]~dbern(pu.tilde[j])
logit(pu.tilde[j])<-eta[1]+eta[2]*x1.tilde[j]
}
# prior information
for(k in 1:J){lam[k]~dgamma(0.01,0.01)}
for(l in 1:4){beta[l]~dnorm(0,0.1)}
for(l in 1:2){eta[l]~dnorm(0,0.1)}
lambda~dnorm(0,0.1)
}
```

B.6 Naive Model with Binary Unmeasured Confounders

```
#########################################################
## Naive model, ignoring binary unmeasured confounder
## n=1000; m=100;
```

```
## Censoring rates: 30%, 60%, 90%
#########################################################

library(rjags)



nds<-function(m,  n, J, b1, b2,b3,b4, lambda,a1,a2,cs,seed){
set.seed(seed)
## Vectors to store results
ave.beta1 = rep(NA, m)
coverage.beta1 = rep(NA,m)
length.beta1 = rep(NA,m)
## Keep the 95% of beta1, which is treatment exposure
b1lb<-rep(NA, m) # Low bound of beta1
b1ub<-rep(NA, m) # Upper bound of beta1

ave.beta2 = rep(NA, m)
coverage.beta2 = rep(NA,m)
length.beta2 = rep(NA,m)
ave.beta3 = rep(NA, m)
coverage.beta3 = rep(NA,m)
length.beta3 = rep(NA,m)

ave.beta4 = rep(NA, m)
coverage.beta4 = rep(NA,m)
length.beta4 = rep(NA,m)
cens.rate<-rep(NA, m)
```

```
for (i in 1:m){

x1<-rbinom(n, 1, 0.4)
z1<-rbinom(n, 1, 0.08)
z2<-rbinom(n, 1, 0.24)

z3<-rnorm(n, mean=61, sd=4)
## Data is similar to Connors(1996)
u<-rbinom(n,1,1/(1+exp(-(cbind(1,x1))%*%c(a1,a2))))
# Generate the survival time
uni<-runif(n, 0, 1)
## Suppose the survival time follow Weibull distribution
time<-1/(0.13)*log(1-(0.13*log(uni))/(0.0035*
exp(b1*x1+b2*z1+b3*z2+b4*z3+lambda*u)))

## Defined the  different censor rate
limit<-0
repeat{
cens.t<- runif(n,0, limit)
dur<- pmin(time,cens.t)
event <- (dur==time)*1
limit<-limit+0.01
if (1-sum(event)/n<=cs){
break
}
}
##  Data driven  intervals
```

```
quan<-quantile(dur,seq(0,1,length=J+1))
a<-rep(0,J+1)
for(t in 2:J){
a[t]<-quan[t]
}
a[J+1]<-100*quan[J+1]
## Observed data
d<-array(0, dim=c(n,J))
for(h in 1:n){
for (k in 1:J){
d[h,k]<-event[h]*((dur[h]-a[k])>=0)*((a[k+1]-dur[h])>0)
}
}
data<-list(n=n,J=J,d=d, dur=dur,a=a,x1=x1, z1=z1,z2=z2, z3=z3)
parameters<-c("beta")
inits<-list(beta=c(rnorm(4,0,1)), lam=c(rgamma(J,1,2)))
inits<-list(inits)
jag.sim<-jags.model(file="~/diss/graph/final/nds_in.txt",data=data,
inits=inits,n.chains =1, n.adapt = 200)

burn.in<-7000
samps <- coda.samples(jag.sim, parameters, n.iter = 20000)
ss.sim<-summary(window(samps, start = burn.in))

ave.beta1[i] = ss.sim$statistics[1, 1]
coverage.beta1[i]<-(b1 >= ss.sim$quantiles[1,1])&
(b1 <= ss.sim$quantiles[1,5])
```

```
length.beta1[i] = ss.sim$quantiles[1,5]-ss.sim$quantiles[1,1]
b1ub[i]<-ss.sim$quantiles[1,5]
b1lb[i]<-ss.sim$quantiles[1,1]

ave.beta2[i] = ss.sim$statistics[2, 1]
coverage.beta2[i]<-(b2 >= ss.sim$quantiles[2,1])&
(b2 <= ss.sim$quantiles[2,5])
length.beta2[i] = ss.sim$quantiles[2,5]-ss.sim$quantiles[2,1]

ave.beta3[i] = ss.sim$statistics[3, 1]
coverage.beta3[i]<-(b3 >= ss.sim$quantiles[3,1])&
(b3 <= ss.sim$quantiles[3,5])
length.beta3[i] = ss.sim$quantiles[3,5]-ss.sim$quantiles[3,1]

ave.beta4[i] = ss.sim$statistics[4, 1]
coverage.beta4[i]<-(b4 >= ss.sim$quantiles[4,1])&
(b4 <= ss.sim$quantiles[4,5])
length.beta4[i] = ss.sim$quantiles[4,5]-ss.sim$quantiles[4,1]

