UiO : Department of Mathematics
University of Oslo

# Multiple imputation for Cox regression with sampled cohort data

**Aleksander Njøs**

Master’s Thesis, Spring 2020

This master's thesis is submitted under the master's programme *Stochastic Modelling, Statistics and Risk Analysis*, with programme option *Statistics*, at the Department of Mathematics, University of Oslo. The scope of the thesis is 30 credits.

The front page depicts a section of the root system of the exceptional Lie group $E_8$, projected into the plane. Lie groups were invented by the Norwegian mathematician Sophus Lie (1842–1899) to express symmetries in differential equations and today they play a central role in various parts of mathematics.

# Abstract

In nested case-control and case-cohort studies of time-to-events, covariate information is collected for all individuals in the sampled cohort. Often information on some of the covariates are easily available for the entire cohort while some can only be collected for a limited amount of individuals; those in the sampled cohort. Multiple imputation, an algorithm for handling missing data, can be used to impute ("fill inn") covariate values, that have not been collected for individuals in the remaining part of the cohort, a small to moderately number of times. Then, Cox regression estimates from each imputed dataset (cohort) can be combined according to Rubin's rules. Multiple imputation used in this setting has previously been shown to give more efficient inferences by utilising more of the available information outside the sampled cohorts. However, in studies with very large cohorts, multiple imputation for the entire cohort might be very demanding or even infeasible.

In this thesis, existing methods for multiple imputation of missing values (by chance) in sampled cohort studies, in their original and an adapted form, are used to impute values in a superset of the sampled cohort. Imputing values missing by design in the superset motivates estimating the regression coefficients with nested case-control or case-cohort estimators. The results from simple simulations experiments show good performance with respect to bias and efficiency. For very large cohorts, the number of controls in a nested case-control superset or the size of the subcohort in a case-cohort superset, determines the size of the part of the cohort that is to be imputed, and superset imputation therefore looks like a promising method when imputation of the entire cohort is not possible.

# Acknowledgements

First, I would like to thank my supervisor, Ørnulf Borgan, for all the help I have received and for coming up with an interesting problem for this thesis. I feel lucky to be one of the many students to have benefited from his enthusiastic teaching and excellent guidance. Further, I would like to thank professors, teachers, student assistants, and fellow students for creating and contributing to engaging atmospheres of learning throughout my years of education. Along the way, I am also grateful to have had the opportunity to apply some of what I have learned in summer jobs. To see how knowledge is used in the world outside university has been very motivating for me. I would also like to thank my friends for enriching life inside and outside studies. Lastly and most importantly I would like to thank my family, especially my parents, for their love and for always supporting me.

Aleksander Njøs,
Oslo, May 2020

# Contents

# CHAPTER 1

# Introduction

The time it takes to occurrences of events is of interest in many applications, e.g. medicine, econometrics, biology, social sciences. To estimate the relationship between the time-to-event and relevant covariates, Cox (proportional hazards) regression is much used. For Cox regression, at each time of an event, the covariates of the individual who experiences the event is compared to the covariates of the other individuals who are still under observation (i.e. at risk) in the cohort. In large cohorts studies the event of interest might only happen to a small proportion of the individuals and the collection of covariate information for all individuals in the cohort might be impossible or very demanding.

Sampled cohorts studies offer a solution to this by comparing the covariates of the individuals who experience the event to a sample of the individuals at risk in the cohort. Then covariate information is only needed for those who experience the event and the sampled controls. Nested case-control studies and case-cohort studies are two well researched and widely applied approaches to sampled cohort studies. The nested case-control and case-cohort sampling designs were proposed by Thomas (1977) and Prentice (1986) respectively. The methods differ in the way controls are selected and how the sampled cohorts are analysed, but in their classical form both use simple random sampling without replacement.

The classical methods for analysing data from sampled cohort studies, only use covariate information for the cases and the controls. However, often some covariates values are easily available for all individuals in a cohort while others are more expensive in the sense that they require more resources to obtain, and these expensive covariates are only collected for the sampled cohort. Different methods have been considered to enable more efficient analyses of nested case-control and case-cohort data by using more of the information from the full cohort. Extensions of the classical sampled cohort designs to use stratified sampling have been studied for nested case-control samples by Langholz and Borgan (1995) and for case-cohort samples by Borgan, Langholz, et al. (2000). Stratified sampling allows the use of additional information available in the cohort to obtain a more efficient sample of controls. Borgan and Samuelsen (2016) give a review of these sampled cohort methods.

Two other methods are inverse probability weighted (IPW) estimators, where the weights are based on information from the full cohort, examined by (Støer and Samuelsen (2013)) for nested case-control studies, and full likelihood inference for the missing data that use the Expectation-Maximization algorithm in nested case-control (Scheike and Juul (2004)) and case-cohort samples (Scheike and Martinussen (2004)). The former approach was shown in simulations to work well compared to the traditional nested case-control estimator in some situations with estimates closer to the full cohort. While the latter approach was shown to give gains in efficiency when the hazard ratios and disease incidence were high.

Another approach, the one that will be considered in this thesis, is the method of multiple imputation. Multiple imputation developed out of the paper on missing data by Rubin (1976). Missing data in statistical analyses has since received increased attention as a potential source of bias and in studies where data are missing by design. A general overview of statistical analysis with missing data is given by Little and Rubin (2019) and an overview of multiple imputation by Carpenter and Kenward (2012). Furthermore, development of the versatile full conditional specification (FCS) method of multiple imputation and the `mice` software package by Buuren and Groothuis-Oudshoorn (2010) has made multiple imputation popular in applications as an algorithm to handle missing data.

Imputation of missing covariates for time-to-event data in Cox regression has been examined by White and Royston (2009) who came up with an approximate imputation model, and Bartlett et al. (2015) have further extended the FCS algorithm to be compatible with a wider range of imputation models, including an alternative to approximate imputation for Cox regression models.

In sampled cohort settings, multiple imputation, can be used to impute ("fill in") values of expensive covariates not collected in the sampled cohort using outcome and covariate information that is available for all individuals in the cohort. Multiple imputation of the full cohort with sampled cohort data was considered by Keogh and White (2013) and was further developed for imputing missing values in nested case-control and case-cohort studies (Keogh, Seaman, et al. (2018) ). Simulations have shown good performance of multiple imputation for the full cohort. For an overview of multiple imputation for sampled cohort data and the aforementioned alternatives see Keogh (2018).

However, when the study cohorts are very large, multiple imputation of missing values for the entire cohort might be challenging or infeasible. Additionally, when the event of interest is rare, imputing values for an excessive amount of controls per case might be unnecessary. A middle way between the classical sampled cohort study and sampled cohort studies with imputation of the full cohort is to consider imputation in only a subset of the cohort. The idea is that values of expensive covariates only need to be collected in the sampled cohort, while more easily obtainable covariates can be collected for a larger part of the cohort, but not necessarily for all individuals. Then multiple imputation can be performed for a subset of the cohort and the imputed dataset can be analysed effectively. This middle way is what we will consider in this thesis.

Thus, the aim of this thesis will be to investigate methods for multiple imputation for Cox regression with sampled cohort data that only use a subset of the cohort for imputation. In order to do this, sampled cohorts of nested case-control and case-cohort studies will be the starting point. The two prominent multiple imputations algorithms for Cox regression, approximate imputation and rejection sampling, will used for imputation. Methods for multiple imputation of only a subset of the cohort will be investigated in 4 simulation settings. The methods will be compared with the classical nested-case control and case-cohort estimators, and imputation of the full cohort.

The rest of the thesis is organised as follows. In Chapter 2] the background knowledge on time-to-event analysis, Cox regression, nested case-control and case-cohort sampling designs will be presented. The methods will be illustrated on a dataset of from a study of the effect of serum-free-light-chain blood measurements on mortality. Then Chapter 3 describes the problem of missing data in statistical studies and the method of multiple imputation. The general full conditional speficication (FCS) and substantive model compatible FCS multiple imputation algorithms will be described. In addition a short overview of how multiple imputation may be performed with non-linear terms, interactions and auxiliary variables will be given. Chapter 4 goes through multiple imputation for Cox regression with cohort and sampled cohort data. The methods will be illustrated on the same real-world dataset as in Chapter 2. In Chapter 5 we investigate how multiple imputation with only a part of the full cohort can be performed. To examine and compare the methods we will use simulations experiments, and the results will be reported. Finishing off, in Chapter 6 we summarise and discuss of the results of the thesis and potential further studies. Two appendices will include histograms of selected estimates from the simulations (Appendix A) and selected code from the simulation experiments (Appendix B).

CHAPTER 2

# Cox regression for cohort and case-control data

This chapter introduces the time-to-event analysis framework, Cox regression and cohort studies. Moreover, the two classical case-control sampling designs, the nested case-control sampling and the case-cohort sampling are described. The presentation of the theory in this chapter is primarily based on the textbook by Aalen, Borgan, and Gjessing (2008) and the overview chapter by Borgan and Samuelsen (2016).

## 2.1 Time-to-event analysis

Time-to-event (or survival) analysis is the study of time to events. On a convenient time scale, the interval from a defined starting point to the observation of an event of interest is called the failure (or survival) time. If an event of interest is not observed then the time from the start to the end of the observation period is called a censored failure time.

Two important concepts for analysing time-to-event data are the survival function and the hazard rate. Let $T$ be a random variable, continuous on $[0, \infty)$, that represents the time to an event of interest. Then the survival function

$$S(t) = P(T > t) \tag{2.1}$$

describes the probability that the failure time is larger than $t$. The hazard rate can be defined as

$$h(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} P(T \in [t, t + \Delta t) \mid T \geq t). \tag{2.2}$$

Thus the hazard rate multiplied by a small time interval, $h(t)\Delta t$, can be thought of as the probability that an event will be observed within $\Delta t$ given that it has not yet been observed just before time $t$.

In a study of time-to-event data there will nearly always be censoring. It can be caused if the event has still not happened at the last time recorded. This can happen if an individual is withdrawn/lost from the study or the end of the study period is reached. Imagining a timeline starting from the left and going to the right, we call this right censoring. An assumption that will be important is that of independent censoring, stating that the processes governing the censoring and the failure times should be independent. More details can be found in Section 2.2.8 in Aalen, Borgan, and Gjessing (2008).

Often failure times and censored failure times are denoted by $t$ and an observed-event indicator $\delta$ is recorded. For a study of a cohort of $n$ individuals a survival data set consist of

$$\{(t_i, \delta_i), i = 1, \ldots, n\}\,. \tag{2.3}$$

Two important non-parametric estimators for censored survival data are the Nelson-Aalen estimator and the Kaplan-Meier estimator. They provide estimates for the cumulative hazard rate and the survival function respectively. Let $\mathcal{R}(t)$ denote the risk set at time $t$, that is the set of individuals entered in the study for whom the event or censoring has not yet occured just before time $t$, and let $|\mathcal{R}(t)|$ be the number of individuals in this risk set. The Nelson-Aalen estimator for the cumulative hazard $H(t) = \int_0^t h(s)ds$ is defined as

$$\hat{H}(t) = \sum_{t_i \leq t} \frac{\delta_i}{|\mathcal{R}(t_i)|} \tag{2.4}$$

with increments at the observed event times. The Kaplan-Meier estimator for the survival function is given by

$$\hat{S}(t) = \prod_{t_i \leq t} \left\{ 1 - \frac{\delta_i}{|\mathcal{R}(t_i)|} \right\} \tag{2.5}$$

and it reduces to one minus the empirical distribution function in the absence of censoring.

## 2.2 Cox regression

To analyse possibly censored survival data (2.3) in a regression setting, where a vector of covariates values $\boldsymbol{x}_i$ is also recorded for each individual $i$, the relation between the covariates and the hazard rate can be described by Cox' semiparametric model

$$h_i(t) = h(t \mid \boldsymbol{x}_i) = h_0(t) \exp(\boldsymbol{\beta}'\boldsymbol{x}_i), \tag{2.6}$$

where the non-parametric $h_0(t)$ is an unknown baseline hazard rate corresponding to the situation when all covariates are zero and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ is a vector of unknown parameters. Although the methods in this chapter have been extended to deal with time-varying covariates we will in this thesis only consider covariates that are fixed with respect to the time. Considering the hazard rate ratio of a unit increase in covariate $j$, while keeping all the other covariates fixed,

$$\frac{\exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_j(x_j + 1) + \cdots + \beta_p x_p)}{\exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_j x_j + \cdots + \beta_p x_p)} = \exp(\beta_j) \tag{2.7}$$

shows that a unit increase in covariate $j$ implies multiplying the hazard rate by $e^{\beta_j}$. We will refer to $e^{\beta_j}$ as the relative risk for covariate (or exposure) $x_j$. A relative risk equal to one means that the risk of event is unaffected by the covariate, while a relative risk larger than one means that the risk is increased and vice versa.

The canonical estimator for the coefficient vector of effects $\boldsymbol{\beta}$ is the maximizer of Cox' partial likelihood,

$$L(\boldsymbol{\beta}) = \prod_{i \in \mathcal{E}} \frac{\exp(\boldsymbol{\beta}'\boldsymbol{x}_i)}{\sum_{k \in \mathcal{R}(t_i)} \exp(\boldsymbol{\beta}'\boldsymbol{x}_k)}, \tag{2.8}$$

where $\mathcal{E} = \{i : \delta_i = 1\}$ is the set of observed (uncensored) survival times and $\boldsymbol{x}_i$ is the covariate vector of individual $i$ with failure time $t_i$. In the denominator $\mathcal{R}(t_i)$ denotes the risk set at time $t_i$ which are all individuals in the cohort still under observation just before time $t_i$. We see that the unknown baseline hazard rate $h_0(t)$ is not used.

It was shown by Andersen and Gill (1982) that this maximizer, $\hat{\boldsymbol{\beta}}$, is distributed as an ordinary maximum likelihood estimator, i.e. normally distributed around the true value of $\boldsymbol{\beta}$ where the inverse of the information matrix,

$$\mathrm{I}(\boldsymbol{\beta}) = -\frac{\partial^2 \log L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}' \partial \boldsymbol{\beta}}, \tag{2.9}$$

evaluated at $\hat{\boldsymbol{\beta}}$ can be used as an estimate of the covariance matrix. Because the partial likelihood have the usual likelihood properties we may apply the standard likelihood-based hypothesis tests; the likelihood ratio test, the Wald test or the score test.

In order to compare with the case-control sampling designs we will elaborate a bit on this. The maximizer of (2.8) is also the solution to the score equation $\boldsymbol{U}(\boldsymbol{\beta}) = 0$. The score function of Cox' partial likelihood, $\boldsymbol{U}(\boldsymbol{\beta}) = \frac{\partial \log L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}$, can be written as

$$\boldsymbol{U}(\boldsymbol{\beta}) = \sum_{i \in \mathcal{E}} \left\{ \boldsymbol{x}_i - \bar{\boldsymbol{x}}(\boldsymbol{\beta}, t_i) \right\}. \tag{2.10}$$

The summand compares the covariate vector of individual $i$ who has failure time $t_i$ with the weighted mean

$$\bar{\boldsymbol{x}}(\boldsymbol{\beta}, t_i) = \frac{\sum_{k \in \mathcal{R}(t_i)} \exp(\boldsymbol{\beta}'\boldsymbol{x}_k)\boldsymbol{x}_k}{\sum_{k \in \mathcal{R}(t_i)} \exp(\boldsymbol{\beta}'\boldsymbol{x}_k)} \tag{2.11}$$

at that time. The weighted mean is the average covariate vector of all individuals in the cohort at risk, weighted with their relative risks. We will return to this weighted mean for case-cohort studies.

A quick review of the statistics for the likelihood based tests and a discussion on assumptions for Cox model follows as they will be used in the example below. As stated we may apply the standard likelihood-based tests in order to test the null hypothesis $\boldsymbol{\beta} = \boldsymbol{\beta}_0$ for a known $\boldsymbol{\beta}_0$. The likelihood ratio test statistic

$$\chi^2_{LR} = 2\{\log L(\hat{\boldsymbol{\beta}}) - \log L(\boldsymbol{\beta}_0)\}, \tag{2.12}$$

the Wald test statistic

$$\chi^2_{W} = (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0)'\mathrm{I}(\hat{\boldsymbol{\beta}})(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0) \tag{2.13}$$

and the score test statistic

$$\chi^2_{SC} = \boldsymbol{U}(\boldsymbol{\beta}_0)'\mathrm{I}(\boldsymbol{\beta}_0)^{-1}\boldsymbol{U}(\boldsymbol{\beta}_0) \tag{2.14}$$

are approximately chi-squared distributed with $p$ degrees of freedom under the null hypothesis. The three tests can be generalized for composite hypotheses, and are equivalent asymptotically.

Considering model checking, the two essential assumptions of Cox' model,

$$h(t|\boldsymbol{x}_i)/h(t|\boldsymbol{x}_k) = \exp\{\boldsymbol{\beta}'(\boldsymbol{x}_i - \boldsymbol{x}_{\boldsymbol{k}})\} \text{ for individuals } i \text{ and } k\text{, and}$$

$$\log h(t|\boldsymbol{x}_i) = \boldsymbol{\beta}'\boldsymbol{x}_i + \log h_0(t)$$

imply that hazard rates are proportional, i.e. that the hazard rate ratio does not change with time. Also, as we saw a special case of in (2.7), that covariates have log-linear effect on the hazard rate. These assumptions are in some cases too strict and should be examined. One way to do this is to estimate the cumulative hazard for different values of the covariates and check for proportionality. Some other methods will be briefly mentioned in the below example.

In the discussion of Cox' 1972 paper, Breslow came up with the following estimator for the cumulative baseline hazard, $H_0(t) = \int_0^t h_0(u)du$,

$$\hat{H}_0(t) = \sum_{t_i \leq t} \frac{\delta_i}{\sum_{k \in \mathcal{R}(t_i)} \exp(\hat{\boldsymbol{\beta}}'\boldsymbol{x}_k)}. \tag{2.15}$$

For fixed covariates the cumulative hazard for an individual with covariates $\boldsymbol{x}_0$ can be estimated by

$$\hat{H}(t|\boldsymbol{x}_0) = \hat{H}_0(t)\exp(\hat{\boldsymbol{\beta}}'\boldsymbol{x}_0). \tag{2.16}$$

In addition to providing cumulative hazards plots, the Breslow estimator is an important building block in further development of Cox model and semiparametric inference with censored data.

The `survival` package for the statistical software `R` includes the function `coxph` for fitting Cox' regression model and the usual likelihood based tests for regression are implemented. It also handles defining splines for covariates to check log-linearity and offers functionality for checking the proportional hazards assumption (with e.g. `cox.zph`). For the programming language `Python`, there exists a library called `lifelines` which is dedicated for survival analysis. In addition `statsmodel` has support for Cox regression, and for the machine learning oriented there is a package that is called `sci-kit survival`. For the methods relevant in this thesis `R` has the most extensive support and we will use the `survival` package to apply Cox regression on a real data set in the following example.

### Example

To illustrate Cox regression for a full cohort, and later for case-control and case-cohort designs, we will study the relationship between blood measurements of non-clonal serum immunoglobulin free light chains (FLCs) and the survival times of people dying from neoplasms (e.g. cancer). The survival or failure time is the period from the blood measurement was made until death from neoplasm. People dying from other causes, in addition to those still alive at the end of the study, will have censored failure times. This dataset is from a cohort of residents in Olmsted County, Minnesota studied by Dispenzieri et al. (2012) and the data is available in the `survival` package.

There are two measurements for FLC, $\lambda$ and $\kappa$, where $\lambda$ is the one most strongly associated with an increased hazard. We will consider only $\lambda$ as a covariate in the Cox model while controlling for the potential confounders gender and age at entry to the study. In later examples, $\kappa$ will be be used as a surrogate for $\lambda$. The FLC measurements will both be converted to a logarithmic scale with base 2.

The full cohort for this example are measurements on $n = 5486$ individuals, out of which $d = 305$ deaths due to neoplasms are observed corresponding to 5-6%. The genders are roughly evenly distributed with 2861 females and 2625 males. The distributions of age when measurements were taken and the FLC measurement are illustrated in the two leftmost panels in Figure 2.1.

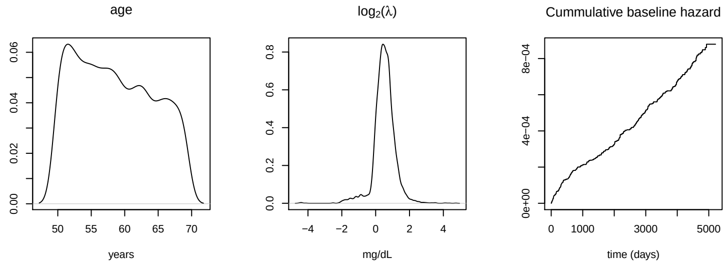


Figure 2.1: Left and centre: Estimated densities of age and $\log_2(\lambda)$ using Gaussian kernels. Right: Estimated cumulative baseline hazard

The result of fitting Cox regression model with the covariates age, sex and $\log_2(\lambda)$ are shown in Table 2.1. All covariates have a significant effect on the survival time at a significance level of 0.05. The p-values for the null hypothesis of $\log_2(\lambda)$ having no effect on survival time is $< 0.0001$. A unit increase in $\log_2(\lambda)$, which corresponds to a doubling in $\lambda$, implies a relative risk of $e^{\beta_{\log_2(\lambda)}} = 1.712$ with a 95% confidence interval $[1.419, 2.065]$. This shows that elevated FLC levels for $\log_2(\lambda)$ are clearly associated with higher death rates. The same goes for being male and having a higher age at entry to the study. The rightmost panel of Figure 2.1 shows that the cumulative baseline hazard is approximately linear which means that the baseline hazard rate is approximately constant.

To test dependency on time the function `cox.zph` adds for each covariate $j$ in turn a time dependent term $\rho x_j g(t)$ to the model and tests the parameter $\rho$'s deviation from zero with a score-test. Choosing $g(t)$ as both the identity

Table 2.1: Table of regression coefficients for full cohort

| | coef | exp(coef) | se(coef) | z | p |
|---|---|---|---|---|---|
| age | 0.066 | 1.068 | 0.010 | 6.485 | <0.001 |
| sexM | 0.250 | 1.284 | 0.115 | 2.170 | 0.030 |
| loglambda | 0.538 | 1.712 | 0.096 | 5.618 | <0.001 |

function and the logarithm indicated that the proportional hazards assumption for $\log_2(\lambda)$ does not hold ($p < 0.01$). Therefore the estimated effect of $\log_2(\lambda)$ should be interpreted as an average effect over time. The log-linear effects assumption, which is examined by fitting splines for both continuous covariates and testing the non-linear effects, seem to hold ($p > 0.35$) for $\log_2(\lambda)$. □

We note that with relatively few cases, the excess of controls in the risk set may not be worth the additional estimation information compared to the cost of collecting the measurements. For example, some biological material might require extensive amount of work to collect and can only be used once. Also summing over the risk set in the denominator of (2.8) can be a costly operation when the cohort is of substantial size. The two cohort sampling techniques, nested case-control and case-cohort design utilise the idea of reducing the risk set with random sampling while hopefully retaining sufficiently effective estimates of the effects of the covariates.

## 2.3 Nested case-control sampling

In a nested case control study $m-1$ controls are sampled independently for each case $i \in \mathcal{E}$ without replacement from those at risk in the full cohort at the time of failure for the case (at $t_i$). Each case $i$ is compared with a sampled risk set $\tilde{\mathcal{R}}(t_i)$ of size $m$ consisting of the case and the $m-1$ controls. Note here that a control for an individual $i$ with failure time $t_i$ is a not a case at that time, but may later be a case itself. Meaning that the the term control used for survival data is more subtle than the traditional use where a control is simply not a case. Thus a nested case-control sample consists of the union of sampled risk sets $\tilde{\mathcal{R}}(t_i)$ for $i \in \mathcal{E}$. For a full cohort of size $n$ with $d$ observed failures the number of unique individuals in a nested case-control sample is less than $d \times m$ since the controls may be sampled in different risk sets and because cases can be sampled as controls for other cases.

In the simplest situation nested case-control sampling is done randomly without replacement. It can be shown that a partial likelihood for Cox' proportional hazards model, can be written

$$L_{ncc}(\boldsymbol{\beta}) = \prod_{i \in \mathcal{E}} \frac{\exp(\boldsymbol{\beta}'\boldsymbol{x}_i)}{\sum_{k \in \tilde{\mathcal{R}}(t_i)} \exp(\boldsymbol{\beta}'\boldsymbol{x}_k)}. \tag{2.17}$$

Under weak regularity conditions,

$$\sqrt{n}\left(\hat{\boldsymbol{\beta}}_{ncc} - \boldsymbol{\beta}\right) \rightarrow^d \mathcal{N}(0, \boldsymbol{\Sigma}_{ncc}^{-1}). \tag{2.18}$$

where

$$\hat{\boldsymbol{\beta}}_{ncc} = \text{argmax}_{\boldsymbol{\beta}} L_{ncc}(\boldsymbol{\beta}). \tag{2.19}$$

Also it can be proven that $n^{-1}\boldsymbol{I}_{ncc}(\hat{\boldsymbol{\beta}}_{ncc})$ is a consistent estimator for $\boldsymbol{\Sigma}_{ncc}$, meaning that the information matrix evaluated at $\hat{\boldsymbol{\beta}}_{ncc}$ can be used to estimate the covariance matrix of the nested case control estimator.

The relative efficiency of $\hat{\boldsymbol{\beta}}_{ncc}$ compared to the full cohort estimator $\hat{\boldsymbol{\beta}}$ is the ratio of the inverse of their respective variances. For the situation where there is only one covariate with true parameter equal to zero, the relative efficiency has been shown to be $\frac{m-1}{m}$ (Goldstein and Langholz 1992). In models with more parameters or when the effects are non-zero the relative efficiency may be lower.

Nested case-control analysis can be done with the `coxph` command in `R` using the $d$ case-control sets as separate strata. Alternatively, one may manipulate the entry and exit times such that only the case and its controls are at risk at the case's failure time, which might be computationally faster.

### Example cont.

Letting the number of controls for each case be 2 ($m = 3$) the results from one nested case-control sample are compared with the results from the full cohort in Table 2.2.

Table 2.2: Estimated regression coeffiecient full cohort and nested case-control sampling for $m = 3$

| | coef | exp(coef) | se(coef) | z | p |
|---|---|---|---|---|---|
| | | Full cohort | | | |
| age | 0.066 | 1.068 | 0.010 | 6.485 | <0.001 |
| sexM | 0.250 | 1.284 | 0.115 | 2.170 | 0.030 |
| loglambda | 0.538 | 1.712 | 0.096 | 5.618 | <0.001 |
| | | Nested case-control | | | |
| age | 0.062 | 1.063 | 0.013 | 4.827 | <0.001 |
| sexM | 0.198 | 1.219 | 0.143 | 1.383 | 0.167 |
| loglambda | 0.477 | 1.611 | 0.129 | 3.705 | <0.001 |

The nested case-control estimate for $\log_2(\lambda)$ is somewhat smaller than the estimate using the full cohort. A 95% CI for the relative risk is $[1.252, 2.072]$. We also see that the estimated coefficient for age is very similar to the estimate for the full cohort while being male is not found to be significantly associated with an increased hazard using only the nested case-control sample. The standard errors for the nested case-control estimates are as expected larger than for the full cohort.

The relative efficiency of the nested case-control estimator for $\log_2(\lambda)$ compared to the full cohort estimator, i.e. the square of the full cohort standard error divided by the square of the nested case-control standard error, is equal to 0.553. For $m = 3$ the total number of unique individuals included in the nested case control study is less than 915 (867) compared to the original 5486. In this sampled cohort, an approximately 35% loss in relative efficiency for sexM obscures the significance ($p < 0.05$) found in the full cohort analysis.

To illustrate the variability of the nested case-control analysis, 1000 nested case-control samplings with two controls per case were performed. Figure 2.2

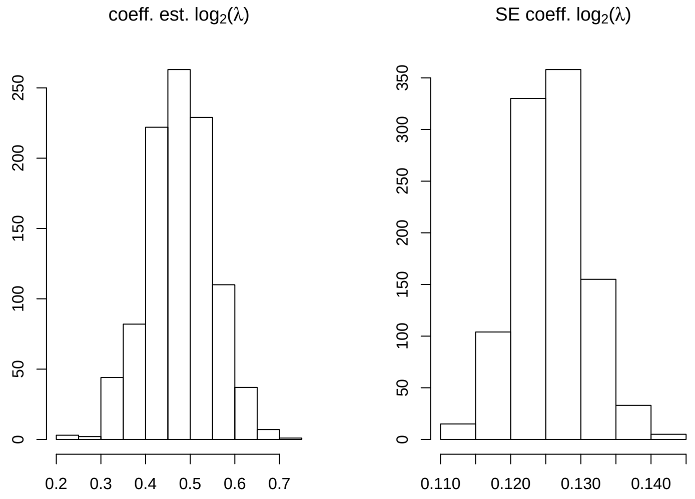


Figure 2.2: Histograms of nested case-control estimated coefficient and standard error for $\log_2(\lambda)$

shows the distribution of the estimates for $\log_2(\lambda)$ along with their standard errors. The mean of the estimates for $\log_2(\lambda)$ is 0.478, which is lower than for the full cohort, but not far off. The mean standard error is 0.126. A 95% interval estimated from the 2.5 and 97.5 percentiles of the distribution of the 1000 simulated estimates is $[0.337, 0.619]$ for $\log_2(\lambda)$ and $[0.116, 0.136]$ for the standard error. □

## 2.4 Case-cohort design

Under a classic case-cohort study design a subcohort $\mathcal{C}$ of size $\tilde{m}$ is sampled randomly from the full cohort. For each failure time $t_i$ the individuals in the subcohort, along with one or more cases not occurring in the subcohort, at risk make up the case-cohort risk set $\mathcal{S}(t_i)$. As a result the number of unique individuals contained in the subcohort in addition to all cases from the full cohort is at most $d + \tilde{m}$, depending on how many of the cases that occur in the subcohort. Several estimation procedures for case-cohort samples have been proposed. We will consider the original case-cohort estimator by Prentice (1986) and inverse probability weighted estimators.

### Prentice's estimator

Prentice's original estimator for the vector of effects, denoted $\hat{\boldsymbol{\beta}}_P$, is the maximizer of

$$L_P(\boldsymbol{\beta}) = \prod_{i \in \mathcal{E}} \frac{\exp(\boldsymbol{\beta}'\boldsymbol{x}_i)}{\sum_{k \in \mathcal{S}(t_i)} \exp(\boldsymbol{\beta}'\boldsymbol{x}_k)}. \tag{2.20}$$

Here the risk set is defined as $\mathcal{S}(t_i) = \mathcal{C}(t_i) \cup \{i\}$, i.e. the individuals at risk in the subcohort at time $t_i$, denoted $\mathcal{C}(t_i)$, added with the case with observed failure at that time (if it was not already sampled in the subcohort). When the subcohort is randomly sampled without replacement it can be shown that

$$\sqrt{n}\left(\hat{\boldsymbol{\beta}}_P - \boldsymbol{\beta}\right) \rightarrow^d \mathcal{N}(0, \boldsymbol{\Sigma}^{-1} + \frac{1-p}{p}\boldsymbol{\Sigma}^{-1}\boldsymbol{\Delta}\boldsymbol{\Sigma}^{-1}) \tag{2.21}$$

where $p = \frac{\tilde{m}}{n}$ is the proportion of the full cohort that is sampled in the subcohort and $\boldsymbol{\Delta}$ is the limit in probability of the covariance matrix of individual score contributions. The practical consequence is as $n$ increases $\hat{\boldsymbol{\beta}}_P$ is approximately normally distributed with expectation $\boldsymbol{\beta}$. As for the covariance matrix one may use $n^{-1}\mathrm{I}_P(\hat{\boldsymbol{\beta}}_P)$ to estimate $\boldsymbol{\Sigma}$ where $\mathrm{I}_P$ is the information matrix obtained from (2.20). Further, $\boldsymbol{\Delta}$ can be estimated by (Borgan, Langholz, et al. 2000)

$$\hat{\boldsymbol{\Delta}} = \frac{1}{\tilde{m}}\sum_{k\in\mathcal{E}} \boldsymbol{Z}_k\boldsymbol{Z}_k' \tag{2.22}$$

where

$$\boldsymbol{Z}_k = \sum_{i\in\mathcal{E}} \left\{\boldsymbol{x}_k - \tilde{\boldsymbol{x}}_k(\hat{\boldsymbol{\beta}}_P, t_i)\right\} \frac{\exp(\hat{\boldsymbol{\beta}}_P'\boldsymbol{x}_k)}{\sum_{k\in\mathcal{S}(t_i)}\exp(\hat{\boldsymbol{\beta}}_P'\boldsymbol{x}_k)\frac{n}{\tilde{m}}}. \tag{2.23}$$

Here the weighted mean is

$$\tilde{\boldsymbol{x}}_k(\hat{\boldsymbol{\beta}}_P, t_i) = \frac{\sum_{k\in\mathcal{S}(t_i)}\exp(\hat{\boldsymbol{\beta}}_P'\boldsymbol{x}_k)\boldsymbol{x}_k}{\sum_{k\in\mathcal{S}(t_i)}\exp(\hat{\boldsymbol{\beta}}_P'\boldsymbol{x}_k)}. \tag{2.24}$$

Thus the uncertainty due to subsampling of the full cohort is expressed in the additional variance term.

