Bayesian Sensitivity Analysis for a Missing Data Model

Incorporating Covariates via a Cox Model

C. Van Vliet

Student number: 4372905

TU Delft

Bayesian Sensitivity Analysis for a Missing Data Model

Incorporating Covariates via a Cox Model

by

C. Van Vliet

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday 2 August, 2023 at 14:00.

Student number:	4372905	
Project duration:	5 December, 2022 – 2 August, 2023	
Thesis committee:	Prof. dr. A. W. Van der Vaart	TU Delft, supervisor
	Dr. J. H. Krijthe,	TU Delft
	B. Eggen MSc.,	TU Delft

TU Delft

Abstract

In problems with missing data, the data are often considered to be missing at random. This assumption can not be checked from the data. We need to assess the sensitivity of study conclusions to violations of non-identifiable assumptions. This thesis performs Bayesian sensitivity analysis for a missing data model with life time outcomes and covariate information. The outcome distribution is modelled through a Cox model, with a beta process prior on the cumulative hazard function. We run experiments in a simulation study to test the performance of the model in scenarios with simulated data of several sample sizes. We show the validity of the model in the context of Bayesian sensitivity analysis, and propose extensions.

Preface

This thesis was written in context of the MSc. programme in Applied Mathematics at Delft University of Technology. The thesis is targeted at students in the same MSc. programme with a background in statistics or machine learning. Elementary knowledge of measure theory is helpful, but not strictly speaking necessary to understand the content. Readers with a less solid background in statistics are advised to thoroughly read chapter 2. Experienced statisticians can skip most (or all) of chapter 2.

I would like to thank my supervisor, Aad van der Vaart, for the countless hours he took to discuss any questions I had in carrying out the research in this thesis. He was always willing to fill in the gaps in my knowledge about more elementary aspects of Bayesian statistics, and he seemed to enjoy giving mini-lectures about more advanced topics on nonparametric Bayesian methods almost as much as I enjoyed receiving them. Also thanks for the freedom in studying new topics! I thoroughly enjoyed learning about nonparametric Bayesian inference and survival analysis. I would also like to thank Bart Eggen, who was always available to discuss theory and implementation details, and provided me with a lot of feedback. Thanks to Jaeyong Lee, who sent me his code to simulate from a beta process -it really helped me get started with the implementation of the sampling scheme.

Lastly, I would like to thank my friends, family, and my wife Sharline, who were always kind enough to ask about my thesis, and were patient enough to sit through my explanations of the research.

C. Van Vliet
Delft, July 2023

Contents

Glossary of Symbols

i.i.d.	independent and identically distributed
c.d.f.	cumulative distribution function
p.d.f.	probability density function
p.m.f.	probability mass function
c.h.f.	cumulative hazard function
a.s.	almost surely
$\mathbb{1}$	indicator function
δ	Dirac measure
$\delta_x(A) = \mathbb{1}_A(x) = \begin{cases} 0, & x \notin A \\ 1, & x \in A \end{cases}$	
$\times$	Cartesian product
$\setminus$	difference of sets
$\propto$	proportional to
$F- = F(u-)$	left limit of F (at u)
$\Delta F = \Delta F(u) = F(u) - F(u-)$	
$a \coloneqq b$	a is defined as b
$\mathbb{N}$	set of natural numbers $\{1, 2, \dots\}$
$\mathbb{N}_0 = \mathbb{N} \cup \{0\}$	
$\mathbb{R}$	set of real numbers
$\mathbb{S}_k$	k-dimensional unit simplex
$\mathfrak{M}(\mathfrak{X})$	space of probability measures on a sample space $\mathfrak{X}$
$\mathfrak{M}_\infty(\mathfrak{X})$	space of positive measures on a sample space $\mathfrak{X}$
$\text{supp}(P)$	support of a probability measure P
$\mathbb{E}[X]$	expectation of X
$\mu f = \int f d\mu$	
$\text{var}[X]$	variance of X
$\text{cov}(X, Y)$	covariance of X and Y
$\mathcal{L}(X)$	law of X
$X \perp\!\!\!\perp Y$	random variables X and Y are independent
$X \sim P$	X has distribution P
$\mathbb{P}_n$	empirical measure
$\text{Be}(a, b)$	beta distribution with shape parameters a and b
$\text{Ber}(p)$	Bernoulli distribution with mean p
$\text{Dir}(k; \alpha_1, \dots, \alpha_k)$	k-dimensional Dirichlet distribution
$\text{Exp}(\lambda)$	exponential distribution with mean $\frac{1}{\lambda}$
$\text{Ga}(a, b)$	gamma distribution with shape a and scale b
$\text{Geom}(p)$	geometric distribution with success probability p
$\text{Nor}(\mu, \sigma^2)$	normal distribution with mean μ and variance σ^2
$\text{Pois}(\lambda)$	Poisson distribution with mean λ
$\text{Unif}(a, b)$	uniform distribution on (a, b)
$\text{BP}(c, \Lambda)$	beta process with concentration function c and baseline c.h.f. Λ
$\text{DP}(\alpha)$	Dirichlet process with base measure α

1 Introduction

Causal questions are at the heart of science. In many applications data are missing, and we need to make assumptions about the missingness to be able to draw conclusions about causality. An important assumption is the assumption that the probability that a value is missing is the same for all data points within groups defined by covariate information present in the observed data, i.e. data are *missing at random* (MAR). Here a problem arises: the MAR assumption can not be checked from the observed data and, moreover, it is quite a strong assumption, and in many applications actually implausible to hold. If the data are not missing at random, but are assumed to be MAR, the study conclusions will paint a distorted picture.

A way to mitigate this problem, is to perform a sensitivity analysis. This is done by modelling deviations from MAR, and investigating the sensitivity of some conclusion about causality to varying magnitudes of deviation from MAR. Often, domain experts have prior knowledge about the nature and magnitude of deviations from MAR. It seems natural to account for this prior knowledge by carrying out a sensitivity analysis within the Bayesian paradigm: *Bayesian sensitivity analysis* (BSA).

Robins, Rotnitzky & Scharfstein (2000) proposed a way to model deviations from MAR in [28], which was applied in the setting of BSA by Scharfstein, Daniels & Robins (2003) and Eggen, Van der Pas & Van der Vaart (2023) [30, 10], who investigated the situation where covariate information is ignored. This thesis extends the approach of these works to a setting with lifetime outcomes, taking into account covariate information by means of a Cox model.

We seek a flexible model for the observed data, and we therefore use nonparametric Bayesian methods. More specifically, a Dirichlet process prior is put on the distribution of the covariates, and a beta process prior is put on the baseline cumulative hazard function in the Cox model. Following [30, 10], the missingness mechanism is modelled in a parametric way, to preserve interpretability of the parameters. In order to draw samples from the posterior, a Gibbs sampling scheme was implemented. A variety of experiments is then conducted in a simulation study, to gauge the usefulness of our model for BSA.

The report is organised in the following way: Chapter 2 discusses concepts and techniques from various areas of statistics that are used in the remaining chapters. An introduction into the topic of Bayesian sensitivity analysis is given in chapter 3, along with examples from the literature. Chapter 4 contains technical details on the statistical model, posterior sampling scheme, and experimental setup of the simulation study. The results of the simulation study are reported in chapter 5. Chapter 6 concludes this thesis with a discussion of the simulation study.

2

Statistical Preliminaries

This chapter aims to provide the reader with necessary background on concepts from statistics that are used or referenced in the remainder of the thesis. Section 2.1 details the framework of causal inference and missing data. Section 2.2 details techniques that are needed for the computation of posterior distributions. An introduction to some subjects from nonparametric Bayesian inference is given in section 2.3, before concluding the chapter with a description of survival analysis[1] in section 2.4.

2.1. Causal Inference & Missing Data

Many applications of statistics require the estimation of the causal effect of some action on a certain outcome. A common example is the setting of medical treatment: does treatment X have a (positive) causal effect on Y? We first use this example to introduce the potential outcomes framework in section 2.1.1. In section 2.1.2 we frame causal inference as a missing data problem.

2.1.1. Potential Outcomes

Consider the setting of medical treatment administration. We take a sample of individuals from a population, and we are interested in the causal effect of some treatment on an outcome (e.g. some cell count in blood, life expectancy). Pertaining to every individual are the following random variables:

$$\begin{aligned} A &= \text{treatment indicator,} \\ Y^0 &= \text{outcome if not treated,} \\ Y^1 &= \text{outcome if treated,} \\ Y &= \text{observed outcome,} \\ Z &= \text{measured covariates,} \end{aligned}$$

where $A = 1$ if the individual is given treatment and $A = 0$ if the individual is not given treatment. Y^1 and Y^0 are called *potential outcomes*, and usually only one of the two is measured for every individual, since it is impossible to simultaneously both treat and not treat an individual. It is possible to model multiple treatment arms, in which case A could take values in $\{0, 1, \dots, k\}$, and we would model corresponding potential outcomes $Y^0, Y^1, \dots, Y^k$. This extension is outside the scope of this text.

We are interested in the difference between the distributions of Y^1 and Y^0, and therefore we define the *average causal effect* (ACE) by

$$\mathbb{E}Y^1 - \mathbb{E}Y^0.$$

Typically, only (A, Y, Z) are observed, and thus, in order to estimate the ACE, we need assumptions linking the observed data to the potential outcomes. We make the following assumptions:

[1] with a strong emphasis on nonparametric Bayesian methods

$$
\begin{aligned}
Y = Y^A, &\quad \textit{Consistency} \text{ (C)},\\
Y^a \perp\!\!\!\perp A|Z,\ \forall a, &\quad \textit{Conditional exchangeability} \text{ (CE)},\\
\Pr(A = a|Z) > 0,\ \forall a, &\quad \textit{Positivity} \text{ (P)}.
\end{aligned}
$$

Let's discuss the interpretation of these assumptions. The consistency assumption (C) states that the observed outcome Y is equal to the potential outcome associated with the treatment indicator A, yielding the expression $Y = AY^1 + (1-A)Y^0$. The positivity assumption (P) restricts the *propensity score* $f(Z) = \Pr(A = 1|Z)$ to $(0, 1)$: for every value of Z, the probability of (not) receiving treatment must be non-zero, i.e. there is no individual for whom it is impossible to (not) receive treatment. The conditional exchangeability[2] assumption (CE) states that within levels of the covariate Z, observation of any of the potential outcomes does not provide any additional information about the treatment assignment, and vice versa.

The following theorem shows that under the three discussed assumptions, the ACE can be obtained from the distribution of the observed data (A, Y, Z).

Theorem 2.1. *If (C), (CE), and (P) hold, we have* $\mathbb{E}Y^1 - \mathbb{E}Y^0 = \mathbb{E}_Z\left(\mathbb{E}(Y|A = 1, Z) - \mathbb{E}(Y|A = 0, Z)\right)$.

Proof. We first show $Y^a|Z \sim Y|A = a, Z$:

$$
\begin{aligned}
\Pr(Y \in B|A = a, Z) &\overset{\text{(P)}}{=} \frac{\Pr(Y \in B, A = a|Z)}{\Pr(A = a|Z)} \overset{\text{(C)}}{=} \frac{\Pr(Y^a \in B, A = a|Z)}{\Pr(A = a|Z)}\\
&\overset{\text{(CE)}}{=} \frac{\Pr(Y^a \in B|Z)\Pr(A = a|Z)}{\Pr(A = a|Z)} = \Pr(Y^a \in B|Z).
\end{aligned}
$$

It follows from the towering property that $\mathbb{E}Y^a = \mathbb{E}_Z\mathbb{E}(Y^a|Z) = \mathbb{E}_Z\mathbb{E}(Y|A = a, Z)$. □

2.1.2. Missing Data

The causal inference setup can be viewed as an instance of a missing data problem. Indeed, for some individuals the variable Y^0 is missing, while for others Y^1 is not observed, resulting in the popular characterisation of causal inference as "missing data *twice*". In the context of missing data, A is interpreted as the mechanism that determines whether a data point is observed $(A = 1)$ or not $(A = 0)$. As mentioned in the previous section, A could be allowed to take values in $\{0, 1, \ldots, k\}$, which in the missing data context models multiple reasons why a data point is (not) missing, but we restrict our attention to a binary A. We are now only interested in estimating $\mathbb{E}Y^1$ (*half* the ACE), and will usually suppress the superscript of the potential outcome of interest, writing Y instead of Y^1. In the context of missing data, a more common name for the (CE) assumption from the previous section, is *missing at random* (MAR) (Rubin, 1976) [29]:

$$Y \perp\!\!\!\perp A|Z.$$

2.2. Bayesian Computation

This section briefly reviews the main ideas of Markov Chain Monte Carlo methods in section 2.2.1, and looks at two important examples of such methods: the Metropolis-Hastings (MH) algorithm (2.2.2) and Gibbs sampling (2.2.3). Throughout this section, we will adhere to the Bayesian setting detailed in section 1.3 of [14], and in [33]. In particular, we assume a dominated model $\{P_\theta : \theta \in \Theta\}$.

2.2.1. Markov Chain Monte Carlo

A well-known expression of the posterior distribution is given by *Bayes' formula*:

$$\Pi(B|X) = \frac{\int_B p_\theta(X)d\Pi(\theta)}{\int p_\theta(X)d\Pi(\theta)}. \tag{2.1}$$

[2] alternative names: unconfoundedness, no unmeasured confounding, ignorability

Usually, the integral in the denominator of 2.1 is intractable, and thus a nice analytical expression for the posterior distribution only rarely exists. If we want to compute properties of the posterior distribution (e.g. mean, mode, credible region), we need methods to approximate the posterior. *Markov Chain Monte Carlo* (MCMC) methods form a class of algorithms that are up to this task.

The idea of MCMC is to simulate (dependent) values from a Markov chain that has as its stationary distribution the posterior distribution of interest. We change notation for convenience, and denote the underlying sample space as $\mathcal{Y}$ with elements y rather than Θ with elements θ. We consider a *time homogeneous Markov chain* $\{Y_n : n \in \mathbb{N} \cup \{0\}\}$ with *state space* $(\mathcal{Y}, \mathscr{Y})$ and transition kernel Q given by

$$Q(y, B) = \int_B q(y, z) d\nu(z) = \Pr(Y_{n+1} \in B | Y_n = y), \tag{2.2}$$

with $y \in \mathcal{Y}$, $B \in \mathscr{Y}$, and q a *transition density* with respect to a σ-finite measure ν on $(\mathcal{Y}, \mathscr{Y})$. Let the *stationary distribution*[3] of the Markov chain be denoted by Π, i.e. for all measurable $B \in \mathscr{Y}$ we have

$$\int_{\mathcal{Y}} Q(y, B) d\Pi(y) = \Pi(B).$$

In this chapter we assume that Π allows a density π with respect to ν. If we could simulate independent values from the Markov chain, the law of large numbers states that the sample average tends to the true mean of the stationary distribution almost surely -but we can only simulate dependent values. Fortunately, the *ergodic theorem* extends the law of large numbers to samples of dependent values, if the Markov chain is well connected in the following sense: for some measure ψ on $(\mathcal{Y}, \mathscr{Y})$, and for every $B \in \mathscr{Y}$ with $\psi(B) > 0$ we must have that

- for any $y \in \mathcal{Y}$, at some point in time B can be reached from y with positive probability[4], and
- if one departs from any $y \in B$, the number of returns to B is infinite almost surely[5].

If the Markov chain satisfies these two conditions, it is called *ergodic*. In practice, we additionally require a Markov chain to not periodically cycle between subsets of $\mathcal{Y}$. The reason for this, roughly speaking, is that not all individual Y_n are necessarily drawn from the stationary distribution, and we want to prevent the chain from periodically revisiting ”problematic” areas that distort the picture of the stationary distribution. Another practice in using MCMC samplers to generate draws from a posterior distribution, is the discardment of the first m values $Y_0, \dots, Y_m$, which is called the *burn-in*. The ergodicity of the chain motivates this practice, since only the first N values of the chain are simulated, and typically $Y_{m+1}, \dots, Y_N$ give a better representation of the stationary distribution than all $Y_0, \dots, Y_N$. Let’s look at two well-known MCMC algorithms in the following sections.

2.2.2. Metropolis-Hastings

Often the target density π is known up to a multiplicative constant, say we know $\tilde{\pi} = c\pi$, but $c \neq 0$ is unknown. The MH algorithm [26, 15] uses a *proposal density* q in conjunction with $\tilde{\pi}$ to create a Markov chain with π as its stationary distribution. We define the MH *acceptance probability* as follows:

$$\alpha(y, z) = \frac{\pi(z) q(z, y)}{\pi(y) q(y, z)} \wedge 1. \tag{2.3}$$

It is clear from 2.3 that it suffices to know $\tilde{\pi}$, and furthermore that a symmetric q vanishes from the expression. Algorithm 1 gives the MH algorithm.

In words, Algorithm 1 achieves the following: First a number of iterations N is chosen. Then the first value of the Markov chain, Y_0, is generated. Next, the algorithm iteratively samples an element from the proposal density q, and an element from the standard uniform distribution. If the uniformly distributed sample is smaller than the acceptance probability given by 2.3, the sample from the proposal density is the next value of the Markov Chain, otherwise the next value of the Markov Chain is set equal to the current value. The important question then remains: How do we choose q?

A popular choice of q is a symmetric density centred around the first argument of q. This version of MH is called *random walk* MH. We could for instance choose $q(y, \cdot) = \phi(\cdot; y, \sigma^2)$, the normal density centred around y with variance σ^2 (a tuning parameter).

[3] which we will assume to exist

[4] the chain is ψ-*irreducible*

[5] if the chain additionally is ψ-irreducible, this requirement makes the chain *Harris recurrent*

Algorithm 1 Metropolis-Hastings

Given: N
Initialise: Y_0
for $n = 1, \dots, N$ **do**
  Sample $Z_n \sim q(Y_{n-1}, \cdot)$
  Sample $U_n \sim \text{Unif}(0, 1)$
  if $U_n < \alpha(Y_{n-1}, Z_n)$ **then**
    $Y_n \coloneqq Z_n$
  else
    $Y_n \coloneqq Y_{n-1}$

2.2.3. Gibbs Sampler

In many cases the sample space has a "nice" product structure $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_M$. The Gibbs sampler exploits this structure by sampling from each of the *full conditional distributions* with densities

$$\pi_m(y_m | y_1, \dots, y_{m-1}, y_{m+1}, \dots, y_M) = \frac{\pi(y_1, \dots, y_M)}{\int \pi(y_1, \dots, y_M) d\mu_m(y_m)}, \tag{2.4}$$

where π is a density with respect to the product measure $\mu_1 \times \cdots \times \mu_M$ on $\mathcal{Y}$. The Gibbs sampler iteratively generates new samples y_m from the conditional densities $\pi_1, \dots, \pi_M$, conditional on the $M-1$ most recent samples of the other components of $(y_1, \dots, y_{m-1}, y_{m+1}, \dots, y_M)$, as is shown in Algorithm 2.

