PROPERTIES OF ESTIMATORS FOR RELATIVE RISKS FROM NESTED CASE-CONTROL STUDIES WITH MULTIPLE OUTCOMES (COMPETING RISKS)

by

NATHALIE C. STØER

THESIS
for the degree of
MASTER OF SCIENCE

Modelling and Data Analysis

Statistics Division, Department of Mathematics
Faculty of Mathematics and Natural Sciences
University of Oslo

May 2010

Abstract

Nested case-control studies (NCC) reduce the cost of large cohort studies, but are statistically less efficient since all information is only available for cases and controls. In particular in a competing risk situation the traditional partial likelihood estimator for NCC can not handle controls sampled for cases of one disease as controls for another disease. This may be especially problematic if one outcome is common, but the other is rare. There has, however, been developed methods based on inverse probability weighting (IPW) that allow for reusing controls (and cases). Also maximum likelihood methods for NCC have been developed.

Furthermore, in addition to the information collected on cases and controls, some information is usually known for the entire cohort, like gender, age, etc. This information can be utilized to obtain more accurate estimates both for the IPW and MLE approaches.

This is a comparison of such methods. It is carried out both on simulated data and on data from the Norwegian Medical Birth Registry with death of cancer being the rare endpoint and death of all other causes being the common. In simulations with only one covariate IPW and MLE methods performed similarly. With two covariates where one covariate was known for the entire cohort methods that utilized this information gave efficiency improvements in particular for the fully observed covariate. If the two covariates were dependent we also found improvement for the covariate only known for cases and controls. Analysis on the Norwegian Medical Birth Registry data showed similar results than the simulation, but due to the high number of controls, at least for cancer endpoint, the improvements are not that pronounced.

Acknowledgements

For one and a half year now, I have been working on this thesis and it has been an incredible instructive period. During this process, there have been several people that have been of great help that I would like to thank. First and foremost I would like to thank my supervisor Professor Sven Ove Samuelsen for giving me an interesting theme for my thesis. He has shown a sincere interest, patiently answered my questions and given me useful feedback. I would also like to thank Professor Norman E. Breslow for taking his time to meet with me on his stay her in Oslo, and Professor Thomas H. Scheike for providing me with his R-code and answering my e-mails about it.

In addition I want to thank my fellow students at B802 for support and encouragement, it wouldn't have been the same without you. And at last my family and friends, who have supported me through the Master program. I have indeed learned a lot and I will bring it all with me into my life and career in the future.

Oslo, May 2010
Nathalie C. Støer

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In survival analysis a group of people, the cohort, is followed from start of study to experience of event or to end of study. During this time one observe the deaths in the cohort and collect information, covariates, from the individuals. For individuals who do not experience the event or disappear from the cohort from other reasons i.e. death from other causes, moving etc. we don't know the actual survival time, only that they lived longer than the observed time. These times are called censored survival times. Because of, that in practice, censoring is always at play, we need special methods for dealing with survival data. The most famous model is Cox-regression model [9]. This model assumes that the hazard rate for an individual takes the form

$$\alpha(t|x_i;\beta) = \alpha_0(t)\exp(\beta_1 x_{i1} + \cdots + \beta_p x_{ip}),$$

where $(x_{i1}, \ldots, x_{ip})$ is the covariate vector of individual i, $(\beta_1, \ldots, \beta_p)$ is the regression coefficients and α_0 the baseline hazard. The expression $\exp(\beta_1 x_{i1} + \cdots + \beta_p x_{ip})$ is a relative risk describing the connection between the covariates and the hazard rate.

Estimation in Cox-regression models is usually based on a partial likelihood, see (2.9) below, which require the knowledge of all covariates for all members of the cohort. This is because at each event time the covariates of the individual who experience the event is compared to the covariates of all the individuals who have not yet experienced it. To collect covariate information from all individuals in a cohort can both be time consuming and very expensive. Especially in epidemiologic studies of rare diseases this can be a problem because one need to follow up a large group of people in order to get enough deaths to obtain precise estimates. As an alternative one can instead use methods which is based on collecting information from the people who experienced the event and a subgroup of people who did not experience it. Two such designs are nested case-control (NCC) [27] and case-cohort(CC) [16], which are the two most common cost-efficient sampling schemes. In a NCC design one samples a small number of controls without replacement from the cohort at each event time. These controls are individuals who was still at risk when the event happened. In a CC design one samples a subcohort from the full cohort at the outset of the study, and those individuals are being used as controls at all event times.

Since most of the statistical information will be contained in the cases, when events are rare this type of studies can still give reliable results. But of course, with fewer observations the variance will increase and we would like to squeeze as much information as possible out of the controls.