## Censored rate
cens.rate[i]<-round(1-sum(event)/n, 2)
}
return(cbind(ave.beta1,coverage.beta1,length.beta1,
ave.beta2,coverage.beta2,length.beta2,
ave.beta3,coverage.beta3,length.beta3,
ave.beta4,coverage.beta4,length.beta4,
b1lb, b1ub, cens.rate))
```

```
}
cs<-c(0.31, 0.61, 0.91)
res<-array(0, dim=c(length(cs),15))
ptm <- proc.time()
for (i in 1:length(cs)){
res[i,]<-colMeans(nds(40,1000,5,-0.3,-1,0.3,0.005, 1.4 ,
-1.3, -1.6, cs[i],104))
}
proc.time() - ptm
############################################################
## Naive model, ignoring binary unmeasured confounder
## This is jags code
## Model the main study data
model
{
for(i in 1:n){
for (k in 1:J){
d[i,k]~dpois(mu[i,k])
mu[i,k]<-(min(dur[i],a[k+1])-a[k])*step(dur[i]-a[k])*lam[k]*
exp(beta[1]*x1[i]+beta[2]*z1[i]+beta[3]*z2[i]+beta[4]*z3[i])
}
}
## Modeling validation data
for(k in 1:J){lam[k]~dgamma(0.01,0.01)}
for(l in 1:4){beta[l]~dnorm(0,0.1)}
}
```

BIBLIOGRAPHY

Austin, Peter C. (2012), "Generating Survival Times to Simulate Cox Proportional Hazards Models with Time-Varying Covariates," *Statistics in Medicine* , 31(29): 3946-3958.

Bender, Ralf, Thomas Augustin, and Maria Blettner (2005), "Generating Survival Times to Simulate Cox Proportional Hazards Models," *Statistics in Medicine*, 24(11): 1713-1723.

Black, N. (1996), "Why We Need Observational Studies to Evaluate the Effectiveness of Health Care," *British Medical Journal*, 312(7040): 1215-1218.

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M.A., Guo, J., Li, P., and Riddell, A. Forthcoming. "Stan: A Probabilistic Programming Language." *Journal of Statistical Software*. www.stat.columbia.edu/ gelman/research/unpublished/ stan-resubmit-JSS1293.pdf

Carroll, Kevin J. (2003), "On the Use and Utility of the Weibull Model in the Analysis of Survival Data," *Controlled Clinical Trials*, 24(6): 682-701.

Carroll, Kevin J. (2007), "Analysis of Progression-Free Survival in Oncology Trials: Some Common Statistical Issues," *Pharmaceutical Statistics*, 6(2): 99-113.

Casella, George, and Roger L. Berger (2002), *Statistical Inference*, Duxbury Thomson Learning.

Chen, Qingxia, Huiyun Wu, Lorraine B. Ware, and Tatsuki Koyama (2014), "A Bayesian Approach for the Cox Proportional Hazards Model with Covariates Subject to Detection Limit," *International Journal of Statistics in Medical Research*, 3(1): 3243.

Chen, Yi-Hau, and Hung Chen (2000), "A Unified Approach to Regression Analysis under Double-Sampling Designs," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(3): 449-460.

Christensen, Ronald, Wesley Johnson, Adam Branscum, and Timothy E. Hanson (2010), *Bayesian Ideas and Data Analysis: An Introduction for Scientists and Statisticians*, CRC Press.

Collett, David (2003), *Modelling Survival Data in Medical Research*, CRC Press.

Connors AF, Jr., Speroff T., Dawson N.V., et al. (1996), "The Effectiveness of Right Heart Catheterization in the Initial Care of Critically Ill Patients," *JAMA*, 276(11): 889-897.

Daniel Paulino, Carlos, Paulo Soares, and John Neuhaus (2003), "Binomial Regression with Misclassification," *Biometrics*, 59(3): 670-675.

Djibuti, Mamuka, Eka Mirvelashvili, Nutsa Makharashvili, and Matthew J. Magee (2014), "Household Income and Poor Treatment Outcome among Patients with Tuberculosis in Georgia: A Cohort Study," *BMC Public Health*, 14: 88.

Faries, Douglas, Xiaomei Peng, Manjiri Pawaskar, et al. (2013), "Evaluating the Impact of Unmeasured Confounding with Internal Validation Data: An Example Cost Evaluation in Type 2 Diabetes," *Value in Health*, 16(2): 259-266.

Gustafson, Paul (2015), *Bayesian Inference for Partially Identified Models: Exploring the Limits of Limited Data*, CRC Press.