Algorithm 2 Gibbs Sampler

Given: N
Initialise: $Y_0 = (Y_{0,1}, \dots, Y_{0,M})$
for $n = 1, \dots, N$ **do**
  for $m = 1, \dots, M$ **do**
    Sample $Y_{n,m} \sim \pi_m(\cdot | \{Y_{n,i}, i < m \neq 1\}, \{Y_{n-1,i}, i > m \neq M\})$

The Gibbs sampler is attractive in many problems with missing data. Suppose the full data consist of $(X_{\text{obs}}, X_{\text{miss}})$ and we only observe X_{obs}. Let's say a model for the full data is given by $p_\theta(x_{\text{obs}}, x_{\text{miss}})$, and the model parameter is equipped with a prior density $\pi(\theta)$. The posterior density $\pi(\theta | x_{\text{obs}})$ is then proportional to the product of $\pi(\theta)$ and the marginal density of X_{obs} given θ, the latter of which is obtained by integrating out X_{miss}:

$$\int p_\theta(x_{\text{obs}}, x_{\text{miss}}) d\mu(x_{\text{miss}}).$$

If this integral can't be evaluated analytically, it needs to be approximated numerically, which can come at a great computational cost. There is a cheaper way of marginalising out X_{miss}: we can generate samples from $\pi(\theta, x_{\text{miss}} | x_{\text{obs}}) \propto p_\theta(x_{\text{obs}}, x_{\text{miss}}) \pi(\theta)$ using a Gibbs sampler, and simply throw away the values of X_{miss}.

2.3. Nonparametric Bayesian Inference

This section discusses some background on concepts from the field of Bayesian nonparametric inference. Nonparametric Bayesian methods can be used to model data in a flexible way, without needing to specify a parameter of fixed size. Rather, these methods allow the model "parameter" to grow with the size of the data. We will specify priors in terms of stochastic processes, which will allow us to place a prior directly on an abstract space (e.g. a space of probability measures), rather than on a real-valued parameter. This text will follow closely the notation and ideas exhibited in [14], but many details will be left out. Throughout this section we will assume an underlying probability space $(\Omega, \mathcal{A}, \Pr)$ that is sufficiently rich[6].

[6] For details on the "richness" requirements, we refer to [14], in particular appendix J.

2.3.1. Dirichlet Process

We want to place a prior on a space of probability measures. To understand what this means, we need some ingredients. Let $(\mathfrak{X}, \mathscr{X})$ be a Polish space, and let $\mathfrak{M} = \mathfrak{M}(\mathfrak{X})$ be the collection of all Borel probability measures on $(\mathfrak{X}, \mathscr{X})$. Now we choose $\mathscr{M}$ to be the smallest σ-field on $\mathfrak{M}$ such that all maps $M \mapsto M(A)$ from $\mathfrak{M}$ to $\mathbb{R}$ are measurable, $\forall A \in \mathscr{X}$. Then we can give the definition of a random measure:

Definition 2.2 (Random measure)**.** A measurable map P from a probability space into $(\mathfrak{M}, \mathscr{M})$ is called a *random measure*.

Now we can place a prior Π on $(\mathfrak{M}, \mathscr{M})$, so that $P \sim \Pi$. A particular prior of interest is the Dirichlet process distribution, first introduced by Ferguson (1973) in [11]:

Definition 2.3 (Dirichlet process; Def. 4.1 in [14])**.** A random measure P on $(\mathfrak{X}, \mathscr{X})$, i.e. a probability measure on $(\mathfrak{M}, \mathscr{M})$, is said to possess a *Dirichlet process* distribution $\mathrm{DP}(\alpha)$ with *base measure* α, if for every finite measurable partition $A_1, \dots, A_k$ of $\mathfrak{X}$,

$$(P(A_1), \dots, P(A_k)) \sim \mathrm{Dir}(k; \alpha(A_1), \dots, \alpha(A_k)). \tag{2.5}$$

It might be intuitively helpful to parameterise the base measure α by two components: $|\alpha|$ and $\bar{\alpha}$. We call $|\alpha| = \alpha(\mathfrak{X})$ the *prior precision* of the *centre measure* $\bar{\alpha} = \alpha/|\alpha|$. The centre measure is the probability measure obtained through normalisation of the base measure. The prior precision can be interpreted as the concentration of a $\mathrm{DP}(\alpha)$ realisation around the centre measure $\bar{\alpha}$. The Dirichlet process is discrete in nature: a $\mathrm{DP}(\alpha)$ realisation is an almost surely discrete measure [11], even if the base measure is absolutely continuous. This makes the Dirichlet process an unsuitable prior for density estimation.

The moments of a Dirichlet process can be expressed in terms of the base measure:

Proposition 2.4 (Moments; Prop. 4.3 in [14])**.** *If $P \sim DP(\alpha)$, then for any measurable functions ψ and ϕ for which the expression on the right-hand side is meaningful,*

$$\mathbb{E}(P\psi) = \int \psi d\bar{\alpha}, \tag{2.6}$$

$$var(P\psi) = \frac{\int (\psi - \int \psi d\bar{\alpha})^2 d\bar{\alpha}}{1 + |\alpha|}, \tag{2.7}$$

$$cov(P\psi, P\phi) = \frac{\int \psi\phi d\bar{\alpha} - \int \psi d\bar{\alpha} \int \phi d\bar{\alpha}}{1 + |\alpha|}. \tag{2.8}$$

The following theorem shows that the Dirichlet process prior is a conjugate prior:

Theorem 2.5 (Conjugacy; Thm. 4.6 in [14])**.** *The $DP(\alpha + \sum_{i=1}^{n} \delta_{X_i})$ is a version of the posterior distribution given an i.i.d.sample $X_1, \dots, X_n$ from the $DP(\alpha)$*[7].

Theorem 2.5 gives an updating rule for the base measure of the Dirichlet process parameterised by α. The updating rule for the alternative parameterisation $(|\alpha|, \bar{\alpha})$ might be more intuitive:

$$|\alpha| \mapsto |\alpha| + n, \tag{2.9}$$

$$\bar{\alpha} \mapsto \frac{|\alpha|}{|\alpha| + n}\bar{\alpha} + \frac{n}{|\alpha| + n}\mathbb{P}_n, \tag{2.10}$$

where $\mathbb{P}_n = \frac{1}{n}\sum_{i=1}^{n} \delta_{X_i}$ is the empirical distribution. Equation 2.10 shows that the posterior centre measure is a convex combination of the prior centre measure and the empirical distribution function, where the weights are determined by the prior precision and the sample size. In light of this observation, $|\alpha|$ is sometimes called the *prior sample size*, and $|\alpha| + n$ is called the *posterior sample size*. If n gets

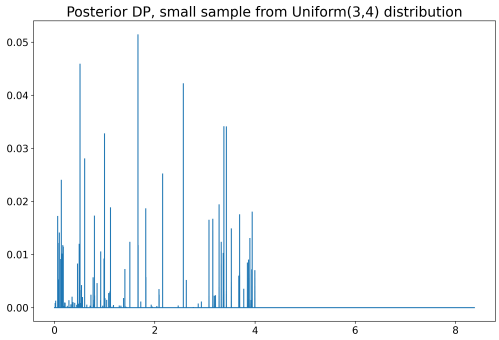

Figure 2.1: Realisation of a DP$(\alpha + \sum_{i=1}^{20} \delta_{X_i})$. $\bar{\alpha}$ = Exp(1) Prior precision = 20, data $X_1, \ldots, X_{20} \overset{\text{i.i.d.}}{\sim}$ Unif$(3,4)$.

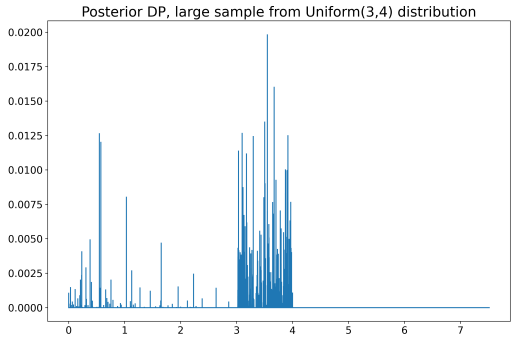

Figure 2.2: Realisation of a DP$(\alpha + \sum_{i=1}^{200} \delta_{X_i})$. $\bar{\alpha}$ = Exp(1) Prior precision = 20, data $X_1, \ldots, X_{200} \overset{\text{i.i.d.}}{\sim}$ Unif$(3,4)$.

very large compared to $|\alpha|$, the data start to dominate the posterior distribution, as illustrated in Figures 2.1 and 2.2.

Since almost every realisation P of a Dirichlet process is a random discrete probability measure, it is possible to express P as the sum of countably many weights at the locations θ_j where P is supported:

$$P = \sum_{j=1}^{\infty} W_j \delta_{\theta_j}, \tag{2.11}$$

where the weights W_j sum to 1. One way to find the correct weights, is through a technique called *stick-breaking*. Stick-breaking works as follows. Since we want to construct a probability measure, the weights of the distribution should sum to 1, so we start with a stick of length 1. We generate a random variable $0 \leq V_1 \leq 1$, break V_1 off the stick, leaving a stick of length $1 - V_1$, and we assign weight V_1 to point θ_1. Next, we generate a random variable $0 \leq V_2 \leq 1$, break V_2 off the stick, leaving a stick of length $(1 - V_2)(1 - V_1)$, and we assign weight $V_2(1 - V_1)$ to point θ_2. Continuing this process yields weights

$$W_j = V_j \prod_{l=1}^{j-1} (1 - V_l). \tag{2.12}$$

Under mild conditions[8] these weights W_j sum to 1. In [32] Sethuraman (1994) showed that, in order to obtain a realisation from the Dirichlet process, the V_j should be realisations of a beta distribution:

Theorem 2.6 (Stick-breaking representation; Thm. 4.12 in [14])**.** *If* $\theta_1, \theta_2, \ldots \overset{\text{i.i.d.}}{\sim} \bar{\alpha}$*, and* $V_1, V_2, \ldots \overset{\text{i.i.d.}}{\sim}$ *Be*$(1, |\alpha|)$ *are independent random variables and* $W_j = V_j \prod_{l=1}^{j-1}(1 - V_l)$*, then* $\sum_{j=1}^{\infty} W_j \delta_{\theta_j} \sim DP(\alpha)$*.*

The stick-breaking representation of the Dirichlet process inspires a simple algorithm for sampling realisations of a Dirichlet process: [8] gives an algorithm that generates realisations of a Dirichlet process posterior by combining Thm. 2.5 and Thm. 2.6:

Remark. *Algorithm 3 approximates the infinite sum from Thm. 2.6 by a sum of* J *terms, so* J *should be chosen fairly large to obtain a good approximation. Furthermore, it is often convenient to pass* $\bar{\alpha}$ *and* $|\alpha|$ *to the algorithm separately.*

2.3.2. Completely Random Measures

In section 2.3.1 the random measure was introduced, which allows us to place a prior on a space of probability measures. One type of random measure that is of particular interest is the completely random measure, introduced by Kingman (1967) in [18]:

[7] Meaning $X_1, \ldots, X_n \overset{\text{ind}}{\sim} P$, where P follows a Dirichlet process prior.
[8] specified in Lemma 3.4 of [14]

Algorithm 3 Dirichlet Process Posterior

Given: $\alpha, J \in \mathbb{N}$, $\mathbb{P}_n$
Sample $V_1, \dots, V_J \overset{\text{i.i.d.}}{\sim} \text{Be}(1, |\alpha| + n)$
for $j = 1, \dots, J$ **do**
  Sample $U \sim \text{Unif}[0, 1]$
  if $U < \frac{|\alpha|}{|\alpha|+n}$ **then**
    Sample $\theta_j \sim \bar{\alpha}$
  else
    Sample $\theta_j \sim \mathbb{P}_n$
  $W_j \coloneqq V_j \prod_{l=1}^{j-1}(1 - V_l)$

Definition 2.7 (Completely random measure; Def. J.5 in [14])**.** A measurable map $\Phi : \Omega \to (\mathfrak{M}_\infty, \mathscr{M}_\infty)$ is a *completely random measure* (CRM) on $\mathfrak{X}$ if the random variables $\Phi(A_1), \dots, \Phi(A_k)$ are mutually independent, for any disjoint sets $A_1, \dots, A_k \in \mathscr{X}$.

CRMs are often used in nonparametric Bayesian inference as (ingredients for) priors. A fundamental example of a completely random measure is the Poisson process.

Definition 2.8 (Poisson process; Def. J.1 in [14])**.** A *Poisson random subset* (PRS) of $\mathfrak{X}$ is a map Π from Ω into the collection of subsets of $\mathfrak{X}$ of at most countably many elements such that $N(A) \coloneqq \text{card}(\Pi \cap A)$ is a random variable for every $A \in \mathscr{X}$ and $N(A_i) \overset{\text{ind}}{\sim} \text{Pois}(\mu(A_i))$, for every finite collection of disjoint sets $A_1, \dots, A_k \in \mathscr{X}$ and a measure μ on $(\mathfrak{X}, \mathscr{X})$, called the *intensity measure*. The stochastic process $N = \{N(A) : A \in \mathscr{X}\}$ is called a *Poisson process* on $\mathfrak{X}$ with intensity measure μ. The corresponding counting measure on the points Π is called a *Poisson random measure*.

One can think of the Poisson process in the following way: The PRS Π randomly places points in the space $\mathfrak{X}$, and the Poisson random measure N counts for a certain set A the size of the overlap with Π. The PRS "is realised" in such a way that said counts are Poisson distributed, and independent for disjoint sets in $\mathfrak{X}$, showing that the Poisson process is a CRM by definition.

Although the distribution of an individual random variable $N(A)$ in the Poisson process with intensity measure μ is clear from def. 2.8, it will be helpful to characterise it in terms of its Laplace transform[9]:

$$\mathbb{E}(e^{-\theta \int f dN}) = \exp\Big(- \int (1 - e^{-\theta f(x)})\, d\mu(x) \Big), \tag{2.13}$$

for some measurable $f : \mathfrak{X} \to \mathbb{R}$ and $\theta > 0$. For $f = \mathbb{1}_A$ the right hand side of 2.13 reduces to $\exp(\mu(A)(e^{-\theta} - 1))$, which we recognise as the moment generating function (MGF) $M_X(-\theta)$ of a random variable $X \sim \text{Pois}(\mu(A))$, which is indeed the distribution of $N(A)$. This characterisation, in conjunction with the following result, will be instrumental in characterising other CRMs.

Proposition 2.9 (CRM decomposition)**.** *Any CRM Φ can, under reasonable conditions*[10]*, be represented in a unique way as*

$$\Phi = \sum_j \Phi(\{a_j\})\delta_{a_j} + \beta + \Psi, \tag{2.14}$$

with $a_1, a_2, \dots \in \mathfrak{X}$ fixed, β a deterministic σ-finite Borel measure on $\mathfrak{X}$, and a CRM

$$\Psi(A) = \sum_{(x,s) \in \Pi^c, x \in A} s = \iint \mathbb{1}_A(x) s \; N^c(dx, ds), \tag{2.15}$$

where Π^c is a PRS on $\mathfrak{X} \times (0, \infty]$ and N^c is the accompanying Poisson random measure associated with a Poisson process, with intensity measure ν^c such that $\nu^c(\{x\} \times (0, \infty]) = 0 \; \forall x \in \mathfrak{X}$, which is independent of the $\Phi(\{a_1\}), \Phi(\{a_2\}), \dots$.

[9] this result is due to a version of the Lévy-Khinchine formula, which is often formulated in terms of characteristic functions (Fourier transform)
[10] for details, consult [14] prop. J.6

In slightly different words, this result states that any CRM can be decomposed into

- a part with atoms at fixed locations, with random masses,
- a deterministic part,
- a part with atoms at random locations, with random masses, given by a Poisson process.

Let's look at Ψ from equation 2.15. It has no deterministic part and no fixed atoms. Application of 2.13 for a nonnegative measurable $f : \mathfrak{X} \to \mathbb{R}$ yields

$$\begin{aligned}\mathbb{E}(e^{-\theta \int f d\Psi}) &= \exp\Big(- \iint (1 - e^{-\theta s f(x)})\, \nu^c(dx, ds)\Big) \\ &= \exp\Big(\iint \Big(\Big(\sum_{j=0}^{\infty} \frac{(-\theta s f(x))^j}{j!} \Big) - 1\Big)\, \nu^c(dx, ds)\Big) \\ &= \exp\Big(\sum_{j=1}^{\infty} \frac{(-\theta)^j}{j!} \iint s^j f^j(x)\, \nu^c(dx, ds)\Big). \end{aligned} \tag{2.16}$$

The expression in 2.16 is instrumental in characterising CRMs, and moreover it inspires a method of simulating a CRM: Ψ is portrayed as a discrete measure with weights s at countably many locations x. From prop. 2.9 we know that (x, s) are the points of a Poisson process, and thus we can simulate a CRM Ψ with intensity measure $\nu(dx, ds)$ by simulating an appropriate Poisson process on $\mathfrak{X} \times \mathbb{R}^+$. We write the intensity measure as $\nu(dx, ds) = \rho_x(ds)\alpha(dx)$, where α is a probability measure on $\mathfrak{X}$ and ρ_x is a transition kernel on $\mathfrak{X} \times \mathbb{R}^+$. In practice we can only generate finitely many locations, so we first draw a sufficiently large number of locations $X_1, \dots, X_k \overset{\text{i.i.d.}}{\sim} \alpha$. Secondly, we generate $V_1, \dots, V_k$ from a standard homogeneous Poisson process on $\mathbb{R}^+$. Finally, we obtain the weights S_i at the locations X_i by transforming the V_i in the following way. Define the transform $L_x(s) = \rho_x((s, \infty))$, then the weights are given by the inverse of the transform as $S_i = L_{X_i}^{-1}(V_i)$. We then obtain the (approximation of the) CRM $\Psi(A) = \sum_{i=1}^{k} S_i \delta_{X_i}(A)$.

In this text, CRMs will mostly be used as priors in the context of survival analysis, and therefore the remainder of this section considers the case where $\mathfrak{X} = \mathbb{R}^+$. Let Φ be a CRM on $\mathbb{R}^+$ that is finite on finite intervals. In this case we can write the distribution function $X(t) = \Phi((0, t])$. Then X is an independent increment process.