### Inverse probability weighted estimators

A weakness of Prentice's estimator is that it does not appear to use all the information available on the cases (for whom we have collected data). In the risk sets, only the case with survival time $t_i$ is added to the subcohort. Thus we may be ignoring information of the other cases at risk at $t_i$ who are not included in the subcohort. The idea of inverse probability weighted (IPW) estimators is to include all cases at risk as controls in the risk set $\tilde{\mathcal{S}}(t_i) = \mathcal{C}(t_i) \cup \{k : \delta_k = 1, t_k \geq t_i\}$ and to use inverse probability weighting to correct for possible bias. The weighted pseudo-likelihood

$$L_W(\boldsymbol{\beta}) = \prod_{i\in\mathcal{E}} \frac{\exp(\boldsymbol{\beta}'\boldsymbol{x}_i)}{\sum_{k\in\tilde{\mathcal{S}}(t_i)}\exp(\boldsymbol{\beta}'\boldsymbol{x}_k)w_k} \tag{2.25}$$

was originally proposed by Kalbfleisch and Lawless (1988). The weights are $w_k = 1$ for cases and $w_k = \frac{1}{p_k}$ for non failures. The $p_k$'s should be the appropriate inclusion probabilities. Letting the inclusion probabilities for non-failures equal the number of non-failures in the subcohort over the number of non-failures in the full cohort we obtain the estimator $\hat{\boldsymbol{\beta}}_W$ of Lin and Ying (1993). They also showed how to estimate the variance with small modifications to the procedure for the Prentice estimator described above. It can be shown that the variance of $\hat{\boldsymbol{\beta}}_W$ is smaller than of $\hat{\boldsymbol{\beta}}_P$ in theory but this will often not matter much in practice.

Both methods can be fitted using the `cch` function in the `survival` package with the method argument specified as either `“Prentice”` or `“LinYing”`.

### Example cont.

In order to compare with nested case-control sampling, the size of the subcohort is chosen to be $\tilde{m} = 593$, slightly less than 610, see Langholz and Thomas (1990) for details. The fitted values for the Prentice and IPW case-cohort estimators are reported in Table 2.3. We note that the standard errors for the IPW estimator are slightly lower than for the original (non-robust) Prentice estimator. For both case-cohort methods age and $\log_2(\lambda)$ are found to be highly significant ($p < 0.001$) and for sexM both p-values are larger than 0.05.

Table 2.3: Estimated regression coefficients full cohort and one case-cohort sampling for $\tilde{m} = 593$

| | coef | exp(coef) | se(coef) | z | p |
|---|---|---|---|---|---|
| | | Full cohort | | | |
| age | 0.066 | 1.068 | 0.010 | 6.485 | <0.001 |
| sexM | 0.250 | 1.284 | 0.115 | 2.170 | 0.030 |
| loglambda | 0.538 | 1.712 | 0.096 | 5.618 | <0.001 |
| | | Prentice estimator | | | |
| age | 0.066 | 1.069 | 0.013 | 5.113 | <0.001 |
| sexM | 0.334 | 1.397 | 0.147 | 2.273 | 0.023 |
| loglambda | 0.498 | 1.645 | 0.125 | 3.984 | <0.001 |
| | | IPW estimator | | | |
| age | 0.068 | 1.070 | 0.013 | 5.261 | <0.001 |
| sexM | 0.342 | 1.407 | 0.146 | 2.343 | 0.019 |
| loglambda | 0.468 | 1.598 | 0.120 | 3.914 | <0.001 |

The variability across 1000 case-cohort samples for the IPW estimator is shown in Figure 2.3. The mean value across the simulations of the Prentice estimate for $\log_2(\lambda)$ is 0.550 and the mean standard error is 0.133. For the IPW estimator they are 0.547 and 0.127 respectively which shows that both case-cohort methods give similar estimates. The mean coefficient estimates and standard errors are higher than the nested case-control results. The 95% intervals for the estimated effect of $\log_2(\lambda)$ and the standard error are $[0.393, 0.721]$ and $[0.117, 0.138]$ for the IPW estimator. The width of the intervals and the histograms indicate more sampling variability for the case-cohort study compared with the nested case-control study. □

As we have seen for both nested case-control and case-cohort studies the results with relatively few number of controls are not far from those from the full cohort. Although, the standard errors are increased and the estimates more variable. Multiple imputation could offer an improvement to this by making use of hypothetically easily available information on the individuals not sampled in the nested case-control or case-cohort study. We could then treat values that are not collected on the individuals outside the sampled cohort as missing data.

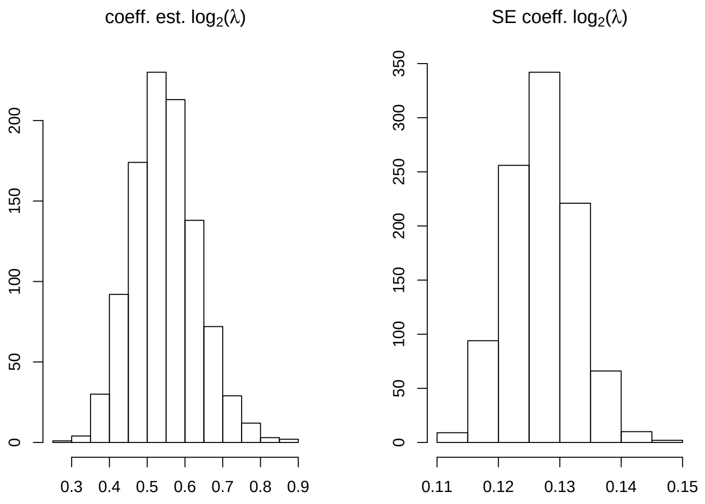


Figure 2.3: Histogram of estimated coefficients and standard errors for $\log_2(\lambda)$ using the IPW estimator.

CHAPTER 3

# Multiple Imputation

This chapter describes the method of multiple imputation (MI) for handling missing data. First we take a look at the problem of missing data in general and the MI algorithm as a way to handle missing data. Then we will consider some of the theory behind multiple imputation. Finally, the broadly applicable method of full conditional specification and substantive model compatible full conditional specification for imputation will be presented. The main sources for this chapter are the textbooks by Little and Rubin (2019) and Carpenter and Kenward (2012), and the article by by Bartlett et al. (2015).

## 3.1 Missing data and multiple imputation

The concept of missing data describes the situation when data intended to be collected have not been done so. A variable is missing if no value has been recorded for it. This could be due to a wide variety of reasons, however, one assumes that the value of the variable would have been possible to observe. Missing data arises across almost all applications, e.g. patients lost to follow up, failure in measurement equipment or inconclusive measurements, non-response in surveys, misplacement of data. Although historically the problem of missing data was heavily ignored, many methods and a growing literature now exist for handling partially observed data. The methods range from the most basic complete-case analysis (deleting all individuals for which data is missing on one or more variables), weighted complete-case analysis, mean-imputation (imputing missing values of a variable with the mean of its observed values), to likelihood based methods, Bayesian iterative simulation methods and multiple imputation.

The proper way to handle missing data will depend on the reasons for why data is missing, the analysis one wishes to perform and the resources available. If there are systematic differences between the values that are missing and the values that are observed, then this handling is non-trivial. For example consider a univariate complete-case analysis of the mean of $n$ survival times for an event of interest, $T_1, \dots, T_n$ for which only $D$ $(< n)$ events have been observed. Discarding the censored survival times and computing the mean value of the observed survival times gives an estimate that is biased downwards.

The general idea of imputation is to use information available in the observed data to fill in the values of the missing data. With more than one variable the idea is to use inferred relationships between the variables from observed data, and then for an individual with missing data, use the observed variables to impute the missing ones. If the assumptions and inferred relationship are correct then imputation avoids throwing away information, making estimates more efficient. On the other hand, treating imputed values equally as the observed values, in effect as if they were known, can give invalid (anti-conservative) inferences. Thus more advanced methods are often necessary such that imputation correctly propagates the uncertainty due to missingness and the uncertainty in the models or methods used for imputation. Multiple imputation is seen as a good trade-off between ease of use (available software packages, required modelling) and robust, efficient estimates with the right amount of uncertainty.

### A note on Bayesian statistics

Multiple imputation can be seen as an approximation to a full Bayesian analysis of incomplete data, and it is derived in a Bayesian setting. In Bayesian statistics the *posterior distribution* of a parameter (vector/matrix) $\boldsymbol{\theta}$ given a vector or matrix of observed data $\boldsymbol{y}$ is calculated by

$$f(\boldsymbol{\theta} \mid \boldsymbol{y}) = \frac{f(\boldsymbol{\theta})f(\boldsymbol{y} \mid \boldsymbol{\theta})}{f(\boldsymbol{y})}, \tag{3.1}$$

where $f(\boldsymbol{y} \mid \boldsymbol{\theta})$ is the *data distribution* and $f(\boldsymbol{\theta})$ is a *prior distribution* expressing our belief about the parameter before the data is seen. When $f(\boldsymbol{y} \mid \boldsymbol{\theta})$ is regarded as a function of $\boldsymbol{\theta}$ for fixed $\boldsymbol{y}$ it is called a likelihood function. The primary task of any Bayesian application is to develop the model $f(\boldsymbol{y}, \boldsymbol{\theta}) = f(\boldsymbol{\theta})f(\boldsymbol{y} \mid \boldsymbol{\theta})$ and perform the computations to summarize $f(\boldsymbol{\theta} \mid \boldsymbol{y})$ (Gelman et al. 2013).

To predict unobserved values $\tilde{\boldsymbol{y}}$ stemming from the same data generating process (or data distribution), we can use the *posterior predictive distribution* of $\tilde{\boldsymbol{y}}$ given the observed data $\boldsymbol{y}$ which can be written

$$f(\tilde{\boldsymbol{y}} \mid \boldsymbol{y}) = \int_{\boldsymbol{\theta}} f(\tilde{\boldsymbol{y}} \mid \boldsymbol{\theta}, \boldsymbol{y}) f(\boldsymbol{\theta} \mid \boldsymbol{y}) d\boldsymbol{\theta}. \tag{3.2}$$

The posterior predictive distribution is an average of the conditional distributions for $\tilde{\boldsymbol{y}}$ over the posterior distribution of $\boldsymbol{\theta}$. It is called predictive because it is the distribution of an observable quantity.

### Missing data mechanisms

A framework for analysis of partially observed data was laid out by Rubin (1976). Consider covariates and outcome variables $Y_1, \dots, Y_p$ referred to as just variables (or items) intended to be collected for $n$ individuals in a sample. Let the complete data matrix $\boldsymbol{y}$ be the $n \times p$ matrix of values for these variables. Now suppose that some of the values (intended to be collected) are not observed. The complete data matrix $\boldsymbol{y}$ can then be partitioned into the observed data denoted $\boldsymbol{y}_O$, and the missing data denoted $\boldsymbol{y}_M$.

Further let the variable $R_j$ be a response indicator for $Y_j$ such that if a value for $Y_j$ on an individual is observed then $R_j = 1$ and if the value is missing then $R_j = 0$ for $j = 1, \dots, p$. We let the matrix $\boldsymbol{r}$ of binary values be the response indicator matrix for the complete data matrix where the indicator value $r_{ij}$ correspond the element $(i, j)$ of $\boldsymbol{y}$. Then $\boldsymbol{r}$ gives the partitioning of $\boldsymbol{y}$.

By sorting the rows and columns in $\boldsymbol{y}$ a useful *missigness pattern* for the entire matrix may be found. For example $r_{ij} = 1 \ \forall j \neq k$ on $i = 1, \dots, m$

and $r_{ij} = 1 \; \forall j$ on $i = m+1, \ldots, n$ describe a univariate missingness pattern for variable $k$, i.e. values are only missing on $Y_k$ for some $(n-m)$ of the $n$ individuals. We say that $Y_k$ is partially observed (or partially missing) in the sample. Another pattern, monotone missingness is when, if data is missing on variable $Y_{ij}$ it is also missing on variables $Y_{i,j+1}, \ldots, Y_{i,p}$ for all $i$. This can simplify methods for imputation. In applications there will often be several missingness patterns describing different partitions of the individuals. However, one or a few missingness patterns will typically dominate, and it is the assumptions about the missingness mechanisms governing these that are important for the study.

*Missingness mechanisms* describe the relationship between the missingness pattern and the values in the data matrix. Let the parameters of the missingness mechanism be $\boldsymbol{\omega}$. Then the general expression for the missingness mechanism is

$$f(\boldsymbol{r} \mid \boldsymbol{y}_M, \boldsymbol{y}_O, \boldsymbol{\omega}). \tag{3.3}$$

The three main assumptions on the mechanism governing the missingness are missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR). Missing completely at random is the assumption that data are missing independently of their underlying values. The MCAR assumption can expressed as

$$f(\boldsymbol{r} \mid \boldsymbol{y}_M, \boldsymbol{y}_O, \boldsymbol{\omega}) = f(\boldsymbol{r} \mid \boldsymbol{\omega}) \tag{3.4}$$

A weaker assumption is that of MAR, which states that

$$f(\boldsymbol{r} \mid \boldsymbol{y}_M, \boldsymbol{y}_O, \boldsymbol{\omega}) = f(\boldsymbol{r} \mid \boldsymbol{y}_O, \boldsymbol{\omega}), \tag{3.5}$$

the probability of values being missing do not depend on the values of the missing variables when conditioned on the observed values. We note that contrary to what it sounds like, the data is not unconditionally missing at random (like MCAR). They are instead assumed missing at random given the observed values.

To give an example of MCAR and MAR, consider blood pressure measurements taken by a physician where some values are missing. Suppose the physician is more likely to record the blood pressure for older people than for younger. If blood pressure generally increases with age, then the blood pressure is by itself not missing completely at random because it is more likely to be missing for lower values. When conditioning on age, which is available for all individuals, the probability that blood pressure measurement is missing for e.g. a 30 year old may be higher than for an 80 year old, but it does no longer depend of the value of the blood pressure. Therefore blood pressure is MAR given age. The assumption of MAR tends to be more reasonable as more observed variables can be conditioned on.

When neither MAR nor MCAR hold, data is said to be missing not at random (MNAR). Methods for analysis under MNAR exists. However, these situations require more modelling assumptions and that the consequences of the model for the missingness mechanism be carried throughout the analysis. They will not be considered in this thesis.

### Ignorable missingness

When inferences from an analysis ignoring the missing data mechanism are equivalent to those from the full analysis including the missing data mechanism then the missingness is said to be *ignorable*. The parameters of interest in the analysis are usually (implied by) those for the distribution of the complete data $f(\boldsymbol{y} \mid \boldsymbol{\theta})$ and the parameters $\boldsymbol{\omega}$ of the missingness mechanism are of no interest. The posterior distribution for the parameters given the observed quantities $\boldsymbol{y}_O$ and $\boldsymbol{r}$ is

$$f(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \boldsymbol{y}_O, \boldsymbol{r}) \propto f(\boldsymbol{\theta}, \boldsymbol{\omega}) f(\boldsymbol{y}_O, \boldsymbol{r} \mid \boldsymbol{\theta}, \boldsymbol{\omega}). \tag{3.6}$$

For inference based on likelihood ratios MAR is a sufficient condition for ignorable missingness. For Bayesian inference, an additional requirement is that the parameters are apriori independent, i.e. $f(\boldsymbol{\theta}, \boldsymbol{\omega}) = f(\boldsymbol{\theta}) f(\boldsymbol{\omega})$. In practice MAR is the important condition. Under ignorable missingness the posterior predictive distribution of the missing values does not depend on the response indicator matrix , i.e.

$$f(\boldsymbol{y}_M \mid \boldsymbol{y}_O, \boldsymbol{r}) = f(\boldsymbol{y}_M \mid \boldsymbol{y}_O). \tag{3.7}$$

This implies

$$f(\boldsymbol{y} \mid \boldsymbol{y}_O, \boldsymbol{r} = \mathbf{1}) = f(\boldsymbol{y} \mid \boldsymbol{y}_O, \boldsymbol{r} = \mathbf{0}) \tag{3.8}$$

showing that the posterior predictive distribution of the data can be estimated from observed data and that the estimated distribution can be used to impute the missing data. Ignorable missingness will be assumed for the methods in this thesis. See chapter 6 in Little and Rubin (2019) and chapter 2 in Van Buuren (2018) for more details on ignorable missingness and its implications.

### Multiple Imputation

The multiple imputation algorithm imputes (fills in) missing values using an *imputation model* and fits a model of interest for analysis, the *analysis model*, on imputed datasets a moderate number of times. The results from each fit are combined to give the final estimates. Figure 3.1 shows the steps in multiple imputation for three imputations.

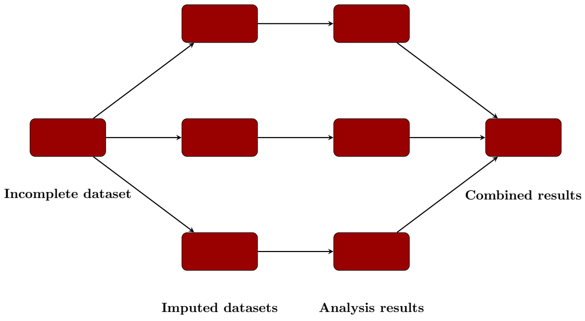


Figure 3.1: Multiple imputation

Let $\boldsymbol{y}_M$ be the matrix of missing data , let $\boldsymbol{y}_O$ be the matrix of observed values and let $\boldsymbol{r}$ be the matrix of response indicators for variables $Y = (Y_1, \ldots, Y_p)$ across $n$ individuals. The algorithm for multiple imputation is as follows

**Algorithm 1** Multiple imputation for missing data

```
1: Impute values $\tilde{\boldsymbol{y}}_M$ of the missing data $\boldsymbol{y}_M$ for $k = 1, \ldots, K$, giving $K$ "complete" (imputed) datasets $\boldsymbol{y}^{(k)} = (\tilde{\mathbf{y}}_M^{(k)}, \mathbf{y}_O)$.
2: Fit the analysis model to each of the $K$ "complete" datasets.
3: Combine the results using Rubin's rules.
```

In the imputation step (1), missing values, $\tilde{\boldsymbol{y}}_M$ are drawn from the posterior predictive distribution of the missing values given the observed data and observed response indicators, which under MAR is,

$$f(\boldsymbol{y}_M \mid \boldsymbol{y}_O, \boldsymbol{r}) = f(\boldsymbol{y}_M \mid \boldsymbol{y}_O) = \int_{\boldsymbol{\theta}} f(\boldsymbol{y}_M \mid \boldsymbol{y}_O, \boldsymbol{\theta}) f(\boldsymbol{\theta} \mid \boldsymbol{y}_O) d\boldsymbol{\theta}. \tag{3.9}$$

In general, to impute values we can

> Draw the parameters from their posterior distribution given the observed data $\boldsymbol{\theta}^{(k)} \sim f(\boldsymbol{\theta} \mid \boldsymbol{y}_O)$
>
> For the missing values draw $\tilde{\boldsymbol{y}}_M \sim f(\boldsymbol{y}_M \mid \boldsymbol{y}_O, \boldsymbol{\theta}^{(k)})$

where $f(\boldsymbol{y}_M \mid \boldsymbol{y}_O, \boldsymbol{\theta})$ is the conditional data distribution implied by the joint data (imputation) model $f(\boldsymbol{y}|\boldsymbol{\theta})$. For each imputation $k$ a new set of parameters $\boldsymbol{\theta}^{(k)}$ is first drawn from the the calculated posterior before the missing values are imputed, making $\tilde{\boldsymbol{y}}_M^{(k)}$ a draw from its posterior predictive distribution.

In (2) each imputed dataset is treated as if no data were missing and the analysis model is fitted (e.g. with maximum likelihood) giving parameter estimates $\hat{\boldsymbol{\beta}}_k$, and their estimated covariance matrix $\hat{\boldsymbol{V}}_k$ for each of the $K$ imputations.

Lastly (3) we use Rubin's rules (Rubin 1987) to combine these into the MI estimate

$$\hat{\boldsymbol{\beta}}_{MI} = \frac{1}{K} \sum_{k=1}^{K} \hat{\boldsymbol{\beta}}_k \tag{3.10}$$

and the estimated total variance

$$\hat{\boldsymbol{V}}_{MI} = \hat{\boldsymbol{W}} + \left(1 + \frac{1}{K}\right) \hat{\boldsymbol{B}}, \tag{3.11}$$

where the within imputation variance and the between imputation variance are respectively given by

$$\hat{\boldsymbol{W}} = \frac{1}{K} \sum_{k=1}^{K} \hat{\boldsymbol{V}}_k, \quad \hat{\boldsymbol{B}} = \frac{1}{K-1} \sum_{k=1}^{K} \left(\hat{\boldsymbol{\beta}}_k - \hat{\boldsymbol{\beta}}_{MI}\right) \left(\hat{\boldsymbol{\beta}}_k - \hat{\boldsymbol{\beta}}_{MI}\right)'. \tag{3.12}$$

The number of imputations is generally quite moderate, e.g. $K = 5$ or $K = 10$, but some authors have suggested that more imputations should be used, up to the lower hundreds.

### Bayesian justification and inference for mutiple imputation

Suppose we are interested in the posterior distribution of the parameters of an analysis model $\boldsymbol{\beta}$ but are missing some data. The posterior distribution of the parameters $\boldsymbol{\beta}$ given the observed data $\boldsymbol{y}_O$ can be written

$$f(\beta|\boldsymbol{y}_O) = \int_{\boldsymbol{y}_M} f(\boldsymbol{\beta}, \boldsymbol{y}_M|\boldsymbol{y}_O)d\boldsymbol{y}_M = \int_{\boldsymbol{y}_M} f(\beta|\boldsymbol{y}_M, \boldsymbol{y}_O)f(\boldsymbol{y}_M|\boldsymbol{y}_O)d\boldsymbol{y}_M. \quad (3.13)$$

To simulate values from $f(\boldsymbol{\beta} \mid \boldsymbol{y}_O)$ one can draw $\tilde{\boldsymbol{y}}_M \sim f(\boldsymbol{y}_M \mid \boldsymbol{y}_O)$ and then draw $\boldsymbol{\beta} \sim f(\boldsymbol{\beta} \mid \tilde{\boldsymbol{y}}_M, \boldsymbol{y}_O)$. When one or both of these distributions are not straightforward to draw from, iterative methods may be needed, e.g. Gibbs sampling. For each draw the iterative sequence needs to converge, and this convergence should be assessed for each sequence. Inference for the observed data posterior of the parameters requires a vast amount of draws, and this can be quite demanding. However, in some situations the posterior mean and variance are enough to describe the posterior distribution. Under regularity conditions, the posterior mean can be written as

$$\mathrm{E}[\boldsymbol{\beta}|\boldsymbol{y}_O] = \mathrm{E}_{f(\boldsymbol{y}_M|\boldsymbol{y}_O)}[\mathrm{E}_{f(\boldsymbol{\beta}|\boldsymbol{y}_M,\boldsymbol{y}_O)}[\boldsymbol{\beta}]], \quad (3.14)$$

and the variance as

$$\begin{aligned}\mathrm{Var}(\boldsymbol{\beta}|\boldsymbol{y}_O) =& \mathrm{E}_{f(\boldsymbol{y}_M|\boldsymbol{y}_O)}[\mathrm{Var}_{f(\boldsymbol{\beta}|\boldsymbol{y}_M,\boldsymbol{y}_O)}(\boldsymbol{\beta})] \\ &+ \mathrm{Var}_{f(\boldsymbol{y}_M|\boldsymbol{y}_O)}(\mathrm{E}_{f(\boldsymbol{\beta}|\boldsymbol{y}_M,\boldsymbol{y}_O)}[\boldsymbol{\beta}]). \quad (3.15)\end{aligned}$$

As the sample size grows to infinity the complete data posterior will typically approach a multivariate normal distribution

$$f(\beta|\boldsymbol{y}_M, \boldsymbol{y}_O) \approx N(\hat{\beta}, \hat{\boldsymbol{J}}^{-1}) \quad (3.16)$$

and does not depend on any reasonably prior for the parameters (see discussion in chapter 2 Carpenter and Kenward (2012)). The mean, $\hat{\boldsymbol{\beta}}$, is the vector of maximum likelihood estimates and the covariance matrix is the inverse of the information matrix $\hat{\boldsymbol{J}}$. Thus the complete data posterior mean and variance can be estimated using the ML estimates when the missing data are imputed.

To approximate the outer expectation we can use Monte Carlo estimation to approximate the integral (3.13) over the the missing values. By first drawing $\tilde{\boldsymbol{y}}_M^{(k)}$ for $k = 1, \dots, K$ from the posterior predictive distribution $f(\boldsymbol{y}_M \mid \boldsymbol{y}_O)$ and then estimating the inner expectation and variance using the ML estimators, one obtains

$$\mathrm{E}[\beta|\boldsymbol{y}_O] \approx \frac{1}{K}\sum_{k=1}^{K} \hat{\boldsymbol{\beta}}_{ML}\left(\tilde{\boldsymbol{y}}_M^{(k)}, \boldsymbol{y}_O\right) = \hat{\beta} \quad (3.17)$$

$$\mathrm{Var}(\beta|\boldsymbol{y}_O) \approx \frac{1}{K}\sum_{k=1}^{K} \hat{\boldsymbol{V}}_{ML}\left(\tilde{\boldsymbol{y}}_M^{(k)}, \boldsymbol{y}_O\right) \quad (3.18)$$

$$+ \frac{1}{K-1}\sum_{k=1}^{K}\left(\hat{\boldsymbol{\beta}}_{ML}\left(\tilde{\boldsymbol{y}}_M^{(k)}, \boldsymbol{y}_O\right) - \hat{\beta}\right)\left(\hat{\boldsymbol{\beta}}_{ML}\left(\tilde{\boldsymbol{y}}_M^{(k)}, \boldsymbol{y}_O\right) - \hat{\beta}\right)' \quad (3.19)$$

Comparing with Rubin's rules (3.11) we can note the similarity except for the extra $\frac{1}{K}$ between imputation variance term, correcting for when the number of imputations is small, included in Rubin's rules.

The MI method relies on that the posterior is asymptotically multivariate normal with a mean and covariance matrix given by likelihood based estimates and their inverse information matrix. It is therefore suggested for logistic regression that one should use the log-odds scale and that a logarithmic scale be used for hazard ratios .

MI is an approximation to the full Bayesian analysis with fewer number of draws due to combining estimates and uncertainty with Rubin's rules and utilizing the speed of ML estimates for computing complete data posterior means and variances. In practice one can ignore that multiple imputation is essentially a Bayesian method, by assuming non-informative priors for parameters.

When the imputations are Bayesian, i.e. the missing values are imputed from their posterior predictive distribution given the observed values, uncertainty about the parameters of the imputation model are propagated correctly. This is achieved by drawing the parameters of the imputation model $\boldsymbol{\theta}^{(k)}$ from their posterior distribution before the missing values are drawn from $f(\boldsymbol{y}_M \mid \boldsymbol{y}_O, \boldsymbol{\theta}^{(k)})$. For a discussion of the frequentist properties of MI see section 2.5 of Carpenter and Kenward (2012) and Rubin (1987).

### Congenial and uncongenial models

However, for the uncertainty about the parameters of the analysis to be valid using multiple imputation the imputation and analysis model are required to be *congenial*. The imputation model and the analysis model are said to be congenial when they correspond. Carpenter and Kenward (2012) give the following explanation of congeniality between imputation and analysis model. Assume MAR and suppose we obtain $K$ imputed data sets from a Bayesian posterior predictive distribution $f(\boldsymbol{y}_M \mid \boldsymbol{y}_O)$, fit our analysis model to each and combine the results for inference using Rubin's rules. Separate to this assume that there exists a full Bayesian procedure for obtaining the posterior of $\boldsymbol{\beta}$ (the parameters of the analysis model) such that if - in addition - we were to use this full Bayesian procedure to impute the missing data, then the imputation distribution would be the same as the posterior predictive distribution $f(\boldsymbol{y}_M \mid \boldsymbol{y}_O)$ used for the multiple imputations. Meng (1994) give a more detailed and extensive definition of congeniality.

When the imputation model (or the imputation model implied by the imputation method) do not align with the analysis model, the models are said to be uncongenial. Uncongeniality tend to result in conservative inferences and can in special cases lead to invalid inferences. Especially inference in the tails can be more sensitive to this. For example, if there is an interaction or non-linear term present in the data, but the imputation model does not include this, then the interaction or non-linear effect will be weakened in the analysis model. On the other hand, an imputation model can be richer than the analysis model when, e.g. it includes additional variables. These *auxilliary variables* in the imputation model are used to impute missing values, but are not included in the analysis model themselves. Nested within an imputation model that is richer than the analysis model there can be an imputation model that is congenial with the analysis model. Multiple imputation using a richer imputation model tend to give more efficient estimates than anticipated.

## 3.2 More on imputing missing values: Imputation using full condition specification (FCS)

It can be challenging to come up with a multivariate distribution that provides a good fit to the data. Often it is easier to specify a conditional distribution for each variable given all the others. This is the idea behind the full conditional specification (FCS) using chained equations approach for imputation. It uses a method similar to a Gibbs sampler.

From here on we will distinguish between response variable(s) and covariates as defined by an analysis model. Let $X_1, \dots, X_q$ be partially observed covariates, let $Z$ be a vector of fully observed covariates and let $Y$ be a fully observed response variable with respect to the analysis model with distribution $f(y \mid x, z, \psi)$, where the goal of the analysis is to draw inferences regarding its parameters.

Further, for $j = 1, \dots, q$, let $\boldsymbol{x}_{j,O}$ and $\boldsymbol{x}_{j,M}$ be the vectors of observed and missing values on $X_j$ across the $n$ individuals, and define the matrices $\boldsymbol{x}_M = (\boldsymbol{x}_{1,M}, \dots, \boldsymbol{x}_{q,M})$ and $\boldsymbol{x}_O = (\boldsymbol{x}_{1,O}, \dots, \boldsymbol{x}_{q,O})$. Let $\boldsymbol{z}$ and $\boldsymbol{y}$ be the matrix and vector of observed values on the fully observed covariates and response across the $n$ individuals.

For the partially observed covariates $j = 1, \dots, q$, specify a conditional density

$$f(x_j | x_{-j}, z, y, \theta_j), \tag{3.20}$$

where $x_{-j}$ is the vector $x_{-j} = (x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_q)$, and a prior $f(\theta_j)$ for its parameter (vector) $\theta_j$. For the $t$th iteration of the algorithm let $\boldsymbol{x}_j^{(t)} = (\boldsymbol{x}_{j,O}, \boldsymbol{x}_{j,M}^{(t)})$ be the vector of all observations across $n$ individuals for covariate $X_j$ with the current vector of imputed missing values $\boldsymbol{x}_{j,M}^{(t)}$. In addition, let $\boldsymbol{x}_{-j}^{(t+1)}$ be the matrix of currently imputed and observed values on all partially observed covariates except $X_j$, i.e. $\boldsymbol{x}_{-j}^{(t+1)} = (\boldsymbol{x}_1^{(t+1)}, \dots, \boldsymbol{x}_{j-1}^{(t+1)}, \boldsymbol{x}_{j+1}^{(t)}, \dots, \boldsymbol{x}_q^{(t)})$. Note that the most current values on each covariate are used.