The traditional NCC estimator only use the case and it's sampled controls at the event time, a more efficient way is to use the the cases and the controls whenever they are at risk. This is possible by using a weighted partial likelihood (WPL) in which the original time-matched controls sets together with all cases are pooled together, are used in the estimation. Using unit weights on both cases and controls will result in biased estimates since the cases always are included in the risk sets while the controls are included with a probability (much) smaller than one. The probability for a member of the cohort being included in a risk set increases with time since the number of chances of being sampled increases with time. It is therefore sensible to give the cases unit weights and weigh the remaining controls by the inverse of their probability of being sampled, this method is applicable both in NCC- and CC-designs. There are a number of ways to estimate the inclusion probabilities, Samuelsen [20] has proposed one estimator, Breslow et al. [4, 5] have suggested another method, mainly for the CC-design, that aims at minimizing the extra variance we get from the sampling. Other possibilities are for instance logistic regression or a logistic generalized additive model (GAM), where the difference between them is that GAM allow the covariates to be included as arbitrary smooth functions in stead of only linearly as is the case with logistic regression.

One situation where there is potentially much to gain by being able to use controls (and cases) over again is in a competing risk situation. Assume that we have a cohort with two competing risks where controls have been sampled for both endpoints. This means that covariate information have been obtained for cases of both types and their sampled controls. If the same covariates are of interest, the same covariates have been obtained for both endpoints, then it seems appealing to be able to use both cases and controls for one endpoint as additional controls when analyzing another endpoint. This is of course especially useful if one endpoint is common, but another is rare, then the number of controls for the rare endpoint can increase drastically, which results in (much) lower variability. Above a weighted partial likelihood could be used when the risk sets together with all the controls are pooled together, this method is also applicable in a competing risk setting where cases and controls from both endpoints now are pooled togehter.

Saarela et al. [19] describe another way of reusing controls. They propose a full likelihood for both the cases and the controls, by parameterizing unknown quantities, like the baseline, the likelihood can be directly optimized with any numerical optimizer. Scheike and Juul [23] have also proposed a full likelihood, but they approach the maximization in a different way, by using the EM-algorithm. Both of these likelihoods can also handle multiple outcomes.

In this thesis we are going to investigate the efficiency improvements by using all controls and the cases of one outcome, when analysing another outcome. We will try to find criterion for when the full likelihood estimator of Saarela et al. is better than the weighted partial likelihood with inverse probability weighting. Further we will try to find out if the approach proposed by Breslow et al. can be useful also in an NCC situation with competing risks. We will evaluate the properties of the variance estimators both by simulation studies and studies of concrete data.

The thesis is outlined as follows:

- In chapter 2 we repeat the basics of survival analysis, look at the competing risk setup and review the NCC-design, CC-design and the weighted partial likelihood. We also go through the likelihood of Saarela et al. and Scheike's likelihood. At last we also go through the calibration method of Breslow et al. and try to generalize it to NCC with multiple endpoints.

- Chapter 3 is about simulations. We first discuss different censoring schemes, then we present and comment on the simulation experiments. The chapter is ended with a brief discussion of the problems that have arisen in connection with the simulations.

- In chapter 4 we try out the different methods on a real data example. First we present the data, then we explain what we have done and how we have done it and at last we present and comment on the results.

- The last chapter is summing up what we have done and what we have found out. We make a conclusion about when it is worth using more complicated models. We also comment on what else that could have been done, but due to the time limit, haven't been done.

Chapter 2

Survival analysis

2.1 Basic concepts

In survival analysis one usually observe the time T to a specific event of a group of people called the cohort. Unlike most other study designs, the observations in a survival analysis are observed over time, we have to wait for the event to actually happen. Because of that, usually T can not be observed for all members of the cohort. Either the study is ended before all individuals have experienced the event or some individuals disappear from the study because of death from other causes, moving etc. This means that we have a mixture of complete and incomplete observations. These incomplete observations are called censored survival times and are denoted C, while the actual survival times are denoted T. Even though one usually can't observe T for all members of the cohort, what we can observe is $\tilde{T} = \min(C, T)$ together with an indicator E, indicating whether the time is an actual survival time or a censored survival time.

Sometimes the individuals may not be observed from start of study, only some time after. The individuals who experience the event before this time is then not included in the study and what we observe are $(V, \tilde{T}, E)$, which is entry time, exit time and an indicator that indicates whether or not the exit time is censored.