Definition 2.10 (Independent increment process)**.** Let $\{X(t) : t \in \mathbb{R}^+\}$ be a stochastic process. If the sample paths of X are non-decreasing and right-continuous, and the increments over disjoint intervals are independent, then X is called an *independent increment process* (IIP).

Since any CRM can be decomposed as in 2.14, we can write the IIP X in the following way:

$$X(t) = \sum_{j : a_j \le t} \Delta X(a_j) + \beta(t) + \int_{(0,t]} \int_{(0,\infty)} s\, N^c(dx, ds), \tag{2.17}$$

where the $\Delta X(a_j)$ are the *fixed jumps* at the fixed atoms of Φ, and $\beta(t)$ is a c.d.f. on $\mathbb{R}^+$. In this text we are not interested in CRMs with a deterministic part, so we set $\beta \equiv 0$. Let's remind ourselves that we are Bayesians. We might specify a prior for X that has no fixed atoms. In this case one can sample from the prior by simulating an appropriate Poisson process. If the prior on X has fixed atoms, then these need to be dealt with separately. The latter is also true for the posterior of X. Regardless of whether the prior on X contains atoms or not, the data will determine fixed atoms and bring about fixed jumps in the posterior of X. Let's look at two suitable priors for IIPs: the gamma process and the beta process.

Example 2.11 (Gamma process)**.** A CRM Φ on $\mathbb{R}^+$ with intensity measure $\nu(dx, ds) = s^{-1} e^{-bs}\, ds\, d\alpha(x)$, with α a σ-finite measure on $\mathbb{R}^+$, is called a *gamma process*. If we set $b = 1$ we obtain the *standard*

gamma process. Using 2.16 we can determine the distribution of the $\Phi(A)$:

$$\begin{aligned}\mathbb{E}(e^{-\theta\int f d\Phi}) &= \exp\left(\sum_{j=1}^{\infty}\frac{(-\theta)^j}{j!}\int_0^{\infty}s^{j-1}e^{-s}\,ds\int f^j(x)\,d\alpha(x)\right)\\ &= \exp\left(\sum_{j=1}^{\infty}\frac{(-\theta)^j}{j!}(j-1)!\int f^j(x)\,d\alpha(x)\right)\\ &= \exp\left(\int\sum_{j=1}^{\infty}\frac{(-\theta f(x))^j}{j}\,d\alpha(x)\right)\\ &= \exp\left(-\int\log(1+\theta f(x))\,d\alpha(x)\right)\\ &\overset{f=\mathbb{1}_A}{=}(1+\theta)^{-\alpha(A)},\end{aligned}$$

which we recognise as the MGF $M_Y(-\theta)$ of a random variable $Y \sim \text{Ga}(\alpha(A), 1)$, thus we conclude that the individual $\Phi(A)$ follow gamma distributions.

Remark. *If we normalise a CRM, we obtain a random probability measure $\Phi/\Phi(\mathfrak{X})$, which is aptly called a normalised completely random measure (NCRM). We previously encountered an NCRM: the Dirichlet process can be obtained by normalising a gamma process. The Dirichlet process itself, however, is not a CRM, because for any partition $A_1, \dots, A_k$ of $\mathfrak{X}$ the $P(A_1), \dots, P(A_k)$ follow a Dirichlet distribution and must therefore sum to 1, which implies that the individual $P(A)$ are negatively correlated and thus not independent.*

Example 2.12 (Beta process)**.** A CRM Φ on $\mathbb{R}^+$ with intensity measure $\nu(dx, ds) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}s^{a-2}(1-s)^{b-1}\mathbb{1}_{(0,1)}(s)\,dx\,ds$, with $a, b > 0$, is called the *standard beta process*. The jump sizes follow beta distributions, and are limited to $(0, 1)$, making the beta process an appealing prior in the context of survival analysis. A more general version of the beta process will be introduced in section 2.4.2.

2.4. Survival Analysis

This section provides the necessary background on survival analysis. First a general introduction is given in section 2.4.1. Then the focus shifts towards the application of nonparametric Bayesian methods in survival analysis: the Beta Process prior is introduced in section 2.4.2, and section 2.4.3 discusses the nonparametric Bayesian version of the Cox model. The main resources for this section are [14] (chapter 13) and [1].

2.4.1. Introduction

Survival analysis is the field of statistics that is concerned with studying the distribution of life times. A life time is the time between an initiating event (e.g. birth, installation of a component, admission to rehab) and a terminal event (e.g. death, failure of a component, relapse). The latter event is generally referred to as the *event of interest*. Since survival times are usually nonnegative, we consider the distribution of a random variable T on $(0, \infty)$, given by its cumulative distribution function F. An interesting feature of survival data, is that often at the moment of data collection the event of interest has not occurred for a number of data subjects. This phenomenon is called *right censoring*.

Definition 2.13 (Right censoring)**.** Let C be a *censoring variable*. $\tilde{T}$ is observed if $\tilde{T} \leq C$, and if $\tilde{T} > C$, C is observed. Put differently, we observe the pair $T = \min(\tilde{T}, C)$ and $\Delta = \mathbb{1}\{\tilde{T} \leq C\}$. An observation with $\Delta = 0$ is called *(right) censored*.

In the remainder of this chapter independent right censoring is assumed, thus we consider data of the form $D_n = \{(\Delta_1, T_1), \dots, (\Delta_n, T_n)\}$. We are generally more interested in the probability that a data subject is still alive by a certain time t, rather than the probability that a death has occurred by that time. For this reason we consider the *survival distribution* $S(t) = 1 - F(t),\ t \geq 0$. Closely linked to the survival distribution is the concept of hazard.

Definition 2.14 (Cumulative hazard function)**.** The *cumulative hazard function* (c.h.f.) corresponding to a survival distribution F is given by

$$H(t) = \int_{(0,t]} \frac{dF}{1-F-}, \tag{2.18}$$

where $F-$ denotes the left limit of F.

If F is absolutely continuous, then so is H, in which case H admits a density given by

$$h(t) = \frac{f(t)}{1-F(t)} = \lim_{\delta \downarrow 0} \frac{1}{\delta} \Pr(t \leq T \leq t+\delta | T \geq t), \tag{2.19}$$

where f is the density of the survival time. h is often called the *hazard rate*, and the right hand side of the preceding display motivates the interpretation of $h(t)$ as the "instantaneous rate of failure at time t, given survival up to time t".

The c.h.f. can be obtained from the survival distribution via the following *product integral*:

$$S(t) = \prod_{(0,t]} (1 - dH) = e^{-H^c(t)} \prod_{u \in (0,t]} (1 - \Delta H(u)), \tag{2.20}$$

where H^c denotes the continuous part of H, and we write $\Delta H(u) = H(u) - H(u-)$ for a jump of H at u. In the special case where $H = H^c$, the product integral reduces to $S = e^{-H}$. Since H might make jumps at certain locations, we define the function $A = -\log S$, which is closely related to the function H. In order to construct estimators for H and S, it is useful to capture the data D_n in two counting processes.

Definition 2.15 (Observed failures & subjects at risk)**.** Let the counting processes $N = \{N(t) : t \geq 0\}$ and $Y = \{Y(t) : t \geq 0\}$, the number of *observed failures* and the number of *subjects at risk*, respectively, be defined by

$$N(t) = \sum_{i=1}^{n} N_i(t) = \sum_{i=1}^{n} \mathbb{1}\{T_i \leq t\}\Delta_i = \sum_{i=1}^{n} \mathbb{1}\{\tilde{T}_i \leq t, \tilde{T}_i \leq C_i\}, \tag{2.21}$$

$$Y(t) = \sum_{i=1}^{n} Y_i(t) = \sum_{i=1}^{n} \mathbb{1}\{T_i \geq t\} = \sum_{i=1}^{n} \mathbb{1}\{\tilde{T}_i \geq t, C_i \geq t\}. \tag{2.22}$$

The processes N and Y are related through the c.h.f. H in the following way: Since Y is the number of subjects at risk, and H is the function that expresses the risk, one would intuitively expect the increase in the number of observed failures, i.e. materialised risk in a small time interval, dN, to be equal to YdH. This is indeed the case, in the sense that $N(t) - \int_0^t Y(s)dH(s)$ is a martingale. It follows that $\int_0^t \mathbb{1}\{Y(s) > 0\}(Y(s))^{-1}dN(s) - H(t)$ is a martingale, which motivates the following classical estimators:

Definition 2.16 (Classical estimators)**.** The *Nelson-Aalen* (NA) estimator for the cumulative hazard function is given by

$$\hat{H}(t) = \int_{(0,t]} \mathbb{1}\{Y > 0\} \frac{dN}{Y}.$$

The *Kaplan-Meier* (KM) estimator for the survival distribution is given by

$$\hat{S}(t) = \prod_{(0,t]} \left(1 - \mathbb{1}\{Y > 0\} \frac{dN}{Y}\right).$$

The latter estimator makes intuitive sense: Survival up to time t is equivalent with not dying in any of the subintervals $(t_{i-1}, t_i]$ for any arbitrary partition $0 = t_0 < t_1 < \cdots < t_k = t$ of $(0, t]$. Since the conditional probability of dying in $(t_{i-1}, t_i]$ given survival past t_{i-1} can be estimated by the number of deaths in the interval, $\Delta N(t_{i-1}, t_i]$, divided by the number at risk at the beginning of the interval, $Y(t_{i-1})$, the resulting product (over arbitrarily small intervals) of survival probabilities is recognised as the KM estimator.

2.4.2. Beta Process Prior

The data D_n are generated by drawing $\tilde{T}_i \overset{\text{i.i.d.}}{\sim} F$ and $C_i \overset{\text{i.i.d.}}{\sim} G$, for $i = 1, \dots, n$. Then there are two options per data point: either $\Delta_i = 1$, meaning the event of interest has occurred at time T_i, or $\Delta_i = 0$, which means survival up to $T_i = C_i$. Following [20], the likelihood of all observations can be formulated in terms of the counting processes from 2.15:

$$\prod_{t \in \mathbb{R}^+} dH(t)^{dN(t)}(1 - dH(t))^{Y(t) - dN(t)}. \tag{2.23}$$

[14, p. 400] points out that the likelihood in 2.23 has an intuitive interpretation as the continuous execution of binomial experiments: the number of subjects at risk $Y(t)$ can be seen as the number of independent experiments at time t, and the number of observed failures $N(t)$ can be seen as the number of successes. Then $dH(t)$ can be understood as the probability of success at time t. Since the beta distribution is a conjugate prior to the binomial distribution, it would be nice if we could put a prior on H, such that the increments of H are independent and follow beta distributions. We encountered such an object in section 2.3.2: the beta process. A more general definition is given here.

Definition 2.17 (Beta process; Def. 13.3 in [14])**.** Let $c : [0, \infty) \to [0, \infty)$ be a measurable function, and let $\Lambda = \Lambda^c + \Lambda^d$ be a cumulative hazard function. A beta process with parameters (c, Λ), denoted $\text{BP}(c, \Lambda)$, is an independent increment process with intensity measure $\nu = \nu^c + \nu^d$ on $(0, \infty) \times (0, 1)$, given by

$$\begin{aligned} \nu^c(dx, ds) &= c(x)s^{-1}(1 - s)^{c(x)-1} d\Lambda^c(x) ds, \\ \nu^d(\{x\}, \cdot) &= \text{Be}(c(x)\Delta\Lambda(x), c(x)(1 - \Delta\Lambda(x))). \end{aligned}$$

If Λ is absolutely continuous with respect to the Lebesgue measure, it admits a hazard rate λ and the intensity measure of the beta process reduces to

$$\nu(dx, ds) = c(x)s^{-1}(1 - s)^{c(x)-1}\lambda(x) dx ds.$$

If $H(t) \sim \text{BP}(c, \Lambda)$, the mean and variance are given by

$$\begin{aligned} \mathbb{E}[H(t)] &= \Lambda(t), \\ \text{var}[H(t)] &= \int_{(0,t]} \frac{1 - \Delta\Lambda}{c + 1} d\Lambda. \end{aligned} \tag{2.24}$$

The concentration function c has an interpretation that is similar in spirit to the interpretation of the prior precision for the Dirichlet process: it reflects the prior belief in Λ. This can be seen from 2.24: if $c(t)$ is large on an interval (τ_1, τ_2), then the variance around the mean $\Lambda(t)$ is small on that interval, meaning the prior function Λ is followed more closely.

In section 2.3.2 we discussed that a CRM without fixed atoms can be simulated by simulating an appropriate Poisson process. Lee & Kim (2004) [20] proposed an efficient algorithm to do this for a beta process. The algorithm is shown in Algorithm 4.

Algorithm 4 Beta Process

Given: $c, \Lambda, \varepsilon, \tau$
$\mu \coloneqq \frac{1}{\varepsilon} \int_0^\tau c d\Lambda$
Sample $M \sim \text{Pois}(\mu)$
for $i = 1, \dots, M$ **do**
  Sample from $\pi(\theta_i) \propto c(\theta) d\Lambda(\theta) \mathbb{1}\{0 \leq \theta \leq \tau\}$
for $i = 1, \dots, M$ **do**
  Sample $s_i \sim \text{Be}(\varepsilon, c(\theta_{(i)}))$

2.4.3. Cox Model

In section 2.4.1 we considered the case where the data D_n are of the form $(\Delta_1, T_1), \dots, (\Delta_n, T_n)$. Sometimes there is additional data available on one or two categorical covariates, like sex or age cohort. In this case the data can be separately analysed per group defined by (combinations of) the covariate(s). Usually, data is available on more than two covariates, and some of these covariates might be numeric. The data D_n are then of the form $(\Delta_1, T_1, Z_1), \dots, (\Delta_n, T_n, Z_n)$. In this case it is no longer feasible to group the data and perform separate analyses, and we need to resort to some kind of regression model.

The *Cox model*, introduced by Cox (1972) in [6], is a semiparametric regression model that assumes that the hazard rate of an individual characterised by a covariate vector $z \in \mathbb{R}^p$ is proportional to some *baseline hazard rate*. The model is therefore often called the *proportional hazards model*.

Definition 2.18 (Cox model: proportional hazards)**.** Let $z \in \mathbb{R}^p$, and let $h_0(t)$ be a baseline hazard rate. Then the *Cox model* postulates that, for some $\beta \in \mathbb{R}^p$, we have

$$h(t|z) = e^{\beta^\top z} h_0(t). \tag{2.25}$$

A consequence of the assumption of proportional hazards, is that one can calculate the *hazard ratio* between two individuals characterised by covariate vectors z_1 and z_2:

$$\frac{h(t|z_1)}{h(t|z_2)} = \frac{\exp(\beta^\top z_1) h_0(t)}{\exp(\beta^\top z_2) h_0(t)} = e^{\beta^\top (z_1 - z_2)}.$$

In particular, if z_1 and z_2 are equal in all coordinates, except the jth coordinate, such that $z_{2j} = z_{1j} + 1$, then the hazard ratio becomes e^{β_j}, which is called the *relative risk* of the jth covariate. If all we are interested in, is determining relative risks, then we need not specify h_0.

In this thesis we are interested in Bayesian inference, and we would like to use the Cox model in conjunction with a nonparametric (beta process) prior on the baseline c.h.f. H. As a consequence, H_0 does not admit a density h_0, so we need to adapt the proportionality assumption from def. 2.18. A possibility is to make the conditional cumulative hazard functions $H(t|Z)$ proportional to H_0, but since H_0 can make jumps, this might result in an $H(t|Z)$ that makes jumps of size larger than 1, whereas a c.h.f. can not make jumps larger than 1. This problem could be remedied by scaling the jumps of H_0 by the smallest factor $e^{-\beta^\top Z}$, but this seems cumbersome. Instead the proportionality assumption of the Cox model in nonparametric Bayesian settings is usually adjusted to mean that the functions $A(t|Z) = -\log(S(t|Z))$ are proportional:

$$A(t|Z) = A(t) e^{\beta^\top Z}. \tag{2.26}$$

In the following we use this definition of proportional hazards to replace the one in def. 2.18. We now place a BP prior on H_0, and an independent prior on β. Since prior information about β is often unavailable, a uniform improper prior, $\pi(\beta) \propto 1$, seems like a decent choice [14, p. 426]. Now that priors are specified, we obtain expressions for the marginal posteriors of H and β in theorems 2.19 and 2.20. To improve readability of the remainder of this section, we introduce some notation:

$$\begin{aligned}
D_n(t) &= \{i : T_i = t, \Delta_i = 1, i = 1, \dots, n\}, \\
R_n(t) &= \{i : t \le T_i, i = 1, \dots, n\}, \\
R_n^+(t) &= R_n(t) \backslash D_n(t), \\
R_n(t, \beta) &= \sum_{j \in R_n(t)} \exp(\beta^\top Z_j), \\
R_n^+(t, \beta) &= \sum_{j \in R_n^+(t)} \exp(\beta^\top Z_j).
\end{aligned}$$

Theorem 2.19 (H posterior in Cox model; Thm. 13.32 & Ex. 13.34 in [14])**.** *If H follows a beta process prior with parameters (c, Λ), where Λ has a density λ, then the posterior of H given β and the data D_n*

is an independent increment process with intensity measure

$$\nu^c_{H|D_n,\beta}(dt,ds) = c(t)s^{-1}(1-s)^{R_n(t,\beta)+c(t)-1}\lambda(t)dtds,$$
$$\nu^d_{H|D_n,\beta}(\{t\},ds) \propto s^{-1}\prod_{i\in D_n(t)}[1-(1-s)^{\exp(\beta^\top Z_i)}](1-s)^{R^+_n(t,\beta)+c(t)-1}.$$

The intensity measure of the posterior of $H|\beta, D_n$ is the sum of a continuous[11] intensity measure $\nu^c_{H|D_n,\beta}$ and a discrete intensity measure $\nu^d_{H|D_n,\beta}$. It is worth noting that $\nu^c_{H|D_n,\beta}$ is the intensity measure of a beta process with parameters $R_n(t,\beta)+c(t)$, and $\frac{c(t)}{R_n(t,\beta)+c(t)}\Lambda(t)$. $\nu^d_{H|D_n,\beta}$ is the intensity measure of a process with jumps at fixed locations -the locations determined by the uncensored data points. The jump sizes of $\nu^d_{H|D_n,\beta}$ however, do not follow beta distributions. Therefore the posterior of H given β and the data D_n is not a beta process. Fortunately, we can deal with both the continuous and the discrete intensity measures separately.