Gustafson, Paul, Alan E. Gelfand, Sujit K. Sahu, et al. (2005), "On Model Expansion, Model Contraction, Identifiability and Prior Information: Two Illustrative Scenarios Involving Mismeasured Variables," *Statistical Science*, 20(2): 111-140.

Handorf, Elizabeth A., Justin E. Bekelman, Daniel F. Heitjan, and Nandita Mitra (2013), "Evaluating Costs with Unmeasured confounding: A Sensitivity Analysis for The Treatment Effect," *The Annals of Applied Statistics*, 7(4): 2062-2080.

Hernandez, Adrian V., Marinus J.C. Eijkemans, and Ewout W. Steyerberg (2006), "Randomized Controlled Trials With Time-to-Event Outcomes: How Much Does Prespecified Covariate Adjustment Increase Power?" *Annals of Epidemiology*, 16(1): 41-48.

Holtz, Timothy H., Maya Sternberg, Steve Kammerer, et al. (2006), "Time to Sputum Culture Conversion in Multidrug-Resistant Tuberculosis: Predictors and Relationship to Treatment Outcome," *Annals of Internal Medicine*, 144(9): 650-659.

Hosmer Jr., David W. , Stanley Lemeshow, and Susanne May (2008), *Applied Survival Analysis: Regression Modeling of Time to Event Data*, John Wiley & Sons.

Howards, Penelope P., Enrique F. Schisterman, Charles Poole, Jay S. Kaufman, and Clarice R. Weinberg (2012), " 'Toward a Clearer Definition of Confounding' Revisited With Directed Acyclic Graphs," *American Journal of Epidemiology*, 176(6): 506-511.

Ibrahim, Joseph G., Ming-Hui Chen, and Debajyoti Sinha (2013), *Bayesian Survival Analysis*, Springer Science & Business Media.

Jepsen, P., S. P. Johnsen, M. W. Gillman, and H. T. Sorensen (2004), "Interpretation of Observational Studies," *Heart*, 90(8): 956960.

Kasza, Jessica, Darren Wraith, Karen Lamb, and Rory Wolfe (2014), "Survival Analysis of Time-to-Event Data in Respiratory Health Research Studies," *Respirology*, 19(4): 483492.

Khan, Azizur Rahman, and Carl Riskin (2005), "China's Household Income and Its Distribution, 1995 and 2002," *The China Quarterly*, 182: 356-384.

Kleinbaum, David G., and Mitchel Klein (2005), *Survival Analysis: A Self-Learning Text. 2nd edition*, New York, NY: Springer.

Klungsyr, Ole, Joe Sexton, Inger Sandanger, and Jan F. Nygrd (2008), "Sensitivity Analysis for Unmeasured Confounding in a Marginal Structural Cox Proportional Hazards Model," *Lifetime Data Analysis*, 15(2): 278-294.

Lawless, Jerald F. (2002), *Statistical Models and Methods for Lifetime Data*, John Wiley & Sons.

Lin, D. Y., B. M. Psaty, and R. A. Kronmal (1998), "Assessing the Sensitivity of Regression Results to Unmeasured Confounders in Observational Studies," *Biometrics*, 54(3): 948-963.

Lin, Hui-Wen, and Yi-Hau Chen (2014), "Adjustment for Missing Confounders in Studies Based on Observational Databases: 2-Stage Calibration Combining Propensity Scores From Primary and Validation Data," *American Journal of Epidemiology*, 130.

Lin, Nan Xuan, Stuart Logan, and William Edward Henley (2013), "Bias and Sensitivity Analysis When Estimating Treatment Effects from the Cox Model with Omitted Covariates: Bias and Sensitivity Analysis for the Cox Model," *Biometrics*, 69(4): 850-860.

Lipsitz, Stuart R., and Joseph G. Ibrahim (1998), "Estimating Equations with Incomplete Categorical Covariates in the Cox Model," *Biometrics*, 54(3): 1002-1013.

Magder, Laurence S., and James P. Hughes (1997), "Logistic Regression when the Outcome is Measured with Uncertainty," *American Journal of Epidemiology*, 146(2): 195-203.

McCandless, Lawrence C., Paul Gustafson, and Peter C. Austin (2009), "Bayesian Propensity Score Analysis for Observational Data," *Statistics in Medicine*, 28(1): 94-112.