The FCS algorithm generates a sequence of draws $\boldsymbol{x}_{j,M}^{(t+1)}$ for $j = 1, \dots, q$ from their conditional posterior predictive distributions given the observed data and the current values of the imputed missing data. When the sequence has converged, a generated set of missing values $\boldsymbol{x}_M^{(t+1)}$ is approximately a draw from the (joint) posterior predictive distribution of the missing data given the observed data, i.e. from $f(\boldsymbol{x}_M \mid \boldsymbol{x}_O, \boldsymbol{z}, \boldsymbol{y})$.

In the situation where individuals are independent, for each individual with a missing value on $X_j$ a scalar $x_j^{(t+1)}$ is drawn from the univariate conditional specified model (3.20) using the observed or current imputed values of covariates $X_1, \dots, X_{j-1}, X_{j+1}, \dots, X_q$ for the individual and the current given value $\theta_j^{(t+1)}$,

$$x_j^{(t+1)} \sim f(x_j | x_{-j}^{(t+1)}, z, y, \theta_j^{(t+1)}). \tag{3.21}$$

At (assumed) convergence, the last imputed values of the missing data and the observed values make up one imputed data set.

**Algorithm 2** Full Conditional Specification/Chained Equations imputation

1. Create inital imputations of the missing values $\boldsymbol{x}_{1,M}^{(0)}, \dots, \boldsymbol{x}_{q,M}^{(0)}$ by an approximate method, e.g. by replacing the missing values on each covariate by randomly drawing from the observed values of the same covariate.

2. For $t = 0, 1, \dots$ draw from the following distributions (up to constants of proportionality)

$$\theta_1^{(t+1)} \sim f(\theta_1) f(\boldsymbol{x}_{1,O} \mid \boldsymbol{x}_{-1}^{(t+1)}, \boldsymbol{z}, \boldsymbol{y}, \theta_1)$$

$$\boldsymbol{x}_{1,M}^{(t+1)} \sim f(\boldsymbol{x}_{1,M} \mid \boldsymbol{x}_{-1}^{(t+1)}, \boldsymbol{z}, \boldsymbol{y}, \theta_1^{(t+1)})$$

$$\theta_2^{(t+1)} \sim f(\theta_2) f(\boldsymbol{x}_{2,O} \mid \boldsymbol{x}_{-2}^{(t+1)}, \boldsymbol{z}, \boldsymbol{y}, \theta_2)$$

$$\boldsymbol{x}_{2,M}^{(t+1)} \sim f(\boldsymbol{x}_{2,M} \mid \boldsymbol{x}_{-2}^{(t+1)}, \boldsymbol{z}, \boldsymbol{y}, \theta_2^{(t+1)})$$

$$\vdots$$

$$\theta_q^{(t+1)} \sim f(\theta_q) f(\boldsymbol{x}_{q,O} \mid \boldsymbol{x}_{-q}^{(t+1)}, \boldsymbol{z}, \boldsymbol{y}, \theta_q)$$

$$\boldsymbol{x}_{q,M}^{(t+1)} \sim f(\boldsymbol{x}_{q,M} \mid \boldsymbol{x}_{-q}^{(t+1)}, \boldsymbol{z}, \boldsymbol{y}, \theta_q^{(t+1)})$$

For this to be equivalent to a Gibbs sampler, we would have drawn each $\theta_j$ from its posterior distribution also given the most recent values of the missing data, i.e. $\theta_j^{(t+1)} \sim f(\theta_j) f(\boldsymbol{x}_{j,O}, \boldsymbol{x}_{j,M}^{(t)} \mid \boldsymbol{x}_1^{(t+1)}, \dots, \boldsymbol{x}_{j-1}^{(t+1)}, \boldsymbol{x}_{j+1}^{(t)}, \dots, \boldsymbol{x}_q^{(t)}, \boldsymbol{z}, \boldsymbol{y}, \theta_j)$. In addition, the specified conditional distributions (3.20) must correspond to a joint distribution $f(x_1, \dots, x_q \mid z, y, \theta)$ where it's parameters $\theta$ (and prior) have the appropriate relation to the $\theta_j$s (and their priors). As mentioned, when these conditions are satisfied the sequence converges to the posterior predictive distribution of the missing data given the observed.

In practice this rarely holds and it is unknown what distribution the sequence may converge to, but empirical studies and simulations have shown that multiple imputation using FCS still can perform quite well. The multiple imputation full conditional specification (chained equations) algorithm is implemented in the package `mice` (Buuren and Groothuis-Oudshoorn 2010) in `R`. More details and applications can be found in the textbook by Van Buuren (2018).

## 3.3 Substantive model compatible (FCS) imputation

Substantive model compatible full conditional specification (SMC-FCS) is an expansion of the FCS algorithm that seeks to ensure *compatibility* between the analysis (substantive) model and the imputation model (assuming that the analysis model is correctly specified). Two conditional models are incompatible if there exists no joint distribution for which the conditionals (for the relevant variables) equal these conditional models. Incompatible analysis and imputation models can lead to biased estimates of the parameters in the analysis model. Compatibility is similar and related to the concept of congeniality (see Bartlett et al. (2015) for more details on this). The SMC-FCS algorithm ensures compatibility by indirectly specifying an imputation model while taking account of the analysis model. Since the implied imputation model is generally non-standard, imputations are made using rejection sampling.

For a partially observed covariate $X_j$ we have that

$$f(x_j \mid x_{-j}, z, y) \propto f(y \mid x, z) f(x_j \mid x_{-j}, z) \tag{3.22}$$

where the vector $x_{-j}$ is defined as in the previous section and $x = (x_j, x_{-j})$. To implicitly specify an imputation model for the missing data on the partially observed covariate, for each $X_j$ using the above, we specify a model $f(x_j \mid x_{-j}, z, \phi_j)$ and a non-informative prior $f(\phi_j)$. For given values of the $\psi$ and $\phi_j$ the missing values for $X_j$ can be imputed from the density proportional to

$$f(y \mid x, z, \psi) f(x_j \mid x_{-j}, z, \phi_j). \tag{3.23}$$

The implied imputation model depends on the both $\phi_j$ and $\psi$.

Similarly as before, let $\boldsymbol{x}_j^{(t)}$ be the vector of current imputed values and observed values on covariate $X_j$ across $n$ individuals and now let $\boldsymbol{x}_{-j}^{(t+1)}$ be the matrix of currently imputed and observed values on all partially observed covariates except $X_j$, i.e. $\boldsymbol{x}_{-j}^{(t+1)} = (\boldsymbol{x}_1^{(t+1)}, \ldots, \boldsymbol{x}_{j-1}^{(t+1)}, \boldsymbol{x}_{j+1}^{(t)}, \ldots, \boldsymbol{x}_q^{(t)})$. Assuming independent priors for the parameters such that $f(\phi_j, \psi) = f(\phi_j) f(\psi)$ the posterior $f(\phi_j, \psi \mid \boldsymbol{x}_j^{(t)}, \boldsymbol{x}_{-j}^{(t+1)}, \boldsymbol{z}, \boldsymbol{y})$ is proportional to

$$f(\phi_j) f(\psi) f(\boldsymbol{y} \mid \boldsymbol{x}_j^{(t)}, \boldsymbol{x}_{-j}^{(t)}, \boldsymbol{z}, \psi) f(\boldsymbol{x}_j^{(t)} \mid \boldsymbol{x}_{-j}^{(t+1)}, \boldsymbol{z}, \phi_j). \tag{3.24}$$

In the $(t+1)$th iteration of the SMC-FCS algorithm one draws (up to constants of proportionality)

$$\psi^{(t+1,j)} \sim f(\psi) f(\boldsymbol{y} \mid \boldsymbol{x}_j^{(t)}, \boldsymbol{x}_{-j}^{(t+1)}, \boldsymbol{z}, \psi) \tag{3.25}$$

$$\phi_j^{(t+1)} \sim f(\phi_j) f(\boldsymbol{x}_j^{(t)} \mid \boldsymbol{x}_{-j}^{(t+1)}, \boldsymbol{z}, \phi_j) \tag{3.26}$$

and then imputes the missing values of $X_j$ from the density proportional to (3.23). Comparing with the FCS algorithm, we note that we condition on $\boldsymbol{x}_{j,M}^{(t)}$ in addition to $\boldsymbol{x}_{j,O}, \boldsymbol{x}_{-j}^{(t+1)}, \boldsymbol{z}$, and $\boldsymbol{y}$ as in a standard Gibbs sampler. However, Bartlett et al. (2015) posit that this might require more iterations before the chain reach convergence. In the chain, imputed missing values of $X_j$ can be generated using rejection sampling.

Rejection sampling is an indirect way of simulating draws from a desired distribution called a *target distribution*. The idea is to draw values from a simpler distribution, denoted a *proposal distribution*, that can take on the same values as the target distribution, and then to accept the draws under a certain condition.

To sample from (3.23) the specified density $f(x_j \mid x_{-j}, z, \phi_j)$ (that is easy to sample from) is a proposal density . Typically this will be a normal distribution for continuous $X_j$ and a logistic model for binary $X_j$. Then one must find an upper bound, $c(y, x_{-j}, z, \psi)$, for the ratio of the target density to the proposal density which does not depend on $X_j$. Here the ratio of the target to the proposal density is proportional to

$$\frac{f(y \mid x, z, \psi) f(x_j \mid x_{-j}, z, \phi_j)}{f(x_j \mid x_{-j}, z, \phi_j)} = f(y \mid x, z, \psi). \tag{3.27}$$

To simulate a value from the density proportional to (3.23) we can then draw a proposal value $x_j^*$ from $f(x_j \mid x_{-j}, z, \phi_j^{(t+1)})$ and value $u$ from a standard uniform distribution (on $(0,1)$) and accept the proposed value if

$$u \leq \frac{f(y \mid x_j^*, x_{-j}, z, \psi^{(t+1,j)})}{c(y, x_{-j}, z, \psi^{(t+1,j)})}. \tag{3.28}$$

If the the proposed draw is rejected new values $x_j^*$ and $u$ are repeatedly sampled until the condition given by the inequality is satisfied.

The `smf-fcs` package in `R` offers imputation for SMC-FCS.

## 3.4 Imputing with non-linear terms, interactions and auxiliary variables

In the standard FCS algorithm only main effects are modelled. Very often non-linear effects or interaction effects are of interest to the analyst. When missing values are imputed with a model with only main effects, the imputation is uncongenial or incompatible with analysis models with non-linear and interaction effects. In these situations the non-linear effect or interaction effect is attenuated. There exists modifications to that can e.g. impute with an interaction, but this methods have generally shown bias toward the null for the interaction effect. On the other hand substantive model compatible FCS where the non-linear or interaction term is included in the substantive/analysis model has been shown to be unbiased in many simulation settings, and is generally preferred when non-linear or interaction effects are believed to be present.

One of the biased methods for handling interaction (or non-linear) effects with FCS is the "impute, then transform" method. It imputes the missing data without the interaction and then derives and adds the interaction term after the missing values have been imputed. Another, modification which have shown to give better results is the method of "passive imputation". This methods imputes with the derived variable which is updated in after each iteration. The derived variable, e.g. the interaction, can then be used to impute missing values for variables other than the two included in the interaction. With only one variable partially observed imputation with an interaction term, which includes the partially observed variable, is equivalent for the two modifications. For an overview and more details on this see Van Buuren (2018). The `mice` package in `R` offers functionality for imputation with non-linear and interaction terms.

Auxiliary variables are fully observed variables related to the partially observed variables, but not of interest in the analysis model. If these are available to the imputer then they can be included in the imputation model for the partially observed variable(s). This has been shown to give more effective estimates. For FCS the auxiliary variable can simply be included as a covariate in the linear or logistic regression models used for imputations. In SMC FCS (rejection sampling) the auxiliary variable can be included in both the the analysis model and the proposal distribution. The auxiliary variable is in some situations a surrogate for a partially observed variable, i.e. if the partially observed variable was fully observed and included in the analysis model, then the surrogate variable would have no effect on the outcome. The SMF FCS algorithm can incoropate this assumption by only including the auxiliary variable in the proposal distribution. However, since this is a stronger assumption, we will only use the former rejection sampling with an auxilliary variable procedure.

In the next chapter we are going to return to the analysis of time-to-event data and look at multiple imputation for Cox regression models.

CHAPTER 4

# Multiple imputation in Cox regression for cohort and sampled cohort data

In this chapter we will consider multiple imputation for cohort data with missing covariates in Cox regression, and for sampled cohort data in nested case-control and case-cohort studies. Adapted versions of approximate imputation and rejection sampling for survival data will be presented and illustrated on the example for the full cohort. Then an overview of current approaches for using multiple imputation for sampled cohort data and the empirical results from simulation studies will be given. The approaches for multiply imputing data with approximate imputation and rejection sampling for sampled cohort data will be elaborated in this setting. Finally, the methods will also here be exemplified.

Approximate imputation for Cox regression was developed by White and Royston (2009). Keogh and White (2013) investigated approximate imputation for sampled cohort data. Substantive model compatible FCS using rejection sampling for Cox regression was proposed in Bartlett et al. (2015) and has been further examined for sampled cohorts in Keogh, Seaman, et al. (2018).

## 4.1 MI for Cox missing covariates regression

This section covers multiple imputation for survival data, and more specifically how multiple imputation may be used when the analysis model is a Cox proportional hazards model. Survival data can be seen as a special case of missing data, where we know the range of the values of the missing data. If the event is not observed then the survival time must be larger than the censoring time. However, in many applications the survival time and censoring status are easily available, and the following will focus on imputation for covariates.

We will assume that only a single covariate $X$ is partially observed while a set of covariates denoted by the vector $Z$ and the survival/censoring time and the censoring status represented by the two component vector $Y = (T, \Delta)$ are assumed fully observed as before. Assuming that the censoring mechanism is random/non-informative (each individual has a censoring time that is independent of their survival time) and that the censoring time distribution is independent of $X$ given $Z$ we have under Cox model that the joint distribution of $T$ and $\Delta$ is

$$f(t,\delta \mid x,z) \propto \left\{h_0(t)e^{\beta x+\gamma' z}\right\}^{\delta} \exp\left\{-H_0(t)e^{\beta x+\gamma' z}\right\} \tag{4.1}$$

for the parameters given by the scalar $\beta$ and vector $\gamma$. Our goal is to impute missing values for $X$ from its conditional distribution given the other covariates, the survival/censoring time and the censoring indicator. Using Bayes theorem gives that the conditional distribution for $X$ given values $z$, $t$ and $\delta$ is

$$f(x \mid z,t,\delta) \propto f(t,\delta \mid x,z)f(x \mid z). \tag{4.2}$$

The proportionality constant here does not depend on $x$ so we can combine it with the proportionality constant in (4.1) into a constant $C(z,t,\delta)$ and take the logarithm of (4.2) to obtain

$$\begin{aligned}\log f(x \mid z,t,\delta) =& \log f(x|z) + \log f(t,\delta \mid x,z) + \log C(z,t,\delta) \qquad (4.3)\\ =& \log f(x|z) + \delta \log h_0(t) + \delta\left(\beta x + \gamma' z\right) - H_0(t)e^{\beta x+\gamma' z}\\ &+ \log C(z,t,\delta).\end{aligned}$$

We see that the conditional distribution $f(x \mid z,t,\delta)$ is non-standard. In order to draw values for $X$ from this non-standard distribution, we will consider using an *approximate imputation model* and using *rejection sampling*.

### Approximate imputation

White and Royston (2009) show how an approximation of $f(x \mid z,t,\delta)$ can yield an imputation model for binary or continuous $X$ by including $z$, $\delta$ and $\hat{H}(t)$ as linear predictors. Here $\hat{H}(t)$ is obtained by the Nelson-Aalen estimator (2.4).

To impute values for binary $X$ we assume a logistic model

$$\operatorname{logit}\{P(X=1 \mid z,t,\delta)\} = \eta_0 + \eta_1' z + \eta_2\delta + \eta_3\hat{H}(t) \tag{4.4}$$

where $\operatorname{logit}(p) = \log\frac{p}{1-p}$ is the log-odds for probability $p$. This model is fitted using the currently imputed data set. The resulting estimates and their corresponding covariance matrix can be used as the mean vector $\hat{\eta} = (\hat{\eta}_0, \hat{\eta}_1, \hat{\eta}_2, \hat{\eta}_3)'$ and covariance matrix $\operatorname{Var}(\hat{\eta})$ of a multivariate normal distribution to create independent draws $\eta^{(k)}$ for $k = 1,\ldots,K$. Then for each individual with missing $X$, an imputed $X^{(k)}$ can be drawn from the Bernoulli distribution with probability

$$P(X^{(k)} = 1 \mid z,t,\delta) = \frac{\exp\left(\eta_0^{(k)} + \eta_1^{(k)\prime} z + \eta_2^{(k)}\delta + \eta_3^{(k)}\hat{H}(t)\right)}{1+\exp\left(\eta_0^{(k)} + \eta_1^{(k)\prime} z + \eta_2^{(k)}\delta_i + \eta_3^{(k)}\hat{H}(t_i)\right)}. \tag{4.5}$$

giving the $k$th set of imputed values.

This corresponds to a step of the FCS MI algorithm (2) in section 3.2 with $q = 1$. We fit the model (4.4) to the current imputed dataset and draw the parameters $\eta$ from their posterior distribution, for which we assume a vague prior and that this posterior is approximately multivariate normal. Then we update the missing values from the conditional distribution given the observed values and the drawn parameter values $\eta^{(k)}$ using the approximate model for $f(x \mid z, t, \delta, \eta)$.

To see why the approximation is reasonable White and Royston (2009) consider a logistic model for the binary $X$ given $z$

$$\operatorname{logit}\{P(X = 1 \mid z)\} = \zeta_0 + \zeta_1' z. \tag{4.6}$$

With $f(x \mid z)$ given by (4.6) and by inserting for $X = 1$ and $X = 0$ in (4.3), illustrated by a slightly non-standard use of notation, it can be seen that we obtain the imputation model

$$\begin{aligned}\operatorname{logit}\{P(X = 1 \mid z, t, \delta)\} &= \log f(x = 1 \mid z, t, \delta) - \log f(x = 0 \mid z, t, \delta) \qquad (4.7)\\ &= \zeta_0 + \zeta_1' z + \beta\delta + \left(1 - e^{\beta}\right) e^{\gamma' z} H_0(t)\end{aligned}$$

where the terms that do not depend on $x$ are cancelled. This model is non-standard due to the $e^{\gamma' z}$ term. However with no $z$ it corresponds exactly to a logistic regression with a coefficient for the rightmost term representing $\left(1 - e^{\beta}\right)$. For a single categorical $z$, the model (4.7) is exactly a logistic regression of $x$ on $z$, $\delta$, $H_0(t)$ and the interaction between $z$ and $H_0(t)$.

For other situations there are no exact results, but one may approximate $e^{\gamma' z} \approx e^{\gamma' \bar{z}}$, where $\bar{z}$ is the sample mean (vector) of $z$, or by a more accurate approximation $e^{\gamma' z} \approx e^{\gamma' \bar{z}} \{1 + \gamma' (z - \bar{z})\}$, for small $\operatorname{Var}(\gamma' z)$. The first approximation suggests imputing $x$ with a logistic regression on $z$, $\delta$ and $H_0(t)$, while the second approximation suggests imputing $x$ with a logistic regression on $z$, $\delta$, $H_0(t)$ and the interaction between $z$ and $H_0(t)$.

Furthermore, the simulations of White and Royston (2009) show that models using the Nelson-Aalen estimator to estimate the cumulative baseline hazard and no interaction terms, performed just as well as more advanced models with interactions and the Breslow estimator for $H_0(t)$. This was also the situation for a normal missing variable.

The imputation model when assuming $X \mid z \sim N(\phi_0 + \phi' z, \sigma_{X,Z})$ is

$$X = \eta_0 + \eta_1' z + \eta_2 \delta + \eta_3 \hat{H}(t) + \epsilon, \tag{4.8}$$

where $\epsilon \sim N(0, \sigma_\epsilon^2)$. Similarly as for a binary variable, the model is fitted to the complete observations and obtaining estimates for the parameters $\hat{\eta}$, their estimated covariance matrix $V = \operatorname{Var}(\hat{\eta})$ and the estimated residual variance $\hat{\sigma}_\epsilon^2$. Then drawing $\sigma_\epsilon^{(k)} = \hat{\sigma}_\epsilon \sqrt{(n_{obs} - J)/g}$ where $n_{obs}$ is the number of individuals with $X$ observed, $J$ is the length of $\hat{\eta}$ and $g$ is drawn from chi-squared distribution with degrees of freedom equal to $n_{obs} - J$. Afterwards draw a vector of independent standard normals $\boldsymbol{u}$ to obtain $\eta^{(k)} = \hat{\eta} + \sigma_\epsilon^{(k)} u' V^{\frac{1}{2}}$ where $V^{\frac{1}{2}}$ is the Cholesky decomposition of the matrix $V$. Lastly $\epsilon^{(k)}$ is drawn from a mean-zero normal distribution with variance $\sigma_\epsilon^{2(k)}$. With the set of drawn parameters, the missing values of $X$ are imputed from (4.8). Again see White and Royston (2009) for a similar argument as the one given for logistic regression.

FCS with an approximate imputation model can easily be performed with the `mice` package after precomputing the Nelson-Aalen estimate.

### Imputation using rejection sampling

To impute missing values according to $f(x \mid z, t, \delta)$, Keogh (2018) summarize the SMC-FCS rejection sampling method of Bartlett et al. (2015) into the following algorithm. For imputation $k = 1 \ldots, K$:

**Algorithm 3** Rejection sampling

Repeat steps $1 - 6$ until convergence

1. For the parameters $\psi = (\beta, \gamma)$ of the analysis model (4.1), draw values $\beta^{(k)}$, $\gamma^{(k)}$ from a joint normal distribution with mean $(\hat{\beta}, \hat{\gamma})$ and covariance matrix $\hat{V}$, where the estimates $\hat{\beta}, \hat{\gamma}$ and covariance matrix $\hat{V}$ are obtained from fitting the Cox model to the current "completed" (imputed) dataset. In the first iteration, appropriate initial values for the missing values should be used.

2. Obtain an estimate of the cummulative baseline hazard $H_0^{(k)}(t)$ by the Breslow estimator (2.15) evaluated at the parameter values $\beta^{(k)}$, $\gamma^{(k)}$ and the current imputed values of $X$.

3. Estimate the parameters $\phi$ of the model (e.g. normal or logistic regression) for the proposal distribution $f(x \mid z)$ and their covariance matrix using the current "completed" dataset. Then draw values of the parameters from their estimated joint posterior distribution.

4. For each indvidual with missing $X$ , draw a potential value $x^*$ from the proposal distribution given the parameters drawn in the previous step.

5. Draw a single standard uniform $u$ and accept the potential imputed value $x^*$ if

$$u \leq \begin{cases} \exp\left(-H_0^{(k)}(t) e^{\beta^{(k)} x^* + \gamma^{(k)\prime} z}\right) & \text{if } \delta = 0 \\ H_0^{(k)}(t) \exp\left(1 + \beta^{(k)} x^* + \gamma^{(k)\prime} z - H_0^{(k)}(t) e^{\beta^{(k)} x^* + \gamma^{(k)\prime} z}\right) & \text{if } \delta = 1 \end{cases} \tag{4.9}$$

6. Repeat steps 4 and 5 until a value $x^*$ is accepted for each individual for whom it is missing, giving an updated set of missing values.

At convergence this gives the $k$th imputed dataset with missing values drawn from their posterior predictive distribution.

For survival data with response $(t, \delta)$ the ratio of target to proposal, cfr. (3.27), is proportional to

$$\frac{f(t, \delta \mid x, z, \psi) f(x \mid z, \phi)}{f(x \mid z, \phi)} = f(t, \delta \mid x, z, \psi) \tag{4.10}$$

For the following argument let a censoring time be denoted $C$ and a failure time be denoted $\tilde{T}$, such that the observed time is $T = \min(C, \tilde{T})$. Under the assumption that $\tilde{T}$ is independent of $C$ given $X, Z$ and that $C$ is independent of $X$ given $Z$, the joint distribution for an individual with censored time is

$$f(T = t, \delta = 0 \mid x, z, \psi) = f(\tilde{T} > t, C = t \mid x, z, \psi) \tag{4.11}$$

$$
\begin{aligned}
&= P(\tilde{T} > t \mid x, z, \psi) f(C = t \mid z) \\
&\leq f(C = t \mid z).
\end{aligned}
$$

Thus the unkown censoring time distribution $f(C = t \mid z)$ is an upper bound of the ratio of target to proposal. For Cox model, dividing by this upper bound gives

$$
\begin{aligned}
\frac{f(T = t, \delta = 0 \mid x, z, \psi)}{f(C = t \mid z)} &= P(\tilde{T} > t \mid x, z, \psi) \qquad (4.12) \\
&= \exp\left(-H_0^{(k)}(t) e^{\beta^{(k)} x + \gamma^{(k)\prime} z}\right)
\end{aligned}
$$

which result in the expression in step 5 for $\delta = 0$. A similiar, although somewhat more complicated argument gives the expression for $\delta = 1$ (Bartlett et al. 2015).

The R package `smc-fcs` implements rejection sampling for Cox models.

## 4.2 Example for the full cohort

To illustrate multiple imputation for cohort data we will again consider the survival times for people dying from neoplasms (as introduced in Section 2.2). We let $\log_2 \lambda$ be partially observed and all the other covariates and the response variables be fully observed. Define the missingness mechanism for $\log_2 \lambda$ as

$$
f(r_\lambda = 0 \mid t, \delta) = \begin{cases} 0.8 & \text{if } t \leq t_{50}, \delta = 0 \\ 0.3 & \text{if } t > t_{50}, \delta = 0 \\ 0 & \text{else} \end{cases}, \qquad (4.13)
$$

where $t_{50}$ is the median follow-up time. Now $\log_2 \lambda$ is MAR given $(t, \delta)$. This means that the probability of a value for $\log_2 \lambda$ being missing is higher for non-events with shorter censoring times than for individuals with longer censoring times, and overall that non-events have missing values while individuals who experience the event do not. This could be plausible if the capacity for measuring $\log_2 \lambda$ is limited and cases are prioritised, such that for individuals newly entered into the study (which would have short censored times at the time of the analysis) the measurements are less likely to have been obtained.

Let $x$ denote the partially observed $\log_2 \lambda$ and let $z$ denote the fully observed covariates age and sex. We then have that

$$
f(t, \delta \mid x, z, r = 1) = \frac{f(r = 1 \mid x, z, t, \delta)}{f(r = 1 \mid x, z)} f(t, \delta \mid x, z). \qquad (4.14)
$$

Since the missingness mechanism depends on the response variables, the complete observation analysis will generally be biased.

Applying the missingness mechanism on our cohort yields 2713 missing values such that the percentage of missing values on $\log_2 \lambda$ is roughly 50%. First an analysis using just the 2773 complete observations is performed. Then missing values of $\log_2 \lambda$ will be imputed $K = 5$ times with the approximate imputation model (4.8) in FCS with maximum 100 iterations and with rejection sampling in SMC-FCS using 100 iterations. For the latter $\log_2 \lambda$ given the other covariates will be assumed normally distributed. Then the models are fitted and the results combined with Rubin's rules.

Table 4.1: Estimated regression coefficients for full cohort, complete observations and multiple imputation using approximate imputation and rejection sampling for $K = 5$. Multiple imputation is performed with and without an auxiliary variable.

| | coef | exp(coef) | se(coef) | z | p |
|---|---|---|---|---|---|
| | | | Full cohort | | |
| age | 0.066 | 1.068 | 0.010 | 6.485 | <0.001 |
| sexM | 0.250 | 1.284 | 0.115 | 2.170 | 0.030 |
| loglambda | 0.538 | 1.712 | 0.096 | 5.618 | <0.001 |
| | | | Complete observations | | |
| age | 0.062 | 1.064 | 0.010 | 6.100 | <0.001 |
| sexM | 0.264 | 1.303 | 0.115 | 2.293 | 0.022 |
| loglambda | 0.614 | 1.848 | 0.098 | 6.296 | <0.001 |
| | | | Approximate imputation | | |
| age | 0.067 | 1.069 | 0.010 | 6.560 | <0.001 |
| sexM | 0.256 | 1.291 | 0.116 | 2.213 | 0.027 |
| loglambda | 0.523 | 1.687 | 0.099 | 5.256 | <0.001 |
| | | | Rejection sampling | | |
| age | 0.067 | 1.069 | 0.010 | 6.532 | <0.001 |
| sexM | 0.259 | 1.296 | 0.116 | 2.240 | 0.025 |
| loglambda | 0.572 | 1.772 | 0.100 | 5.734 | <0.001 |
| | | | Approximate imputation auxilliary | | |
| age | 0.066 | 1.069 | 0.010 | 6.499 | <0.001 |
| sexM | 0.249 | 1.283 | 0.116 | 2.153 | 0.031 |
| loglambda | 0.522 | 1.686 | 0.103 | 5.081 | <0.001 |
| | | | Rejection sampling auxilliary | | |
| age | 0.067 | 1.069 | 0.010 | 6.569 | <0.001 |
| sexM | 0.252 | 1.287 | 0.116 | 2.180 | 0.029 |
| loglambda | 0.525 | 1.691 | 0.099 | 5.280 | <0.001 |

Results are shown in Table 4.1. We see that the complete observations analysis yields a markedly higher effect for $\log_2 \lambda$. The standard errors do not indicate that information on about 50% of the individuals have been discarded, which is likely because none of the cases are missing. Approximate imputation reduces the overestimated effect of $\log_2 \lambda$ from the complete observations analysis. Compared to the full cohort analysis, the estimate is slightly lower. The standard errors for the fully observed variables are very nearly the same as for the full cohort analysis while there is an additional uncertainty in the standard error for $\log_2 \lambda$. Rejection sampling give fairly similar results for age and sex as for the full cohort and approximate imputation. The estimate for $\log_2 \lambda$ is closer to the full cohort than the complete cases, but is for this partially observed cohort further away than approximate imputation.

To illustrate the use of an auxiliary variable for imputation we will include $\log_2 \kappa$ in the imputation model. The auxiliary variable $\log_2 \kappa$ is, as mentioned in the example in Section 2.2, another FLC measurement and it is moderately correlated with $\log_2 \lambda$. The analysis model will still only include age, sex and $\log_2 \lambda$ (although for rejection sampling the imputation model will be compatible with an analysis model with $\log_2 \kappa$ included). The correlation between the two FLC measurements is 0.68. The coefficient estimate and standard error for $\log_2 \kappa$ are both lower (0.464 and 0.087 ) than for $\log_2 \lambda$ if it were to replace it in the full cohort analysis. The estimates for age would be unchanged while the effect for being male would be slightly higher (0.227) with $\log_2 \kappa$ in the model instead of the original $\log_2 \lambda$. When both $\log_2 \lambda$ and $\log_2 \kappa$ are in the full cohort analysis model their estimates are both attenuated to 0.330 and 0.252 respectively. Applying approximate imputation and rejection sampling on the same partially observed single cohort with $\log_2 \kappa$ as an auxiliary variable give the results in the lower part of Table 4.1. Using an auxiliary variable that is predictive of $\log_2 \lambda$ give results that are much more similar for the two methods. The estimate for $\log_2 \lambda$ is considerably reduced for rejection sampling and slightly reduced for approximate imputation.

Repeating 1000 simulations of $\boldsymbol{r}$ according to (4.13) and analysing the resulting data sets with complete observations, approximate imputation sampling and rejection sampling give the results in the upper part of Table 4.2. The relative differences between the methods are similar to the results in Table 4.1. We note that there is still indication of bias (compared to the full cohort) for the estimated effects using the complete observations, and that there now is less bias for the estimated effect of $\log_2 \lambda$ from rejection sampling than from approximate imputation.