Algorithm 5 H Posterior: Beta Process

Given: $c, \Lambda, \varepsilon, \tau, \beta, D_n$
$\mu \coloneqq \frac{1}{\varepsilon}\int_0^\tau c d\Lambda$
Sample $M \sim \text{Pois}(\mu)$
for $i = 1, \dots, M$ **do**
  Sample from $\pi(\theta_i) \propto c(\theta)d\Lambda(\theta)\mathbb{1}\{0 \le \theta \le \tau\}$
for $i = 1, \dots, M$ **do**
  Sample $s_i \sim \text{Be}(\varepsilon, R_n(\theta_{(i)},\beta) + c(\theta_{(i)}))$

Algorithm 6 H Posterior: Fixed Jumps

Given: N, β, D_n
Initialise: $U_0 = (U_{0,1}, \dots, U_{0,k^d})$
for $n = 1, \dots, N$ **do**
  for $i = 1, \dots, k_d$ **do**
    $V_i \coloneqq -\log(1-u_{n-1,i})$
    Sample $Y_i \sim \text{Geom}(1-e^{-V_i})$
    for $j = 1, \dots, k_i$ **do**
      Sample from $\pi(w_{ij}) \propto \exp(-e^{\beta_n^\top z_{i(j)}}V_i w_{ij})\mathbb{1}\{0 < w_{ij} < 1\}$
    Sample $V_i \sim \text{Ga}(k_i+1, c(t_i)+R^+_n(t_i,\beta_n)+y_i+\sum_{j=1}^{k_i}W_{ij}e^{\beta_n^\top z_{i(j)}})$
    $U_{n,i} = 1-e^{-V_i}$
$(u_1, \dots, u_{k^d}) \coloneqq (U_{N,1}, \dots, U_{N,k^d})$

Modification of Algorithm 4 to simulate the jumps of the beta process with intensity measure $\nu^c_{H|D_n,\beta}$ yields Algorithm 5. The algorithm generates jump locations $\theta_{(1)} < \dots < \theta_{(M)}$, forming the set $\mathcal{T}_c = \cup_{i=1}^M\{\theta_i\}$, and corresponding jump sizes $s_1, \dots, s_M$. In [19] Laud, Damien & Smith (1998) proposed an MCMC algorithm to simulate the jumps at fixed locations of an IIP with intensity measure $\nu^d_{H|D_n,\beta}$ (Algorithm 6). We denote the set of distinct uncensored observations by $\mathcal{T}_d$, with size $\text{card}(\mathcal{T}_d) = k^d$ and elements $t^d_1 < \dots < t^d_{k^d}$. Algorithm 6 then generates jump sizes $u_1, \dots, u_{k^d}$, taking into account the number of uncensored ties k_i for $i = 1, \dots, k_d$. Once these two parts are simulated, we can write the marginal posterior of the baseline c.h.f. as the sum of the sample paths of the two parts as

$$H(t) = \sum_{i=1}^M s_i\mathbb{1}_{[0,t]}(\theta_{(i)}) + \sum_{j=1}^{k^d}u_j\mathbb{1}_{[0,t]}(t^d_j) = \sum_{t^c\in\mathcal{T}_c}\Delta H(t^c)\mathbb{1}_{[0,t]}(t^c) + \sum_{t^d\in\mathcal{T}_d}\Delta H(t^d)\mathbb{1}_{[0,t]}(t^d), \qquad (2.27)$$

which can be used to find the marginal posterior of β:

[11] This should not be confused with "having a continuous sample path": An IIP with continuous intensity measure does not generally have a continuous sample path.

Theorem 2.20 (β posterior in Cox model)**.** *Consider the same setting as in theorem 2.19. The posterior of β given H and the data D_n is proportional to*

$$\pi(\beta) \prod_{t^d \in \mathcal{T}_d} \left((1 - \Delta H(t^d))^{R_n^+(t^d,\beta)} \prod_{j \in D_n(t^d)} (1 - (1 - \Delta H(t^d))^{\exp(\beta^\top Z_j)}) \right) \prod_{t^c \in \mathcal{T}_c} (1 - \Delta H(t^c))^{R_n(t^c,\beta)}. \quad (2.28)$$

Proof. The proof is a clarification[12] of the result in [20]. The assumption in 2.26 allows us to express the survival function $S(t|Z)$ in two ways, that are necessarily equivalent:

$$S(t|Z) = \left(\prod_{(0,t]} (1 - dH(\cdot|Z)) \right) = \left(\prod_{(0,t]} (1 - dH) \right)^{\exp(\beta^\top Z)},$$

implying $dH(t|Z) = 1 - (1 - dH(t))^{\exp(\beta^\top Z)}$ for all t. Now we can write the (partial) likelihood of β in the following way:

$$\begin{aligned} L(\beta|H, D_n) &= \prod_{i=1}^{n} \prod_{t \in [0,\tau]} dH(t|Z_i)^{dN_i(t)} (1 - dH(t|Z_i))^{Y_i(t) - dN_i(t)} \\ &= \prod_{i=1}^{n} \prod_{t \in [0,\tau]} (1 - (1 - dH(t))^{\exp(\beta^\top Z)})^{dN_i(t)} (1 - dH(t))^{\exp(\beta^\top Z)(Y_i(t) - dN_i(t))}. \end{aligned}$$

This expression of the likelihood in terms of the counting processes $Y_i(t)$ and $N_i(t)$ can cause confusion in the case of uncensored ties in the data. We want to allow each distinct observation to have its subject-specific hazard. The product integral can be eliminated from the expression by observing that $dH(t) > 0 \iff t \in \mathcal{T}_c \cup \mathcal{T}_d$. Furthermore we know that $dN_i(t) > 0 \iff t \in \mathcal{T}_d \iff D_n(t) \neq \emptyset$, so we obtain

$$\begin{aligned} L(\beta|H, D_n) &= \prod_{t \in \mathcal{T}_c \cup \mathcal{T}_d} \left(\prod_{j \in D_n(t)} (1 - (1 - \Delta H(t))^{\exp(\beta^\top Z_j)}) \prod_{k \in R_n^+(t)} (1 - \Delta H(t))^{\exp(\beta^\top Z_k)} \right) \\ &= \prod_{t^d \in \mathcal{T}_d} \left(\prod_{j \in D_n(t^d)} (1 - (1 - \Delta H(t^d))^{\exp(\beta^\top Z_j)}) \prod_{k \in R_n^+(t^d)} (1 - \Delta H(t^d))^{\exp(\beta^\top Z_k)} \right) \\ &\quad \cdot \prod_{t^c \in \mathcal{T}_c} \prod_{l \in R_n(t^c)} (1 - \Delta H(t^c))^{\exp(\beta^\top Z_l)}, \end{aligned}$$

and the result follows. □

Since the expression in eq. 2.28 is proportional to the posterior of β given H and D_n, the MH algorithm (Alg. 1) can be used to sample from the posterior, by substituting the expression into eq. 2.3.

[12] correction of the result: [20] appears to have a misplaced bracket

3

Bayesian Sensitivity Analysis

This chapter explains the practice of Bayesian sensitivity analysis, and illustrates it with examples from the literature. Section 3.1 introduces the concept of sensitivity analysis, motivates its usage in missing data problems, and places it in the Bayesian framework. Sections 3.2 and 3.3 describe the two prevalent modelling approaches in Bayesian sensitivity analysis, the unmeasured confounder approach and the selection bias approach, respectively. The chapter concludes with some ideas for interesting research directions in the field of Bayesian sensitivity analysis in section 3.4.

3.1. Bayesian Sensitivity Analysis for Missing Data

In studies with missing data, we need to make assumptions that are non-identifiable from the observed data, to draw valid inferences. The assumption investigated in this thesis is *missing at random* (MAR; conditional exchangeability (CE) in the context of causal inference), introduced in section 2.1. Depending on the context, we will use MAR and CE interchangeably. Since the MAR assumption can not be checked from the observed data, it is sensible to test how sensitive inferences regarding the outcome of interest drawn under the MAR assumption are to *deviations* from MAR (i.e. *missing not at random* (MNAR)). This practice is called *sensitivity analysis*.

In sensitivity analysis, a *sensitivity model* for the data is specified. This model, usually governed by a *sensitivity parameter* α, expresses how the missing data are believed to be missing. The special case where $\alpha = 0$ corresponds to the MAR assumption, and $\alpha \neq 0$ constitutes MNAR. The data are then analysed by varying α over a plausible range of values, usually determined with the help of expert opinions [30]. Every study has some quantity of interest, and sensitivity analysis gauges the changes in this quantity as α is being varied.

Bayesian sensitivity analysis (BSA) is exactly what it says on the tin: a Bayesian approach to sensitivity analysis. The main idea of the Bayesian paradigm is to equip model parameters with priors, and adjust the prior beliefs about these parameters by computing posterior distributions given the observed data. In this way, domain experts can express their beliefs about plausible values of α in a prior on α, instead of just a range of values. This approach has two great advantages when compared to alternative approaches to sensitivity analysis. First of all, knowledge of domain experts can be very naturally incorporated in the statistical model through the specification of priors [30]. Secondly, BSA yields posterior distributions, and not just point estimates of quantities of interest. These posteriors form a summary of the analysis [30, 10].

Within the field of BSA, there are many model elements that can be varied. An example is the scale on which outcomes and covariates are measured (e.g. binary, continuous). Another example is the way distributions are modelled. If it is well-known that a variable follows a certain parametric distribution, a choice can be made for a parametric model. If little is known about distributional forms of a variable, or if flexibility is required, it might be desirable to model certain aspects nonparametrically (or semiparametrically). In BSA, as in other areas of causal inference, the choice is often made for

flexible nonparametric Bayesian methods [27].

3.2. Unmeasured Confounder Approach

The unmeasured confounder approach was first used in 1959 in [5] and has been applied in many forms ever since. The approach assumes there is an unmeasured confounder U, such that $Y^a \perp\!\!\!\perp A|Z$ fails, but $Y^a \perp\!\!\!\perp A|U,Z$ is true. One then proceeds to model $(Y,U)|A,Z$, possibly by using the factorisation $P(Y,U|A,Z) = P(Y|A,U,Z)P(U|A,Z)$ [21], which gives a model for $Y|A,Z$ after marginalising out the latent confounder U. If priors are specified for $P(Y|A,U,Z)$ and $P(U|A,Z)$, a posterior is obtained for $\mathbb{E}Y^a = \mathbb{E}_{U,Z}\mathbb{E}(Y^a|U,Z) = \mathbb{E}_{U,Z}\mathbb{E}(Y|A=a,U,Z)$ by Thm. 2.1. Robins, Rotnitzky & Scharfstein (2000) argue in [28] that one should take the unmeasured confounder approach if there exists a known concrete confounder U that, for some reason, was not measured, and "there exists reasonable historical knowledge about the magnitude of assocation of U with both the outcome (conditional on treatment and measured confounders) and the treatment (conditional on measured confounders)". They state that the selection bias approach (discussed in section 3.3) is preferable if either the nature of U, or the magnitude of assocation of U with A and Y is unknown, because they believe domain experts can more easily form opinions about the association between Y^a and A, than about the scale (continuous/discrete) and dimensionality of U, and the magnitude of association of U with A and Y. In the following we describe some applications of the unmeasured confounder approach in the literature.

In [23] A, U, and Y are modelled as binary random variables, and Z is a vector of measured covariates. They model the distributions as follows:

$$\Pr(Y=1|A,U,Z) = \Psi(\beta_0 + \beta_1 A + \alpha U + \eta^\top Z), \tag{3.1}$$
$$\Pr(U=1|A,Z) = \Psi(\gamma_0 + \gamma_1 A + \xi^\top Z), \tag{3.2}$$

with $\Psi(x) = (1+e^{-x})^{-1}$. The model is non-identifiable from the observed data (A,Y,Z), and the assumptions about unmeasured confounding are captured by the parameters $\alpha, \gamma_0, \gamma_1$, and ξ. The prior beliefs about these parameters are then specified in a prior $\pi(\alpha,\gamma_0,\gamma_1,\xi) = \pi(\alpha)\pi((\gamma_0,\gamma_1))\pi(\xi)$. For all parameters zero-centred Gaussian priors were chosen. Then a Gibbs sampler was implemented to sample from the posterior of $(\alpha,\beta_0,\beta_1,\gamma_0,\gamma_1,\eta,\xi,U|A,Y,Z)$, and integrate out the latent variable U, as described in section 2.2.3.

Other papers take very similar approaches: In [22] an almost identical modelling approach is taken, but with zero-centred uniform priors on the sensitivity parameters. [25] again uses the same logistic regression model, but extends the model to distinguish between confounding and non-confounding covariates. The latter work follows an idea from [17] that measured confounders could be informative about unmeasured confounders: They view measured and unmeasured confounders as exchangeable, which reduces the difficulty of prior specification for individual bias parameters.

A slightly different approach is taken by [2, 7], who model the dependence of A on U and Z, rather than the dependence of U on A and Z:

$$\begin{aligned} Y|A,U,Z &\sim \text{Nor}(\beta A + \alpha U + \eta^\top Z, \sigma^2), \\ A|U,Z &\sim \text{Ber}(\Phi(\gamma U + \xi^\top Z)), \\ U &\sim \text{Ber}(p), \end{aligned}$$

where Φ denotes the standard normal c.d.f. . [7] extend the model by modelling the response surface in a nonparametric way using Bayesian additive regression trees (BART).

$$\begin{aligned} Y|A,U,Z &\sim \text{Nor}(\mu(A,Z) + \alpha_1 U, \sigma^2), \\ A|U,Z &\sim \text{Ber}(\Phi(\alpha_2 U + \beta^\top Z)), \\ \mu(A,Z),\sigma^2|A,Z &\sim \text{BART}(A,Z), \\ U &\sim \text{Ber}(p). \end{aligned}$$

BART, first introduced in [3], is an ensemble of regression trees. If BART is used for regression of Y on X, then each of the regression trees partitions the space of the X, and predicts a value $\hat{Y}$ for each element of the partition. For all the trees these $\hat{Y}$ are added together, to obtain a mean function μ. Then for data $(X_1, Y_1), ..., (X_n, Y_n)$ the Y_i are independently normally distributed with mean $\mu(X_i)$ and a common variance σ^2, which is also learned by BART. BART is a Bayesian method, because all parameters that govern the construction of the trees, as well as the common variance σ^2, are equipped with priors.

Another big difference between [2, 7] and [23], is that the former do not place priors on the sensitivity parameter (α_1, α_2). Gibbs sampling is again used to sample from the posterior, but this is done for a range of combinations of α_1 and α_2, resulting in a sort of "hybrid" BSA.

3.3. Selection Bias Approach

The selection bias approach is based on the observation that a violation of $Y^a \perp\!\!\!\perp A|Z$ implies that the distribution of $A|, Y^a, Z$ is not free of Y^a. The approach was first suggested by Robins et al. (2000) in [28], who provided two advantages in comparison to the unmeasured confounder approach. Firstly, the selection bias approach uses fewer sensitivity parameters, and is therefore computationally cheaper. Secondly, the choices regarding scale and dimensionality of U are avoided.

We illustrate the selection bias approach, and focus on just the one potential outcome Y^1, suppressing the superscript for ease of notation. We denote the full data by (A, Y, Z), and the observed data by (A, AY, Z)[1]. The law of the observed data can be identified with (1) the law of the covariates Z, $\mathcal{L}(Z) = P_Z$, (2) the propensity score $f(Z) = \Pr(A = 1|Z)$, and (3) the law of the observed outcome of interest, $\mathcal{L}(Y|A = 1, Z) = P_1(Y|Z)$. The aim of BSA is to assess sensitivity of study conclusions to deviations from MAR, by modelling the missing outcome of interest as MNAR rather than MAR. Saying data are MNAR is equivalent to saying $P_1(Y|Z)$ is different from the law of the missing outcome of interest, $P_0(Y|Z)$. The selection bias approach models $P_0(Y|Z)$ by specifying a *sensitivity function* $q(x, y)$ and assuming the following relationship holds

$$dP_0(Y|Z) \propto e^{-q(Y,Z)} dP_1(Y|Z). \tag{3.3}$$

The relationship in equation 3.3 requires P_0 and P_1 to have the same support -an again non-identifiable assumption. Using 3.3 we find an expression for $\mathcal{L}(Y|Z)$:

$$\begin{aligned} P(Y \in B|Z) &= f(Z)P_1(Y \in B|Z) + (1 - f(Z))P_0(Y \in B|Z) \\ &= \int_B \left(f(Z) + (1 - f(Z)) \frac{e^{-q(y,Z)}}{\int e^{-q(y,Z)} dP_1(y|Z)} \right) dP_1(y|Z). \end{aligned} \tag{3.4}$$

Equation 3.4 shows that the *functional of interest*, $\mathbb{E}Y = \mu$, can be estimated if we have estimators for P_1, f, and P_Z, and if we know the sensitivity function q. We could therefore specify priors for these elements. This is the first of two possible parameterisations for this model. The following lemma from [28] motivates the other possible parameterisation.

Lemma 3.1. *Let $(\mathcal{Y}, \mathscr{Y})$ be a Polish space. Given a measurable function $g : \mathcal{Y} \to [0, 1]$, a number $p \in (0, 1)$, and a probability distribution P_1 on $\mathcal{Y}$, there exists a law for a random vector (A, Y) with values in $\{0, 1\} \times \mathcal{Y}$ such that*

$$\Pr(A = 1|Y) = g(Y), \tag{3.5}$$

$$\Pr(A = 1) = p, \tag{3.6}$$

$$Y|A = 1 \sim P_1, \tag{3.7}$$

if and only if

$$\int \frac{1}{g} dP_1 = \frac{1}{p}. \tag{3.8}$$

[1] the analysis can be extended to include the other potential outcome Y^0 by instead applying the approach "twice" to the observed data $(A, AY^1, (1 - A)Y^0, Z)$

Moreover, this law is unique and satisfies

$$\Pr(Y \in B) = p \int_B \frac{1}{g} dP_1, \tag{3.9}$$

$$\Pr(Y \in B | A = 0) = \frac{p}{1-p} \int_B \frac{1}{g} dP_1, \tag{3.10}$$

for any $B \in \mathscr{Y}$.

If we apply lemma 3.1 conditionally on Z, we obtain the following:

Corollary 3.2. *If for some measurable function* $g : \mathcal{Y} \times \mathcal{Z} \to [0,1]$ *we have that*

$$\int \frac{1}{g(y,z)} dP_1(y|z) = \frac{1}{f(z)}, \quad \forall z \in \mathcal{Z}, \tag{3.11}$$

then

$$\Pr(A = 1 | Y = y, Z = z) = g(y,z). \tag{3.12}$$

This result allows us to model the probability of being observed given Y and Z as follows

$$\Pr(A = 1 | Y, Z) = \Psi(\eta(Z) + q(Y,Z)), \tag{3.13}$$

where $\Psi(x) = (1 + e^{-x})^{-1}$ is the logistic function. The function q in 3.13 is the same sensitivity function as in 3.3, and in fact lemma 3.1 shows the two equations to be equivalent. The function $\eta(Z)$ can be solved from 3.11, and it is interpreted as the intercept of the logistic regression of A on Y given Z. In light of all this, the second strategy for specifying priors is by putting independent priors on P (i.e. $\mathcal{L}(Y|Z)$), η, q, and P_Z. We will refer to this as the second parameterisation. Let's look at some examples of this approach in the literature.