McCandless, Lawrence C., Paul Gustafson, and Adrian Levy (2007), "Bayesian Sensitivity Analysis for Unmeasured Confounding in Observational Studies," *Statistics in Medicine*, 26(11): 2331-2347.

McCandless, Lawrence C., Sylvia Richardson, and Nicky Best (2012), "Adjustment for Missing Confounders Using External Validation Data and Propensity Scores," *Journal of the American Statistical Association*, 107(497): 40-51.

McInturff, Pat, Wesley O. Johnson, David Cowling, and Ian A Gardner (2004), "Modelling Risk When Binary Outcomes Are Subject to Error," *Statistics in Medicine*, 23(7): 1095-1109.

Millet, Juan-Pablo, Evelyn Shaw, ngels Orcau, et al. (2013), "Tuberculosis Recurrence after Completion Treatment in a European City: Reinfection or Relapse?" *PLoS ONE*, 8(6).
http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3679149/

Mitra, Nandita, and Daniel F. Heitjan (2007), "Sensitivity of the Hazard Ratio to Nonignorable Treatment Assignment in an Observational Study," *Statistics in Medicine*, 26(6): 1398-1414.

Mudholkar, Govind S., Deo Kumar Srivastava, and Georgia D. Kollia (1996), "A Generalization of the Weibull Distribution with Application to the Analysis of Survival Data," *Journal of the American Statistical Association*, 91(436): 1575-1583.

Paulino D., Paulo Soares, and John Neuhaus (2003) "Binomial Regression with Misclassification," *Biometrics* 59(3): 670675.

Rosenbaum, Paul R., and Donald B. Rubin (1983), "The Central Role of the Propensity Score in Observational Studies for Causal Effects," *Biometrika*, 70(1): 41-55.

Rothman, Kenneth J., Sander Greenland, and Timothy L. Lash (2008), *Modern Epidemiology*, Lippincott Williams & Wilkins.

Schneeweiss, Sebastian (2006), "Sensitivity Analysis and External Adjustment for Unmeasured Confounders in Epidemiologic Database Studies of Therapeutics," *Pharmacoepidemiology and Drug Safety*, 15(5): 291-303.

Song, Jae W., and Kevin C. Chung (2010), "Observational Studies: Cohort and Case-Control Studies," *Plastic and Reconstructive Surgery*, 126(6): 2234-2242.

Stamey, James D., Daniel P. Beavers, Douglas Faries, Karen L. Price, and Seaman (2014), "Bayesian Modeling of Cost-Effectiveness Studies with Unmeasured Confounding: A Simulation Study," *Pharmaceutical Statistics*, 13(1): 94-100.

Steenland, K. and Greenland S. (2004), "Monte Carlo Sensitivity Analysis and Bayesian Analysis of Smoking as an Unmeasured Confounder in a Study of Silica and Lung Cancer," *American Journal of Epidemiology*, 160(4): 384-392.

Sturmer, Til, Sebastian Schneeweiss, Jerry Avorn, and Robert J. Glynn. (2005), "Adjusting Effect Estimates for Unmeasured Confounding with Validation Data Using Propensity Score Calibration," *American Journal of Epidemiology*, 162(3): 279-289.

Swindell, William R. (2009), "Accelerated Failure Time Models Provide a Useful Statistical Framework for Aging Research," *Experimental Gerontology*, 44(3): 190-200.

VanderWeele, Tyler J. (2008), "Sensitivity Analysis: Distributional Assumptions and Confounding Assumptions," *Biometrics*, 64(2): 645-649.

VanderWeele, Tyler J., and Ilya Shpitser (2013), "On the Definition of a Confounder," *Annals of Statistics*, 41(1): 196-220.

Wahed, Abdus S., The Minh Luong, and Jong-Hyeon Jeong (2009), "A New Generalization of Weibull Distribution with Application to a Breast Cancer Data Set," *Statistics in Medicine*, 28(16): 2077-2094.

Weinberg, Clarice R. (1993), "Toward a Clearer Definition of Confounding," *American Journal of Epidemiology*, 137(1): 1-8.

Zhao, Yanlin, Shaofa Xu, Lixia Wang, et al. (2012), "National Survey of Drug-Resistant Tuberculosis in China," *The New England Journal of Medicine*, 366(23): 2161-2170.