For imputation with an auxiliary variable both methods yield lower estimated effect for $\log_2 \lambda$ than the full cohort. There is little difference between approximate imputation with and without an auxiliary variable except for slightly more efficient estimates. Rejection sampling without an auxiliary variable overestimates the effect while the effect is underestimated when the auxiliary variable is included. The underestimation might be because $\log_2 \kappa$ still seem to be associated with the hazard rate when $\log_2 \lambda$ is in the model, i.e. it is not a proper surrogate variable for $\log_2 \lambda$. Note, however, that the full cohort estimate is included in the 95% intervals for all imputation methods. □

Table 4.2: Estimated regression coefficients of 1000 runs for multiple imputation by approximate imputation and rejection sampling for $K = 5$, and for complete observations analysis.

| | coef | 95% interval | se(coef) | 95% interval | rel.efficiency |
|---|---|---|---|---|---|
| | | | Full cohort | | |
| age | 0.066 | - | 0.010 | - | 1 |
| sexM | 0.250 | - | 0.115 | - | 1 |
| loglambda | 0.538 | - | 0.096 | - | 1 |
| | | | Complete observations | | |
| age | 0.059 | [0.055, 0.064] | 0.010 | [0.010, 0.010] | 0.993 |
| sexM | 0.274 | [0.226, 0.322] | 0.115 | [0.115, 0.116] | 0.999 |
| loglambda | 0.583 | [0.532, 0.636] | 0.097 | [0.093, 0.100] | 0.981 |
| | | | Approximate imputation | | |
| age | 0.067 | [0.065, 0.068] | 0.010 | [0.010, 0.010] | 0.993 |
| sexM | 0.252 | [0.236, 0.268] | 0.116 | [0.115, 0.116] | 0.993 |
| loglambda | 0.506 | [0.447, 0.567] | 0.101 | [0.096, 0.110] | 0.893 |
| | | | Rejection sampling | | |
| age | 0.066 | [0.065, 0.068] | 0.010 | [0.010, 0.010] | 0.992 |
| sexM | 0.249 | [0.233, 0.264] | 0.116 | [0.115, 0.117] | 0.992 |
| loglambda | 0.551 | [0.495, 0.604] | 0.101 | [0.095, 0.109] | 0.905 |
| | | | Approximate imputation auxilliary | | |
| age | 0.067 | [0.066, 0.068] | 0.010 | [0.010, 0.010] | 0.995 |
| sexM | 0.252 | [0.240, 0.263] | 0.116 | [0.115, 0.116] | 0.996 |
| loglambda | 0.507 | [0.464, 0.551] | 0.101 | [0.097, 0.106] | 0.902 |
| | | | Rejection sampling auxilliary | | |
| age | 0.067 | [0.066, 0.068] | 0.010 | [0.010, 0.010] | 0.995 |
| sexM | 0.251 | [0.239, 0.263] | 0.116 | [0.115, 0.116] | 0.996 |
| loglambda | 0.507 | [0.468, 0.548] | 0.101 | [0.097, 0.107] | 0.902 |

## 4.3 Overview of multiple imputation for sampled cohort data

For nested case-control and case-cohort studies follow-up times and event indicators are assumed to be observed for the full cohort. Measurements on all relevant covariates for each individual in the nested case-control sample or the case-control sample are collected. One covariate (or more) intended to be collected can be expensive in the sense that obtaining its measurement demands extensive resources which is motivating the need for a sampled cohort study because it is unfeasible to collect for everyone. Meanwhile there will often be information available for all individuals in the full cohort on easily observed covariates such as age, gender and in some situations a surrogate for an expensive covariate. The expensive variable is then missing by design in the full cohort. Multiple imputation is a promising approach for utilising more of the information available to the analyst.

Multiple imputation using approximate imputation and rejection sampling was investigated by Keogh and White (2013) for sampled cohort data. They assumed one continuous expensive covariate missing by design on all individuals in the full cohort not sampled in the nested case-control or case-cohort sample. All follow-up times, event indicators and other covariates or a surrogate were assumed completely observed in the full cohort. Missing values for the expensive covariate were then multiply imputed for full cohort analyses where the results were combined according to Rubin's rules. Compared with the traditional nested case-control and case-cohort analysis multiple imputation was in simulation studies found to result in gains in efficiency, particularly for the covariates observed in the full cohort and when the number of controls in a nested case-control study was small or the case-cohort subcohort was small compared to the full cohort. Of the two methods, approximate imputation was found to give biased estimates for the parameters for large effect sizes and in the presence of an interaction term between the missing expensive covariate and a fully observed covariate in the cohort. This is not unexpected because the approximate imputation model without an interaction term does not contain a nested imputation model that is congenial with an analysis model with the interaction term. The same applies to the situation with non-linear terms of the missing covariate. Rejection sampling gave no apparent biases.

Multiple imputation is a way to make use of the full cohort data available. An advantage with multiple imputation is that handling of data missing by chance, i.e. not by design, can easily be incorporated into the anaysis. Keogh, Seaman, et al. (2018) investigated MI with data missing by design on one expensive covariate $X_1$ only observed in the sampled cohort and data missing by chance (10% and 50% missingness) on a cheap covariate $X_2$ observed for the full cohort where the probability of being missing was dependent on a binary fully observed confounder $Z$, the censoring status and their interaction. Their results showed that MI handles missing data well in nested case-control and case-cohort studies when the imputation model is approximately correctly specified. Relative to a complete observation analysis MI gives bias correction and gains in efficiency. They also considered an intermediate approach where they imputed for the full cohort but only used the sampled cohort for fitting analysis models. This was shown to be more robust to misspecification of imputation model.

Also for values of the cheap covariate missing by chance in the sampled cohort they considered imputing only in the sampled cohort and fitting the analysis model with the traditional nested case-control and case-cohort estimators on the "complete" sampled cohort. This approach was found to reduce bias and give gains in efficiency compared to the complete observation sampled cohort analysis.

From the mentioned studies MI can be expected to give gains in efficiency over standard sampled cohort analysis when information on the full cohort is available. When data is missing by chance MI can give bias correction and gain in efficiency compared to a complete observation analysis. However, it comes at a price of potential bias if imputation model is misspecified. The effect of misspecification may be non-negligible when a large fraction of missing data is imputed as it is if data for the full cohort is imputed.

## 4.4 Full cohort multiple imputation for nested case-control and case-cohort samples

The sampled cohort in a nested case-control study consists of all cases and, for each case, a small number of controls $m - 1$ that are sampled from the risk set at the each case's failure time. The nested case-cohort sample is sampled according to observed event indicator $\delta$ and follow-up time $t$. We again assume that data is missing only on an expensive variable $x$ for those individuals not sampled in the nested case-control study. The missingness mechanism, which is by design, means that if $\delta = 1$ then $x$ is not missing, but if $\delta = 0$ then $x$ is missing depending on the follow-up time. For example, an individual that is censored early (such that it is not at risk at most of the observed event times) will be less likely to be included in the sampled cohort than an individual that is censored later. We know that $x$ is MAR given $t$ and $\delta$.

Both approximate imputation and rejection sampling can be directly applied to nested case-control data. The Nelson-Aalen estimate, which incorporates $t$ in the approximate model can be estimated from the full cohort information on $t$ and $\delta$. In the rejection sampling algorithm proposed values are drawn from $f(x \mid z, \phi^{(t+1)})$. Estimation of the parameters of this distribution using the nested case-control sample can for the early iterations be biased which leads to potentially many rejected draws. This might affect the time, but the draws we end at up with at convergence are still approximately from the target distribution.

The case-cohort sample consists of the sampled subcohort $\mathcal{C}$ and of all cases in the cohort. Missing values $x$ outside the case-cohort sample are MAR given $\delta$. Compared to the nested case-control sample, another difference is that the subcohort $\mathcal{C}$ is a random sample from the full cohort which we will consider further in the next chapter. Both multiple imputation algorithms may be applied directly in the same way as for nested case-control samples when imputing the full cohort.

## 4.5 Example sampled cohort

The sampled cohorts in this section have the same setup as in chapter 2. Nested case-control sampling is performed with 2 controls ($m = 3$) and case-cohort sampling with a subcohort size of 593 randomly sampled individuals from the full cohort. The percentage of individuals with missing $\log_2 \lambda$ value is about 84%. The same setup as the previous example with 100 iterations and 5 imputations is used. Approximate imputation and rejection sampling without and with an auxiliary variable are performed.

In Table 4.3 we see the results of mutiply imputing the full cohort from the same single nested case-control sample as in the example of Section 2.3. Without an auxiliary variable both imputation methods yield estimates closer to the full cohort than the traditional nested case-control estimate and with smaller standard errors. The estimates for $\log_2 \lambda$ are both downward biased compared to the full cohort estimate. With an auxiliary variable both methods give results close to the full cohort with the estimate of $\log_2 \lambda$ from approximate imputation lower and the estimate from rejection sampling higher than the full cohort estimate.

Table 4.3: Nested case-control sampling with 2 controls ($m = 3$): Estimated regression coefficients for full cohort, complete observations and multiple imputation using approximate imputation and rejection sampling for $K = 5$.

| | coef | exp(coef) | se(coef) | z | p |
|---|---|---|---|---|---|
| | | Full cohort | | | |
| age | 0.066 | 1.068 | 0.010 | 6.485 | <0.001 |
| sexM | 0.250 | 1.284 | 0.115 | 2.170 | 0.030 |
| loglambda | 0.538 | 1.712 | 0.096 | 5.618 | <0.001 |
| | | Traditional nested case-control | | | |
| age | 0.062 | 1.063 | 0.013 | 4.827 | <0.001 |
| sexM | 0.198 | 1.219 | 0.143 | 1.383 | 0.167 |
| loglambda | 0.477 | 1.611 | 0.129 | 3.705 | <0.001 |
| | | Approximate imputation | | | |
| age | 0.065 | 1.068 | 0.010 | 6.306 | <0.001 |
| sexM | 0.253 | 1.288 | 0.118 | 2.147 | 0.032 |
| loglambda | 0.508 | 1.661 | 0.130 | 3.904 | 0.001 |
| | | Rejection sampling | | | |
| age | 0.068 | 1.070 | 0.010 | 6.639 | <0.001 |
| sexM | 0.265 | 1.304 | 0.117 | 2.271 | 0.023 |
| loglambda | 0.509 | 1.663 | 0.150 | 3.380 | 0.001 |
| | | Approximate imputation auxilliary | | | |
| age | 0.066 | 1.069 | 0.010 | 6.419 | <0.001 |
| sexM | 0.267 | 1.306 | 0.117 | 2.290 | 0.022 |
| loglambda | 0.514 | 1.672 | 0.114 | 4.516 | <0.001 |
| | | Rejection sampling auxilliary | | | |
| age | 0.067 | 1.070 | 0.010 | 6.594 | <0.001 |
| sexM | 0.253 | 1.288 | 0.117 | 2.162 | 0.031 |
| loglambda | 0.552 | 1.737 | 0.121 | 4.562 | <0.001 |

The results of multiply imputing the full cohort from 1000 nested case-control samples are shown in table Table 4.4. The estimated log hazard ratios for $\log_2 \lambda$ are closer to zero than the traditional nested-case control estimate for imputation without an auxiliary variable, but the mean standard error and width of the 95% intervals reduced. Estimation for the fully observed variables are more efficient and very similar to the full cohort estimates. With an auxiliary variable the estimated coefficient for $\log_2 \lambda$ is closer to the full cohort and the relative efficiency increased for both methods.

For the case-cohort sample from Section 2.4 the multiple imputation results are presented in Table 4.5. With this sample the complete observations analysis estimates for $\log_2 \lambda$ are even lower than for the nested case-control sample. All imputation estimates for $\log_2 \lambda$ are lower than the estimate from the full cohort, but the two with an auxiliary variables in the imputation model are lifted closer to the full cohort estimate. Will will see below that compared to the averaged results over 1000 repetitions the results from this single case-cohort is somewhat

Table 4.4: Nested case-control sampling with 2 controls ($m = 3$): Estimated regression coefficients of 1000 runs for multiple imputation by approximate imputation and rejection sampling for $K = 5$, and for complete observations analysis.

| | coef | 95 % interval | se(coef) | 95% interval | efficiency |
|---|---|---|---|---|---|
| Full cohort | | | | | |
| age | 0.066 | - | 0.010 | - | 1 |
| sexM | 0.250 | - | 0.115 | - | 1 |
| loglambda | 0.538 | - | 0.096 | - | 1 |
| Traditional nested case-control | | | | | |
| age | 0.064 | [0.049, 0.081] | 0.013 | [0.012, 0.014] | 0.615 |
| sexM | 0.226 | [0.061, 0.403] | 0.148 | [0.142, 0.153] | 0.607 |
| loglambda | 0.478 | [0.337, 0.617] | 0.126 | [0.116, 0.136] | 0.577 |
| Approximate imputation | | | | | |
| age | 0.066 | [0.063, 0.069] | 0.010 | [0.010, 0.011] | 0.960 |
| sexM | 0.249 | [0.213, 0.281] | 0.118 | [0.116, 0.122] | 0.962 |
| loglambda | 0.454 | [0.328, 0.575] | 0.116 | [0.093, 0.154] | 0.667 |
| Rejection sampling | | | | | |
| age | 0.066 | [0.063, 0.069] | 0.010 | [0.010, 0.011] | 0.958 |
| sexM | 0.251 | [0.212, 0.283] | 0.118 | [0.116, 0.122] | 0.961 |
| loglambda | 0.470 | [0.336, 0.594] | 0.117 | [0.093, 0.157] | 0.660 |
| Approximate imputation auxilliary | | | | | |
| age | 0.067 | [0.064, 0.069] | 0.010 | [0.010, 0.010] | 0.979 |
| sexM | 0.248 | [0.218, 0.276] | 0.117 | [0.115, 0.119] | 0.978 |
| loglambda | 0.488 | [0.393, 0.592] | 0.109 | [0.094, 0.133] | 0.767 |
| Rejection sampling auxilliary | | | | | |
| age | 0.067 | [0.064, 0.069] | 0.010 | [0.010, 0.011] | 0.979 |
| sexM | 0.248 | [0.217, 0.276] | 0.117 | [0.115, 0.119] | 0.978 |
| loglambda | 0.494 | [0.391, 0.595] | 0.109 | [0.094, 0.135] | 0.760 |

from the mean, but it illustrates the variability of the methods.

The results for many case-cohort samples given in Table 4.6 follow the same pattern. The imputation methods give estimates for the partially observed $\log_2 \lambda$ that are downward biased, but less so when imputing with an auxiliary variable. On the other hand, the estimated relative efficiency is higher with imputation. The reason for the downward bias could be due to convergence issues, the low number of imputations (5) or a degree of model misspecification.

This illustrates the danger of imputing a large percentage of missing values in real world settings. Still the gain in efficiency, especially for the fully observed variables, is clear, and with a good auxiliary variable the potential bias of the partially observed variable might be acceptable. □

In the next chapter we will examine these methods further using simulations studies and investigate the possibility of imputing only a part of the full cohort.

Table 4.5: Case-cohort sampling with a subcohort of size $\tilde{m} = 593$: Estimated regression coefficients for full cohort, complete observations and multiple imputation using approximate imputation and rejection sampling for $K = 5$. Multiple imputation is performed with and without an auxiliary variable.

| | coef | exp(coef) | se(coef) | z | p |
|---|---|---|---|---|---|
| | | Full cohort | | | |
| age | 0.066 | 1.068 | 0.010 | 6.485 | <0.001 |
| sexM | 0.250 | 1.284 | 0.115 | 2.170 | 0.030 |
| loglambda | 0.538 | 1.712 | 0.096 | 5.618 | <0.001 |
| | | Traditional case-cohort (IPW) | | | |
| age | 0.068 | 1.070 | 0.013 | 5.261 | <0.000 |
| sexM | 0.342 | 1.407 | 0.146 | 2.343 | 0.019 |
| loglambda | 0.468 | 1.598 | 0.120 | 3.914 | <0.001 |
| | | Approximate imputation | | | |
| age | 0.069 | 1.071 | 0.010 | 6.555 | <0.001 |
| sexM | 0.259 | 1.296 | 0.118 | 2.198 | 0.028 |
| loglambda | 0.405 | 1.499 | 0.091 | 4.438 | <0.001 |
| | | Rejection sampling | | | |
| age | 0.068 | 1.070 | 0.010 | 6.574 | <0.001 |
| sexM | 0.264 | 1.302 | 0.116 | 2.268 | 0.023 |
| loglambda | 0.335 | 1.398 | 0.104 | 3.223 | 0.001 |
| | | Approximate imputation auxilliary | | | |
| age | 0.068 | 1.070 | 0.010 | 6.590 | <0.001 |
| sexM | 0.243 | 1.275 | 0.118 | 2.053 | 0.040 |
| loglambda | 0.412 | 1.511 | 0.106 | 3.904 | <0.001 |
| | | Rejection sampling auxilliary | | | |
| age | 0.067 | 1.070 | 0.010 | 6.593 | <0.001 |
| sexM | 0.249 | 1.283 | 0.116 | 2.146 | 0.032 |
| loglambda | 0.384 | 1.468 | 0.107 | 3.605 | <0.001 |

Table 4.6: Case-cohort sampling: Estimated regression coefficients of 1000 runs for multiple imputation by approximate imputation and rejection sampling for $K = 5$.

| | coef | 95 % interval | se(coef) | 95% interval | efficiency |
|---|---|---|---|---|---|
| | | Full cohort | | | |
| age | 0.066 | - | 0.010 | - | 1 |
| sexM | 0.250 | - | 0.115 | - | 1 |
| loglambda | 0.538 | - | 0.096 | - | 1 |
| | | Traditional case-cohort (IPW) | | | |
| age | 0.067 | [0.052, 0.083] | 0.013 | [0.012, 0.013] | 0.633 |
| sexM | 0.248 | [0.075, 0.424] | 0.147 | [0.145, 0.149] | 0.617 |
| loglambda | 0.547 | [0.393, 0.721] | 0.127 | [0.117, 0.138] | 0.567 |
| | | Approximate imputation | | | |
| age | 0.066 | [0.063, 0.069] | 0.010 | [0.010, 0.011] | 0.960 |
| sexM | 0.249 | [0.212, 0.285] | 0.118 | [0.116, 0.122] | 0.961 |
| loglambda | 0.454 | [0.327, 0.595] | 0.115 | [0.092, 0.150] | 0.683 |
| | | Rejection sampling | | | |
| age | 0.067 | [0.064, 0.069] | 0.010 | [0.010, 0.011] | 0.963 |
| sexM | 0.257 | [0.219, 0.288] | 0.117 | [0.115, 0.121] | 0.967 |
| loglambda | 0.410 | [0.276, 0.548] | 0.115 | [0.092, 0.156] | 0.675 |
| | | Approximate imputation auxilliary | | | |
| age | 0.066 | [0.064, 0.069] | 0.010 | [0.010, 0.010] | 0.978 |
| sexM | 0.247 | [0.217, 0.274] | 0.117 | [0.115, 0.119] | 0.977 |
| loglambda | 0.502 | [0.405, 0.605] | 0.108 | [0.092, 0.133] | 0.777 |
| | | Rejection sampling auxilliary | | | |
| age | 0.066 | [0.064, 0.069] | 0.010 | [0.010, 0.010] | 0.979 |
| sexM | 0.248 | [0.218, 0.276] | 0.117 | [0.116, 0.119] | 0.976 |
| loglambda | 0.496 | [0.396, 0.601] | 0.108 | [0.093, 0.135] | 0.775 |

CHAPTER 5

# Imputing only a subset of the full cohort and simulation studies

In this chapter multiple imputation in a subset of the full cohort will be explored. The multiple imputation methods for missing data (by chance) in sampled cohort studies of Keogh, Seaman, et al. (2018) will here be applied in a new setting where data will be missing by design in a superset of the sampled cohort. We will denote this superset method A. This setting addresses how to use multiple imputation with only a subset of the full cohort when it can be very large. The sampled cohort MI methods will be slightly adapted and considered in a previously unexamined setting (superset method B). The methods will be examined with simulation studies and compared to the classical sampled cohort methods and the full cohort MI methods for nested case-control and case-cohort samples (Section 4.4). The simulation setup is guided by Morris, White, and Crowther (2019).

## 5.1 Imputing only a subset of the full cohort

In typical nested case-control and case-cohort studies a small fraction of the full cohort is included in the sampled cohort. The starting point for the investigation of the methods described in this chapter will be sampled cohorts for either nested-case control or case-cohort studies. With sampled cohorts it is meant both the cases and the sampled controls. In the sampled cohorts all values are observed while expensive covariate values are missing in the remaining part of the full cohort.

In the previous chapter we used the fully observed variables in the sampled cohort and the fully observed cheap covariates values in the remaining part of the full cohort to gain imputed datasets of the full cohort. However, as mentioned there, when imputing a large fraction of missing data, multiple imputation is less robust to misspecification of the imputation model. Therefore the gain in efficiency might come at the expense of stronger modelling assumptions. For very large cohorts multiple imputation of many variables, where each imputation requires a sequence of iterations to reach convergence, the computational demand of imputing for the full cohort might be prohibitively large. Also, information on an excessive amount of non-cases might be superfluous. These are arguments for imputing missing values in only a part of the full cohort.

The most straightforward approach is then to impute missing values in a random subset of the full cohort, e.g. for 25%, 50% or 75% of the full cohort. The imputed dataset (the sampled cohort study and the imputed missing values for the subset of the full cohort) can then be analysed with Cox model as in full cohort imputation. However, this might introduce bias into the parameter estimates because the imputed dataset is not a representative subsample of the full cohort.

## 5.2 Superset multiple imputation

Instead of imputing the full cohort, we here propose to impute a superset of the sampled cohort. The idea is that the imputed datasets, which then are supersets of the sampled cohort, can be analysed effectively with a sampled cohort estimator. This will be called superset nested case-control (multiple) imputation and superset case-cohort imputation since we will impute missing values such that we obtain supersets of the nested case-case control or case-cohort sample. The situation is illustrated by Figure 5.1.

When obtaining the sampled cohort and the superset we must make sure that the superset is indeed a superset of the sampled cohort. For a nested case-control sample with one control we could first draw a nested case-control superset of e.g. three controls per case and then draw one control from each of the three to make up the nested case-control sample. Or one could first draw one control per case and then draw two additional controls from the corresponding risk sets with the first control removed. In this simple scheme, both ways should be equivalent. For a case-cohort sample with a subcohort of 750 individuals one could first draw a superset with a subcohort of say 1750 individuals and then draw the intended subcohort of 750 from those, or first draw 750 and then draw an additional 1000 from the full cohort with the 750 removed. Since the subcohort in case-cohort studies can be decided in advance the latter might be preferred.

With the same covariates and missingness pattern as in chapter 4 we wish to impute values for $X$ from its conditional distribution given the outcome and the other covariates. Since the superset (or any nested case-control or case-cohort sample) is sampled with respect to the $\delta$ and $t$ the missing values (by design) for $X$ are MAR, and $f(x \mid t, \delta, z, s = 1, \theta)$, where $s = 1$ is an indicator for being in the superset, is equal to $f(x \mid t, \delta, z, \theta)$ . (More specifically we have that $\delta = 0$ here as well).

### Superset imputation method A

Keogh, Seaman, et al. (2018) developed methods for imputing values missing by chance in sampled cohort studies. The methods for imputing missing values only in the sampled cohort immediately adapts to supersets (larger sampled cohorts). For approximate imputation the Nelson-Aalen estimator, $\hat{H}(t)$, in (4.4) or (4.8), can be fitted using the times and censoring status for all individuals in the full cohort.

For rejection sampling with nested case-control supersets, the classical nested case-control estimator, the maximiser of (2.17), is used to obtain the substantive/analysis model parameters in step 1 of the rejection sampling algorithm (Algorithm 3). These drawn parameters, $\hat{\beta}_{ncc}$, are then used in a modified step 2 to obtain the cumulative baseline hazard estimate. The

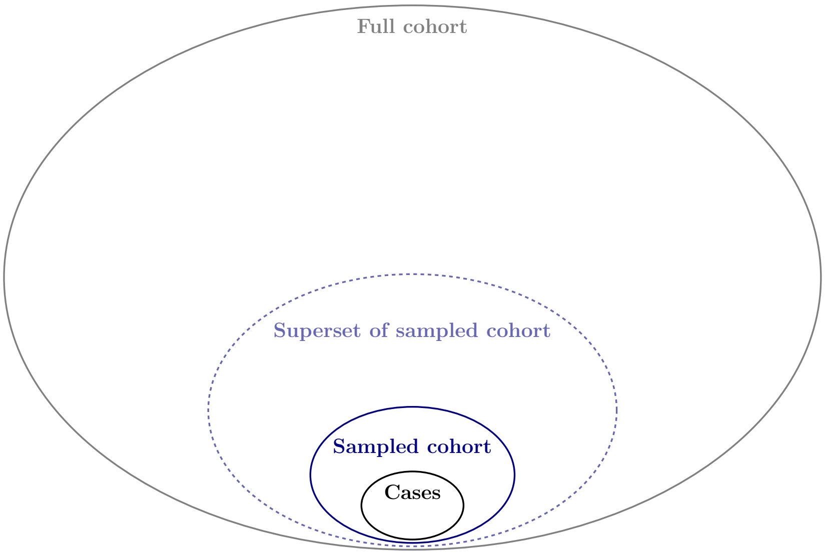


Figure 5.1: Illustration of superset imputation

cumulative baseline hazard in the full cohort can be estimated with the Breslow type estimator of Langholz and Borgan (1997). For a nested case-control superset it can be written

$$\hat{H}_0^{ncc}(t) = \sum_{t_i \leq t} \frac{\delta_i}{\sum_{k \in \tilde{\mathcal{R}}_s(t_i)} \frac{|\mathcal{R}(t_i)|}{m_s} \exp(\hat{\beta}_{ncc,x} x_k + \hat{\beta}_{ncc,z_1} z_{1,k} + \hat{\beta}_{ncc,z_2} z_{2,k})} \tag{5.1}$$

where $\tilde{\mathcal{R}}_s(t_i)$ is the nested case-control superset risk set at event time $t_i$, $|\mathcal{R}(t_i)|$ is the number at risk in the full cohort at $t_i$ and $m_s - 1$ is the number of controls in the superset. Since $f(x \mid z, s = 1)$ will tend to differ from $f(x \mid z)$ only the non-events can be used to estimate $f(x \mid z)$ (assuming rare events). Then, in step 5 the estimate of the substantive model parameters, $\hat{\beta}_{ncc}$, and the cumulative baseline hazard, $\hat{H}_0^{ncc}(t)$, are used to reject/accept proposed values.

For case-cohort supersets the substantive model parameters, $\hat{\beta}_{cch}$, can be obtained with Prentice's estimator, the maximiser of (2.20), which is more in accordance with the following cumulative baseline hazard estimator, $\hat{H}_0^{cch}(t)$, than the IPW estimator:

$$\hat{H}_0^{cch}(t) = \sum_{t_i \leq t} \frac{\delta_i}{\sum_{k \in \tilde{\mathcal{S}}_s(t_i)} \exp(\hat{\beta}_{cch,x} x_k + \hat{\beta}_{cch,z_1} z_{1,k} + \hat{\beta}_{cch,z_2} z_{2,k}) \frac{n}{|\tilde{\mathcal{S}}_s(0)|}}. \tag{5.2}$$

Here $\frac{n}{|\tilde{\mathcal{S}}_s(0)|}$ is the inverse superset subcohort sampling fraction. To estimate $f(x \mid z)$ only the subcohort can be used (since the subcohort is a random sample of the full cohort). Then, in the acceptance/rejection step, we use $\hat{\beta}_{cch}$ and $\hat{H}_0^{cch}(t)$.

Finally, for both superset designs with approximate imputation or rejection sampling, each imputed superset is analysed with its appropriate sampled cohort estimator, the classical nested case-control or the Prentice estimator, and the results are combined according to Rubin's rules.

These methods will be referred to as Superset A methods and are just the methods of Keogh, Seaman, et al. (2018) applied in the superset sampling design setting. The aim is to estimate the imputation model for $X$, $f(x \mid t, \delta, z, \theta)$, using more available information in the remaining part of the full cohort without imputing for the full cohort. This is achieved by estimating the cumulative hazard of the full cohort (population), or the cumulative baseline hazard and the substantive model parameters of the full cohort (population). The superset A methods are effective in that they use information on time and censoring status for all individuals in the remaining full cohort for approximate imputation (Nelson-Aalen estimate) and for rejection sampling with nested case-control samples (size of full cohort risk sets). Rejection sampling for case-cohort uses the sampling fraction.

In `R` these methods are implemented as `smcfcs.nestedcc` and `smcfcs.casecohort` in the `smcfcs` package. Also note that Keogh, Seaman, et al. (2018) mentions another way of estimating $f(x \mid z)$ and there exists a more IPW-like estimator for the full cohort cumulative hazard (see section 17.9 of Borgan and Samuelsen (2016)).

### Superset imputation method B

A more naive approach is to consider imputing $X$ from its distribution in the superset $f(x \mid t, \delta, z, s = 1, \theta)$ using estimates of the cumulative hazard, or the cumulative baseline hazard and the parameters in the substantive model, of the superset. For approximate imputation this implies fitting the Nelson-Aalen estimate to all individuals in the superset (for whom time and outcome status are available). There will be some tied event times, but with about the same number of events this will be rare as the cohort size increase.

For rejection sampling it means estimating $f(x \mid z, s = 1)$ and accepting proposed values with a probability that is compatible with the analysis model $f(t, \delta \mid x, z, \phi, s = 1)$. The analysis model influences rejection sampling through its estimated parameters and the cumulative baseline hazard estimate. Treating the superset as "a full cohort" or "a study population" means that the cumulative baseline hazard should be estimated for the superset, and the $\beta$ estimates of the substantive model could be just the Cox model estimates for the currently imputed superset. That is, the $\hat{\beta}$ estimates are obtained from maximising the full Cox likelihood (2.8) and the cumulative baseline hazard estimated with the Breslow estimator (2.15) where the risk sets $\mathcal{R}(t_i)$ are the individuals at risk in the superset instead of the full cohort. There will be some tied events, which will be handled with the Efron approach (see documation of `smcfcs` package).

These will be the Superset B methods. After imputing the supersets, the nested case-control or IPW case-cohort estimators are fitted and the results combined according to Rubin's rules. It is unclear whether this naive approach will give unbiased or efficient estimates. Two questions are whether the Cox model which we use in rejection sampling is compatible with the sampled cohort estimators and whether it is sufficient to impute missing values of $X$ using only the information available in the subcohort (except for the total number of individuals in the full cohort).

We will see how the superset imputation methods perform in the simulations below.

## 5.3 Simulation setup

The main aim of the simulation study is to compare full cohort imputation using either a nested case-control sample or a case cohort sample with the superset imputation techniques described in the previous sections. The methods will be compared to each other in a setting where it would be convenient to use multiple imputation. The estimates of interest are the log hazard ratios in Cox regression model and their standard errors.

Furthermore, we will consider one setup where the partially observed covariate is not related to the other covariates used for imputation. Then a setting will be examined using an auxiliary variable for imputation. Lastly, a setting with an interaction term will be explored, mainly in order to see how approximate imputation and rejection sampling differ.

The settings will be simple to better be able to gain an insight into the operating characteristics in basic situations. Additionally, the simulation settings are motivated by the FLC example and the studies referenced in the overview section for sampled cohort methods of chapter 4. The parameters of interest are as mentioned the log hazard ratios of Cox proportional model and their standard errors. Some covariates are fully observed and one covariate is deemed expensive and is only observed in the sampled cohort.

We will generate $n_{obs}$ data points from the following data generating mechanism. Further the simulations will be repeated $n_{sim} = 1000$ times. This is in order to reduce the Monte Carlo standard errors resulting from a limited number of simulations. The number of imputations will be $K = 10$ which is within the commonly recommended range. The data generating mechanism is

$$
\begin{aligned}
&Z_1 \sim \text{Bernoulli}(p_{z_1}) \\
&Z_2 \sim N(\mu_{z_2}, 1) \\
&X \sim N(a_x + b_x z_1 + c_x z_2, 1) \\
&V = X + \eta \text{ where } \eta \sim N(0, \sigma_\eta^2) \text{ such that } \operatorname{corr}(X, V) \approx 0.8(0.79).
\end{aligned}
$$

The event times will be generated according to a hazard rate

$$
h(t \mid x, z) = \lambda \tau t^{\tau-1} \exp(\beta x + \beta_{z_1} z_1 + \beta_{z_2} z_2 + \beta_{12} z_1 z_2) \tag{5.3}
$$

by drawing $n_{sim}$ standard uniform variables and transforming them according to

$$
T_e = \left( -\frac{1}{\lambda} \log(U) \exp(\beta_x x + \beta_{z_1} z_1 + \beta_{z_2} z_2 + \beta_{12} z_1 z_2) \right)^{\frac{1}{\tau}}. \tag{5.4}
$$

The resulting times will follow a Weibull distribution.