[30, 10] apply the selection bias approach to a missing data problem without covariates. In the absence of covariates, the MAR assumption is called *missing completely at random* (MCAR). Both papers consider outcomes Y in $\mathbb{R}^+$, modelling CD4 counts of HIV patients, and both papers specify a sensitivity function $q(y) = \alpha \log(y)$. [10] investigate both parameterisations. For the first parameterisation, they specify the hierarchy

$$\begin{aligned} Y|A = 1 &\sim \text{DP}(a), \\ A|f &\sim \text{Ber}(f), \\ f &\sim \text{Be}(\beta_1, \beta_2). \end{aligned}$$

The distribution of Y is then given by $P = fP_1 + (1-f)P_0$. In the second parameterisation, the hierarchy becomes

$$\begin{aligned} A|Y &\sim \text{Ber}(\Psi(\eta + q_\alpha(Y))), \\ Y &\sim \text{DP}(a), \\ \eta &\sim \text{Unif}(u_1, u_2). \end{aligned}$$

For both parameterisations a prior on q is specified by placing a prior on α. In the first parameterisation the posterior computation is easy, because the posterior distribution of P is again a Dirichlet process. In the second parameterisation, a Gibbs sampler is used to sample from the posterior, and relation 3.3 is used to sample the missing outcomes.

An important result from [10], is that the second parameterisation is in some sense favourable over the first, because in the second parameterisation the posterior of q might differ from the prior, whereas in the first parameterisation the posterior of q is equal to the prior on q. This indicates, that the second parameterisation allows the posterior of q -and therefore the posterior of the functional of interest μ- to learn from the data, which instills the hope that part of the non-identifiable sensitivity parameter α can be learned from the data.

Another example that takes a similar modelling approach is [12]. They consider a class of models that they call ”logistic selection with mixtures of exponential families” (logistic-mEF models). Notably, BART and DP mixtures belong to this class. An example of a hierarchy that they use is

$$\begin{aligned} Y^a|A=a,Z &\sim \text{Nor}(\mu_a(Z),\sigma^2),\\ \mu_a(Z),\sigma^2|A=a,Z &\sim \text{BART}(Z|A=a),\\ A|Y^a,Z &\sim \text{Ber}(\Psi(\eta_a(Z)+q_a(Y^a))). \end{aligned}$$

Similar to the approach of [2, 7], no prior is specified on q, and again α is varied over a range of plausible values.

3.4. Directions for Research in Bayesian Sensitivity Analysis

Of the two discussed approaches to BSA, the selection bias approach offers the most opportunities for research, because the approach appears to still be less widely adopted than the unmeasured confounder approach. The approach of [30, 10] can be extended to the case that includes covariates. As [10] already note, this could be done by modelling the dependence of Y on Z through a dependent Dirichlet process, or through BART (similar to [12], but with a prior on q). Another option is to investigate survival outcomes with covariates -possibly in combination with independent right censoring. In this setting covariate-dependence can be introduced through the Cox model. Neither of the two approaches to BSA has many (if any) occurences in the literature. In [24] the unmeasured confounder approach is applied to the setting of mediation analysis. They specify fully parametric survival models. Their method could be applied to the missing data (or causal inference) and improved by using nonparametric Bayesian methods.

4

Methods

This chapter describes and motivates the methods used. First the statistical model is specified in section 4.1. Secondly, section 4.2 describes the steps of a Gibbs sampler to sample from the posterior distribution. Section 4.3 describes a way to extend the model and Gibbs sampler to handle independently right censored data. The chapter is concluded with a section that outlines the experiments of the simulation study.

4.1. Model

A Bayesian sensitivity analysis (BSA) is carried out to assess sensitivity of study conclusions under the MAR assumption, to violations of said assumption. To this end, the missing data are modelled as MNAR using equation 3.13 (or 3.3), following the second parameterisation discussed in section 3.3. We consider a model with life times $T \in \mathbb{R}^+$, and for simplicity we consider the case without right censoring. The full data are of the form $D_n = \{(\Delta_1, T_1, Z_1), \dots, (\Delta_n, T_n, Z_n)\}$, where Δ_i indicates whether the i-th life time T_i is observed[1] or not ($\Delta_i = 1$, respectively $\Delta_i = 0$). Z_i is the covariate characterising the i-th observation, and is for simplicity chosen to take its value in $(0, 1)$. The life times given the covariates are modelled through their c.h.f. through the Bayesian version of the Cox model, given by (2.26). The latter relation is used to find the conditional c.d.f. of T:

$$A(t|Z) = A(t)e^{\beta^\top Z} \iff S(t|Z) = S(t)^{\exp(\beta^\top Z)}$$
$$\iff F(t|\beta, H, Z) = 1 - \Big(\prod_{(0,t]}(1 - dH)\Big)^{\exp(\beta^\top Z)}.$$

A BP(c, Λ) prior is placed on the baseline c.h.f. H, and an independent improper uniform prior is placed on the regression coefficient β, following the choice in [14, p. 426]. The law of the covariates, P_Z, follows a DP(a) prior. The functions q and η in the sensitivity model are indexed by parameters α respectively γ, i.e. $q = q_\alpha$ and $\eta = \eta_\gamma$. These parameters are equipped with independent priors. The following Bayesian hierarchy summarises the generation of the full data:

- $P_Z \sim \text{DP}(a) \perp\!\!\!\perp \alpha \perp\!\!\!\perp \beta \perp\!\!\!\perp \gamma$,
- $Z_i | P_Z \overset{\text{i.i.d.}}{\sim} P_Z, \quad i = 1, \dots, n$,
- $H \sim \text{BP}(c, \Lambda)$,
- $F(t_i|\beta, H, Z_i) = 1 - \Big(\prod_{(0,t]}(1 - dH)\Big)^{\exp(\beta^\top Z_i)}, \quad i = 1, \dots, n$,
- $\Delta_i | \alpha, \gamma, T_i, Z_i \sim \text{Ber}(\Psi(\eta_\gamma(Z_i) + q_\alpha(T_i, Z_i))), \quad i = 1, \dots, n$,

with $\Psi(x) = (1 + e^{-x})^{-1}$. Since the covariates are restricted to $(0, 1)$, the standard uniform distribution is chosen as base measure a, so $\bar{a} = \text{Unif}(0, 1)$ and the prior precision $|a| = 1$, to allow the data to

[1] Δ takes the role of A from chapter 3, because A is used here to denote the negative logarithm of the survival distribution.

determine the posterior. The covariates might not be uniformly distributed in reality, but for increasing sample sizes the effect of the choice of base measure a vanishes. For a similar reason, the concentration function c is chosen equal to 1, to reflect that the prior belief in $\Lambda = t$ (corresponding to a constant hazard rate $\lambda = 1$) is the same for all t, and to allow the data to determine most of the posterior of H.

Given the full data, a strategy for sampling from the posterior can be formulated using the theory from chapter 2, as is done in the following section. In our case, however, the data are not fully observed. Only the data $X_n = \{(\Delta_1, \Delta_1 T_1, Z_1), \dots, (\Delta_n, \Delta_n T_n, Z_n)\}$ are observed[2]. To remedy this, we use theory from chapter 3 to augment the observed data by sampling values T_{miss} from the distribution of the missing data $\mathcal{L}(T|\Delta = 0, Z)$. The posterior quantities of interest are

- The sensitivity parameter α
- The functional of interest $\mu = \mathbb{E}T$

It will be interesting to investigate if (part of) the sensitivity parameter can be learned from the observed data, and if the posterior of the functional of interest will concentrate around its "true" value for increasing sample sizes.

4.2. Sampling from Posterior

Similar to [30, 10], we use a Gibbs sampling scheme to sample from the posterior distribution of (α,β,γ, H,P_Z, T_{miss}) given the data. This section describes the successive sampling steps of the algorithm. Each step generates a sample from a full conditional distribution. Most steps rely on theory and algorithms discussed in chapters 2 and 3. Details on implementation can be found in appendix B.

4.2.1. Dirichlet process

Since a priori $P_Z \sim \text{DP}(a)$, and we observe values $Z_1, \dots, Z_n$, the posterior distribution of P_Z given the data is a $\text{DP}(a + \sum_{i=1}^n \delta_{Z_i})$. We sample from the posterior distribution using an adaptation of Algorithm 3. $P_Z|Z_1, \dots, Z_n$ plays an important role in the computation of the functional of interest μ.

4.2.2. Imputation missing data

As stated in the previous section, we need full data for simulation from the posterior distribution. The missing values T_{miss} are distributed according to $P_0(t|\beta, H, Z)$. Assuming the sensitivity model is true, we can sample from this distribution using the following relationship:

$$dP_0(t|\beta, H, Z) \propto \frac{e^{-q(t,Z)}}{1 + e^{-\eta(Z)-q(t,Z)}} dF(t|\beta, H, Z). \tag{4.1}$$

Since at every iteration we draw a sample[3] from the marginal posterior of H that jumps at finitely many locations, we can see that $F(t|\beta, H, Z)$ is the c.d.f. associated with a discrete measure. Therefore $dF(t|\beta, H, Z)$ can be written as the sum of weights at certain locations. The locations are given by H, and the weights are the jumps $\Delta F(t|\beta, H, Z)$ at said locations, given by

$$\begin{aligned}\Delta F(t|\beta, H, Z) &= \Delta H(t|\beta, Z) \Big(\prod_{u \in (0,t)} (1 - \Delta H(u)) \Big)^{\exp(\beta^\top Z)} \\ &= \Big(1 - (1 - \Delta H(t))^{\exp(\beta^\top Z)}\Big) \Big(\prod_{u \in (0,t)} (1 - \Delta H(u)) \Big)^{\exp(\beta^\top Z)}.\end{aligned}$$

Then $P_0(t|\beta, H, Z)$ is just a re-weighting of $dF(t|\beta, H, Z)$, and thus also a discrete measure.

4.2.3. Cumulative hazard function

We specified a $\text{BP}(c, \Lambda)$ prior on the baseline c.h.f. H. It follows from theorem 2.19 that the posterior of H given β and the full data is an IIP with known intensity measure $\nu_{H|D_n,\beta} = \nu^c_{H|D_n,\beta} + \nu^d_{H|D_n,\beta}$.

[2] by convention we say we observe $T_i = 0$ if $\Delta_i = 0$, but in reality no value is observed for said T_i

[3] which is an approximation

Since in the previous step we imputed the T_{miss} to complete the observed data, we carry on as if we actually observed the full data. We sample from the beta process with intensity measure $\nu^c_{H|D_n,\beta}$ using algorithm 5. An adaptation of algorithm 6 is used to sample from the IIP with intensity measure $\nu^d_{H|D_n,\beta}$. The algorithm simulates jumps at fixed locations -the uncensored T values. In most applications, these locations are indeed fixed, but in our context the locations may vary per iteration, because the sampled T_{miss} may very well be different every iteration. Since the model does not allow for right censoring, all values T_i -including the imputed T_{miss}- are considered uncensored.

4.2.4. Metropolis-Hastings steps

We sample from the full conditional distributions of (α, γ) and β using MH steps. The likelihood of (α, γ) given the full data is the likelihood of a logistic regression, given by

$$\prod_{i=1}^{n} \left(\frac{1}{1 + \exp(-\eta_\gamma(Z_i) - q_\alpha(T_i, Z_i))} \right)^{\Delta_i} \left(\frac{1}{1 + \exp(\eta_\gamma(Z_i) + q_\alpha(T_i, Z_i))} \right)^{1-\Delta_i}.$$

The likelihood of β given H and the full data is given by theorem 2.20. For both MH steps we use a Gaussian proposal kernel with a variance that is tuned to achieve an acceptance rate[4] of between $\frac{1}{4}$ and $\frac{1}{3}$. During the first 99 iterations of the Gibbs sampler multiple MH steps are simulated (linearly decreasing from 100 to 2), to achieve a quicker burn-in. When multiple MH steps are taken during one iteration of the Gibbs sampler, only the last sample is used and the rest are discarded. Starting from the hundreth iteration of the Gibbs sampler, only one MH step is taken per iteration (per full conditional distribution).

4.2.5. Computation functional of interest

At every iteration an approximation of the functional of interest μ is computed. This is done by evaluating the integral

$$\mu \approx \iint t \; dF(t|\beta, H, Z) dP_Z, \tag{4.2}$$

writing P_Z instead of $P_Z|Z_1, \ldots, Z_n$. The double integral in 4.2 is easy to evaluate, since both integrals are with respect to discrete measures. For every given $Z \in \text{supp}(P_Z)$, the weights $\Delta F(t|\beta, H, Z)$ at $t : \Delta H(t) > 0$ are given by $W_{t|Z}$. Denoting the weights of P_Z by W_Z we can write the integral as

$$\mu \approx \sum_{Z \in \text{supp}(P_Z)} W_Z \left(\sum_{t:\Delta H(t)>0} W_{t|Z} \cdot t \right).$$

4.3. Extension to independent right censoring

The current model and sampling algorithm can be modified to be applicable to data with simultaneously both "ordinary" missingness and independent right censoring. Let the observed data be of the form $X_n = \{(\Delta_1^m, \Delta_1^m\Delta_1^c, \Delta_1^m T_1, Z_1), \ldots, (\Delta_n^m, \Delta_n^m\Delta_n^c, \Delta_n^m T_n, Z_n)\}$, where Δ^m is the missingness indicator, $T = \min(\tilde{T}, C)$ with C some independent censoring variable, and Δ^c is the right censoring indicator ($\Delta^c = 0$ means the value is right censored). Then almost all steps of the Gibbs sampler remain unchanged, since the beta process prior on H is compatible with independently right censored data, and the right censored data occur in the posteriors of H and β in a natural way.

A change needs to be made only with regard to the MH steps to sample from the full conditional of (α, γ) (or, more generally, (q, η)). The sensitivity function q is not designed to handle right censored data, as a right censored observation T underestimates the true value $\tilde{T}$. This can be remedied by adding an extra step to the Gibbs sampler. In the added step, the data are augmented by replacing the observed[5] right censored values (where $T = C$) by samples from the distribution of $T|\beta, H, Z$ conditional on the event $\{T \geq C\}$. Even if C is larger than all other observed uncensored life times, the distribution of $T|\beta, H, Z$ still has support to the right of C by virtue of the beta process part of the posterior of H.

[4] according to [13] an acceptance rate of 0.234 is optimal, but this is result is mainly interesting for high-dimensional targets: a slightly higher acceptance rate seems fine in practice for low-dimensional targets.

[5] we assume that missing data are never independently right censored. If a value is missing, it is imputed according to section 4.2.2

4.4. Experimental Design

This section outlines the experiments that are conducted with the aforementioned sampling strategy. First the simulation of data sets is described, then the various experiments are detailed.

4.4.1. Data generation

We run experiments on simulated data. Data sets $D_n = \{(\Delta_1, T_1, Z_1), \dots, \Delta_n, T_n, Z_n)\}$ of sizes $n = 100$, $n = 1000$, and $n = 10000$ are simulated under the assumption that the sensitivity model is true. The data sets are generated in the following way. We choose "true" values $\alpha_0 = 1.5$, $\beta_0 = 1$, $\gamma_0 = 2$, $\lambda_0 = 1$. Then we sample n i.i.d. realisations of

$$\begin{aligned} Z &\sim \text{Unif}(0,1), \\ T|Z &\sim \text{Exp}(\lambda_0 e^{\beta_0 Z}), \\ \Delta|T,Z &\sim \text{Ber}(\Psi(\eta_{\gamma_0}(Z) + q_{\alpha_0}(T,Z))), \end{aligned}$$

with $\Psi(x) = (1 + e^{-x})^{-1}$. The distribution of $T|Z$ is in accordance with the Cox model. The true value of the functional of interest is then

$$\mu_0 = \mathbb{E}_Z \mathbb{E}(T|Z) = \mathbb{E}_Z(\lambda_0^{-1} e^{-\beta_0 Z}) = \int_0^1 e^{-z} dz = 1 - \frac{1}{e} \approx 0.63.$$

For Experiments 1-3 the same data sets are used, with $\eta_\gamma(Z) = \gamma^\top Z$ and $q_\alpha(T,Z) = \alpha T$. This choice of q_α, in light of the logistic regression formulation, means that a unit increase in the value of T implies an increase of α in the log-odds of being observed. η_γ has a similar interpretation. The data for Experiment 4 are simulated slightly differently: first of all, a nonlinear sensitivity function $q_\alpha(T,Z) = \alpha T \mathbb{1}_{(0.5,1]}(Z)$ is used, and secondly, $\gamma_0 = 0.5$ to ensure enough data will end up missing ($\Delta = 0$). Thus the Z_i and $T_i|Z_i$ are the same for all experiments, but different $\Delta_i|T_i, Z_i$ are simulated for Experiment 4.

When the data sets are used in any of the experiments described in the following sections, all data points where $\Delta = 0$ are considered missing, so instead of D_n, observed data $X_n = \{(\Delta_1, \Delta_1 T_1, Z_1), \dots, (\Delta_n, \Delta_n T_n, Z_n)\}$ are used.

4.4.2. Experiments

Four experiments are carried out to test the model and the performance of the Gibbs sampling scheme. Unless mentioned otherwise

- the experiments are carried out on the data sets X_n for $n = 100$, $n = 1000$, and $n = 10000$, and
- the initial values of the Gibbs sampler are chosen to be the prior means.

In all experiments the posterior distributions of α and μ are the main objects of interest.

In Experiment 1 the sensitivity parameter is equipped with a normal prior, $\alpha \sim \text{Nor}(\alpha_0, 0.5)$. When the model is used for BSA, a domain expert would usually express their beliefs about selection bias in the form of such a prior. This experiment tests the case where the domain expert correctly specifies the prior, meaning that the prior mean is equal to the true value α_0. Similarly we specified a prior for $\gamma \sim \text{Nor}(\gamma_0, 0.5)$.

In Experiment 2 the sensitivity parameter is fixed to the true value, $\alpha = \alpha_0$. In this case the model is identifiable and thus the posteriors are expected to converge to point masses (at the true values) for increasing sample size, as was shown for the missing data model without covariates in [10].

Experiment 3 tests the performance of the model if the prior on α is misspecified. It can not always be expected that a domain expert correctly specifies the magnitude of selection bias. An important reason is that the sensitivity model is usually not true. This experiment assesses the ability of the posterior sensitivity function q to learn from the data. To this end the prior for α from Experiment 1 is changed to $\alpha \sim \text{Nor}(0.5, 0.5)$.