The event times $T_e$'s will be generated with scale parameter $\lambda_e = 4.0 \times 10^{-7}$ and shape parameter $\tau_e = 4$. The dropout times $T_c$'s will be generated from a Weibull distribution with parameters $\lambda_c = 2 \times 10^{-5}$ and $\tau_c = 4$, and assumed independent of $X$, $Z_1$ and $Z_2$. The maximum follow-up time will be 15 years. Thus the observed follow-up times are $T = \min(T_e, T_c, 15)$ with corresponding event indicator $D$. The parameters are chosen such that the proportion of individuals who experience the event is about 5% and the proportion that drops out is 62% and the remaining 33% are administratively censored at maximum follow-up time. The sampled cohorts will be approximately 10% of the full cohort (corresponding to 1 control per case).

An overview of the simulation experiments is given in the list below.

1. The standard setting will be imputation for a full cohort of size $n_{obs} = 5000$ and imputation for supersets of size 20% and 40% of the full cohort. The effect size for the binary variable will be equal to $\beta_{z_1} = 1$ and the effect sizes for the continuous variables $\beta_x = 1$ and $\beta_{z_2} = 0.5$. The data will be imputed without a surrogate variable $V$, and generated and analysed without interaction, i.e. $\beta_{12} = 0$ between $X$ and $Z_1$. Let $a_x = 0$, $b_x = c_x = 0.25$ and let $\mu_{z_2} = 0$ such that $\text{cor}(X, Z_1) \approx 0.12$ and $\text{cor}(X, Z_2) \approx 0.24$. Note that the sampled cohort size, about 10% of the cohort, will not be changed. (i-ii)

2. The no correlation setting will be imputation when $X$ is not correlated with the other covariates, i.e. $b_x = c_x = 0$. Here $\lambda_e = 5.5 \times 10^{-7}$ such that the proportion of events is still around 5%. (iii)

3. The auxiliary setting will be imputation with a surrogate variable $V$ with the same parameters (and the same simulated and sampled cohorts) as in the standard setting (i). Rejection sampling will be performed with $V$ in the substantive model, i.e. imputing with a richer imputation model. (iv)

4. The interaction setting will be imputation with an interaction in the true data generating mechanism and analysis model, $\beta_{12} = 0.5$. Here we set $\lambda_e = 2.5 \times 10^{-7}$. The impute, then transform/passive imputation (Section 3.4) will be used for approximate imputation. (v)

Cohorts will be sampled of roughly comparable size for nested case-control and case-cohort studies. The same sampled cohorts will be used for the traditional methods, the MI full cohort and the MI superset methods. The estimators are the traditional nested case-control estimator and the case-cohort IPW estimator. For full cohort the Cox model will be fitted. For superset imputation the nested case-control and the case-cohort estimator will be examined. Both approximate imputation and rejection sampling (SMC imputation) will be done throughout.

The measures of performance will be the bias $E[\hat{\beta}] - \beta$, the empirical standard error $\sqrt{\text{Var}(\hat{\beta})}$ estimated by $\sqrt{\frac{1}{n_{sim}-1} \sum_{i=1}^{n_{sim}} (\hat{\beta}_i - \bar{\beta})^2}$ where $\bar{\beta}$ is the mean estimate for $\beta$, the model standard error $E\sqrt{\widehat{\text{Var}}(\hat{\beta})}$ estimated by $\frac{1}{n_{sim}} \sum_{i=1}^{n_{sim}} \sqrt{\widehat{\text{Var}}(\hat{\beta}_i)}$ , and 95% confidence interval obtained by assuming a normal distribution. The relative efficiency of the parameter estimate compared to the traditional nested case-control or case-cohort estimator, and to the full cohort estimator will also be given. Also the mean squared error $\text{E}[(\hat{\beta} - \beta)]^2$ estimated by $\frac{1}{n_{sim}} \sum_{i=1}^{n_{sim}} (\hat{\beta} - \beta)^2$ which can be decomposed into a squared bias term and a variance term offers some insight on the bias-variance trade-off, interesting particularly with respect to prediction. Although as Morris, White, and Crowther (2019) note that for biased methods the MSE tend to vary more with the sample size $n_{obs}$ than the bias or standard errors alone. The maximum simulation error or Monte Carlo standard error estimates will also be reported.

## 5.4 Simulation results

### Generated data

In the standard and auxiliary settings, (i) and (iv), the $n_{sim} = 1000$ generated cohorts have a mean number of events of 261.5 (5.2% of the cohort) and the correlations between the covariates have a mean across the generated cohorts as given in Table 5.1 . We see that the two covariates $z_1$ and $z_2$ are mildly correlated with $x$ and not correlated with each other, while the surrogate variable is moderately to strongly correlated with $x$.

Table 5.1: Average correlation between the covariates across simulated cohorts

| | $x$ | $z_1$ | $z_2$ | $v$ |
|---|---|---|---|---|
| $x$ | 1.00 | 0.12 | 0.24 | 0.79 |
| $z_1$ | 0.12 | 1.00 | 0.00 | 0.10 |
| $z_2$ | 0.24 | 0.00 | 1.00 | 0.19 |
| $v$ | 0.79 | 0.10 | 0.19 | 1.00 |

From one of the generated cohorts a histograms of the event times, the dropout times and the follow-up times for all individuals in the cohort are shown in the Figure 5.2. These show the similar Weibull shapes of the event and dropout times, and the spike of administrative censoring at the end of maximum follow-up time.

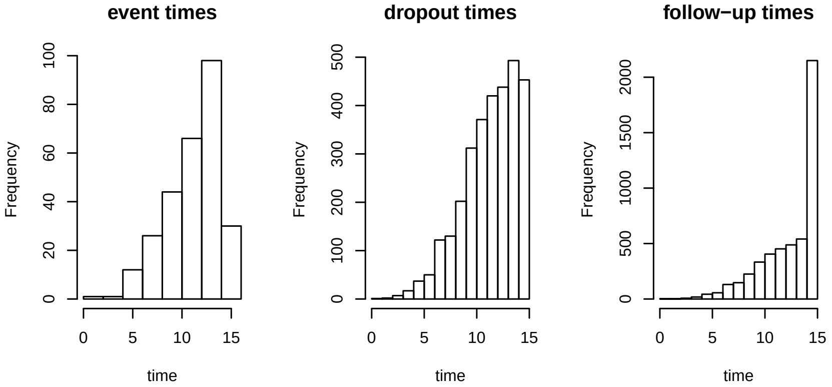


Figure 5.2: Histograms of times

### Simulation results for nested case-control samples

For the standard setup (i), $n_{sim} = 1000$ simulations gave the results in Table 5.2. The maximum estimated Monte Carlo standard error for the estimates are for bias $< 0.01$, empirical standard error $< 0.01$, mean squared error $< 0.01$ and coverage $< 0.02$. The results are rounded to 3 significant figures.

The classical Cox regression for the full generated cohort of $n_{obs} = 5000$ individuals gives effectively unbiased estimates. We see that the empirical standard errors are very close to the model standard errors, and that the 95% confidence intervals are close to nominal. For the classical nested case-control estimator for a nested case-control sample with one control per case ($m = 2$) we obtain approximately unbiased estimates, except for $\beta_x$ which show a little upward bias (0.027), and the standard errors are roughly doubled, leading to a relative efficiency compared to the full cohort estimate of 0.19 for $\beta_x$, 0.28 for $\beta_{z_1}$ and 0.23 for $\beta_{z_1}$. The model and empirical standard errors are closely similar and the coverage is also close to nominal.

Approximate imputation of the full cohort seem to slightly underestimate the parameters of the larger effect sizes, $\beta_x = 1$ and $\beta_{z_1} = 1$, while the estimate for the continuous weaker $\beta_{z_2}$ is close to unbiased. The superset imputation methods give less bias than imputing the full cohort and are close to unbiased for all effects. Both superset methods give very similar results (the methods differ only in how the cumulative hazard has been estimated). The standard errors are larger than when imputing the full cohort, but lower than for the classical nested case-control estimator. The relative efficiencies for $\beta_x, \beta_{z_1}, \beta_{z_2}$ are $0.27, 0.48, 0.40$. Compared to imputing the full cohort there is especially less efficiency for the parameters of the fully observed variables ($Z_1$ and $Z_2$), but there is also less efficiency for the partially observed variable $X$. This is expected since we only impute covariate values in a superset of about 20% of the individuals. Nonetheless, the MSE estimates for $X$ and $Z_2$ with superset imputation are not far from those of the full cohort imputation.

Rejection sampling for the full cohort is approximately unbiased and results in smaller MSE's than approximate imputation of the full cohort. The coverage is close to nominal. With rejection sampling there is more difference between the superset methods. Method A has a non-negligible bias of 0.159, overestimating $\beta_x$, and has undercoverage for $\beta_x$. Superset method B seem to slightly underestimate $\beta_x$. The relative efficiencies for $\beta_x, \beta_{z_1}, \beta_{z_2}$ are $0.30, 0.50, 0.42$ with method B. Of the superset methods, method B with rejection sampling has the least MSE.

Histograms of the estimated effects of $X$ from the standard setting (i) are shown in Appendix A. They are fairly normally distributed and show no unreasonably extreme outliers. Though, note that the histograms for the classical nested case-control and superset method A with rejection sampling show a few very high values.

For situation (ii), when the superset is increased from 3 controls per case to 7 controls, the relative efficiencies are markedly increased as we can see in Table 5.3. The relative efficiencies for approximate imputation superset method A and B increase to about $0.37, 0.63$ and 0.56, and for rejection sampling superset method B to about $0.41, 0.64$ and 0.56. A nested case-control superset with 7 controls consists here of about 40% of the full cohort, compared to about 20% with 3 controls. The relative efficiencies, of superset method A and B for approximate imputation and superset method B for rejection sampling, compared to the full cohort imputation methods are increased from about 63-64% to 87-88% for $\beta_x$, from 65-68% to 85-88% for $\beta_{z_1}$ and from 57-58% to 77-78% for $\beta_{z_2}$. The difference between $Z_1$ and $Z_2$ might be due to difference between the true effect sizes or the difference between imputing a binary and continuos variable.

For $\beta_x$ the approximate imputation superset methods A and B, and rejection sampling B, show a overall less bias and smaller standard errors with more controls, but there is now an indication of undercoverage. Examining the histograms for $\beta_x$ in Appendix A, we see no clear difference between the distributions with an increased number of controls.

Imputing the superset with rejection sampling method $A$ is increasedly biased with a larger superset. This method was previously considered in a different setting, for missingness by chance in the subcohort (values missing for both cases and non-cases), with a larger cohort size and for binary variables, and there are possible modifications to it that might make it perform better here (see Keogh, Seaman, et al. (2018)).

The difference in the full cohort imputation estimates here with respect to those using 3 controls is just due to the stochastic variability of the imputation algorithms, e.g. drawing parameters and missing values (and resulting convergence), since they have been applied on the exact same cohorts with the same sampled cohorts.

Table 5.2: Nested case-control (i): standard setup of 1000 simulations.

| | | **Standard** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Approximate imputation | | | Rejection sampling | | |
| | | Full cohort | NCC | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | 0.001 | 0.027 | -0.037 | 0.013 | 0.013 | -0.001 | 0.159 | -0.030 |
| | $\beta_{z_1}$ | 0.000 | 0.017 | -0.030 | 0.002 | 0.001 | 0.000 | 0.027 | 0.010 |
| | $\beta_{z_2}$ | 0.000 | 0.015 | -0.018 | 0.003 | 0.002 | 0.001 | 0.002 | 0.012 |
| ModelSE | $\beta_x$ | 0.066 | 0.153 | 0.104 | 0.127 | 0.127 | 0.097 | 0.130 | 0.121 |
| | $\beta_{z_1}$ | 0.145 | 0.274 | 0.169 | 0.209 | 0.209 | 0.170 | 0.219 | 0.206 |
| | $\beta_{z_2}$ | 0.065 | 0.138 | 0.077 | 0.103 | 0.102 | 0.077 | 0.107 | 0.101 |
| EmpSE | $\beta_x$ | 0.064 | 0.154 | 0.102 | 0.125 | 0.125 | 0.099 | 0.142 | 0.112 |
| | $\beta_{z_1}$ | 0.148 | 0.277 | 0.164 | 0.206 | 0.204 | 0.171 | 0.223 | 0.202 |
| | $\beta_{z_2}$ | 0.066 | 0.141 | 0.075 | 0.100 | 0.100 | 0.078 | 0.108 | 0.098 |
| RelEff | $\beta_x$ | 1 | 0.191 | 0.427 | 0.275 | 0.274 | 0.477 | 0.261 | 0.301 |
| | $\beta_{z_1}$ | 1 | 0.284 | 0.742 | 0.485 | 0.485 | 0.733 | 0.442 | 0.497 |
| | $\beta_{z_2}$ | 1 | 0.226 | 0.717 | 0.406 | 0.406 | 0.722 | 0.371 | 0.416 |
| MSE | $\beta_x$ | 0.004 | 0.024 | 0.012 | 0.016 | 0.016 | 0.010 | 0.045 | 0.013 |
| | $\beta_{z_1}$ | 0.022 | 0.077 | 0.028 | 0.042 | 0.042 | 0.029 | 0.050 | 0.041 |
| | $\beta_{z_2}$ | 0.004 | 0.020 | 0.006 | 0.010 | 0.010 | 0.006 | 0.012 | 0.010 |
| Cov | $\beta_x$ | 0.964 | 0.955 | 0.923 | 0.945 | 0.953 | 0.954 | 0.787 | 0.946 |
| | $\beta_{z_1}$ | 0.948 | 0.948 | 0.957 | 0.960 | 0.963 | 0.947 | 0.952 | 0.960 |
| | $\beta_{z_2}$ | 0.947 | 0.943 | 0.946 | 0.952 | 0.957 | 0.942 | 0.949 | 0.955 |

Table 5.3: Nested case-control (ii): Standard with larger superset of 1000 simulations.

**Standard 7 controls**

| | | | | Approximate imputation | | | Rejection sampling | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Full cohort | NCC | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | 0.001 | 0.027 | -0.036 | 0.005 | 0.005 | 0.000 | 0.336 | -0.018 |
| | $\beta_{z_1}$ | 0.000 | 0.017 | -0.029 | -0.008 | -0.008 | 0.000 | 0.063 | 0.006 |
| | $\beta_{z_2}$ | 0.000 | 0.015 | -0.019 | -0.001 | -0.001 | 0.000 | 0.005 | 0.008 |
| ModelSE | $\beta_x$ | 0.066 | 0.153 | 0.104 | 0.110 | 0.109 | 0.098 | 0.112 | 0.104 |
| | $\beta_{z_1}$ | 0.145 | 0.274 | 0.169 | 0.183 | 0.183 | 0.171 | 0.200 | 0.182 |
| | $\beta_{z_2}$ | 0.065 | 0.138 | 0.077 | 0.087 | 0.087 | 0.077 | 0.096 | 0.087 |
| EmpSE | $\beta_x$ | 0.064 | 0.154 | 0.102 | 0.119 | 0.120 | 0.099 | 0.137 | 0.107 |
| | $\beta_{z_1}$ | 0.148 | 0.277 | 0.164 | 0.185 | 0.186 | 0.171 | 0.232 | 0.186 |
| | $\beta_{z_2}$ | 0.066 | 0.141 | 0.074 | 0.088 | 0.088 | 0.078 | 0.108 | 0.088 |
| RelEff | $\beta_x$ | 1 | 0.191 | 0.421 | 0.369 | 0.370 | 0.471 | 0.350 | 0.409 |
| | $\beta_{z_1}$ | 1 | 0.284 | 0.739 | 0.632 | 0.631 | 0.727 | 0.529 | 0.638 |
| | $\beta_{z_2}$ | 1 | 0.226 | 0.723 | 0.557 | 0.558 | 0.722 | 0.466 | 0.562 |
| MSE | $\beta_x$ | 0.004 | 0.024 | 0.012 | 0.014 | 0.014 | 0.010 | 0.132 | 0.012 |
| | $\beta_{z_1}$ | 0.022 | 0.077 | 0.028 | 0.034 | 0.035 | 0.029 | 0.058 | 0.035 |
| | $\beta_{z_2}$ | 0.004 | 0.020 | 0.006 | 0.008 | 0.008 | 0.006 | 0.012 | 0.008 |
| Cov | $\beta_x$ | 0.964 | 0.955 | 0.937 | 0.936 | 0.922 | 0.951 | 0.179 | 0.937 |
| | $\beta_{z_1}$ | 0.948 | 0.948 | 0.953 | 0.951 | 0.946 | 0.949 | 0.901 | 0.947 |
| | $\beta_{z_2}$ | 0.947 | 0.943 | 0.944 | 0.948 | 0.949 | 0.945 | 0.914 | 0.948 |

The results of imputing $X$ with no correlation between $X$ and the other covariates are given in Table 5.4. As expected we see that the methods are close to unbiased. However, there are slightly more efficient estimates and smaller MSE's here than for the imputation methods in the standard setting. Although the generated cohorts are not the same, this is perhaps unexpected for $\beta_x$ especially. This could illustrate that in this simple simulation setting with only values of $X$ missing, only for controls, it is mainly the time and censoring information through the estimated cumulative hazard, or cumulative baseline hazard, that is important to obtain the gains in efficiency seen in the standard setting, when imputing missing values of $X$ for the MI algorithm. We remember that in the standard setting the correlation between $X$ and the other covariates is mild, 0.21 and 0.24 for $Z_1$ and $Z_2$ respectively. Next we will see what happens when imputation is performed with an auxiliary variable that is more strongly correlated with $X$.

The results of auxiliary variable imputation for the same sampled cohorts as in the standard case with 3 controls are displayed in Table 5.5. Using an auxiliary variable that is a true surrogate of $X$ for imputation give reduction of bias and gains in efficiency compared to both the standard setting (i) and the no correlation setting (iii). The gain is most substantial for full cohort imputation as there is a fully observed auxiliary covariate for all individuals in the entire cohort, that we make use of.

For the superset methods there is also a clear gain in efficiency and this is expected to increase with a larger superset. We note rejection sampling superset method A is now considerably less biased with an auxiliary variable and that method B is approximately unbiased. Overall, this suggests that when an auxiliary variable is available it should strongly be considered used. However, as we noted in Section 4.5 when the auxiliary variable is not a true surrogate of $X$, i.e. the response is independent of the auxiliary variable when the other variables in the analysis model are included, there could be some bias in the parameter estimates. In situations where it is unclear whether the auxiliary variable is a surrogate or not, superset imputation might be safer than full cohort imputation.

Table 5.4: Nested case-control (iii): $X$ not correlated with $Z_1$ and $Z_2$ of 1000 simulations.

**No correlation**

| | | | | Approximate imputation | | | Rejection sampling | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Full cohort | NCC | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | 0.005 | 0.034 | -0.025 | 0.017 | 0.016 | 0.008 | 0.165 | -0.018 |
| | $\beta_{z_1}$ | 0.009 | 0.019 | -0.019 | 0.006 | 0.004 | 0.004 | 0.058 | 0.002 |
| | $\beta_{z_2}$ | 0.000 | 0.008 | -0.015 | 0.004 | 0.003 | 0.000 | 0.029 | 0.003 |
| ModelSE | $\beta_x$ | 0.065 | 0.143 | 0.102 | 0.120 | 0.120 | 0.097 | 0.124 | 0.116 |
| | $\beta_{z_1}$ | 0.136 | 0.251 | 0.162 | 0.195 | 0.194 | 0.163 | 0.205 | 0.192 |
| | $\beta_{z_2}$ | 0.062 | 0.125 | 0.075 | 0.095 | 0.095 | 0.074 | 0.100 | 0.094 |
| EmpSE | $\beta_x$ | 0.068 | 0.147 | 0.103 | 0.122 | 0.121 | 0.104 | 0.138 | 0.110 |
| | $\beta_{z_1}$ | 0.139 | 0.255 | 0.162 | 0.195 | 0.196 | 0.166 | 0.218 | 0.193 |
| | $\beta_{z_2}$ | 0.064 | 0.126 | 0.072 | 0.093 | 0.092 | 0.074 | 0.104 | 0.091 |
| RelEff | $\beta_x$ | 1 | 0.211 | 0.431 | 0.299 | 0.298 | 0.466 | 0.282 | 0.320 |
| | $\beta_{z_1}$ | 1 | 0.300 | 0.719 | 0.496 | 0.497 | 0.711 | 0.449 | 0.509 |
| | $\beta_{z_2}$ | 1 | 0.254 | 0.707 | 0.435 | 0.436 | 0.711 | 0.393 | 0.445 |
| MSE | $\beta_x$ | 0.005 | 0.023 | 0.011 | 0.015 | 0.015 | 0.011 | 0.046 | 0.012 |
| | $\beta_{z_1}$ | 0.019 | 0.065 | 0.027 | 0.038 | 0.038 | 0.027 | 0.051 | 0.037 |
| | $\beta_{z_2}$ | 0.004 | 0.016 | 0.005 | 0.009 | 0.009 | 0.005 | 0.012 | 0.008 |
| Cov | $\beta_x$ | 0.933 | 0.948 | 0.923 | 0.943 | 0.946 | 0.921 | 0.735 | 0.945 |
| | $\beta_{z_1}$ | 0.954 | 0.953 | 0.940 | 0.949 | 0.951 | 0.939 | 0.928 | 0.952 |
| | $\beta_{z_2}$ | 0.945 | 0.953 | 0.950 | 0.960 | 0.961 | 0.954 | 0.940 | 0.959 |

Table 5.5: Nested case-control (iv): Auxilliary variable of 1000 simulations.

| | | **Auxilliary variable imputation** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Approximate imputation | | | Rejection sampling | | |
| | | Full cohort | NCC | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | 0.001 | 0.027 | -0.027 | 0.011 | 0.010 | 0.000 | 0.097 | -0.006 |
| | $\beta_{z_1}$ | 0.000 | 0.017 | -0.019 | 0.002 | 0.002 | -0.001 | 0.012 | 0.007 |
| | $\beta_{z_2}$ | 0.000 | 0.015 | -0.01 | 0.003 | 0.003 | 0.001 | 0.001 | 0.007 |
| ModelSE | $\beta_x$ | 0.066 | 0.153 | 0.088 | 0.117 | 0.116 | 0.087 | 0.122 | 0.114 |
| | $\beta_{z_1}$ | 0.145 | 0.274 | 0.156 | 0.202 | 0.202 | 0.156 | 0.207 | 0.201 |
| | $\beta_{z_2}$ | 0.065 | 0.138 | 0.071 | 0.100 | 0.100 | 0.071 | 0.102 | 0.099 |
| EmpSE | $\beta_x$ | 0.064 | 0.154 | 0.084 | 0.116 | 0.117 | 0.087 | 0.133 | 0.109 |
| | $\beta_{z_1}$ | 0.148 | 0.277 | 0.155 | 0.198 | 0.198 | 0.158 | 0.207 | 0.197 |
| | $\beta_{z_2}$ | 0.066 | 0.141 | 0.071 | 0.098 | 0.098 | 0.073 | 0.102 | 0.098 |
| RelEff | $\beta_x$ | 1 | 0.191 | 0.572 | 0.323 | 0.324 | 0.590 | 0.298 | 0.336 |
| | $\beta_{z_1}$ | 1 | 0.284 | 0.859 | 0.515 | 0.516 | 0.860 | 0.490 | 0.520 |
| | $\beta_{z_2}$ | 1 | 0.226 | 0.842 | 0.429 | 0.430 | 0.849 | 0.410 | 0.434 |
| MSE | $\beta_x$ | 0.004 | 0.024 | 0.008 | 0.013 | 0.014 | 0.008 | 0.027 | 0.012 |
| | $\beta_{z_1}$ | 0.022 | 0.077 | 0.024 | 0.039 | 0.039 | 0.025 | 0.043 | 0.039 |
| | $\beta_{z_2}$ | 0.004 | 0.020 | 0.005 | 0.010 | 0.010 | 0.005 | 0.010 | 0.010 |
| Cov | $\beta_x$ | 0.964 | 0.955 | 0.952 | 0.957 | 0.958 | 0.943 | 0.876 | 0.956 |
| | $\beta_{z_1}$ | 0.948 | 0.948 | 0.945 | 0.953 | 0.952 | 0.944 | 0.947 | 0.953 |
| | $\beta_{z_2}$ | 0.947 | 0.943 | 0.945 | 0.963 | 0.959 | 0.938 | 0.955 | 0.963 |

For the last setting, the more complex situation when there is an interaction term between $X$ and $Z_1$, the results are reported in Table 5.6. As expected approximate imputation clearly attenuates the interaction effect, resulting in bias for the other covariate effects as well. The attenuation is less dramatic when only imputing the superset compared to the full cohort imputation. For rejection sampling the missing values are imputed compatible with an analysis model including the interaction term. Full cohort rejection sampling show nearly unbiased results and overall very good performance here. The superset B rejection sampling methods perform best of the superset methods, but we see that it displays some signs of bias, particularly in underestimating the interaction effect. Now, the superset A give nearly unbiased estimates for the interaction effect while the bias for $\beta_x$ is about the same as in the standard setting.

In situations with interaction effects (and this has also been shown for non-linear terms in more general settings) approximate imputation give biased results while rejection sampling is preferred. Here the bias in the superset B method is overall less than the classical nested case-control estimates, has considerably more efficient estimates and comparable overall coverage. Therefore, in situations where full cohort information is not possible, the superset method B, and possibly a modification of superset method A (to make use of more of the full cohort information), seem to offer real improvement over the classical nested case-control estimator from these limited simulations.

Table 5.6: Nested case-control (v): Interaction term of 1000 simulations.

| | | **Interaction term** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Approximate imputation | | | Rejection sampling | | |
| | | Full cohort | NCC | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | -0.003 | 0.064 | 0.186 | 0.130 | 0.130 | 0.015 | 0.159 | 0.009 |
| | $\beta_{z_1}$ | 0.016 | 0.053 | 0.239 | 0.084 | 0.083 | 0.022 | 0.035 | 0.025 |
| | $\beta_{z_2}$ | 0.003 | 0.021 | -0.022 | 0.012 | 0.012 | 0.006 | 0.007 | 0.026 |
| | $\beta_{xz_1}$ | 0.006 | -0.01 | -0.371 | -0.154 | -0.152 | -0.018 | 0.050 | -0.053 |
| ModelSE | $\beta_x$ | 0.158 | 0.304 | 0.197 | 0.252 | 0.252 | 0.198 | 0.261 | 0.245 |
| | $\beta_{z_1}$ | 0.275 | 0.446 | 0.287 | 0.350 | 0.350 | 0.284 | 0.358 | 0.348 |
| | $\beta_{z_2}$ | 0.066 | 0.166 | 0.084 | 0.120 | 0.121 | 0.082 | 0.128 | 0.118 |
| | $\beta_{xz_1}$ | 0.177 | 0.376 | 0.213 | 0.304 | 0.304 | 0.219 | 0.317 | 0.299 |
| EmpSE | $\beta_x$ | 0.165 | 0.317 | 0.178 | 0.234 | 0.231 | 0.205 | 0.270 | 0.233 |
| | $\beta_{z_1}$ | 0.281 | 0.484 | 0.275 | 0.349 | 0.348 | 0.292 | 0.366 | 0.353 |
| | $\beta_{z_2}$ | 0.065 | 0.175 | 0.08 | 0.121 | 0.120 | 0.084 | 0.136 | 0.118 |
| | $\beta_{xz_1}$ | 0.184 | 0.393 | 0.148 | 0.247 | 0.245 | 0.227 | 0.320 | 0.276 |
| RelEff | $\beta_x$ | 1 | 0.290 | 0.662 | 0.410 | 0.411 | 0.658 | 0.384 | 0.433 |
| | $\beta_{z_1}$ | 1 | 0.393 | 0.922 | 0.624 | 0.624 | 0.944 | 0.599 | 0.631 |
| | $\beta_{z_2}$ | 1 | 0.162 | 0.621 | 0.303 | 0.301 | 0.662 | 0.267 | 0.313 |
| | $\beta_{xz_1}$ | 1 | 0.230 | 0.708 | 0.348 | 0.348 | 0.671 | 0.320 | 0.360 |
| MSE | $\beta_x$ | 0.027 | 0.105 | 0.066 | 0.071 | 0.070 | 0.042 | 0.098 | 0.054 |
| | $\beta_{z_1}$ | 0.079 | 0.236 | 0.133 | 0.128 | 0.128 | 0.086 | 0.135 | 0.125 |
| | $\beta_{z_2}$ | 0.004 | 0.031 | 0.007 | 0.015 | 0.015 | 0.007 | 0.018 | 0.015 |
| | $\beta_{xz_1}$ | 0.034 | 0.155 | 0.159 | 0.085 | 0.083 | 0.052 | 0.105 | 0.079 |
| Cov | $\beta_x$ | 0.937 | 0.951 | 0.869 | 0.962 | 0.963 | 0.945 | 0.944 | 0.962 |
| | $\beta_{z_1}$ | 0.941 | 0.942 | 0.915 | 0.954 | 0.955 | 0.938 | 0.947 | 0.951 |
| | $\beta_{z_2}$ | 0.955 | 0.951 | 0.962 | 0.952 | 0.955 | 0.939 | 0.951 | 0.955 |
| | $\beta_{xz_1}$ | 0.942 | 0.946 | 0.61 | 0.979 | 0.976 | 0.939 | 0.953 | 0.975 |

### Simulation results for case-cohort samples

In the standard setup (i) the results for case-cohort are similar to nested case-control and shown in Table 5.7. The maximum Monte Carlo standard error is less than 0.01 (except for the coverage estimate in the interaction setting (v) where it is 0.016). The traditional case-cohort IPW estimator slightly underestimates the standard errors leading to some undercoverage. Both full cohort imputation methods are approximately unbiased and have good coverage. Approximate imputation gives slightly more bias and undercumulative baseline of the effects. For approximate imputation the superset methods are again very similar to each other. For $\beta_x$ the model standard errors are higher than for the classic IPW estimator while the empirical standard errors are lower, closer to the full cohort imputation, leading to the relative efficiency being estimated lower than the classical IPW estimator. For rejection sampling the superset method A here shows only a very slight upward bias. Although the empirical standard error is larger than the model error and there is a clear undercoverage for $\beta_x$, but also the traditional case-cohort estimator suffers from this here, and it is reduced with imputation. We remember that superset A method for rejection sampling uses the Prentice estimator instead of the IPW estimator. The superset B method for rejection sampling show little bias and some overcoverage. All superset methods show less MSE than the classical IPW estimator, except superset A for rejection sampling.

Increasing the superset with an additional 1000 controls give the results of setup (ii) in Table 5.8. The approximate imputation superset methods are slightly more biased, but have more correct standard errors than the standard setting. The MSE's are slightly reduced with a larger superset, and the coverage improved. With rejection sampling the superset methods also show less biased and more correct standard errors except superset A for $\beta_x$ for whom the model standard error is increasedly underestimated. Since this is present in the classical case-cohort estimates as well, which has asymptotically unbiased variance estimates, this could be due to the finite sample size (a larger sample size $n_{obs}$ will affect the superset A methods). In addition, the difference between the superset methods for rejection sampling could be partially due to the difference between the robust standard error estimates of Prentice's estimator and the standard error estimates of the IPW estimator.

Table 5.7: Case-cohort (i): standard setup of 1000 simulations. Results rounded to 3 significant figures

|  |  | **Standard** |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|
|  |  |  |  | Approximate imputation |  |  | Rejection sampling |  |  |
|  |  | Full cohort | CCH | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | 0.001 | 0.034 | -0.033 | -0.007 | -0.008 | -0.002 | 0.023 | -0.010 |
|  | $\beta_{z_1}$ | 0.000 | 0.011 | -0.038 | -0.019 | -0.019 | 0.000 | 0.008 | 0.023 |
|  | $\beta_{z_2}$ | 0.000 | 0.021 | -0.024 | -0.011 | -0.011 | 0.000 | 0.009 | 0.019 |
| ModelSE | $\beta_x$ | 0.066 | 0.127 | 0.108 | 0.137 | 0.137 | 0.100 | 0.143 | 0.122 |
|  | $\beta_{z_1}$ | 0.145 | 0.264 | 0.171 | 0.219 | 0.220 | 0.171 | 0.233 | 0.216 |
|  | $\beta_{z_2}$ | 0.065 | 0.130 | 0.078 | 0.109 | 0.109 | 0.077 | 0.117 | 0.104 |
| EmpSE | $\beta_x$ | 0.064 | 0.143 | 0.101 | 0.108 | 0.107 | 0.098 | 0.165 | 0.103 |
|  | $\beta_{z_1}$ | 0.148 | 0.288 | 0.166 | 0.200 | 0.198 | 0.173 | 0.232 | 0.204 |
|  | $\beta_{z_2}$ | 0.066 | 0.142 | 0.075 | 0.098 | 0.097 | 0.077 | 0.118 | 0.098 |
| RelEff | $\beta_x$ | 1 | 0.278 | 0.399 | 0.241 | 0.240 | 0.457 | 0.223 | 0.299 |
|  | $\beta_{z_1}$ | 1 | 0.304 | 0.726 | 0.441 | 0.440 | 0.729 | 0.395 | 0.455 |
|  | $\beta_{z_2}$ | 1 | 0.256 | 0.701 | 0.360 | 0.360 | 0.716 | 0.316 | 0.393 |
| MSE | $\beta_x$ | 0.004 | 0.022 | 0.011 | 0.012 | 0.011 | 0.010 | 0.028 | 0.011 |
|  | $\beta_{z_1}$ | 0.022 | 0.083 | 0.029 | 0.040 | 0.040 | 0.030 | 0.054 | 0.042 |
|  | $\beta_{z_2}$ | 0.004 | 0.021 | 0.006 | 0.010 | 0.010 | 0.006 | 0.014 | 0.010 |
| Cov | $\beta_x$ | 0.964 | 0.903 | 0.934 | 0.982 | 0.986 | 0.941 | 0.912 | 0.972 |
|  | $\beta_{z_1}$ | 0.948 | 0.919 | 0.945 | 0.962 | 0.966 | 0.946 | 0.963 | 0.961 |
|  | $\beta_{z_2}$ | 0.947 | 0.915 | 0.948 | 0.967 | 0.971 | 0.955 | 0.958 | 0.959 |

Table 5.8: Case-cohort (ii): Standard with larger superset of 1000 simulations.

**Standard subcohort size of** 1750

| | | | | Approximate imputation | | | Rejection sampling | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Full cohort | CCH | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | 0.001 | 0.034 | -0.034 | -0.024 | -0.025 | -0.002 | 0.018 | -0.007 |
| | $\beta_{z_1}$ | 0.000 | 0.011 | -0.038 | -0.031 | -0.032 | 0.001 | 0.006 | 0.018 |
| | $\beta_{z_2}$ | 0.000 | 0.021 | -0.025 | -0.018 | -0.017 | 0.000 | 0.006 | 0.014 |
| ModelSE | $\beta_x$ | 0.066 | 0.127 | 0.108 | 0.121 | 0.120 | 0.099 | 0.122 | 0.108 |
| | $\beta_{z_1}$ | 0.145 | 0.264 | 0.170 | 0.189 | 0.190 | 0.172 | 0.200 | 0.189 |
| | $\beta_{z_2}$ | 0.065 | 0.130 | 0.078 | 0.092 | 0.092 | 0.078 | 0.096 | 0.088 |
| EmpSE | $\beta_x$ | 0.064 | 0.143 | 0.101 | 0.101 | 0.101 | 0.097 | 0.155 | 0.096 |
| | $\beta_{z_1}$ | 0.148 | 0.288 | 0.168 | 0.179 | 0.178 | 0.172 | 0.205 | 0.184 |
| | $\beta_{z_2}$ | 0.066 | 0.142 | 0.074 | 0.084 | 0.083 | 0.078 | 0.096 | 0.085 |
| RelEff | $\beta_x$ | 1 | 0.278 | 0.400 | 0.309 | 0.314 | 0.461 | 0.307 | 0.385 |
| | $\beta_{z_1}$ | 1 | 0.304 | 0.732 | 0.591 | 0.589 | 0.720 | 0.535 | 0.592 |
| | $\beta_{z_2}$ | 1 | 0.256 | 0.708 | 0.510 | 0.511 | 0.713 | 0.469 | 0.547 |
| MSE | $\beta_x$ | 0.004 | 0.022 | 0.011 | 0.011 | 0.011 | 0.010 | 0.024 | 0.009 |
| | $\beta_{z_1}$ | 0.022 | 0.083 | 0.030 | 0.033 | 0.033 | 0.030 | 0.042 | 0.034 |
| | $\beta_{z_2}$ | 0.004 | 0.021 | 0.006 | 0.007 | 0.007 | 0.006 | 0.009 | 0.008 |
| Cov | $\beta_x$ | 0.964 | 0.903 | 0.944 | 0.973 | 0.976 | 0.946 | 0.869 | 0.966 |
| | $\beta_{z_1}$ | 0.948 | 0.919 | 0.942 | 0.959 | 0.964 | 0.945 | 0.942 | 0.955 |
| | $\beta_{z_2}$ | 0.947 | 0.915 | 0.957 | 0.957 | 0.965 | 0.949 | 0.953 | 0.957 |

Table 5.9 show the results for situation (iii). Without correlation between $X$ and the other covariates the classical case-cohort IPW estimator clearly underestimates the standard error of $\beta_x$. For imputation of the full cohort or the superset, with approximate imputation methods A and B and rejection sampling method B, the results are similar to the standard situation. Superset method A with rejection sampling here markedly underestimates the association of $X$ on the relative risk, and overestimates the standard error as before. Although the MSE of the fully observed covariates are, as the other methods, lower than without imputation.

Imputing using an auxilliary variable give gains in efficiency for $\beta_x$, and also for $\beta_{z_1}$ and $\beta_{z_1}$ (Table 5.10). The confidence interval of the superset methods are less conservative and closer to nominal. Compared with the increased subcohort size (ii), the MSE estimates are smaller with an increased superset than with an auxiliary variable, also for $\beta_x$.

With an interaction term we see from Table 5.11 the classical estimator slightly overestimates $\beta_x$ and $\beta_{z_1}$, and slightly underestimates $\beta_{xz_1}$. Approximate imputation clearly underestimates the interaction effect with a bias of $-0.392$ when imputing the full cohort and $-0.273$ and $-0.276$ when imputing the supersets. The estimates for the individual effects that are part of the interaction are clearly overestimated. The bias is less for the approximate superset methods than approximate full cohort imputation, but are still non-negligible. Imputing the superset with rejection sampling give very similar results between method A and B. Method B has more bias for the interaction than method A. The MSE is slightly lower for method B for all effects than method A.

Table 5.9: Case-cohort (iii): $X$ not correlated with $Z_1$ and $Z_2$ of 1000 simulations.

| | | **No correlation** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Approximate imputation | | | Rejection sampling | | |
| | | Full cohort | CCH | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | 0.005 | 0.044 | -0.024 | -0.001 | -0.002 | 0.005 | -0.202 | 0.002 |
| | $\beta_{z_1}$ | 0.009 | 0.035 | -0.026 | -0.003 | -0.004 | 0.008 | -0.062 | 0.029 |
| | $\beta_{z_2}$ | 0.000 | 0.018 | -0.017 | -0.008 | -0.009 | 0.001 | -0.035 | 0.010 |
| ModelSE | $\beta_x$ | 0.065 | 0.125 | 0.107 | 0.134 | 0.133 | 0.100 | 0.140 | 0.120 |
| | $\beta_{z_1}$ | 0.136 | 0.250 | 0.164 | 0.210 | 0.210 | 0.165 | 0.208 | 0.205 |
| | $\beta_{z_2}$ | 0.062 | 0.122 | 0.075 | 0.104 | 0.104 | 0.075 | 0.106 | 0.099 |
| EmpSE | $\beta_x$ | 0.068 | 0.147 | 0.106 | 0.113 | 0.113 | 0.105 | 0.114 | 0.108 |
| | $\beta_{z_1}$ | 0.139 | 0.272 | 0.159 | 0.188 | 0.189 | 0.163 | 0.185 | 0.194 |
| | $\beta_{z_2}$ | 0.064 | 0.140 | 0.072 | 0.091 | 0.090 | 0.075 | 0.096 | 0.094 |
| RelEff | $\beta_x$ | 1 | 0.277 | 0.395 | 0.247 | 0.250 | 0.439 | 0.234 | 0.300 |
| | $\beta_{z_1}$ | 1 | 0.299 | 0.706 | 0.430 | 0.430 | 0.697 | 0.441 | 0.448 |
| | $\beta_{z_2}$ | 1 | 0.267 | 0.696 | 0.365 | 0.364 | 0.698 | 0.355 | 0.399 |
| MSE | $\beta_x$ | 0.005 | 0.024 | 0.012 | 0.013 | 0.013 | 0.011 | 0.054 | 0.012 |
| | $\beta_{z_1}$ | 0.019 | 0.075 | 0.026 | 0.035 | 0.036 | 0.027 | 0.038 | 0.038 |
| | $\beta_{z_2}$ | 0.004 | 0.020 | 0.005 | 0.008 | 0.008 | 0.006 | 0.010 | 0.009 |
| Cov | $\beta_x$ | 0.933 | 0.868 | 0.939 | 0.972 | 0.972 | 0.931 | 0.662 | 0.969 |
| | $\beta_{z_1}$ | 0.954 | 0.919 | 0.948 | 0.967 | 0.969 | 0.958 | 0.956 | 0.966 |
| | $\beta_{z_2}$ | 0.945 | 0.902 | 0.954 | 0.972 | 0.969 | 0.952 | 0.953 | 0.963 |

Table 5.10: Case-cohort (iv): Auxiliary variable of 1000 simulations.

| | | **Auxiliary variable imputation** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Approximate imputation | | | Rejection sampling | | |
| | | Full cohort | CCH | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | 0.001 | 0.034 | -0.026 | -0.005 | -0.006 | -0.001 | 0.021 | 0.010 |
| | $\beta_{z_1}$ | 0.000 | 0.011 | -0.020 | -0.008 | -0.007 | 0.002 | 0.004 | 0.015 |
| | $\beta_{z_2}$ | 0.000 | 0.021 | -0.013 | -0.003 | -0.003 | 0.000 | 0.006 | 0.010 |
| ModelSE | $\beta_x$ | 0.066 | 0.127 | 0.090 | 0.121 | 0.121 | 0.088 | 0.132 | 0.114 |
| | $\beta_{z_1}$ | 0.145 | 0.264 | 0.157 | 0.206 | 0.206 | 0.157 | 0.217 | 0.204 |
| | $\beta_{z_2}$ | 0.065 | 0.130 | 0.071 | 0.101 | 0.102 | 0.071 | 0.110 | 0.099 |
| EmpSE | $\beta_x$ | 0.064 | 0.143 | 0.088 | 0.103 | 0.102 | 0.089 | 0.145 | 0.102 |
| | $\beta_{z_1}$ | 0.148 | 0.288 | 0.154 | 0.197 | 0.198 | 0.157 | 0.218 | 0.200 |
| | $\beta_{z_2}$ | 0.066 | 0.142 | 0.071 | 0.097 | 0.097 | 0.072 | 0.113 | 0.097 |
| RelEff | $\beta_x$ | 1 | 0.278 | 0.548 | 0.306 | 0.305 | 0.576 | 0.259 | 0.341 |
| | $\beta_{z_1}$ | 1 | 0.304 | 0.854 | 0.497 | 0.496 | 0.853 | 0.451 | 0.505 |
| | $\beta_{z_2}$ | 1 | 0.256 | 0.836 | 0.416 | 0.415 | 0.842 | 0.360 | 0.433 |
| MSE | $\beta_x$ | 0.004 | 0.022 | 0.008 | 0.011 | 0.011 | 0.008 | 0.022 | 0.010 |
| | $\beta_{z_1}$ | 0.022 | 0.083 | 0.024 | 0.039 | 0.039 | 0.025 | 0.048 | 0.040 |
| | $\beta_{z_2}$ | 0.004 | 0.021 | 0.005 | 0.009 | 0.009 | 0.005 | 0.013 | 0.009 |
| Cov | $\beta_x$ | 0.964 | 0.903 | 0.947 | 0.971 | 0.970 | 0.941 | 0.930 | 0.971 |
| | $\beta_{z_1}$ | 0.948 | 0.919 | 0.950 | 0.955 | 0.954 | 0.944 | 0.954 | 0.953 |
| | $\beta_{z_2}$ | 0.947 | 0.915 | 0.936 | 0.952 | 0.955 | 0.945 | 0.936 | 0.952 |

Table 5.11: Case-cohort (v): Interaction term of 1000 simulations.

**Interaction term**

| | | | | Approximate imputation | | | Rejection sampling | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Full cohort | CCH | Full cohort | Superset (A) | Superset (B) | Full cohort | Superset (A) | superset (B) |
| Bias | $\beta_x$ | -0.003 | 0.076 | 0.218 | 0.178 | 0.176 | 0.009 | 0.066 | 0.045 |
| | $\beta_{z_1}$ | 0.016 | 0.054 | 0.238 | 0.174 | 0.175 | 0.022 | 0.033 | 0.046 |
| | $\beta_{z_2}$ | 0.003 | 0.017 | -0.036 | -0.018 | -0.017 | 0.003 | 0.008 | 0.034 |
| | $\beta_{xz_1}$ | 0.006 | -0.039 | -0.392 | -0.273 | -0.276 | -0.010 | -0.021 | -0.074 |
| ModelSE | $\beta_x$ | 0.158 | 0.244 | 0.198 | 0.237 | 0.238 | 0.202 | 0.270 | 0.240 |
| | $\beta_{z_1}$ | 0.275 | 0.374 | 0.286 | 0.348 | 0.348 | 0.283 | 0.370 | 0.338 |
| | $\beta_{z_2}$ | 0.066 | 0.135 | 0.086 | 0.129 | 0.129 | 0.083 | 0.138 | 0.116 |
| | $\beta_{xz_1}$ | 0.177 | 0.281 | 0.214 | 0.297 | 0.297 | 0.223 | 0.325 | 0.281 |
| EmpSE | $\beta_x$ | 0.165 | 0.296 | 0.181 | 0.214 | 0.215 | 0.207 | 0.287 | 0.238 |
| | $\beta_{z_1}$ | 0.281 | 0.404 | 0.277 | 0.315 | 0.319 | 0.293 | 0.376 | 0.340 |
| | $\beta_{z_2}$ | 0.065 | 0.153 | 0.079 | 0.101 | 0.102 | 0.083 | 0.146 | 0.103 |
| | $\beta_{xz_1}$ | 0.184 | 0.334 | 0.146 | 0.201 | 0.205 | 0.230 | 0.348 | 0.265 |
| RelEff | $\beta_x$ | 1 | 0.442 | 0.657 | 0.467 | 0.466 | 0.636 | 0.401 | 0.456 |
| | $\beta_{z_1}$ | 1 | 0.547 | 0.927 | 0.632 | 0.632 | 0.944 | 0.577 | 0.668 |
| | $\beta_{z_2}$ | 1 | 0.249 | 0.604 | 0.270 | 0.271 | 0.638 | 0.235 | 0.325 |
| | $\beta_{xz_1}$ | 1 | 0.414 | 0.702 | 0.371 | 0.368 | 0.651 | 0.324 | 0.410 |
| MSE | $\beta_x$ | 0.027 | 0.093 | 0.080 | 0.077 | 0.077 | 0.043 | 0.087 | 0.059 |
| | $\beta_{z_1}$ | 0.079 | 0.166 | 0.133 | 0.13 | 0.132 | 0.086 | 0.142 | 0.117 |
| | $\beta_{z_2}$ | 0.004 | 0.024 | 0.007 | 0.011 | 0.011 | 0.007 | 0.021 | 0.012 |
| | $\beta_{xz_1}$ | 0.034 | 0.113 | 0.175 | 0.115 | 0.118 | 0.053 | 0.121 | 0.076 |
| Cov | $\beta_x$ | 0.937 | 0.877 | 0.831 | 0.929 | 0.926 | 0.939 | 0.941 | 0.946 |
| | $\beta_{z_1}$ | 0.941 | 0.935 | 0.913 | 0.959 | 0.960 | 0.937 | 0.953 | 0.951 |
| | $\beta_{z_2}$ | 0.955 | 0.892 | 0.949 | 0.986 | 0.980 | 0.952 | 0.944 | 0.954 |
| | $\beta_{xz_1}$ | 0.942 | 0.892 | 0.562 | 0.947 | 0.940 | 0.934 | 0.945 | 0.959 |

Overall, multiple imputation when the fully observed sampled cohort is a case-cohort sample give results that resemble those from when the sampled cohort is a nested case-control sample. Full cohort imputation is the most comparable, since the superset methods use their respective sampled cohort estimators. In the standard situation MSE for $\beta_x$, $\beta_{z_1}$ and $\beta_{z_2}$ using approximate imputation of the full cohort from a nested case-control sample are 0.012, 0.028, 0.006. For case-cohort the MSE estimates are 0.011, 0.029 and 0.006. For both sampled cohorts the effects are a bit underestimated with approximate imputation. The standard errors are slightly overestimated for nested case-control and slightly more overestimated for case-cohort. For imputation using rejection sampling for the full cohort, the MSE's are 0.010, 0.029, 0.006 for nested case-control and 0.010, 0.030, 0.006 for case-cohort samples. There is a slight overcoverage for $\beta_x$ for nested-case-control and a slight undercoverage for case-cohort. In the standard case, imputing the full cohort evens out any differences between the sampled cohorts and the imputation results are very similar. Full cohort imputation give similar results also when there is no correlation and when an auxiliary variable is being used. With an interaction the MSE estimates are also similar, being 0.042,0.086, 0.007, 0.052 for nested case-control and 0.043, 0.086, 0.007, 0.053 for case-cohort, both using rejection sampling. The results in the different settings are very similar for both sampling designs and it does not seem to matter whether the controls are matched on time or sampled randomly in the subcohort when the full cohort is imputed.

Comparing the superset methods we see for approximate imputation in the standard case (i) that for both nested case-control supersets and case-cohort superset the estimates are approximately unbiased. The standard errors for nested case-control are smaller and more correct than for case-cohort. Therefore, the relative efficiencies are higher and the confidence interval closer to nominal for nested case-control superset than the case-cohort superset. The same holds for rejection sampling superset B. For the superset A method using rejection sampling the nested case-control sample give a bias in the estimate for $\beta_x$ of 0.162 while the case-cohort superset A estimates are close to unbiased, but have some undercoverage for $\beta_x$. For the situation where the superset consists of 7 controls or a subcohort of 1750 individuals the bias and undercoverage is further increased.

With no correlation the superset A method using rejection sampling overestimates the effect of $\beta_x$ for the nested case-control superset, while it for the case-cohort superset clearly underestimates the effect. While for imputation with an auxiliary variable the nested case-control estimate is less biased and the case-cohort estimate close to unbiased. It is unclear what the explanation for the biased estimates of superset method A with rejection sampling is. Perhaps in certain settings, finite sample bias in the sampled cohort estimators is propagated and increased through the rejection sampling algorithm since they are used in the both the cumulative baseline hazard and the substantive model compatible acceptance probability. There could also be an issue with estimating the population cumulative baseline hazard using the Breslow type estimators (5.1) and (5.2). In the nested case-control design, the superset A method gave biased estimates for $\beta_x$ in all situations, while in the case-cohort design the bias was greatest in the no correlation setting, where the imputations were made mostly based on the time and censoring information.

Lastly, for imputation with an interaction using rejection sampling the

MSE estimates are 0.112, 0.134, 0.018, 0.106 for method A nested case-control superset and 0.087, 0.142, 0.021, 0.121 for the case-cohort superset. For both sampling methods the effects for $X$ and $Z_1$ are overestimated, and this is most clear for the nested case-control sampling. For method B the MSE estimates for the nested case-control superset are 0.054, 0.119, 0.015, 0.079 and for the case-cohort superset they are 0.059, 0.117, 0.012, 0.076. Both superset B results using rejection sampling show a little undercumulative baseline of the interaction effect with a bias of $-0.07$. This can be a sign of uncongeniality for superset method B, but needs further investigation. Despite this sign of weakness, superset method B for rejection sampling seems to show the best performance, of the alternatives to full cohort imputation, overall in the simulation experiments.

# CHAPTER 6

# Discussion and further work

In this thesis multiple imputation for sampled cohort data with Cox regression has been investigated. More specifically, we have looked at how multiple imputation for a only subset of the cohort can be performed. Two methods for multiple imputation in a superset of the nested case-control or case-cohort sample, superset method A and superset method B, have been considered. The results of carried out simulation experiments show improved performance compared to the classical methods of nested case-control and case-cohort sampling. The superset methods (except method A using rejection sampling for nested case-control supersets) show very little bias in the standard setting and with a clear gain in efficiency, especially for the variables fully observed in the superset/cohort. Compared to imputation of the full cohort, there is slightly less bias with approximate superset imputation and slightly more bias with superset imputation using rejection sampling (method B), but the main difference when only the superset is imputed is loss in efficiency.

When the superset sizes are increased, there is a further clear gain in efficiency. For superset method B with rejection sampling the small bias in the setting with smallest superset size is further decreased. For the approximate imputation methods the tendency seen in the full cohort imputation to underestimate larger effects is slightly increased with increased superset size. In the setting with an auxiliary variable (surrogate of $X$) the performance is also improved when only imputing the superset, although less than the improvement when imputing the entire cohort. All superset methods (except method A with rejection sampling) are very close to unbiased and are more efficient when imputing with the auxiliary variable. Also when imputing with no correlation between the partially observed variable and the other covariates there is gain in efficiency with the superset methods compared to the classical sampled cohort analysis. For the estimated effect of $X$ this is mainly with the nested case-control supersets. For the simulation experiment with an interaction term, approximate imputation for the superset and approximate imputation for the full cohort are both clearly biased. As seen in other studies, approximate imputation underestimates the interaction effect. In this setting rejection sampling imputation (method B) of only the superset give some downward bias compared to imputing the full cohort for the interaction effect, but overall less bias and lower MSE's than the classical sampled cohort estimators.

In all, imputing only a superset of the sampled cohort (except with method A using rejection sampling) give good performance with little absolute bias and clear gains in efficiency compared to the classical nested case-control and case-cohort methods in simple settings. In the more advanced setting with an interaction term the approximate imputation superset methods show non-negligible bias (as do full cohort approximate imputation), and rejection sampling superset method B show some bias compared to imputing the entire cohort with rejection sampling.

There are a number of things that would have been of interest to explore further, but for which there was not enough time in work on this thesis. First of all the systematic bias of superset method A using rejection sampling was somewhat unexpected since this method arguably has a stronger theoretical basis than the more naive superset B method. Some modifications to method A are mentioned in Keogh, Seaman, et al. (2018). This and possibly other modifications would be of interest investigate. Furthermore, it requires more study to determine if the small downward bias of superset method B using rejection sampling for the interaction term is a result of uncongeniality between the imputation model (implied by the Cox regression estimates in the superset) and the sampled cohort analysis model.

To be fair, the superset imputation results should have been compared to nested case-control and case-cohort studies that also utilise more of the cohort information. Stratified sampling is one way that it is natural to investigate further, and how multiple imputation perform in comparison with stratified sampled cohort studies. Because the imputed supersets are treated as if they were sampled cohorts (of a larger size) when fitting the analysis models, it should be possible to modify these methods using developed work from stratified sampling. Then multiple imputation could possibly also be improved with stratified sampling. One could have considered stratified sampling for both the sampled cohort and the superset, or simple random sampling for the sampled cohort and stratified sampling for the superset.

Another limitation of this thesis is that only one cohort size was investigated. Since superset imputation is a solution to using multiple imputation when it is not possible to multiply impute for the entire cohort it should be investigated for larger cohort sizes, say for a couple of million individuals. Computationally the superset methods are very scalable to larger cohort sizes. Especially superset method B can easily be applied on very large cohorts as the computational demand and the amount of individuals for which covariate information needs to collected is decided by the number of controls or the subcohort size of the superset. Superset method A is also believed to be scalable as computing the estimates of the cumulative hazard or baseline cumulative hazard is in generally much faster than the iterative imputation algorithms. Superset method A uses more of the full cohort information and it is likely that it will be more affected performance-wise with larger cohorts than superset method B.

In addition, only one sampled cohort size was consider, about 10% of the cohort, and only two superset sizes, about 20% and 30% of the full cohort. The results showed that increasing the superset gave better performance, but to get a sense of how performance changes with different constellations of sampled cohort, superset and cohort sizes more simulation experiments are needed. Although, results from studies on the behaviour of classical nested case-control and case-cohort estimators under different sampled cohort and full cohort sizes could likely give some guidance for the superset methods. The methods are most relevant in the situation with rare events in large cohort, but the sensitivity of the performance with respect to incidence rate could also be explored.

In the design of simulations studies, what parts of the parameter space and what settings to explore is not obvious. The simulations in this thesis have been guided by previous research on related methods and from the running FLC example considered throughout. Previous studies of multiple imputation in the full cohort and alternative methods like the likelihood and IPW approaches have shown performance varying with e.g. weak and strong effect sizes. Here two moderate/strong and one weaker effect size were only considered in the same data-generating mechanism. More complex settings, e.g. with more variables, and settings with closer resemblance to real studies would be of interest to consider. An advantage of multiple imputation is the natural incorporated handling of missing values (by chance, not by design) that is common in applications, but missingness by chance in addition to the missingness by design was not investigated here. It is possible that superset method A would perform better in more complex settings and with missingness arising by chance (but MAR). Another, aspect is model misspecification, e.g. of the proportionality of Cox regression model or of imputation model. Further, it remains to get some insight into what situations the superset method would be preferable to existing, comparable methods. Still the superset method did overall show better performance compared to the classic sampled cohort methods. Ultimately to be used in applications, further study of the operating characteristics of the superset imputation methods are needed. Simulation settings for investigating the methods with regard to application should then be tailored to the problem at hand. With this in mind, and with respect to reproducibility, the code for the simulation study carried out will be available at https://github.com/a-njos/simulations_master_thesis.

In summary, this thesis, though much based on the work Keogh, Seaman, et al. (2018), presents and investigates an original way of using multiple imputation for Cox regression with sampled cohorts where imputation is only done for part of the cohort. Much research has been done on nested case-control and case-cohort sampling designs (Thomas (1977), Prentice (1986), Langholz and Borgan (1995), Borgan, Langholz, et al. (2000), Borgan and Samuelsen (2016) and the superset design attempts to make use of this work. Two superset methods have been considered for nested case-control and for case-cohort supersets, and have been compared to full cohort analysis, full cohort imputation and classical sampled cohort methods. Based on the simulation experiments performed, multiple imputation of a superset of the sampled cohort seems like a promising alternative to existing methods for analysing time to occurrences of rare events in large cohorts.

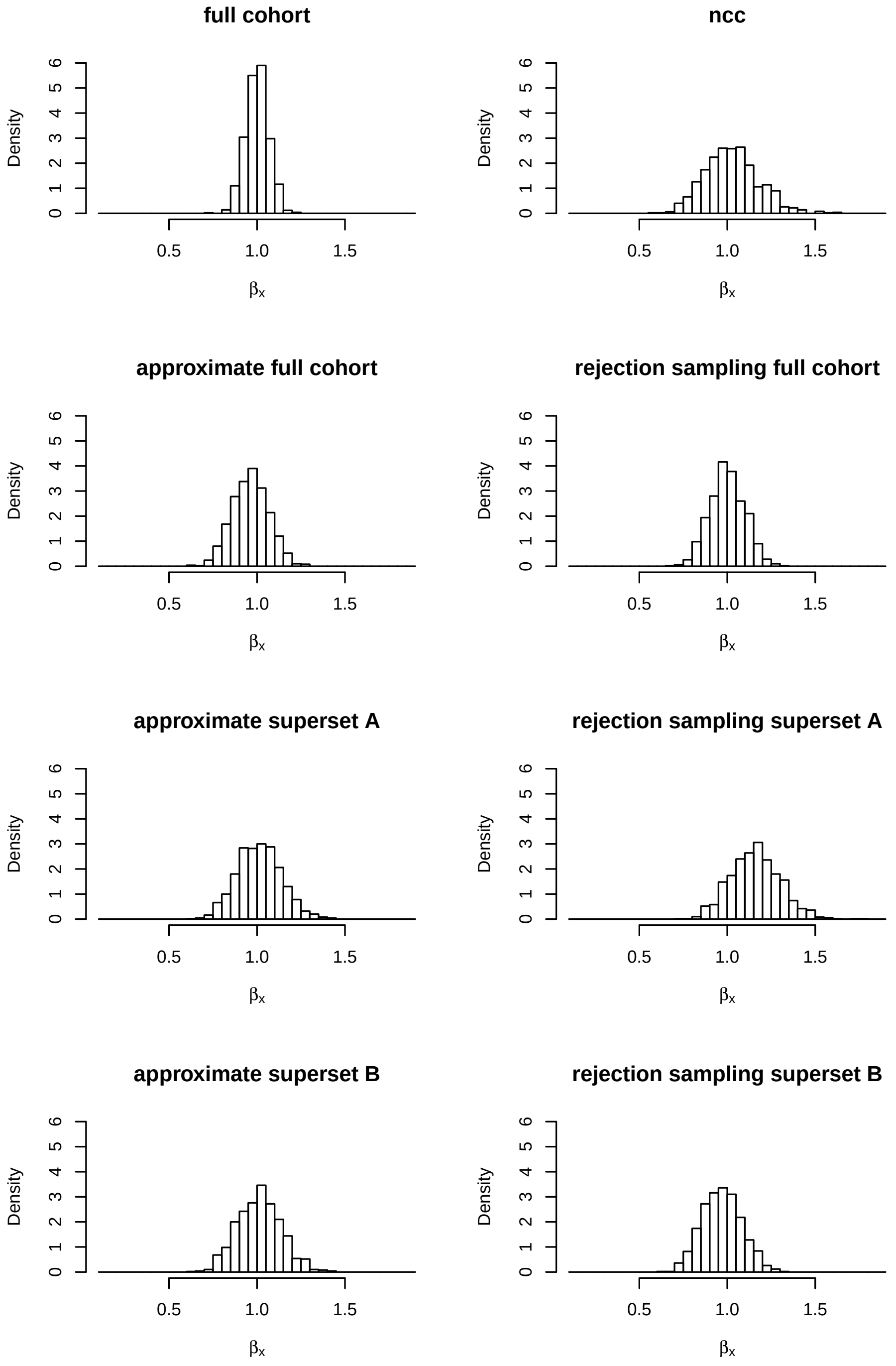
# Appendices

# APPENDIX A

# Histograms of simulation estimates

This appendix includes histograms of the estimates for $\beta_x$ from the standard setting (i-ii), and the estimates of $\beta_x$ and $\beta_{xz_1}$ from the interaction setting(v). The scale in the histograms is the same for standard setting (i) and (ii) for both designs. In the interaction setting (v) the scale for $\beta_x$ are the same for both designs and the scale for $\beta_{xz_1}$ are the same for both designs.