Experiment 4 repeats the first experiment with a different sensitivity function: $q_\alpha(T,Z) = \alpha T \mathbb{1}_{(0.5,1]}(Z)$. This q introduces an interaction between T and Z, in that the log-odds of being observed are increased by αT if $Z > 0.5$.

5 Results

5.1. Generated Data

The data sets were generated according to the hierarchy in section 4.4.1. Some descriptive statistics for the data sets are shown in Table 5.2. For Experiments 1-3 the data were generated with $q(t,z) = \alpha t$. The table shows that larger values for T are more likely to be observed, which is indeed the case because α_0 is positive. For Experiment 4 the data were generated with $q(t,z) = \alpha t \mathbb{1}_{(0.5,1]}(z)$. The table shows a smaller average value for observed T than for missing T. This again makes sense, since the cases where $Z > 0.5$ have a heightened chance of being observed, and since $\beta_0 = 1$ higher values for Z imply a larger rate parameter for the exponential distribution. The expected value of the exponential distribution is the inverse of the rate parameter, so this implies that mainly smaller T values have a heightened probability of being observed.

Statistics D_n	$n = 100$	$n = 1000$	$n = 10000$
Mean T	0.546	0.629	0.638
Mean $T\|\Delta = 1$ Exp. 1-3	0.576	0.679	0.690
Mean $T\|\Delta = 0$ Exp. 1-3	0.431	0.352	0.371
Number observed Exp. 1-3	79	844	8393
Mean $T\|\Delta = 1$ Exp. 4	0.538	0.579	0.620
Mean $T\|\Delta = 0$ Exp. 4	0.570	0.763	0.692
Number observed Exp. 4	76	732	7463

Table 5.1: Descriptive statistics for the generated data sets

5.2. Posterior Distributions

In all four experiments, the Gibbs sampler was run for 5000 iterations, which appears to be enough for convergence. We chose a burn-in period of 500 iterations, and thus the first 500 samples were discarded.

The results of Experiment 1 are summarised in Figure 5.2. For increasing n, the variance of the marginal posteriors decrease, but they are not always exactly concentrated around the true values. The marginal posteriors of α and γ appear correlated, as can be seen in Figure 5.1. Similarly, it can be seen that, for all sample sizes, if the posterior mode of α is slightly lower than $\alpha_0 = 1.5$, then the posterior mode of γ is slightly higher than γ_0 -and vice versa. Despite this fact, it appears that the functional of interest can be recovered pretty accurately: for all sample sizes the posterior concentrates around the sample mean of T in the full data D_n.

In Experiment 2 the sensitivity parameter is fixed $\alpha = \alpha_0$. It can be seen from Figure 5.3 that with this fixed value for α the posterior of γ does concentrate around the true value γ_0. No difference can be seen in the ability of the sampler to learn the correct functional of interest: for all sample sizes the posterior of μ looks the same in the first two experiments, with roughly the same 90% credible intervals (Table 5.2).

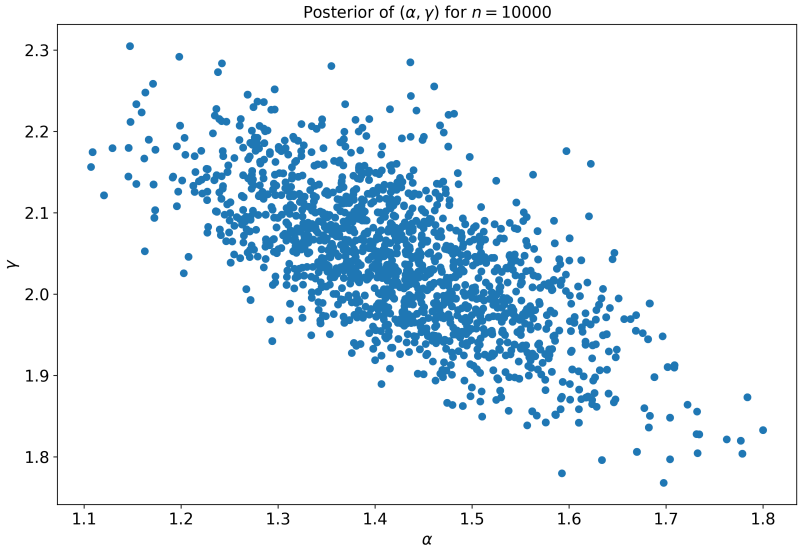

Figure 5.1: Posterior distribution of (α, γ) in Experiment 1. Same correlated behaviour is seen for all sample sizes -also in Experiments 3 and 4.

The results of Experiment 3 are shown in Figure 5.4. The misspecification of the prior on α is seen in the posterior of α for small n, but for increasing n the posterior resembles the one from the first experiment. The same dependence between the posteriors of α and γ is seen here. Even with a misspecified prior on α the functional of interest can be recovered accurately: Table 5.2 shows a slightly larger 90% credible interval for $n = 100$ in comparison with the other experiments, but for $n = 1000$ and $n = 10000$ this effect vanishes.

The results of Experiment 4 are shown in Figure 5.5. Though very similar to the results of the first experiment, it appears the Gibbs sampler learns the wrong value for β for $n = 1000$, causing an underestimation of the posterior of the functional of interest. For all sample sizes the posterior of the sensitivity parameter α concentrates around α_0.

90% credible interval μ	$n = 100$	$n = 1000$	$n = 10000$
Exp. 1: prior on α	$[0.470, 0.692]$	$[0.583, 0.660]$	$[0.628, 0.653]$
Exp. 2: fixed $\alpha = \alpha_0$	$[0.464, 0.664]$	$[0.593, 0.665]$	$[0.627, 0.650]$
Exp. 3: misspecified prior	$[0.487, 0.724]$	$[0.588, 0.663]$	$[0.628, 0.653]$
Exp. 4: nonlinear q	$[0.475, 0.686]$	$[0.550, 0.629]$	$[0.625, 0.654]$

Table 5.2: 90% credible intervals of the posterior of the functional of interest for all four experiments

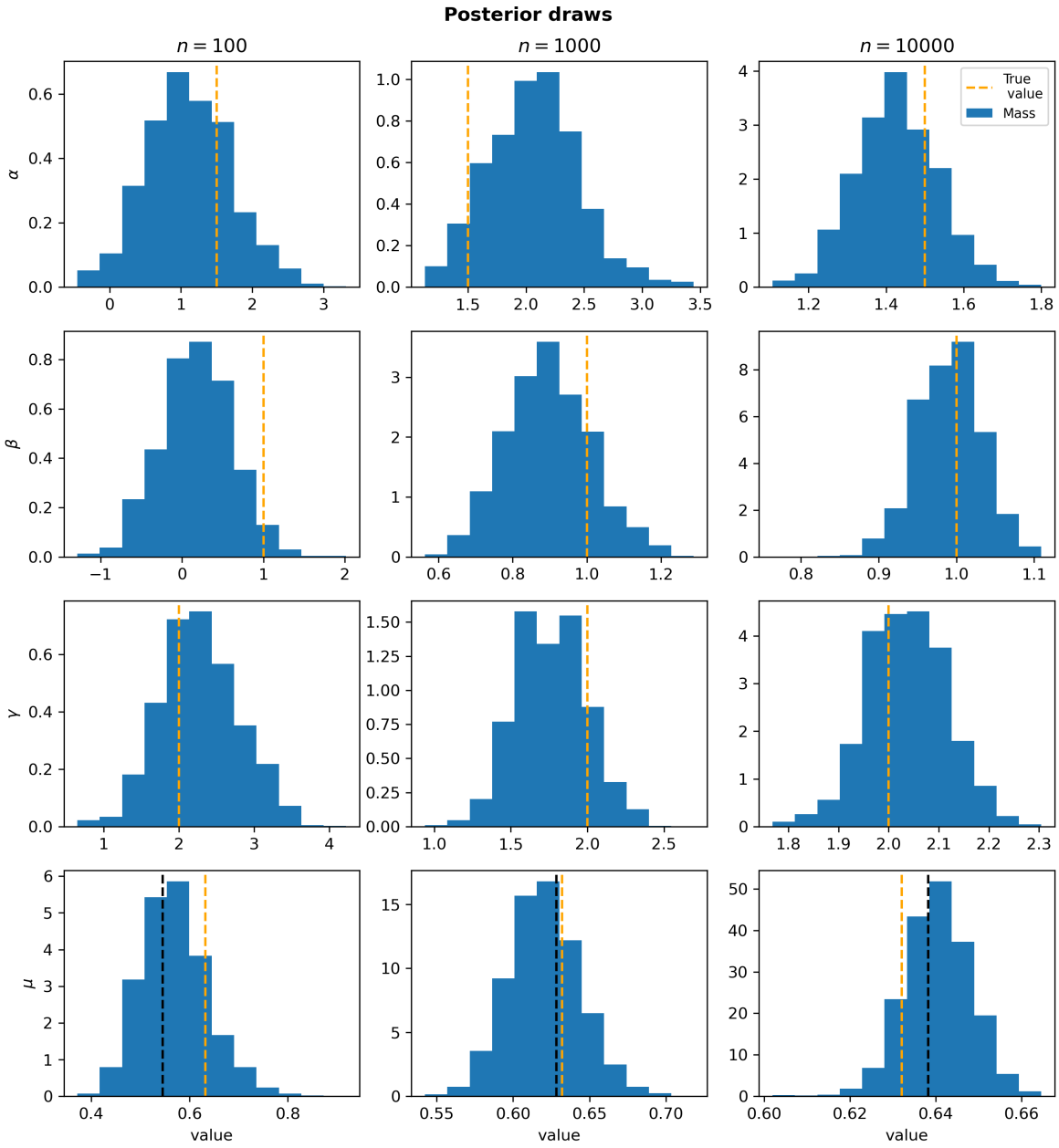

Figure 5.2: Marginal posteriors of α, β, γ, and μ in Experiment 1. The black dashed lines indicate the sample means of T of the full data.

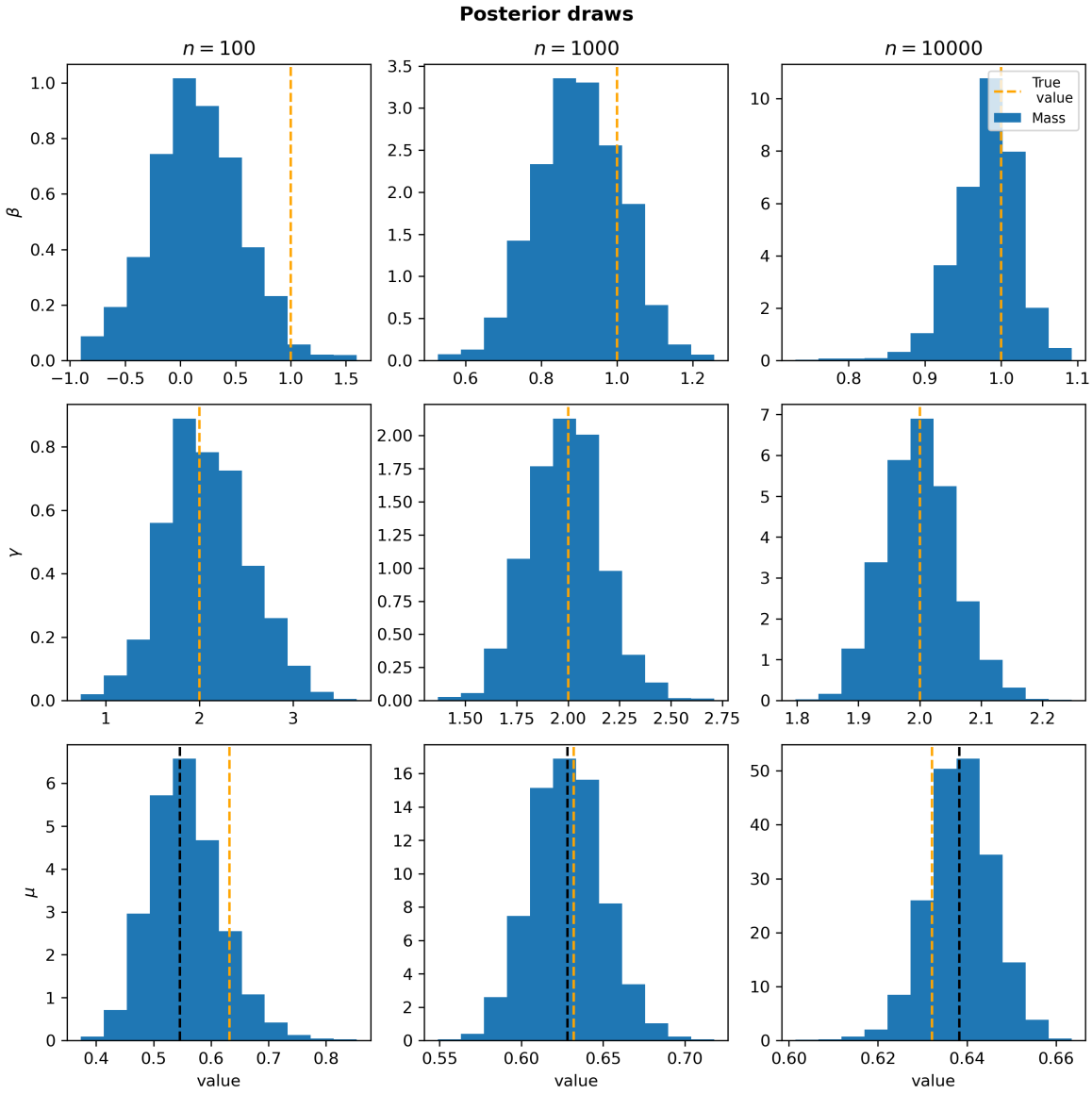

Figure 5.3: Marginal posteriors of α, β, γ, and μ in Experiment 2. The black dashed lines indicate the sample means of T of the full data.

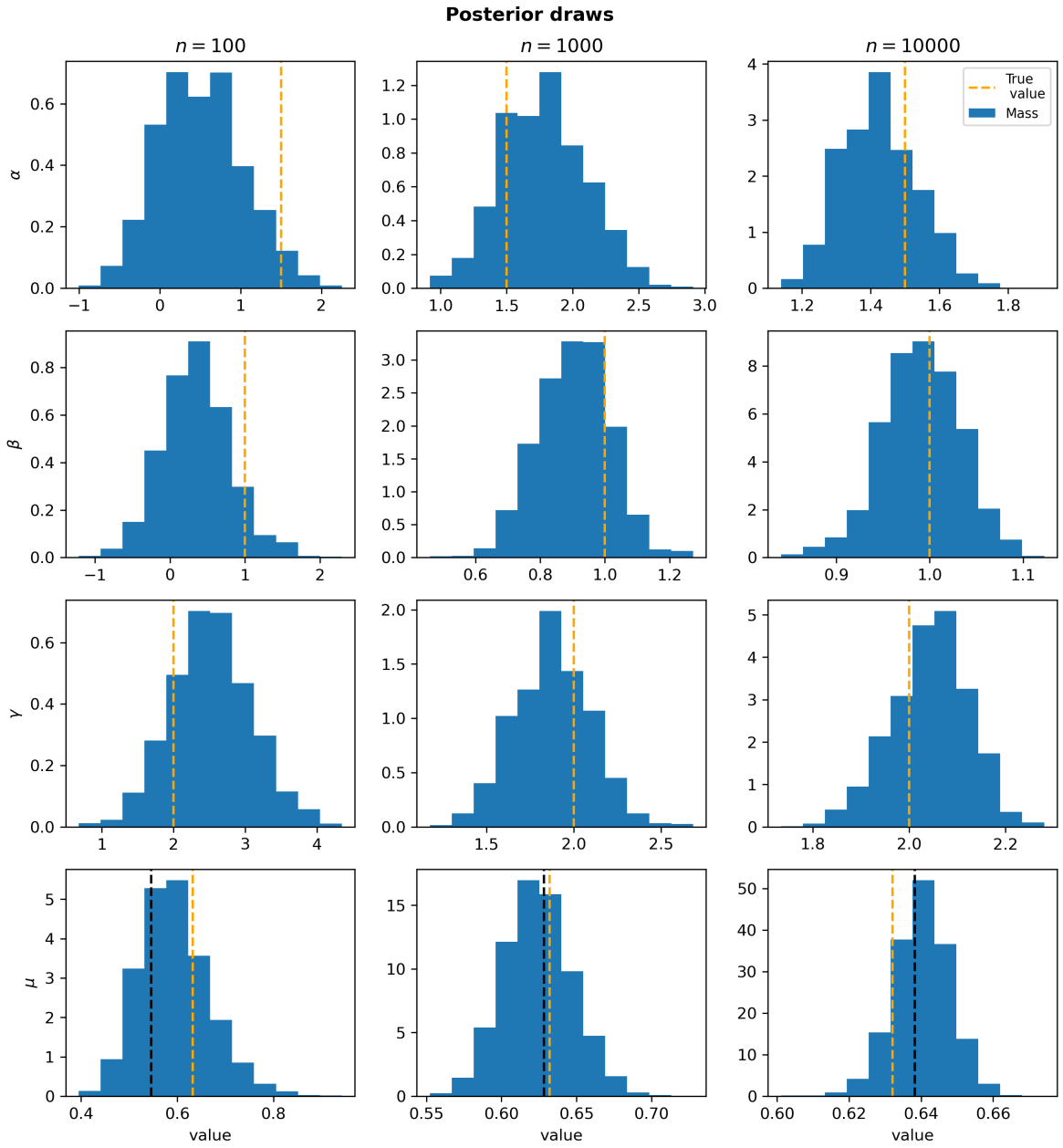

Figure 5.4: Marginal posteriors of α, β, γ, and μ in Experiment 3. The black dashed lines indicate the sample means of T of the full data.