## Nested case-control design


full cohort

ncc

approximate full cohort

rejection sampling full cohort

approximate superset A

rejection sampling superset A

approximate superset B

rejection sampling superset B

Figure A.1: Histograms of estimates for $\beta_x$ in standard setting (i) with nested case-control design

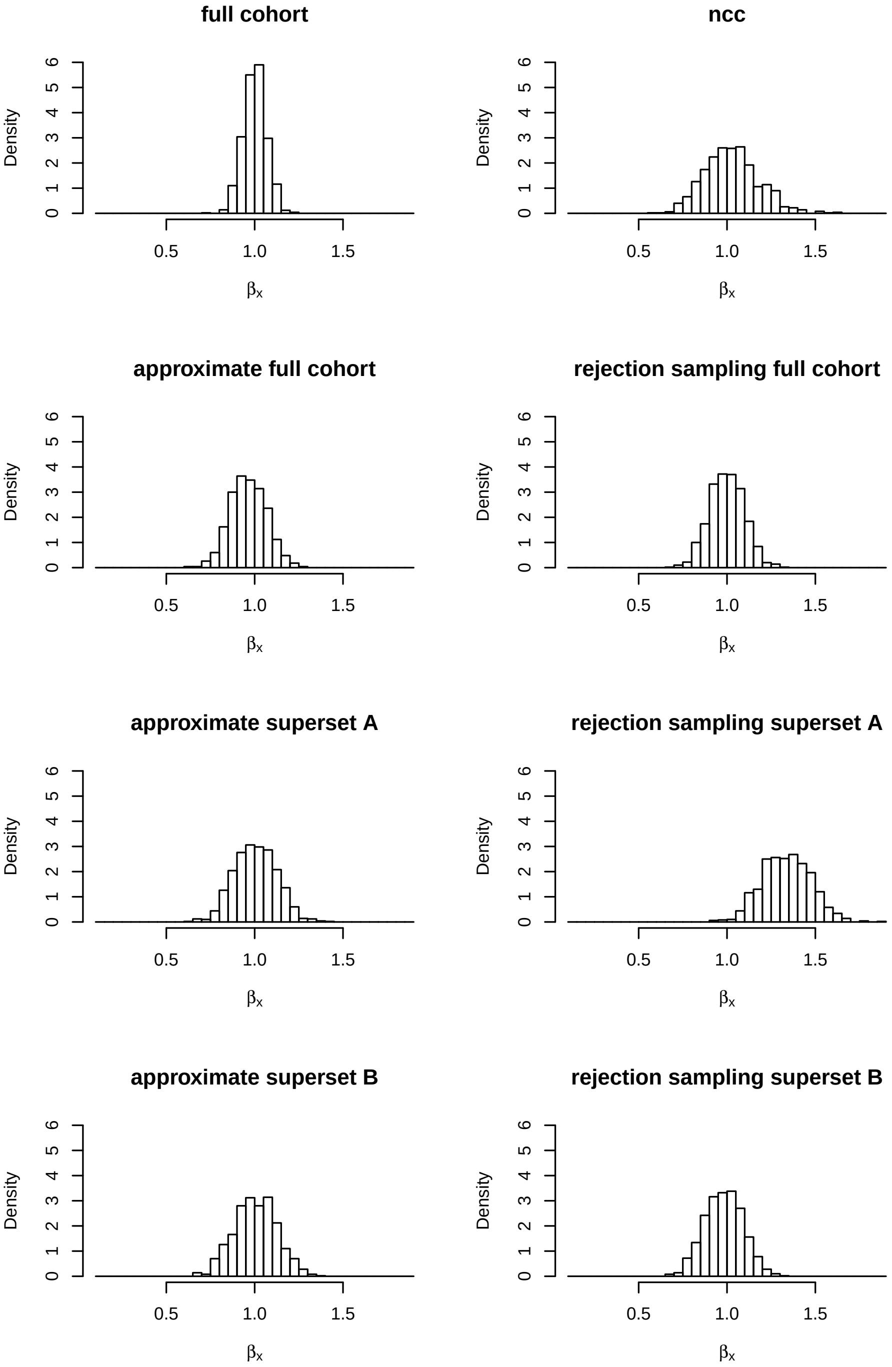


Figure A.2: Histograms of estimates for $\beta_x$ in standard setting 7 controls (ii) nested case-control design

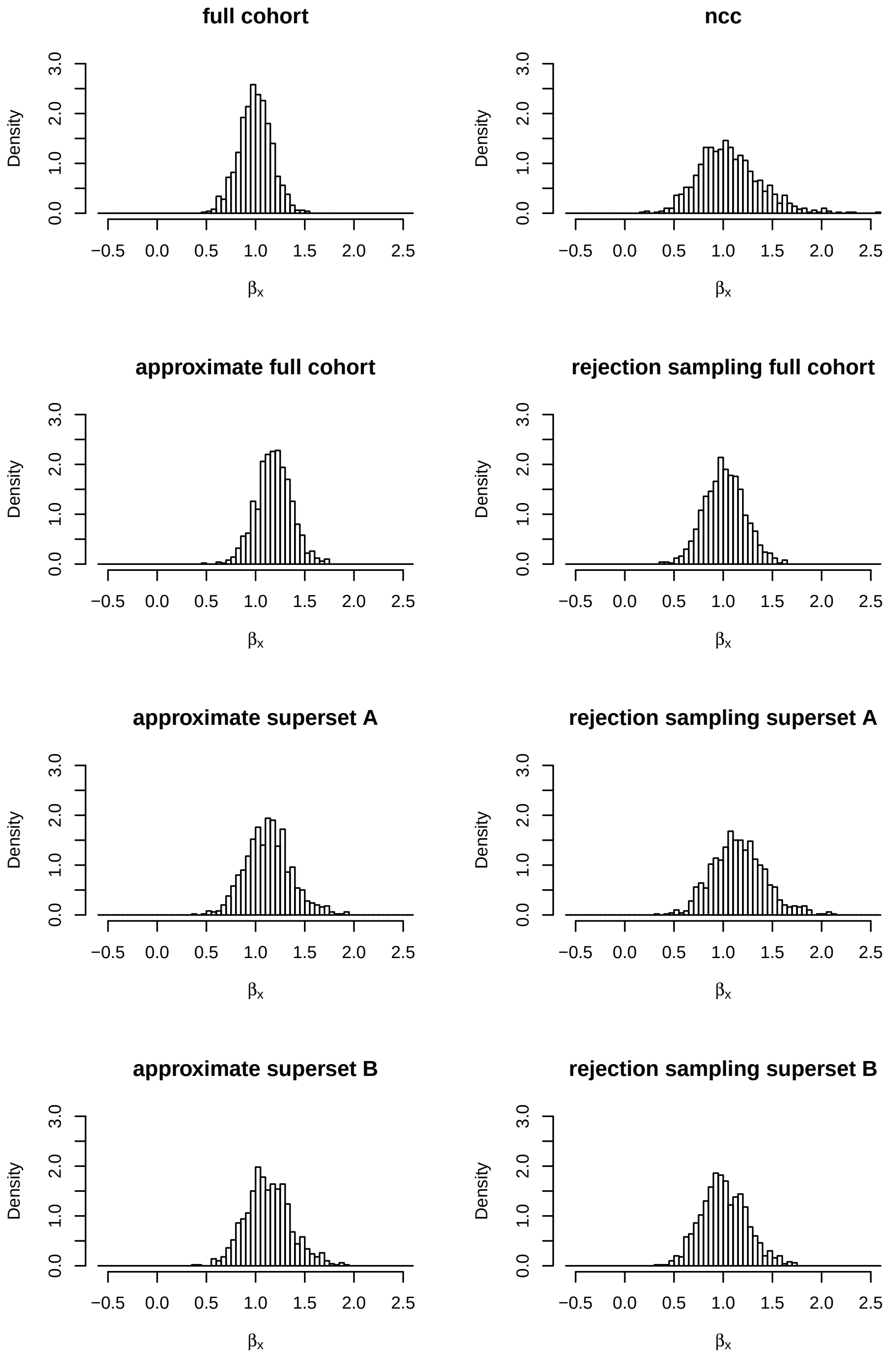


Figure A.3: Histograms of estimates for $\beta_x$ in interaction setting (v) with nested case-control design

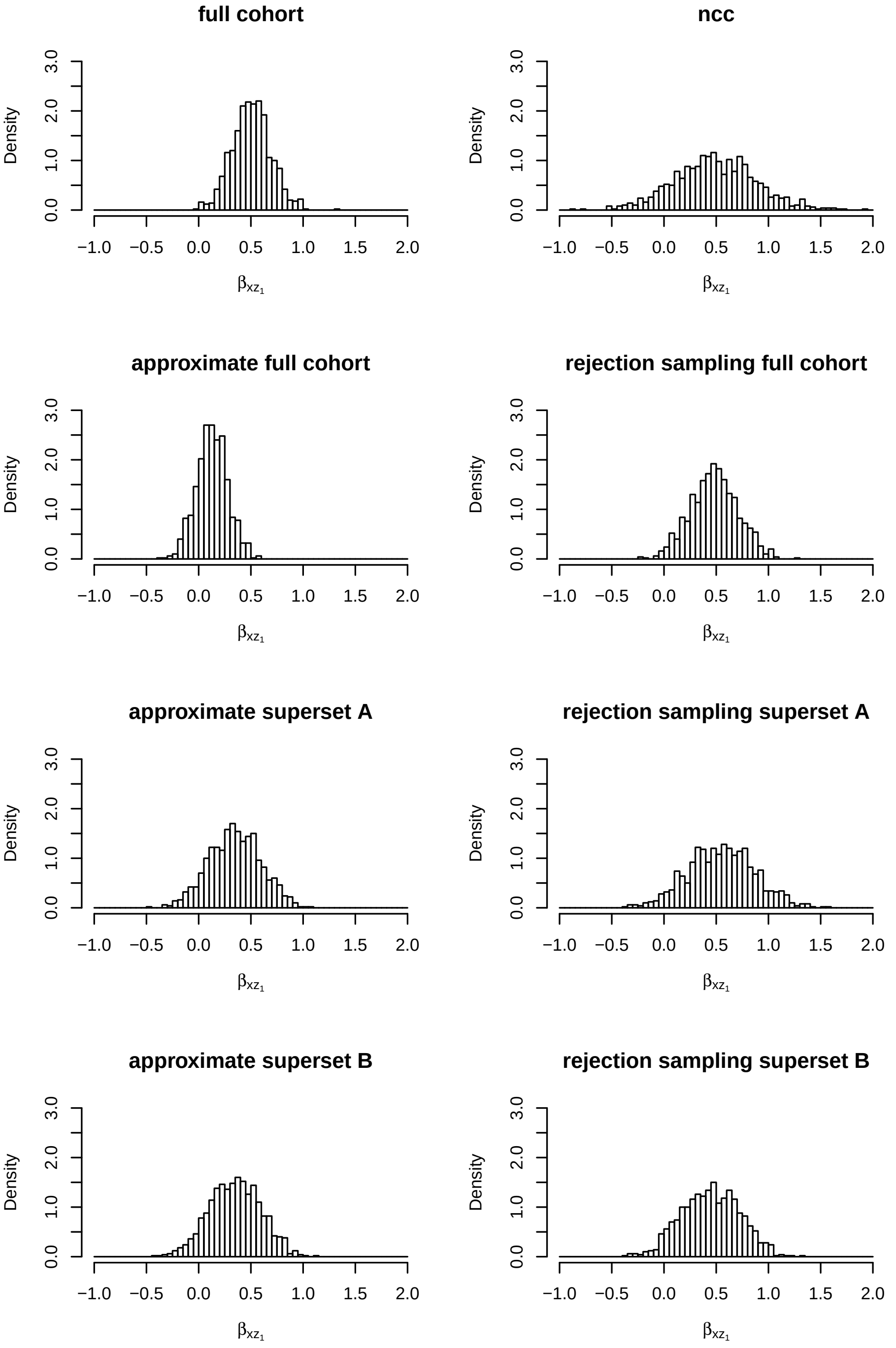
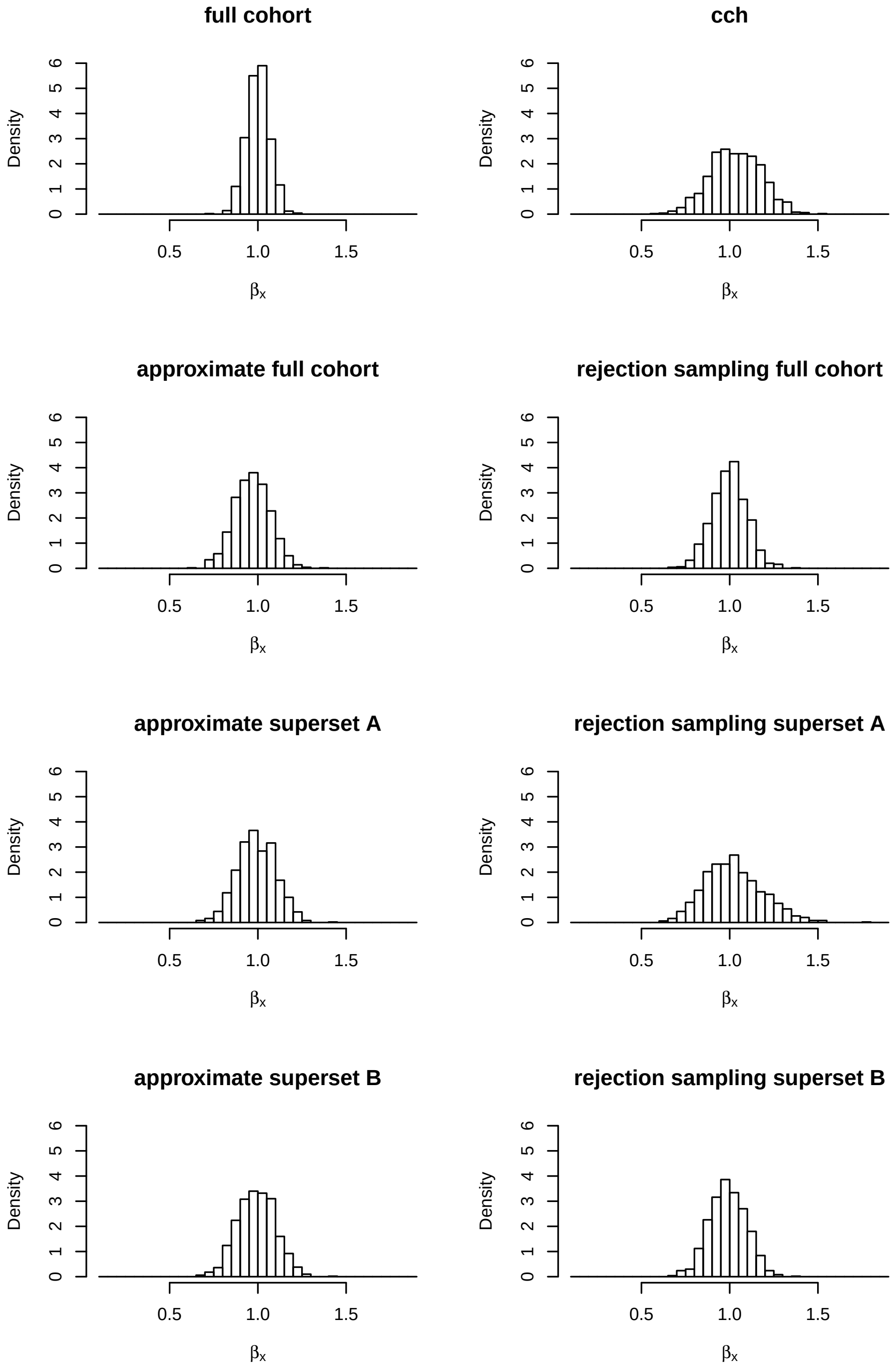


Figure A.4: Histograms of estimates for $\beta_{xz_1}$ in interaction setting (v) with nested case-control design

### Case-cohort design



full cohort

cch

approximate full cohort

rejection sampling full cohort

approximate superset A

rejection sampling superset A

approximate superset B

rejection sampling superset B

Figure A.5: Histograms of estimates for $\beta_x$ in standard setting (i) with case-cohort design

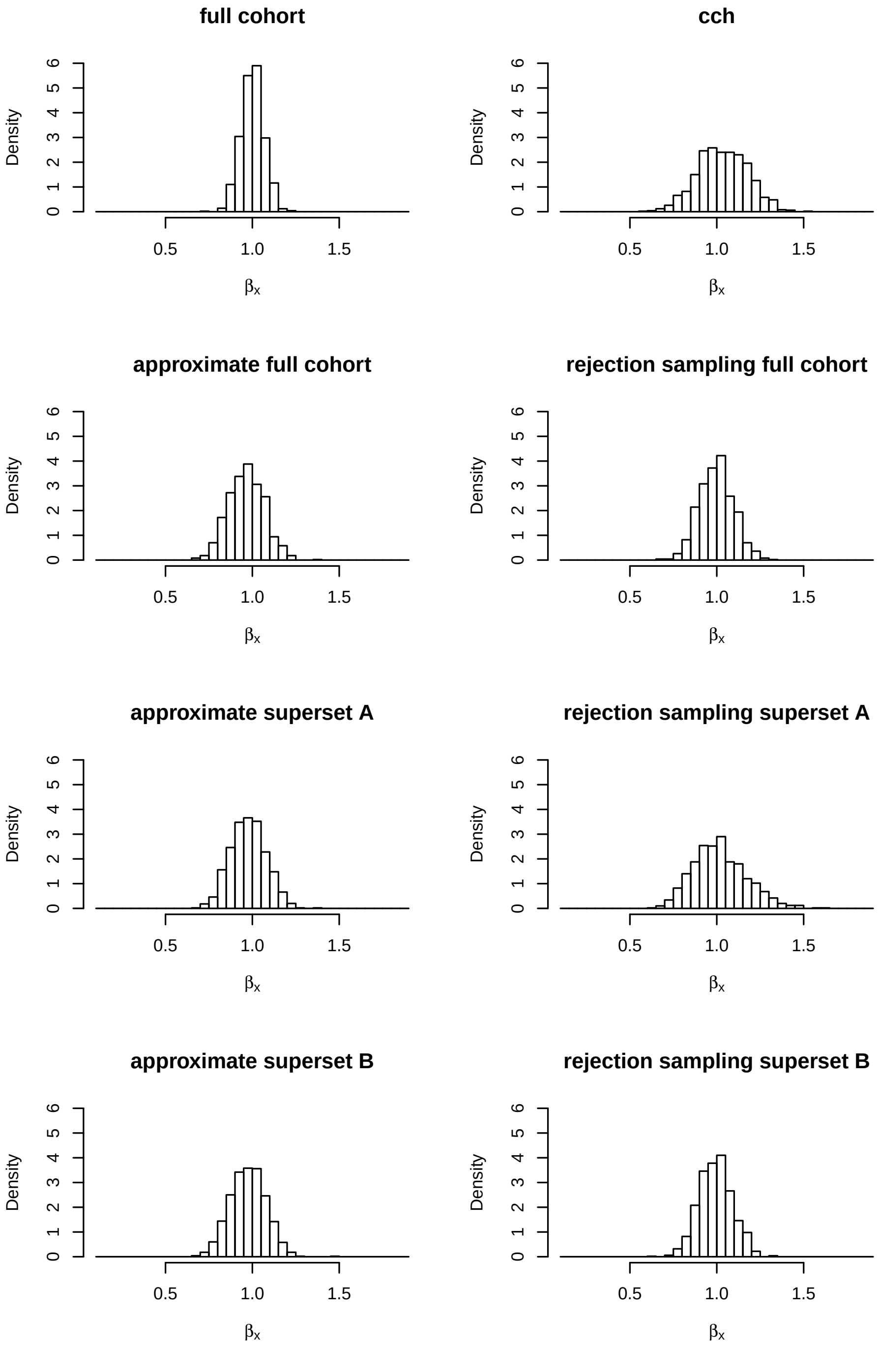


Figure A.6: Histograms of estimates for $\beta_x$ in standard setting (ii) with case-cohort design

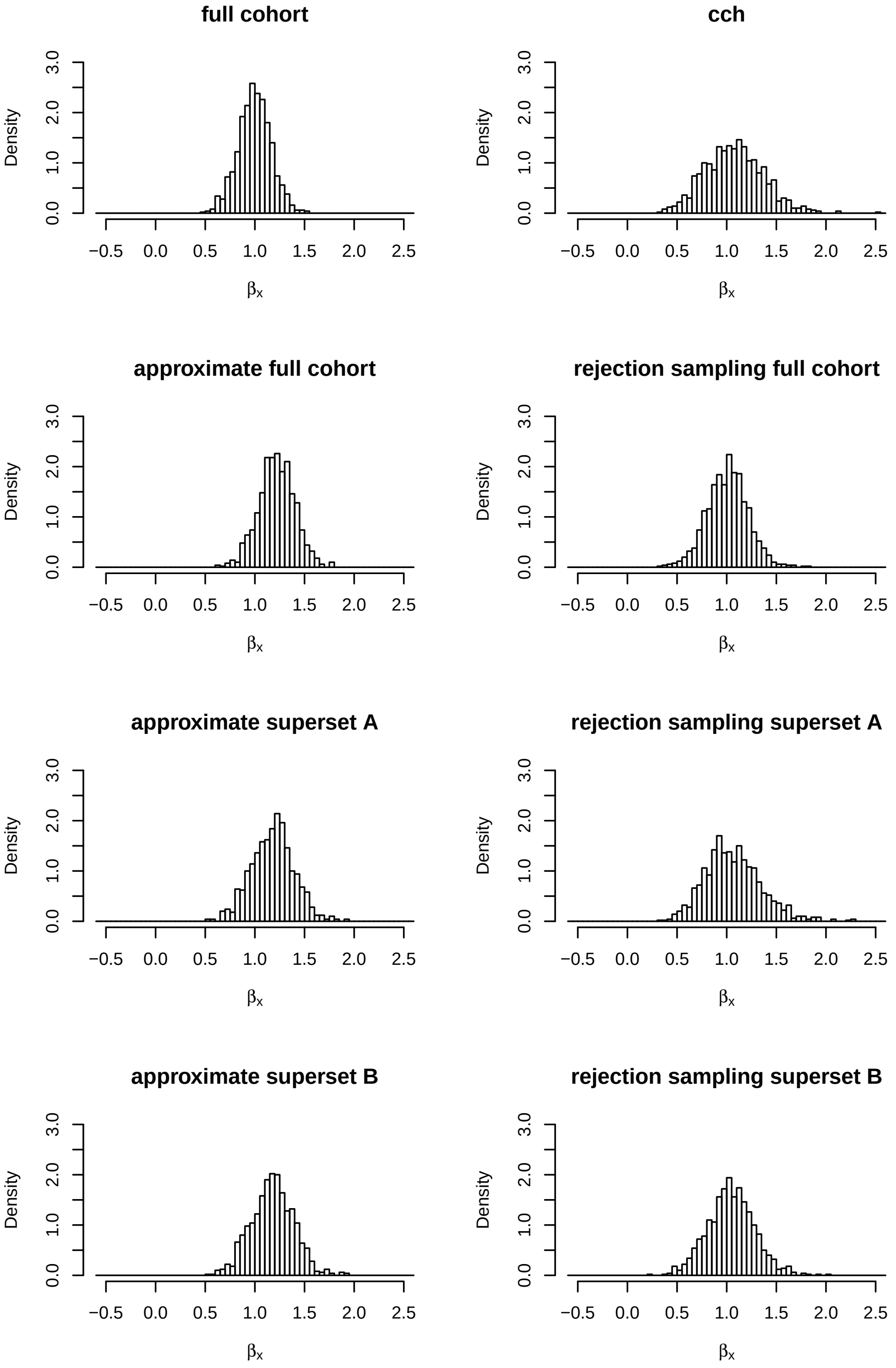


Figure A.7: Histograms of estimates for $\beta_x$ in interaction setting (v) with case-cohort design

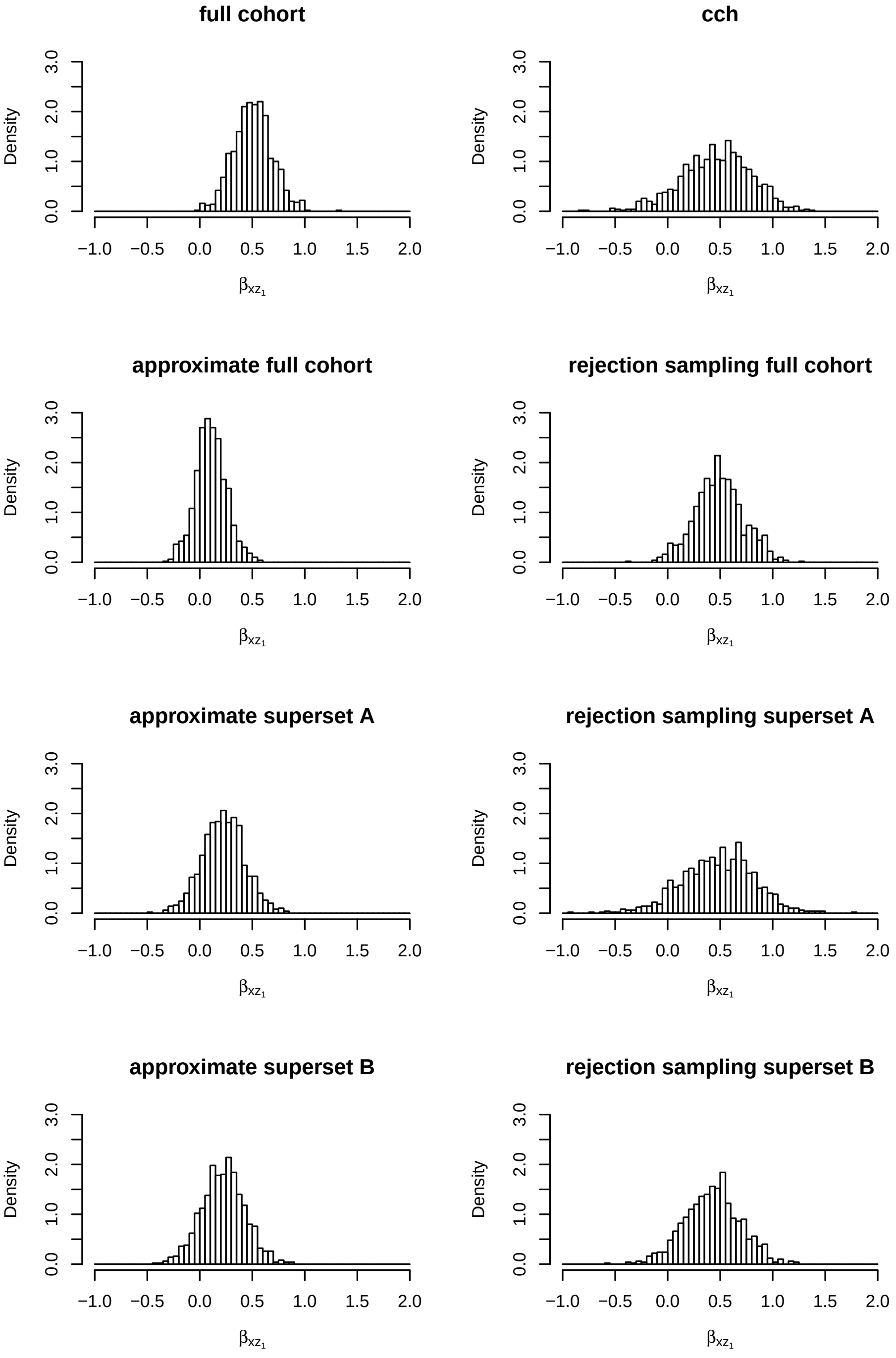


Figure A.8: Histograms of estimates for $\beta_{xz_1}$ in interaction setting (v) with case-cohort design

# APPENDIX B

# R code

This appendix includes selected R code for the simulations experiments. The first script generates and samples nested case-control and case-cohort samples and supersets. The second script analyses the nested case-control data and the third script analyses the case-cohort data in the standard setting. The simulations have been run in R version 3.6.0 (2019-04-26) – "Planting of a Tree" (Copyright (C) 2019 The R Foundation for Statistical Computing), on platform: x86_64-apple-darwin15.6.0 (64-bit), and with version 2.44-1.1 of the survival package, version 1.40 of the smcfcs, version 2.4 of the mitools and version 3.6.0 of mice.

```
# ------------------------------
# GENERATE FULL-COHORT DATA
# ------------------------------
# Code based on https://github.com/ruthkeogh/MI-CC by Ruth H.Keogh
# ------------------------------
sim_setup = "standard"     #  "standard","aux","int","nocorr"
sim_idx = 1
filepath = paste0("simulations/",sim_setup[sim_idx],"/")
nsim = 1000
# ------------------------------
# Set parameter values, etc
n = 5000     # cohort size
close.time=15  # Maximum follow-up time

beta.x=1   # log hazard ratio for X
beta.z1=1   # log hazard ratio for  binary Z1
beta.z2=0.5     # log hazard ratio for continous Z2
beta_int = ifelse(sim_setup[sim_idx]=="int",0.5,0) # interaction X and Z1

beta_x_drop =0    # log hazard ratio for X
beta_z1_drop =0     # log hazard ratio for Z1
beta_z2_drop  =0      # log hazard ratio for Z2

p_z1 = 0.5   # Bernoulli probability for Z1
mu_z2 = 0    # mean for normal Z2

a_x = 0      # constant for X
b_x=0.25   # influence of Z1 on X
c_x=0.25   # influence of Z2 on X

if(sim_setup[sim_idx]=="nocorr"){
  b_x=0
  c_x=0
}
```

```
eta_v = 0.8 #  Gaussian standard deviation for auxiliary
lambda=NA  # Weibull baseline scale for event of interest
if(sim_setup[sim_idx]=="standard"){
  lambda=0.00000040
}else if(sim_setup[sim_idx]=="int"){
  lambda=0.00000025
}else if(sim_setup[sim_idx]=="nocorr"){
  lambda=0.00000055
}
kappa = 4  # Weibull baseline shape
lambda_drop = 0.00002    # Weibull baseline scale dropout time
kappa_drop=4     # Weibull baseline shape for droput time
# -----------------------------
# starting data generating process
set.seed(123)
for(j in seq(1,nsim)){
  # Generate id numbers
  id=seq(1,n)
  # -----------------------------
  # Generate covariates
  z1=rbinom(n,1,p_z1)
  z2=rnorm(n,mu_z2,1)
  x=rnorm(n,a_x +b_x*z1+c_x*z2,1)
  v = x + rnorm(n,0,eta_v)
  # -----------------------------
  # Generate potential event times
  u=runif(n,0,1)
  t.event=(-log(u)*(1/lambda)*
             exp(-(beta.x*x+beta.z1*z1+
                     beta.z2*z2+x*z1*beta_int)))^(1/kappa)
  # -----------------------------
  # Generate potential drop-out time
  u=runif(n,0,1)
  t.drop=(-log(u)*(1/lambda_drop)*
            exp(-(beta_x_drop*x+beta_z1_drop*z1+
                    beta_z2_drop*z2)))^(1/kappa_drop)
  # -----------------------------
  # Generate time for event or drop out
  t=pmin(t.event,t.drop,close.time)
  cause=1*(t==t.event)+2*(t==t.drop)+3*(t==close.time)
  # 1: event, 2: drop out, 3: administrative censoring
  d=ifelse(cause==1,1,0)
  # -----------------------------
  # the full-cohort data with no missingness
  cohort=data.frame(id,t,d,x,z1,z2,v,cause)
  saveRDS(cohort,file=paste0(filepath,"cohort",j,".rds"))

  # -----------------------------
  # GENERATE NESTED CASE-CONTROL DATA
  # -----------------------------
  # Generate nested-case-control superset sample
  n.controls.super = 3 # number of controls per case
  n.controls.super.ext = 7
  n.controls.ncc = 1
  ncc.super.ext = NULL
  ncc.super=NULL
  ncc=NULL
  no.sample=0
  for (i in which(cohort$d==1))
  {
    # Select control(s) for nested case-control
    possible.controls=which(cohort$t>=cohort$t[i])
  if (length(possible.controls)>=n.controls.super.ext){
    controls.super.ext=sample(possible.controls,n.controls.super.ext)
    controls.super=sample(controls.super.ext,n.controls.super)
    controls.ncc=sample(controls.super,n.controls.ncc)
    ncc.super.ext=rbind(ncc.super.ext,cohort[i,])
    ncc.super.ext=rbind(ncc.super.ext,cohort[controls.super.ext,])
    ncc.super=rbind(ncc.super,cohort[i,])
    ncc.super=rbind(ncc.super,cohort[controls.super,])
    ncc=rbind(ncc,cohort[i,])
    ncc=rbind(ncc,cohort[controls.ncc,])
    no.sample=no.sample+1}
}
ncc.super.ext$setno=rep(1:no.sample,each=n.controls.super.ext+1)
ncc.super.ext$case=rep(c(1,rep(0,n.controls.super.ext)),no.sample)
ncc.super$setno=rep(1:no.sample,each=n.controls.super+1)
ncc.super$case=rep(c(1,rep(0,n.controls.super)),no.sample)
ncc$setno=rep(1:no.sample,each=n.controls.ncc+1)
ncc$case=rep(c(1,rep(0,n.controls.ncc)),no.sample)
#-----------------------
# generate indicator of being in the nested case-control sample
cohort.ncc = cohort
cohort.ncc$in.ncc <- cohort.ncc$id%in%ncc$id

cohort.super = ncc.super
cohort.super$in.ncc<-cohort.super$id%in%ncc$id

cohort.super.ext = ncc.super.ext
cohort.super.ext$in.ncc<-cohort.super.ext$id%in%ncc$id

#-----------------------
# make x missing in those outside the nested case-control sample
cohort.ncc$x<-ifelse(cohort.ncc$id%in%ncc$id,cohort.ncc$x,NA)
cohort.super$x<-ifelse(cohort.super$id%in%ncc$id,cohort.super$x,NA)
cohort.super.ext$x<-ifelse(cohort.super.ext$id%in%ncc$id,
                           cohort.super.ext$x,NA)
# ------------------------------
# save NCC data sets
# NCC sample
saveRDS(ncc,file=paste0(filepath,"ncc",j,".rds"))

# NCC within full cohort
saveRDS(cohort.ncc,file=paste0(filepath,"cohort.ncc",j,".rds"))

# NCC within superset ncc
saveRDS(cohort.super,file=paste0(filepath,"cohort.super.ncc",j,".rds"))

# NCC within superset extended ncc
saveRDS(cohort.super.ext,
        file=paste0(filepath,"cohort.super.ext.ncc",j,".rds"))

# ------------------------------
# GENERATE CASE-COHORT DATA
# ------------------------------
cohort.caco=cohort
# ------------------------------
# Generate subcohort
n.subco=250
cohort.caco$subco<-c(rep(1,n.subco),rep(0,n-n.subco))

n.subco.super = 750
cohort.caco$subco.super <- c(rep(1,n.subco.super),rep(0,n-n.subco.super))
```

```
  n.subco.super.ext = 1750
  cohort.caco$subco.super.ext <- c(rep(1,n.subco.super.ext),
                                 rep(0,n-n.subco.super.ext))
  # -----------------------------
  #make x1 missing in those outside the case-cohort sample
  cohort.caco$x<-ifelse(cohort.caco$subco==1|cohort.caco$d==1,
                        cohort.caco$x,NA)
  # -----------------------------
  # Generate data-set which is just the case-cohort substudy
  caco=cohort.caco[cohort.caco$subco==1|cohort.caco$d==1,]

  # Generate data-set which is the case-cohort supersets
  cohort.super.caco=cohort.caco[cohort.caco$subco.super==1|
                                cohort.caco$d==1,]

  cohort.super.ext.caco=cohort.caco[cohort.caco$subco.super.ext==1|
                                    cohort.caco$d==1,]

  cohort.super.caco$entertime=ifelse(cohort.super.caco$d==1 &
                                   cohort.super.caco$subco.super==0,
                                   cohort.super.caco$t-0.001,0)

  cohort.super.ext.caco$entertime=ifelse(cohort.super.ext.caco$d==1&
                                  cohort.super.ext.caco$subco.super.ext==0,
                                  cohort.super.ext.caco$t-0.001,0)

  # -----------------------------
  # save case-cohort data sets
  saveRDS(caco,file=paste0(filepath,"caco",j,".rds"))

  saveRDS(cohort.caco,file=paste0(filepath,"cohort.caco",j,".rds"))

  saveRDS(cohort.super.caco,
          file=paste0(filepath,"cohort.super.caco",j,".rds"))

  saveRDS(cohort.super.ext.caco,
          file=paste0(filepath,"cohort.super.ext.caco",j,".rds"))
}
```

```
# Analyse ncc data
# Code based on https://github.com/ruthkeogh/MI-CC by Ruth H.Keogh
# packages ---------------------
library(survival)
library(mice)
library(smcfcs)
library(mitools)
# setup -----------------------
setup=  "standard"
filepath=paste0("simulations/",setup,"/")
nimp= 10
n.it = 100
npara=3
nsim= 1000
nmethods = 8
res_mat = matrix(NA,nrow=nsim,ncol=2*npara*nmethods)

# standard formulas
formula_full = "Surv(t,d)~x+z1+z2"
formula_ncc = "Surv(t,case)~x+z1+z2+strata(setno)"
sm_formula_full = "Surv(t,d)~x+z1+z2"
sm_formula_ncc = "Surv(t,case)~x+z1+z2+strata(setno)"
predictors_aprx = c("z1","z2","d","chaz")
predictors_rs = c("z1","z2")

#-------------------------------
#===============================
# Run analyses
#===============================
set.seed(1001)
for(j in seq(1,nsim)){
  #===============================
  # cox analysis using full cohort data
  #===============================
  # # load data set
  cohort = readRDS(file=paste0(filepath,"cohort",j,".rds"))
  model=coxph(as.formula(formula_full),data=cohort)
  res_full = c(model$coefficients,sqrt(diag(model$var)))

  #===============================
  # traditional analysis using nested case-control sample
  #===============================
  # load data set
  ncc = readRDS(file=paste0(filepath,"ncc",j,".rds"))

  # fit the model
  model = coxph(as.formula(formula_ncc),data=ncc)
  res_ncc = c(model$coefficients,sqrt(diag(model$var)))

  #===============================
  # MI-approx:  full-cohort approach
  #===============================
  cohort.ncc= readRDS(file=paste0(filepath,"cohort.ncc",j,".rds"))

  # Compute Nelson-Aalen estimate of the cumulative hazard
  cohort.ncc$chaz=nelsonaalen(cohort.ncc,t,d)

  # predictor matrix which determines the imputation models for x1
  pred.mat=matrix(0,nrow=dim(cohort.ncc)[2],ncol=dim(cohort.ncc)[2])
  colnames(pred.mat)=names(cohort.ncc)
  rownames(pred.mat)=names(cohort.ncc)
  pred.mat["x",predictors_aprx]=1
```

```
# method of imputation for x1
method.vec=rep("",dim(cohort.ncc)[2])
method.vec[which(colnames(cohort.ncc)=="x")]="norm"

# perform the imputation
imp<-mice(cohort.ncc, m = nimp, method = method.vec,
          predictorMatrix = pred.mat,
          maxit = n.it, diagnostics = FALSE, printFlag = F)

# Fit the analysis model in each imputed data set
models<-with(imp,coxph(as.formula(formula_full)))

# Combine estimates across the imputed data sets using Rubin’s Rules
summary_aprx = summary(pool(models))
res_aprx = c(summary_aprx[,"estimate"],summary_aprx[,"std.error"])

#================================
#MI-SMC: full-cohort approach
#================================
cohort.ncc= readRDS(file=paste0(filepath,"cohort.ncc",j,".rds"))

# predictor matrix which determines the imputation models for x1
pred.mat=matrix(0,nrow=dim(cohort.ncc)[2],ncol=dim(cohort.ncc)[2])
colnames(pred.mat)=names(cohort.ncc)
rownames(pred.mat)=names(cohort.ncc)
pred.mat["x",predictors_rs]=1

# method of imputation for x1
method.vec=rep("",dim(cohort.ncc)[2])
method.vec[which(colnames(cohort.ncc)=="x")]="norm"

# perform the imputation
imp <- smcfcs(cohort.ncc, smtype="coxph", smformula=sm_formula_full,
              method=method.vec,predictorMatrix=pred.mat,m = nimp,
              numit =n.it, rjlimit = 10000,noisy=F)

# obtain estimates from imputed data sets and combine using Rubin’s Rules
impobj <- imputationList(imp$impDatasets)
models <- with(impobj, coxph(as.formula(formula_full)))
coef = MIcombine(models)$coefficients
se = sqrt(diag(MIcombine(models)$variance))
res_rej = c(coef,se)
#================================
#MI-approx:  superset A ncc
#================================
cohort.ncc= readRDS(file=paste0(filepath,"cohort.super.ncc",j,".rds"))

# Nelson-Aalen estimate of the cumulative hazard for full cohort
cohort$chaz=nelsonaalen(cohort,t,d)

#add cumulative hazard into superset ncc data
cohort.merge<-cohort[,c("id","chaz")]
cohort.ncc<-merge(cohort.ncc,cohort.merge,by.x="id")

# predictor matrix for the imputation models for x1 (not incl. outcome)
pred.mat=matrix(0,nrow=dim(cohort.ncc)[2],ncol=dim(cohort.ncc)[2])
colnames(pred.mat)=names(cohort.ncc)
rownames(pred.mat)=names(cohort.ncc)
pred.mat["x",predictors_aprx]=1

# method of imputation for x1
```

```
method.vec=rep("",dim(cohort.ncc)[2])
method.vec[which(colnames(cohort.ncc)=="x")]="norm"

# perform the imputation
imp<-mice(cohort.ncc, m = nimp, method = method.vec,
          predictorMatrix = pred.mat,
          maxit = n.it, diagnostics = FALSE, printFlag = F)

# Fit the analysis model in each imputed data set
models<-with(imp,coxph(as.formula(formula_ncc)))

# Combine estimates across the imputed data sets using Rubin’s Rules
summary_aprx = summary(pool(models))
res_aprx_sup = c(summary_aprx[,"estimate"],summary_aprx[,"std.error"])

#===============================
# MI-approx: (naive) Superset B ncc
#===============================
cohort.ncc= readRDS(file=paste0(filepath,"cohort.super.ncc",j,".rds"))

# Nelson-Aalen estimate of the cumulative hazard for the superset ncc
cohort.ncc$chaz = nelsonaalen(cohort.ncc,t,d)

# predictor matrix which determines the imputation models for x1
pred.mat=matrix(0,nrow=dim(cohort.ncc)[2],ncol=dim(cohort.ncc)[2])
colnames(pred.mat)=names(cohort.ncc)
rownames(pred.mat)=names(cohort.ncc)
pred.mat["x",predictors_aprx]=1

#method of imputation for x1
method.vec=rep("",dim(cohort.ncc)[2])
method.vec[which(colnames(cohort.ncc)=="x")]="norm"

#perform the imputation
imp<-mice(cohort.ncc, m = nimp, method = method.vec,
          predictorMatrix = pred.mat,
          maxit = n.it, diagnostics = FALSE, printFlag = F)

# Fit the analysis model in each imputed data set
models<-with(imp,coxph(as.formula(formula_ncc)))

# Combine estimates across the imputed data sets using Rubin’s Rules
summary_aprx = summary(pool(models))
res_aprx_sup_nai = c(summary_aprx[,"estimate"],summary_aprx[,"std.error"])

#===============================
#MI-SMC: superset A ncc
#===============================
cohort.ncc= readRDS(file=paste0(filepath,"cohort.super.ncc",j,".rds"))

# Compute number at risk at each event time using the full cohort data
nrisk.fit<-survfit(Surv(t,d)~1,data=cohort)
ord.t.d1<-order(cohort$t[cohort$d==1])
# number at risk at each unique event time
numrisk<-summary(nrisk.fit,censored=F)$n.risk

# add numbers at risk time into the nested case-control data set
cohort.ncc$numrisk<-NA
cohort.ncc$numrisk[cohort.ncc$case==1][ord.t.d1]<-numrisk

# assign number at to every individual in each set
cohort.ncc$numrisk<-ave(cohort.ncc$numrisk, cohort.ncc$setno,
                          FUN = function(x) sum(x, na.rm=T))

  #predictor matrix which determines the imputation models for x
  pred.mat=matrix(0,nrow=dim(cohort.ncc)[2],ncol=dim(cohort.ncc)[2])
  colnames(pred.mat)=names(cohort.ncc)
  rownames(pred.mat)=names(cohort.ncc)
  pred.mat["x",predictors_rs]=1

  #method of imputation for x1
  method.vec=rep("",dim(cohort.ncc)[2])
  method.vec[which(colnames(cohort.ncc)=="x")]="norm"

  #perform the imputation
  imp<-smcfcs.nestedcc(cohort.ncc,smformula=sm_formula_ncc,
                       set="setno",event="d",nrisk="numrisk",
                       method=method.vec, predictorMatrix=pred.mat,
                       m=nimp,numit=n.it,rjlimit=1000,noisy=F)

  # obtain estimates and combine using Rubin’s Rules
  impobj <- imputationList(imp$impDatasets)
  models <- with(impobj, coxph(as.formula(formula_ncc)))
  coef = MIcombine(models)$coefficients
  se = sqrt(diag(MIcombine(models)$variance))
  res_rej_sup = c(coef,se)

  #================================
  # MI-SMC: (NAIVE) superset B ncc
  #================================
  cohort.ncc= readRDS(file=paste0(filepath,"cohort.super.ncc",j,".rds"))
  # predictor matrix which determines the imputation models for x1
  pred.mat=matrix(0,nrow=dim(cohort.ncc)[2],ncol=dim(cohort.ncc)[2])
  colnames(pred.mat)=names(cohort.ncc)
  rownames(pred.mat)=names(cohort.ncc)
  pred.mat["x",predictors_rs]=1

  # method of imputation for x1
  method.vec=rep("",dim(cohort.ncc)[2])
  method.vec[which(colnames(cohort.ncc)=="x")]="norm"

  # perform the imputation
  imp <- smcfcs(cohort.ncc, smtype="coxph", smformula=sm_formula_full,
                method=method.vec,predictorMatrix=pred.mat,m =nimp,
                numit = n.it, rjlimit = 10000,noisy=F)

  # obtain estimates from imputed data sets and combine using Rubin’s Rules
  impobj <- imputationList(imp$impDatasets)
  models <- with(impobj, coxph(as.formula(formula_ncc)))
  coef = MIcombine(models)$coefficients
  se = sqrt(diag(MIcombine(models)$variance))
  res_rej_sup_nai = c(coef,se)
  #--------------------------------
  #================================
  # Add results from simulation j
  res_mat[j,] = c(res_full,res_ncc,res_aprx,res_aprx_sup,res_aprx_sup_nai,
                  res_rej,res_rej_sup,res_rej_sup_nai)
}
#======== END FOR LOOP ==========

#================================
# Performance measurements
#================================
true_parameters = matrix(c(1,1,0.5),nrow=3)
```

```
par_idx = c()
se_idx = c()
for(k in seq(0,nmethods-1)){
  par_idx= c(par_idx,seq(1,npara)+2*npara*k)
  se_idx = c(se_idx,seq(1+npara,2*npara)+2*npara*k)
}
parameters = res_mat[,par_idx]
parameter_se = res_mat[,se_idx]
# Bias
para_mean = apply(parameters,2,mean)
mean_mat = matrix(para_mean,nrow=nsim,ncol=npara*nmethods,byrow=T)
bias = apply(parameters,2,mean)-rep(true_parameters,nmethods)
bias_mat = matrix(bias,nrow=npara,ncol=nmethods,byrow=F)
# Model SE
model_se = apply(parameter_se,2,mean)
model_se_mat = matrix(model_se,nrow=npara,ncol=nmethods,byrow=F)
# Empirical se
emp_se = sqrt((1/(nsim-1))*apply((parameters-mean_mat)^2,2,sum))
emp_se_mat = matrix(emp_se,nrow=npara,ncol=nmethods,byrow=F)
# MSE
mse = apply((parameters-matrix(true_parameters,ncol=nmethods*npara,
                               nrow=nsim,byrow=T))^2,2,mean)
mse_mat = matrix(mse,nrow=npara,ncol=nmethods,byrow=F)
# 95 percent coverage
lower = parameters -1.96*parameter_se
higher = parameters +1.96*parameter_se
in_int = matrix(true_parameters,ncol=nmethods*npara,nrow=nsim,byrow = T) >=
  lower & matrix(true_parameters,ncol=nmethods*npara,nrow=nsim,byrow = T) <=
  higher
cover_mat = matrix(apply(in_int,2,mean),nrow=npara,ncol=nmethods,byrow=F)

# relative efficience (compared to full cohort)
rel_eff= apply(matrix(parameter_se[,seq(1,npara)],nrow=nsim,
                      ncol=nmethods*npara)^2/parameter_se^2,2,mean)
rel_eff = matrix(rel_eff,nrow=npara,ncol=nmethods,byrow=F)

### table of results
tab_res = rbind(bias_mat,model_se_mat,emp_se_mat,rel_eff,mse_mat,cover_mat)
tab_res2 = cbind(c("Bias","","","ModelSE","","","EmpSE","","",
                   "RelEff","","","MSE","","","Cov","",""),
                 matrix(rep(c(" $\\beta_x$"," $\\beta_{z_1}$","
                              $\\beta_{z_2}$"),6)),
                 round(tab_res,3))
print(tab_res2)
# Monte Carlo SE of estimates:
mc_bias = sqrt((1/(nsim*(nsim-1)))*apply((parameters-matrix(true_parameters,
                                      ncol=nmethods*npara,
                                      nrow=nsim,byrow=T))^2,2,sum))
mc_bias_mat = matrix(mc_bias,nrow=npara,ncol=nmethods,byrow=F)
mc_emp_se = (1/sqrt(2*(nsim-1)))*emp_se_mat
mc_mse = sqrt(apply(((parameters-matrix(true_parameters,ncol=nmethods*npara,
                                    nrow=nsim,byrow=T))^2
                     -matrix(mse_mat,ncol=nmethods*npara,
                             nrow=nsim,byrow=T))^2,2,sum)/(nsim*(nsim-1)))
mc_cover = sqrt((cover_mat*(1-cover_mat))/nsim)
```

```
# Analyse case-cohort data
# Code based on https://github.com/ruthkeogh/MI-CC by Ruth H.Keogh
# packages ---------------------
library(survival)
library(mice)
library(smcfcs)
library(mitools)
# setup -----------------------
setup = "standard"
filepath=paste0("simulations/",setup,"/")
nimp=10
n.it = 100
npara=3
nsim=1000
nmethods = 8
res_mat = matrix(NA,nrow=nsim,ncol=2*npara*nmethods)
# interaction formulas
formula = "Surv(t,d)~x+z1+z2"
sm_formula = "Surv(t,d)~x+z1+z2"
sm_formula_caco = "Surv(entertime,t,d)~x+z1+z2"
predictors_aprx = c("z1","z2","d","chaz")
predictors_rs = c("z1","z2")
#===============================
# Run analyses
#===============================
set.seed(1001)
for(j in seq(1,nsim)){
  #===============================
  # cox analysis using full cohort data
  #===============================
  # load data set
  cohort = readRDS(file=paste0(filepath,"cohort",j,".rds"))
  model=coxph(as.formula(formula),data=cohort)
  res_full = c(model$coefficients,sqrt(diag(model$var)))
  # size of full cohort
  n = dim(cohort)[1]
  #===============================
  # traditional case-control analysis
  #===============================
  # load data set
  caco= readRDS(file=paste0(filepath,"caco",j,".rds"))
  # fit the model
  model=cch(as.formula(formula), data=caco,
            subcoh=~subco, id=~id, method="LinYing", cohort.size=n)
  res_caco = c(model$coefficients,sqrt(diag(model$var)))
  #===============================
  # MI-approx:  full-cohort approach
  #===============================
  cohort.caco= readRDS(file=paste0(filepath,"cohort.caco",j,".rds"))
  # Compute Nelson-Aalen estimate of the cumulative hazard
  cohort.caco$chaz=nelsonaalen(cohort.caco,t,d)
  # predictor matrix which determines the imputation models for x1
  pred.mat=matrix(0,nrow=dim(cohort.caco)[2],ncol=dim(cohort.caco)[2])
  colnames(pred.mat)=names(cohort.caco)
  rownames(pred.mat)=names(cohort.caco)
  pred.mat["x",predictors_aprx]=1
  # method of imputation for x1
  method.vec=rep("",dim(cohort.caco)[2])
  method.vec[which(colnames(cohort.caco)=="x")]="norm"
  # perform the imputation
  imp<-mice(cohort.caco, m = nimp, method = method.vec,
            predictorMatrix = pred.mat,
          maxit = n.it, diagnostics = FALSE, printFlag = F)
# Fit the analysis model in each imputed data set
models<-with(imp,coxph(as.formula(formula)))
# Combine estimates across the imputed data sets using Rubin’s Rules
summary_aprx = summary(pool(models))
res_aprx = c(summary_aprx[,"estimate"],summary_aprx[,"std.error"])
#================================
# MI-SMC: full-cohort approach
#================================
cohort.caco= readRDS(file=paste0(filepath,"cohort.caco",j,".rds"))
#predictor matrix which determines the imputation models for x1
pred.mat=matrix(0,nrow=dim(cohort.caco)[2],ncol=dim(cohort.caco)[2])
colnames(pred.mat)=names(cohort.caco)
rownames(pred.mat)=names(cohort.caco)
pred.mat["x",predictors_rs]=1
# method of imputation for x1
method.vec=rep("",dim(cohort.caco)[2])
method.vec[which(colnames(cohort.caco)=="x")]="norm"
# perform the imputation
imp <- smcfcs(cohort.caco, smtype="coxph", smformula=sm_formula,
              method=method.vec,predictorMatrix=pred.mat,m = nimp,
              numit = n.it, rjlimit = 10000,noisy=F)
# estimates from imputed data sets and combine using Rubin’s Rules
impobj <- imputationList(imp$impDatasets)
models <- with(impobj, coxph(as.formula(formula)))
coef = MIcombine(models)$coefficients
se = sqrt(diag(MIcombine(models)$variance))
res_rej = c(coef,se)
#================================
# MI-approx: superset A cch
#================================
cohort.caco=readRDS(file=paste0(filepath,"cohort.super.caco",j,".rds"))
# Compute Nelson-Aalen estimate of the cumulative hazard for full cohort
cohort$chaz=nelsonaalen(cohort,t,d)
# add cumulative hazard into ncc data
cohort.merge<-cohort[,c("id","chaz")]
cohort.caco<-merge(cohort.caco,cohort.merge,by.x="id")
# predictor matrix which determines the imputation models for x1
pred.mat=matrix(0,nrow=dim(cohort.caco)[2],ncol=dim(cohort.caco)[2])
colnames(pred.mat)=names(cohort.caco)
rownames(pred.mat)=names(cohort.caco)
pred.mat["x",predictors_aprx]=1
# method of imputation for x1
method.vec=rep("",dim(cohort.caco)[2])
method.vec[which(colnames(cohort.caco)=="x")]="norm"
#perform the imputation
imp<-mice(cohort.caco, m = nimp, method = method.vec,
          predictorMatrix = pred.mat,
          maxit = n.it, diagnostics = FALSE, printFlag = F)
# Fit the analysis model in each imputed data set
models <- vector("list", nimp)
for (k in 1:nimp){
  model=cch(as.formula(formula),data=complete(imp,k),
            subcoh=~subco.super, id=~id, method="LinYing", cohort.size=n)
  models[[k]] = model
}
# Combine estimates across the imputed data sets using Rubin’s Rules
res_aprx_sup=c(MIcombine(models)$coef,
               sqrt(diag(MIcombine(models)$variance)))
#================================
#MI-approx: (naive) superset B cch
#================================
  cohort.caco=readRDS(file=paste0(filepath,"cohort.super.caco",j,".rds"))
  # Compute Nelson-Aalen estimate of the cumulative hazard for superset
  cohort.caco$chaz = nelsonaalen(cohort.caco,t,d)
  # predictor matrix which determines the imputation models for x1
  pred.mat=matrix(0,nrow=dim(cohort.caco)[2],ncol=dim(cohort.caco)[2])
  colnames(pred.mat)=names(cohort.caco)
  rownames(pred.mat)=names(cohort.caco)
  pred.mat["x",predictors_aprx]=1
  # method of imputation for x1
  method.vec=rep("",dim(cohort.caco)[2])
  method.vec[which(colnames(cohort.caco)=="x")]="norm"
  #perform the imputation
  imp<-mice(cohort.caco, m = nimp, method = method.vec,
            predictorMatrix = pred.mat,
            maxit = n.it, diagnostics = FALSE, printFlag = F)
  # Fit the analysis model in each imputed data set
  models <- vector("list", nimp)
  for (k in 1:nimp){
    model=cch(as.formula(formula),data=complete(imp,k),subcoh=~subco.super,
              id=~id, method="LinYing", cohort.size=n)
    models[[k]] = model
  }
  # Combine estimates across imputed data sets using Rubin’s Rules
  res_aprx_sup_nai=c(MIcombine(models)$coef,
                     sqrt(diag(MIcombine(models)$variance)))
  #================================
  #MI-SMC: superset A cch
  #================================
  cohort.caco=readRDS(file=paste0(filepath,"cohort.super.caco",j,".rds"))
  # predictor matrix imputation models for x1 (not incl. outcomes)
  pred.mat=matrix(0,nrow=dim(cohort.caco)[2],ncol=dim(cohort.caco)[2])
  colnames(pred.mat)=names(cohort.caco)
  rownames(pred.mat)=names(cohort.caco)
  pred.mat["x",predictors_rs]=1
  # method of imputation for x1
  method.vec=rep("",dim(cohort.caco)[2])
  method.vec[which(colnames(cohort.caco)=="x")]="norm"
  # sampling fraction
  my.sampfrac = sum(cohort.caco$subco.super==1)/n
  # perform the imputation
  imp <- smcfcs.casecohort(cohort.caco,smformula=sm_formula_caco,
                           sampfrac=my.sampfrac,in.subco="subco.super",
                           method=method.vec,predictorMatrix=pred.mat,
                           m=nimp,numit=100,rjlimit=10000,noisy=FALSE)
  # estimates from imputed data sets and combine using Rubin’s Rules
  impobj <- imputationList(imp$impDatasets)
  models <- with(impobj,
                 coxph(as.formula(paste0(sm_formula_caco,"+cluster(id)"))))
  coef = MIcombine(models)$coefficients
  se = sqrt(diag(MIcombine(models)$variance))
  res_rej_sup = c(coef,se)
  #================================
  #MI-SMC: superset B cch
  #================================
  cohort.caco=readRDS(file=paste0(filepath,"cohort.super.caco",j,".rds"))
  #predictor matrix which determines the imputation models for x1
  pred.mat=matrix(0,nrow=dim(cohort.caco)[2],ncol=dim(cohort.caco)[2])
  colnames(pred.mat)=names(cohort.caco)
  rownames(pred.mat)=names(cohort.caco)
  pred.mat["x",predictors_rs]=1
  #method of imputation for x1
  method.vec=rep("",dim(cohort.caco)[2])
```

```
  method.vec[which(colnames(cohort.caco)=="x")]="norm"
  #perform the imputation
  imp <- smcfcs(cohort.caco, smtype="coxph", smformula=sm_formula,
                method=method.vec,predictorMatrix=pred.mat,m = nimp,
                numit = n.it, rjlimit = 10000,noisy=F)
  # Fit the analysis model in each imputed data set
  models <- vector("list", nimp)
  for (k in 1:nimp){
    model=cch(as.formula(formula),data=imp$impDatasets[[k]],
              subcoh=~subco.super,id=~id, method="LinYing",cohort.size=n)
    models[[k]] = model
  }
  # Combine estimates across the imputed data sets using Rubin’s Rules
  res_rej_sup_nai=c(MIcombine(models)$coef,
                    sqrt(diag(MIcombine(models)$variance)))
  #--------------------------------
  #================================
  # Add results from simulation j
  res_mat[j,] = c(res_full,res_caco,res_aprx,
                  res_aprx_sup,res_aprx_sup_nai,
                  res_rej,res_rej_sup,res_rej_sup_nai)
} #======== END FOR LOOP ==========
```

# Bibliography

Aalen, O., Borgan, Ø., and Gjessing, H. (2008). *Survival and Event History Analysis: a Process Point of View*. Springer Science & Business Media.

Andersen, P. K. and Gill, R. D. (1982). "Cox's regression model for counting processes: A large sample study". In: *The Annals of Statistics* vol. 10, no. 4, pp. 1100–1120.

Bartlett, J. W., Seaman, S. R., White, I. R., Carpenter, J. R., and Initiative*, A. D. N. (2015). "Multiple imputation of covariates by fully conditional specification: Accommodating the substantive model". In: *Statistical Methods in Medical Research* vol. 24, no. 4, pp. 462–487.

Borgan, Ø., Langholz, B., Samuelsen, S. O., Goldstein, L., and Pogoda, J. (2000). "Exposure stratified case-cohort designs". In: *Lifetime data analysis* vol. 6, no. 1, pp. 39–58.

Borgan, Ø. and Samuelsen, S. O. (2016). "Nested Case-Control and Case-Cohort Studies". In: *Handbook of Survival Analysis*. Ed. by Klein, J. P., Van Houwelingen, H. C., Ibrahim, J. G., and Scheike, T. H. CRC Press. Chap. 17, pp. 343–367.

Buuren, S. v. and Groothuis-Oudshoorn, K. (2010). "mice: Multivariate imputation by chained equations in R". In: *Journal of Statistical Software*, pp. 1–68.

Carpenter, J. and Kenward, M. (2012). *Multiple Imputation and its Application*. John Wiley & Sons.

Dispenzieri, A., Katzmann, J. A., Kyle, R. A., Larson, D. R., Therneau, T. M., Colby, C. L., Clark, R. J., Mead, G. P., Kumar, S., Melton III, L. J., et al. (2012). "Use of nonclonal serum immunoglobulin free light chains to predict overall survival in the general population". In: *Mayo Clinic Proceedings*. Vol. 87. 6. Elsevier, pp. 517–523.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis*. CRC press.

Goldstein, L. and Langholz, B. (1992). "Asymptotic theory for nested case-control sampling in the Cox regression model". In: *The Annals of Statistics*, pp. 1903–1928.

Kalbfleisch, J. and Lawless, J. (1988). "of Reliability in Field–Performance Studies". In: *Technometrics* vol. 30, no. 4, pp. 365–378.

Keogh, R. H. (2018). "Multiple Imputation for Sampled Cohort Data". In: *Handbook of Statistical Methods for Case-Control Studies*. Ed. by Borgan, Ø., Breslow, N., Chatterjee, N., Gail, M. H., Scott, A., and Wild, C. J. CRC Press. Chap. 20, pp. 373–390.

Keogh, R. H., Seaman, S. R., Bartlett, J. W., and Wood, A. M. (2018). “Multiple imputation of missing data in nested case-control and case-cohort studies”. In: *Biometrics* vol. 74, no. 4, pp. 1438–1449.

Keogh, R. H. and White, I. R. (2013). “Using full-cohort data in nested case–control and case–cohort studies by multiple imputation”. In: *Statistics in medicine* vol. 32, no. 23, pp. 4021–4043.

Langholz, B. and Borgan, Ø. (1995). “Counter-matching: a stratified nested case-control sampling method”. In: *Biometrika* vol. 82, no. 1, pp. 69–79.

— (1997). “Estimation of absolute risk from nested case-control data”. In: *Biometrics*, pp. 767–774.

Langholz, B. and Thomas, D. C. (1990). “Nested case-control and case-cohort methods of sampling from a cohort: a critical comparison”. In: *American journal of epidemiology* vol. 131, no. 1, pp. 169–176.

Lin, D. and Ying, Z. (1993). “Cox regression with incomplete covariate measurements”. In: *Journal of the American Statistical Association* vol. 88, no. 424, pp. 1341–1349.

Little, R. J. A. and Rubin, D. B. (2019). *Statistical Analysis with Missing Data*. John Wiley & Sons.

Meng, X.-L. (1994). “Multiple-imputation inferences with uncongenial sources of input”. In: *Statistical Science*, pp. 538–558.

Morris, T. P., White, I. R., and Crowther, M. J. (2019). “Using simulation studies to evaluate statistical methods”. In: *Statistics in medicine* vol. 38, no. 11, pp. 2074–2102.

Prentice, R. L. (1986). “A case-cohort design for epidemiologic cohort studies and disease prevention trials”. In: *Biometrika* vol. 73, no. 1, pp. 1–11.

Rubin, D. B. (1987). *Multiple Imputation for Nonresponse in Surveys*. NY Wiley.

Rubin, D. B. (1976). “Inference and missing data”. In: *Biometrika* vol. 63, no. 3, pp. 581–592.

Scheike, T. H. and Juul, A. (2004). “Maximum likelihood estimation for Cox’s regression model under nested case-control sampling”. In: *Biostatistics* vol. 5, no. 2, pp. 193–206.

Scheike, T. H. and Martinussen, T. (2004). “Maximum likelihood estimation for Cox’s regression model under case–cohort sampling”. In: *Scandinavian journal of statistics* vol. 31, no. 2, pp. 283–293.

Støer, N. C. and Samuelsen, S. O. (2013). “Inverse probability weighting in nested case-control studies with additional matching—a simulation study”. In: *Statistics in medicine* vol. 32, no. 30, pp. 5328–5339.

Thomas, D. C. (1977). “Addendum to: Methods of cohort analysis: Appraisal by application to asbestos mining. By FDK Liddell, JC McDonald and DC Thomas”. In: *Journal of the Royal Statistical Society, Series A* vol. 140, pp. 469–491.

Van Buuren, S. (2018). *Flexible Imputation of Missing Data*. Chapman and Hall/CRC.

White, I. R. and Royston, P. (2009). “Imputing missing covariate values for the Cox model”. In: *Statistics in Medicine* vol. 28, no. 15, pp. 1982–1998.