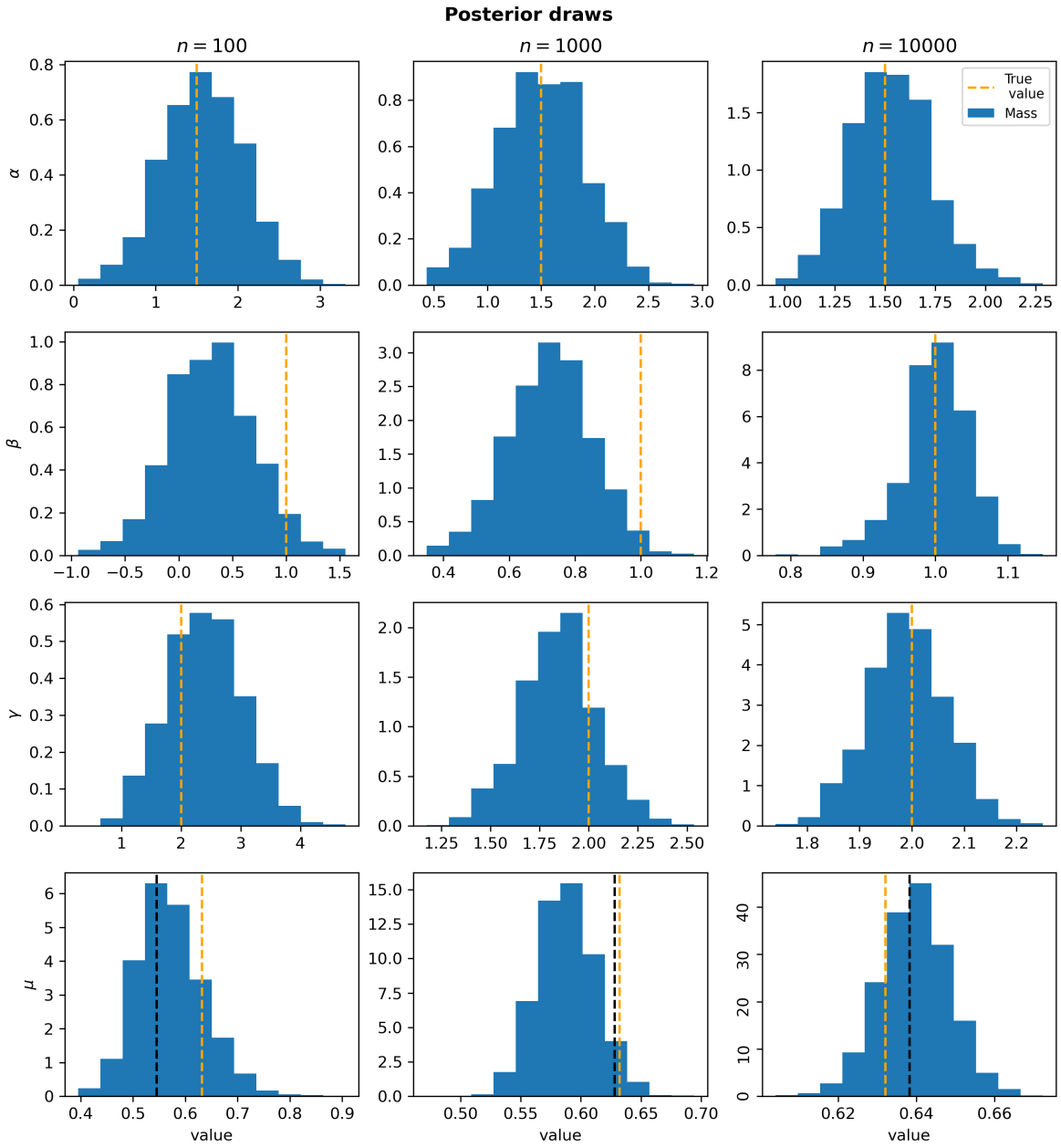

Figure 5.5: Marginal posteriors of α, β, γ, and μ in Experiment 4. The black dashed lines indicate the sample means of T of the full data.

6

Discussion

The results give confidence in the use of the studied model for Bayesian sensitivity analysis (BSA). The model is applicable to survival data with missing data, but without independent right censoring. A description was provided on how the current model can be extended so it can be applied to survival data with simultaneously both "ordinarily" missing data[1] and independent right censoring. It seems that the functional of interest μ can be accurately learned from the observed data. For large sample sizes the role of the prior on the sensitivity function q is shown to become less important. If the prior on the sensitivity function is misspecified, this results in a larger variance in the posterior of the functional of interest for smaller sample sizes, but this effect diminishes for increasing sample sizes. Another interesting result, is the fact the the 90% credible intervals of μ do not differ in length whether a(n incorrect) prior is specified for α, or whether α is fixed at its true value. It seems that we can conclude that our model can find the correct posterior (concentrated around the true values) for α and μ if the functional form of q is correctly specified. It is unclear from our results if in the case of a misspecified functional form of q the posterior of the functional of interest still concentrates around the true value. This is a topic for future research. In a similar vein, it can't be expected that the functional form of η can be correctly specified. Future research into this model might put a more flexible prior on η, such as a Gaussian process prior. If a Gaussian process prior is chosen for η the model should be adjusted by choosing $\Psi = \Phi$ (the *probit link function*), and applying MCMC algorithm proposed in [4].

In light of the findings of [10] in the case without covariates, our results are somewhat surprising. We would expect a prior on a non-identifiable sensitivity parameter to have quite a noticeable effect on the posterior of the functional of interest. Furthermore, we would expect the credible intervals to be smaller if the sensitivity parameter is fixed at its true value. The reason the current model performs better than expected when compared to the case with no covariates, could be attributed to the additional information that is present in the covariates. Indeed if $\alpha = \alpha_0$, the posterior of γ concentrates around γ_0, indicating that a large part of the missingness can be learned from Z. This needs to be tested in future research by reducing the amount of information in the covariates. This can be done by considering a less informative binary covariate, or leaving the covariates out of the model altogether, and investigating if similar results arise.

The dependence in the posteriors of α and γ paints a slightly distorted picture, but fortunately this effect does not seem to hamper the ability of the sampler to learn μ. Following [10] the dependence in the (α, γ) posterior could be remedied by centering the sensitivity function. If the sensitivity function takes the form $q_\alpha = \alpha g(t,z)$, then it might be properly centered by adjusting it so that $q_\alpha = \alpha(g(t,z) - \mathbb{E}_{P_0}[g(t,z)])$.

Although the tests in this thesis were carried out with a one-dimensional covariate, the current model can be applied to data with high-dimensional covariate vectors $Z \in \mathbb{R}^p$. For large p, the MH steps for sampling from the full conditional distributions of β and γ might suffer in efficiency. This can be remedied by either

- applying dimensionality reduction techniques, such as principal components analysis, to Z, or

[1] which can be seen as a harsh form of informative right censoring

- using a more sophisticated MCMC sampler.

An example is Hamiltonian MCMC [9], which is an adjustment of the MH algorithm that interprets the negative log target density as the potential energy of a particle, and generates new proposals by simulating the Hamiltonian dynamics of said particle. Another option is the No U-turn Sampler [16], which is an extension of Hamiltonian MCMC that automatically stops the simulation of the dynamics as soon as the simulated path starts to turn back on itself (for efficiency reasons). Both samplers show good performance on high-dimensional targets. Both samplers require the negative log target densities to be differentiable, which is the case for the present choice of (q, η), and partial likelihood of β.

This thesis carried out BSA to deviations from the MAR assumption. As was alluded to, the assumption of independent right censoring is another non-identifiable assumption that arises often in survival analysis. If this assumption is violated, we speak of informative right censoring. The current model might be extended to perform BSA to deviations from the independent right censoring assumption, by building on works like [31], who propose a censoring bias function q.

A

Probability Distributions

This section serves as a reference on probability distributions used in the main text.

Bernoulli Distribution

A random variable X follows a Bernoulli distribution, $X \sim \text{Ber}(\theta)$, if its p.m.f. is given by

$$p(x) = \theta^x (1-\theta)^{1-x} \mathbb{1}_{\{0,1\}}(x),$$

where $\theta \in (0,1)$. The mean and variance of X are given by

$$\begin{aligned} \mathbb{E}[X] &= \theta, \\ \text{var}[X] &= \theta(1-\theta). \end{aligned}$$

Beta Distribution

A random variable X follows a beta distribution, $X \sim \text{Be}(\alpha, \beta)$, if its p.d.f. is given by

$$p(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1} \mathbb{1}_{(0,1)}(x),$$

where $\Gamma(\alpha)$ denotes the Gamma function given by $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} dt$, and $\alpha, \beta > 0$. The mean and variance of X are given by

$$\begin{aligned} \mathbb{E}[X] &= \frac{\alpha}{\alpha+\beta}, \\ \text{var}[X] &= \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}. \end{aligned}$$

Dirichlet Distribution

A random variable $X = (X_1, \dots, X_k)$ follows a Dirichlet distribution, $X \sim \text{Dir}(k; \alpha)$, if its p.d.f. is given by

$$p(x) = \frac{\Gamma(\alpha_1 + \cdots + \alpha_k)}{\Gamma(\alpha_1) \dots \Gamma(\alpha_k)} \prod_{i=1}^{k} x_i^{\alpha_i - 1} \mathbb{1}_{\mathbb{S}_k}(x),$$

with $k \in \mathbb{N}\backslash\{1\}$, $\alpha = (\alpha_1, \dots, \alpha_k) > 0$, where $\Gamma(\alpha)$ denotes the Gamma function given by $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} dt$, and where $\mathbb{S}_k$ denotes the k-dimensional unit simplex, given by

$$\mathbb{S}_k = \left\{ s = (s_1, \dots, s_k) : s_j \geq 0, j \in \{1, \dots, k\}, \sum_{j=1}^{k} s_j = 1 \right\}.$$

Exponential Distribution

A random variable X follows an exponential distribution, $X \sim \text{Exp}(\lambda)$, if its p.d.f. is given by

$$p(x) = \lambda e^{\lambda x} \mathbb{1}_{\mathbb{R}^+}(x),$$

where $\lambda > 0$. The mean and variance of X are given by

$$\mathbb{E}[X] = \frac{1}{\lambda},$$
$$\text{var}[X] = \frac{1}{\lambda^2}.$$

Gamma Distribution

A random variable X follows a gamma distribution, $X \sim \text{Ga}(\alpha, \beta)$, if its p.d.f. is given by

$$p(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x} \mathbb{1}_{\mathbb{R}^+}(x),$$

where $\Gamma(\alpha)$ denotes the Gamma function given by $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} dt$, and $\alpha, \beta > 0$. The mean and variance of X are given by

$$\mathbb{E}[X] = \frac{\alpha}{\beta},$$
$$\text{var}[X] = \frac{\alpha}{\beta^2}.$$

Geometric Distribution

A random variable X follows a Geometric distribution, $X \sim (\theta)$, if its p.m.f. is given by

$$p(x) = (1-\theta)^{x-1} \theta \mathbb{1}_{\mathbb{N}}(x),$$

where $\theta \in (0, 1)$. The mean and variance of X are given by

$$\mathbb{E}[X] = \frac{1}{\theta},$$
$$\text{var}[X] = \frac{1-\theta}{\theta^2}.$$

Poisson Distribution

A random variable X follows a Poisson distribution, $X \sim \text{Pois}(\lambda)$, if its p.m.f. is given by

$$p(x) = \frac{\lambda^x e^{-\lambda}}{x!} \mathbb{1}_{\mathbb{N}_0}(x),$$

where $\lambda > 0$. The mean and variance of X are given by $\mathbb{E}[X] = \text{var}[X] = \lambda$.

Uniform Distribution

A random variable X follows a Uniform distribution, $X \sim \text{Unif}(a, b)$, if its p.d.f. is given by

$$p(x) = \frac{\mathbb{1}_{(a,b)}(x)}{b-a},$$

where $-\infty < a < b < \infty$. The mean and variance of X are given by

$$\mathbb{E}[X] = \frac{1}{2}(a+b),$$
$$\text{var}[X] = \frac{1}{12}(b-a)^2.$$

B

Python Code

This appendix contains the `python` code for the implementation of the Gibbs sampling scheme in 4.2.

```
# -*- coding: utf-8 -*-
"""
Created on Fri Dec 23 12:26:00 2022

@author: chris
"""
#%% Imports

import numpy as np
import numpy.random as rnd
import math
import time
import pandas as pd
import datetime as dt

#%% Functions

def sortData(Delta,T,Z):
    sortKey = T.argsort()
    Delta_sorted = Delta[sortKey]
    T_sorted = T[sortKey]
    Z_sorted = Z[sortKey]

    return Delta_sorted,T_sorted,Z_sorted


def expBetaZ(beta,Z):
    inproduct = beta@np.transpose(Z)
    terms = np.exp(inproduct)

    return terms


def q(t,z,alpha):
    n = z.shape[0]
    indicator = np.reshape(np.where(z>0.5,1,0),(n))
    return alpha*t*indicator # alpha*np.log(t)


def eta(z,gamma):
    # selection bias in Z
    return gamma@np.transpose(z)


def DP(Z_emp,prior_precision=1,J=10000):

    #initialise some auxilliary variables, and weights, locations, generate Beta r.v.'s for
                                                        stickbreaking
```

```
    n,p = Z_emp.shape[0], Z_emp.shape[1]
    B = rnd.beta(1, prior_precision + n, J)
    locations = np.zeros((J,p))
    B_ = np.ones(J) - B
    weights = np.zeros(J)

    # centre measure: Nor(mu,Sigma)
    #mu = np.zeros(p)
    #Sigma = np.eye(p)

    # stickbreaking procedure
    for j in range(J):
        U = rnd.uniform(0,1)
        if U < (prior_precision / (prior_precision + n)):
            locations[j] = rnd.uniform(0,1,p) # rnd.multivariate_normal(mu,Sigma)
        else:
            index = rnd.choice(n)
            locations[j] = Z_emp[index]
        weights[j] = B[j]*np.product(B_[0:j])

    weights = weights/np.sum(weights)

    # reduces size of arrays weights,locations
    groupedLocations, indices = np.unique(locations,return_inverse=True,axis=0)
    summedWeights = np.zeros(groupedLocations.shape[0])
    for i in range(groupedLocations.shape[0]):
        filteredIndices = np.where(indices==i)
        summedWeights[i] = np.sum(weights[filteredIndices])
    weights,locations = summedWeights,groupedLocations

    weights = weights/np.sum(weights)

    return (weights,locations)


def R_n(locations,T,eBetaZ,plus=False):
    # locations and T,Z should be sorted
    # plus is a boolean that selects either R_n (plus=False) or R_nPLus (plus=True)
    array = np.array([np.sum(eBetaZ[i:]) for i in range(eBetaZ.size)]) # = R_n(T)
    indices = np.array([-1])
    bigger_than_T = 0
    if locations.size > 0:
        if plus:
            indices = np.array([np.argmax(T>i) for i in locations])
            bigger_than_T = np.argmax(locations>=T[-1])
        else:
            indices = np.array([np.argmax(T>=i) for i in locations])
            bigger_than_T = np.argmax(locations>T[-1])
    R_n = array[indices]
    if (bigger_than_T != 0):
        R_n[bigger_than_T:] = 0

    return R_n


def concentration(t,k=1,b=0.5):
    # concentration function, parameter of a Beta Process
    array = k*np.exp(-b*t)

    return k # array


def BP(T,expBetaZ,eps=0.01,tau=6,k=1,b=0.5,a=1):
    # generating beta process with c(t)=k*e^{-b*t}, dA0(t) = a dt on [0,tau]
    # using Lee-Kim algorithm with epsilon.

    mu = a*k*tau/eps # (1-math.exp(-b*tau))*k*a/(eps*b)
    M = rnd.poisson(mu)
    jumpLocations = rnd.uniform(0,1,M)*tau # -np.log(1 -(1-math.exp(-b*tau))*rnd.uniform(0,1,
                                                M))/b
    jumpLocations = np.sort(jumpLocations)
```

```
    R_n_ = R_n(jumpLocations,T,expBetaZ)
    jumpSizes = rnd.beta(eps,R_n_ + concentration(jumpLocations,k,b))
    nearOne = rnd.uniform(1-eps,1,M)
    jumpSizes = np.where(jumpSizes<1,jumpSizes,nearOne)

    return (jumpSizes,jumpLocations)


def fixedJumps(T,eBetaZ,numIter=250):
    # MCMC loop for generating the discrete part of the posterior of the
    # cumulative hazard posterior in a Cox model.
    # The jumps occur at locations determined by the (completed/imputed) data.
    # T might have ties!

    jumpLocations,counts = np.unique(T,return_counts=True)
    cumCounts = np.append(np.zeros(1),np.cumsum(counts))
    numJumps = jumpLocations.size
    jumpSizes_init = rnd.uniform(0,0.1,numJumps)
    jumpSizes = jumpSizes_init
    R_nPlus = R_n(jumpLocations,T,eBetaZ,plus=True)

    for i in range(numIter):

        v = -np.log(1 - jumpSizes)
        y = rnd.geometric(1 - np.exp(-v))

        vTies = np.repeat(v,counts)
        const = -eBetaZ*vTies
        w = (np.log(1 + (np.exp(const) - 1)*rnd.uniform(0,1,T.size)))/const

        w_expBetaZ = w*eBetaZ
        summed_w_expBetaZ = np.zeros(jumpLocations.size)
        for i in range(jumpLocations.size):
            firstIndex = int(cumCounts[i])
            secondIndex = int(cumCounts[i+1]-1)
            summed_w_expBetaZ[i] = np.sum(w_expBetaZ[firstIndex:secondIndex])

        scaleGamma = 1/(concentration(jumpLocations) + R_nPlus + y + summed_w_expBetaZ)
        scaleGamma = np.where(scaleGamma > 0, scaleGamma,1)
        v = rnd.gamma(counts+1,scaleGamma)
        jumpSizes = 1 - np.exp(-v)

        # the following lines are a remedy to a very rare occurrence
        # in simulations, where jumps of size 1 are generated
        nearOne = rnd.uniform(1-0.01,1,numJumps)
        jumpSizes = np.where(jumpSizes<1,jumpSizes,nearOne)

    return (jumpSizes,jumpLocations,counts,jumpSizes_init)


def PgivenZ(H,expBetaZ):

    # extract jump sizes and locations from H
    jumpSizesC,jumpLocationsC,jumpSizesD,jumpLocationsD = H[0],H[1],H[2],H[3]
    jumpSizes = np.append(jumpSizesC,jumpSizesD)
    locations = np.append(jumpLocationsC,jumpLocationsD)

    #sort jump sizes and locations
    sortKey = np.argsort(locations)
    jumpSizes,locations = jumpSizes[sortKey],locations[sortKey]
    jumpSizes[-1] = 1

    oneMinusJumps = 1 - jumpSizes
    cumprod = np.cumprod(oneMinusJumps)
    S_base = cumprod
    N_jumps = locations.size
    weights = np.zeros((expBetaZ.size,N_jumps))
    for i in range(expBetaZ.size):
        S = S_base**(expBetaZ[i])
        F = 1 - S
```

```
        F_weights = np.diff(F)
        weights[i] = np.append(F[0],F_weights)

    return (weights,locations)


def alt_logLikelihood_beta(beta,T,Z,H):
    print(beta)
    eBetaZ = expBetaZ(beta,Z)
    pGivenZ = PgivenZ(H,eBetaZ)
    weights,locations = pGivenZ[0],pGivenZ[1]
    indices = np.array([np.argwhere(locations==t) for t in T])

    fetchedWeights = np.zeros(Z.size)
    for i in range(Z.size):
        fetchedWeights[i] = weights[i,indices[i]]

    logL_beta = np.sum(np.log(fetchedWeights))
    print(logL_beta)

    return logL_beta


def P0givenZ(H,alpha,beta,gamma,Z_miss):
    # make P0(t|Z) using relationship
    # dP0(t|Z) \propto exp(-q)/(1+exp(-eta - q)) * dP(t|Z)

    eBetaZ = expBetaZ(beta,Z_miss)
    weights,locations = PgivenZ(H,eBetaZ)
    for i in range(Z_miss.shape[0]):
        numerator = np.exp(-q(locations,Z_miss[i],alpha))
        denominator = (1 + np.exp(-eta(Z_miss[i],gamma) - q(locations,Z_miss[i],alpha)))
        scaleFactors = numerator/denominator
        scaledWeights = weights[i]*scaleFactors
        sumScaledWeights = np.sum(scaledWeights)
        weights[i] = scaledWeights/sumScaledWeights

    return weights,locations


def SampleMissingT(H,alpha,beta,gamma,Z_miss):
    # sample missing T using P0(t|Z)

    weights,locations = P0givenZ(H,alpha,beta,gamma,Z_miss)
    T_imputed = np.zeros(Z_miss.shape[0])

    for i in range(Z_miss.shape[0]):
        T_imputed[i] = rnd.choice(locations,p=weights[i])

    return T_imputed


def logPrior_beta(beta):

    # specify prior mean mu
    mu = np.array([1])

    #specify prior precision SigmaInverse
    sigma2 = 0.25
    SigmaInverse = np.eye(beta.size)/sigma2

    expression = -(np.transpose(beta - mu)@SigmaInverse@(beta - mu))/2

    return expression


def logPrior_alpha_gamma(x,alphaFixed=False):

    # specify prior mean mu for all parameters
    if not alphaFixed:
        mu_alpha = np.array([1.5]) # 1.5 / 0.5
```

```
    mu_gamma = np.array([2])
    if alphaFixed:
        mu = mu_gamma
    else:
        mu = np.hstack((mu_alpha,mu_gamma))

    # specify prior precision SigmaInverse for all parameters
    sigma2 = 0.5
    SigmaInverse = np.eye(x.size)/sigma2

    #expression = -(np.transpose(x - mu)@SigmaInverse@(x - mu))/2
    lower_gamma, upper_gamma = -3,7
    if not alphaFixed:
        alpha,gamma = x[0],x[1]
        logPrior_alpha = -(alpha - mu_alpha[0])**2 / sigma2
    else:
        gamma = x[0]

    if (gamma < upper_gamma) and (gamma > lower_gamma):
        logPrior_gamma = 0
    else:
        logPrior_gamma = -1e50

    expression = logPrior_gamma
    if not alphaFixed:
        expression += logPrior_alpha

    return expression


def logLikelihood_beta(beta,T,Z,H):

    jumpSizesC,jumpLocationsC,jumpSizesD_unique,jumpLocationsD_unique,counts = H[0],H[1],H[2]
                                                    ,H[3],H[4]
    eBetaZ = expBetaZ(beta,Z)

    #print(beta)

    # resize array with fixed jump sizes
    jumpSizesD = np.repeat(jumpSizesD_unique,counts)
    #jumpLocationsD = np.repeat(jumpLocationsD_unique,counts)

    # calculate R_n and R_nPlus
    R_n_ = R_n(jumpLocationsC,T,eBetaZ)
    R_nPlus = R_n(jumpLocationsD_unique,T,eBetaZ,plus=True)

    # calculate likelihood elements
    logL_betaC = np.sum(R_n_*np.log(1 - jumpSizesC))
    logL_betaD = np.sum(np.log(1 - (1 - jumpSizesD)**eBetaZ)) + np.sum(R_nPlus*np.log(1 -
                                                    jumpSizesD_unique))

    return logL_betaC + logL_betaD


def logLikelihood_alpha_gamma(x,Delta,T,Z,alphaFixed=False,fixed_alpha=0):

    # unpack arguments
    if alphaFixed:
        alpha = fixed_alpha
        gamma = x
    else:
        alpha,gamma = x[0],x[1:]

    argument = eta(Z,gamma) + q(T,Z,alpha)
    logL_alpha_gamma = np.sum(-Delta*np.log(1+np.exp(-argument)) - (1 - Delta)*np.log(1+np.
                                                    exp(argument)))

    return logL_alpha_gamma


def MH_alpha_gamma(previous,Delta,T,Z,recyclables=False,alphaFixed=False,fixed_alpha=0):
```

```
    # Implements the Metropolis-Hastings algorithm to sample from the
    # posterior of (alpha,gamma)|full (completed) data

    # proposal kernel
    sigma2 = 1 # 0.1K: 1; 1K: 0.2 ; 10K: 0.01 (0.08)
    Sigma = sigma2*np.eye(previous.size)
    proposal = rnd.multivariate_normal(previous,Sigma)

    # calculate acceptance probability, use recyclables from previous iteration
    if type(recyclables)=='tuple':
        logPrior_previous = recyclables[0]
        logLikelihood_previous = recyclables[1]
    else:
        logPrior_previous = logPrior_alpha_gamma(previous,alphaFixed)
        logLikelihood_previous = logLikelihood_alpha_gamma(previous,Delta,T,Z,alphaFixed,
                                                           fixed_alpha)

    logPrior_proposal = logPrior_alpha_gamma(proposal,alphaFixed)
    logLikelihood_proposal = logLikelihood_alpha_gamma(proposal,Delta,T,Z,alphaFixed,
                                                       fixed_alpha)

    logAcceptance = logPrior_proposal - logPrior_previous + logLikelihood_proposal -
                                                       logLikelihood_previous
    accepted = (math.log(rnd.uniform(0,1)) < logAcceptance)

    if accepted:
        return proposal, accepted, (logPrior_proposal,logLikelihood_proposal)
    else:
        return previous, accepted, (logPrior_previous,logLikelihood_previous)


def MH_beta(previous,Delta,T,Z,H,recyclables=False):
    # Implements the Metropolis-Hastings algorithm to sample from the
    # posterior of beta|H,full (completed) data

    # proposal kernel
    sigma2 = 0.4 # 0.1K: 0.4 ; 1K: 0.05 ; 10K: 0.005
    Sigma = sigma2*np.eye(previous.size)
    proposal = rnd.multivariate_normal(previous,Sigma)

    # calculate acceptance probability, use recyclables from previous iteration
    if type(recyclables)=='tuple':
        logPrior_previous = recyclables[0]
    else:
        logPrior_previous = logPrior_beta(previous)

    logLikelihood_previous = logLikelihood_beta(previous,T,Z,H)
    logPrior_proposal = logPrior_beta(proposal)
    logLikelihood_proposal = logLikelihood_beta(proposal,T,Z,H)

    logAcceptance =  logLikelihood_proposal - logLikelihood_previous # + logPrior_proposal -
                                                       logPrior_previous
    #acceptance = math.exp(logAcceptance)
    accepted = (math.log(rnd.uniform(0,1)) < logAcceptance)

    if accepted:
        return proposal, accepted, (logPrior_proposal,)
    else:
        return previous, accepted, (logPrior_previous,)



def integralT(Pz,H,beta):

    weightsZ,locationsZ = Pz[0],Pz[1]
    eBetaZ = expBetaZ(beta,locationsZ)
    weightsT,locationsT = PgivenZ(H,eBetaZ)
    inproduct = weightsT@locationsT
    ET = weightsZ@inproduct

    return ET
```

```
#%% Main

def main():

    rnd.seed(1)

    #enter & initialise parameters
    numIter = 5000 # 5000
    MH_iter = 100 # 100
    alpha_0 = 1.5 # 1.5 / 0.5
    beta_0 = np.array([1])
    gamma_0 = np.array([2])
    alphaFixed = False
    resultsFile = "results_n100_Z1d_exponential_qNonlinear_Par2.csv"
    dataFile = "fullData_n100_Z1d_exponential_qNonlinear_Par2.csv"

    #load & process data
    df = pd.read_csv(dataFile)
    data = df.to_numpy()
    n,p = data.shape[0],data.shape[1]-3
    Delta,T_full,Z = data[:,1],data[:,2],data[:,3:]
    Z = np.reshape(Z,(n,p))

    observed = np.where(Delta==1)
    missing = np.where(Delta==0)
    Delta_obs = Delta[observed]
    Delta_miss = Delta[missing]
    T_obs = T_full[observed]
    T_miss = T_full[missing]
    Z_obs = Z[observed]
    Z_miss = Z[missing]
    N_miss = T_miss.size

    ET_0 = np.mean(T_obs)

    #run Gibbs sampling scheme:

    # initialise lists to track parameter values

    alphaList = [alpha_0]
    betaList = [beta_0]
    gammaList = [gamma_0]

    if alphaFixed:
        sampleMH_alpha_gamma = gammaList[0]
    else:
        sampleMH_alpha_gamma = np.hstack((np.array([alphaList[0]]),gammaList[0]))
    sampleMH_beta = betaList[0]

    recyclables_alpha_gamma = False
    recyclables_beta = False

    ET = [ET_0]
    N_acceptedMH_alpha_gamma = 0
    N_acceptedMH_beta = 0
    N_MH_alpha_gamma = 0
    N_MH_beta = 0

    # start timer
    tic = time.perf_counter()

    for i in range(numIter):

        # 1. Sample Pz|Z ~ DP(a + n*Z_emp)
        Pz = DP(Z)

        # 2. Sample missing T from P0(t|Z)
        if i==0:
            T_imp = rnd.choice(T_obs,N_miss) # first imputation of missing T from empirical
```

```
                                                    distribution given by T_obs
        else:
            if alphaFixed:
                T_imp = SampleMissingT(H,alpha_0,betaList[i],gammaList[i],Z_miss)
            else:
                T_imp = SampleMissingT(H,alphaList[i],betaList[i],gammaList[i],Z_miss)

        Delta = np.append(Delta_obs,Delta_miss)
        T = np.append(T_obs,T_imp)
        Z = np.append(Z_obs,Z_miss,axis=0)
        Delta, T, Z = sortData(Delta,T,Z)
        eBetaZ = expBetaZ(betaList[i],Z)

        # 3. Sample H|beta,data
        jumpSizesC,jumpLocationsC = BP(T,eBetaZ)
        jumpSizesD,jumpLocationsD,counts,jumpSizes_initD = fixedJumps(T,eBetaZ)
        H = (jumpSizesC,jumpLocationsC,jumpSizesD,jumpLocationsD,counts)

        # 4. Sample (alpha,beta,gamma) with MH steps
        for j in range(max(MH_iter - i, 1)):
            sampleMH_alpha_gamma,accepted_alpha_gamma,recyclables_alpha_gamma =
                                                            MH_alpha_gamma(
                                                            sampleMH_alpha_gamma, Delta, T, Z,
                                                            False,alphaFixed,alpha_0)
            N_acceptedMH_alpha_gamma += accepted_alpha_gamma
            N_MH_alpha_gamma += 1

        if not alphaFixed:
            alphaList.append(sampleMH_alpha_gamma[0])
        gammaList.append(sampleMH_alpha_gamma[-p:])

        for k in range(max(MH_iter - i, 1)):
            #print("iteration " + str(k))
            sampleMH_beta,accepted_beta,recyclables_beta = MH_beta(sampleMH_beta,Delta,T,Z,H,
                                                            recyclables_beta)
            N_acceptedMH_beta += accepted_beta
            N_MH_beta += 1

        betaList.append(sampleMH_beta)

        # 5. Compute ET (functional of interest)
        ET.append(integralT(Pz,H,betaList[i+1]))


        # print progress
        if ((i+1)%100 == 0):
            print(str(i+1) + " completed.")
            toc = time.perf_counter()
            print("Time elapsed: " + str(dt.timedelta(seconds=round(toc-tic))))
            print("Avg. MH acceptance ratio (alpha,gamma): " + str(round(
                                                            N_acceptedMH_alpha_gamma /
                                                            N_MH_alpha_gamma,3)))
            print("Avg. MH acceptance ratio beta: " + str(round(N_acceptedMH_beta / N_MH_beta
                                                            ,3)))

            # save data to csv
            if not alphaFixed:
                alpha = np.array(alphaList)
                alpha = np.reshape(alpha,(i+2,1))
            beta = np.array(betaList)
            beta = np.reshape(beta,(i+2,1))
            gamma = np.array(gammaList)
            gamma = np.reshape(gamma,(i+2,1))
            ETarray = np.array(ET)
            ETarray = np.reshape(ETarray,(i+2,1))
            if alphaFixed:
                results = np.hstack((beta,gamma,ETarray))
            else:
                results = np.hstack((alpha,beta,gamma,ETarray))
            pd.DataFrame(results).to_csv(resultsFile)
```

```
if __name__ == '__main__':
    main()
else:
    print(”File is not run as main script”)
```

Bibliography

[1] Odd Aalen, Ornulf Borgan, and Hakon Gjessing. *Survival and event history analysis: a process point of view*. Springer Science & Business Media, 2008.

[2] Nicole Bohme Carnegie, Masataka Harada, and Jennifer L Hill. “Assessing sensitivity to unmeasured confounding using a simulated potential confounder”. In: *Journal of Research on Educational Effectiveness* 9.3 (2016), pp. 395–420.

[3] Hugh A Chipman, Edward I George, and Robert E McCulloch. “BART: Bayesian additive regression trees”. In: (2010).

[4] Nidhan Choudhuri, Subhashis Ghosal, and Anindya Roy. “Nonparametric binary regression using a Gaussian process prior”. In: *Statistical Methodology* 4.2 (2007), pp. 227–243.

[5] Jerome Cornfield et al. “Smoking and lung cancer: recent evidence and a discussion of some questions”. In: *Journal of the National Cancer institute* 22.1 (1959), pp. 173–203.

[6] David R Cox. “Regression models and life-tables”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 34.2 (1972), pp. 187–202.

[7] Vincent Dorie et al. “A flexible, interpretable framework for assessing sensitivity to unmeasured confounding”. In: *Statistics in medicine* 35.20 (2016), pp. 3453–3470.

[8] Hani Doss. “Bayesian nonparametric estimation for incomplete data via successive substitution sampling”. In: *The Annals of Statistics* (1994), pp. 1763–1786.

[9] Simon Duane et al. “Hybrid monte carlo”. In: *Physics letters B* 195.2 (1987), pp. 216–222.

[10] Bart Eggen, Stéphanie L. van der Pas, and Aad W. van der Vaart. *Bayesian sensitivity analysis for a missing data model*. 2023. arXiv: 2305.06816 [math.ST].

[11] Thomas S Ferguson. “A Bayesian analysis of some nonparametric problems”. In: *The annals of statistics* (1973), pp. 209–230.

[12] Alexander M Franks, Alexander D’Amour, and Avi Feller. “Flexible sensitivity analysis for observational studies without observable implications”. In: *Journal of the American Statistical Association* (2019).

[13] Andrew Gelman, Walter R Gilks, and Gareth O Roberts. “Weak convergence and optimal scaling of random walk Metropolis algorithms”. In: *The annals of applied probability* 7.1 (1997), pp. 110–120.

[14] Subhashis Ghosal and Aad Van der Vaart. *Fundamentals of nonparametric Bayesian inference*. Vol. 44. Cambridge University Press, 2017.

[15] W Keith Hastings. “Monte Carlo sampling methods using Markov chains and their applications”. In: (1970).

[16] Matthew D Hoffman, Andrew Gelman, et al. “The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo.” In: *J. Mach. Learn. Res.* 15.1 (2014), pp. 1593–1623.

[17] Marshall M Joffe. *Confounding by indication: the case of calcium channel blockers*. 2000.

[18] John Kingman. “Completely random measures”. In: *Pacific Journal of Mathematics* 21.1 (1967), pp. 59–78.

[19] Purushottam W Laud, Paul Damien, and Adrian FM Smith. *Bayesian nonparametric and covariate analysis of failure time data*. Springer, 1998.

[20] Jaeyong Lee and Yongdai Kim. “A new algorithm to generate beta processes”. In: *Computational statistics & data analysis* 47.3 (2004), pp. 441–453.

[21] Danyu Y Lin, Bruce M Psaty, and Richard A Kronmal. “Assessing the sensitivity of regression results to unmeasured confounders in observational studies”. In: *Biometrics* (1998), pp. 948–963.

[22] Lawrence C McCandless and Paul Gustafson. “A comparison of Bayesian and Monte Carlo sensitivity analysis for unmeasured confounding”. In: *Statistics in medicine* 36.18 (2017), pp. 2887–2901.

[23] Lawrence C McCandless, Paul Gustafson, and Adrian Levy. “Bayesian sensitivity analysis for unmeasured confounding in observational studies”. In: *Statistics in medicine* 26.11 (2007), pp. 2331–2347.

[24] Lawrence C McCandless and Julian M Somers. “Bayesian sensitivity analysis for unmeasured confounding in causal mediation analysis”. In: *Statistical Methods in Medical Research* 28.2 (2019), pp. 515–531.

[25] Lawrence C McCandless et al. “Hierarchical priors for bias parameters in Bayesian sensitivity analysis for unmeasured confounding”. In: *Statistics in medicine* 31.4 (2012), pp. 383–396.

[26] Nicholas Metropolis et al. “Equation of state calculations by fast computing machines”. In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092.

[27] Arman Oganisian and Jason A Roy. “A practical introduction to Bayesian estimation of causal effects: Parametric and nonparametric approaches”. In: *Statistics in medicine* 40.2 (2021), pp. 518–551.

[28] James M Robins, Andrea Rotnitzky, and Daniel O Scharfstein. “Sensitivity analysis for selection bias and unmeasured confounding in missing data and causal inference models”. In: *Statistical models in epidemiology, the environment, and clinical trials*. Springer, 2000, pp. 1–94.

[29] Donald B Rubin. “Inference and missing data”. In: *Biometrika* 63.3 (1976), pp. 581–592.

[30] Daniel O Scharfstein, Michael J Daniels, and James M Robins. “Incorporating prior beliefs about selection bias into the analysis of randomized trials with missing outcomes”. In: *Biostatistics* 4.4 (2003), pp. 495–512.

[31] Daniel O Scharfstein and James M Robins. “Estimation of the failure time distribution in the presence of informative censoring”. In: *Biometrika* 89.3 (2002), pp. 617–634.

[32] Jayaram Sethuraman. “A constructive definition of Dirichlet priors”. In: *Statistica sinica* (1994), pp. 639–650.

[33] Botond Szabó and Aad van der Vaart. *Bayesian Statistics (lecture notes)*. Sept. 2022.