2.1.1 Survival function and hazard rate

There are two basic concepts that all survival analysis rely on, this is the survival function and the hazard rate. The survival function is defined as

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

where T is the survival time. This is the probability that the event has not yet happened at time t. Note that

$$S(t) = 1 - F(t)$$

where $F(t) = P(T \leq t)$ is the cumulative distribution function of T

The hazard rate is defined as

$$\alpha(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} P(t \leq T < t + \Delta t | T \geq t). \tag{2.2}$$

$\alpha(t)\Delta t$ can be thought of as the "probability" of experiencing the event in the next small time interval $[t, t + \Delta t)$, given that the event has not yet happened. The interpretation of the hazard rate is the instantaneous risk of experiencing the event.

There are some basic mathematical connections between the survival function and the cumulative hazard rate. Define

$$A(t) = \int_0^t \alpha(s)ds \tag{2.3}$$

to be the cumulative hazard rate. Then it follows from (2.2) and (2.3) that

$$\alpha(t) = A'(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} \frac{S(t) - S(t + \Delta t)}{S(t)} = -\frac{S'(t)}{S(t)}. \tag{2.4}$$

Another basic relation between the survival function and the hazard rate is

$$S(t) = \exp\left\{-\int_0^t \alpha(s)ds\right\} = \exp(-A(t)) \tag{2.5}$$

This can be seen by integrating on both sides of (2.4) and using the fact that $S(0) = 1$, we get

$$-\log(S(t)) = \int_0^t \alpha(s)ds$$

and the result follows from this. From (2.4) we also get the relation

$$\alpha(t) = \frac{f(t)}{S(t)} \tag{2.6}$$

since $S(t) = 1 - F(t)$, and if we differentiate on both sides we get that $S'(t) = -f(t)$ and the relation follows from this.

2.1.2 Likelihood for survival data

From (2.6) we know that the distribution of the survival times is

$$f(t|x; \psi, \beta) = \alpha(t|x; \psi, \beta)S(t|x; \psi, \beta)$$

here x is covariates, ψ are parameters describing the baseline while β are regression parameters. If we knew the actual survival time of every individual the likelihood would simply be

$$\prod_{i=1}^{n} \alpha(t_i|x_i; \psi, \beta)S(t_i|x_i; \psi, \beta),$$

but we don't. We only know $\tilde{T}_i = \min(T_i, C_i)$ together with an indicator E_i indicating whether $\tilde{T}_i$ it's a survival time or a censored survival time. If t_i is a censored survival time we don't know the hazard rate only that the individual lived longer than t_i. This means that the only thing we know is the survival function up to t_i. Thereby the full likelihood for survival data looks like

$$L(\psi, \beta) = \prod_{i=1}^{n} [\alpha(t_i|x_i; \psi, \beta)]^{1_{E_i=1}} S(t_i|x_i; \psi, \beta), \tag{2.7}$$

where $1_{E_i=1}$ is an indicator function indicating whether or not $E_i = 1$. This is a bit cumbersome notation since E_i itself is an indicator, but when we move on to multiple outcomes it is needed because then E_i can take more values than 0 and 1. Therefor in order to be consistent with the notation we use it here as well.

2.2 Proportional hazards models

Usually, the main purpose of a survival analysis is to determine which covariates are important for the survival time, and how they influence it, this calls for regression models. The most common models are the relative risk regression models

$$\alpha(t|x_i; \beta) = \alpha_0(t) r(\beta, x_i)$$

here $r(\cdot)$ is a relative risk function connecting the covariates to the hazard rate. The most famous of these models is Cox's proportional hazard model which is given by

$$\alpha(t|x_i; \beta) = \alpha_0(t) \exp(\beta_1 x_{i1} + \cdots + \beta_p x_{ip}), \tag{2.8}$$

Here $(\beta_1, \ldots, \beta_p)$ are the regression parameters, $(x_{i1}, \ldots, x_{ip})$ is the covariate vector belonging to individual i and α_0 is the baseline hazard, the hazard when all covariates are equal to zero. Even though Cox's proportional hazard model is the most famous, there are other possibilities for $r(\cdot)$, for instance the linear relative risk function $r(\beta, x_i) = \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$ or the excess relative risk model $r(\beta, x_i) = \prod_{j=1}^{p} \{1 + \beta_j x_{ij}\}$.

Cox's proportional hazard model is a semi-parametric model because the risk function $\exp(\beta_1 x_{i1} + \cdots + \beta_p x_{ip})$ is a parametric expression, while the baseline hazard is unspecified, and can basically take any form. Because of that, Cox [7] suggested using the product over the conditionally probabilities of given that it did happen an event at t_i, it happen with individual i,

$$L(\beta_1, \ldots \beta_p) = \prod_{E_i=1} \frac{\exp(\beta_1 x_{i1} + \ldots + \beta_p x_{ip})}{\sum_{j \in \mathcal{R}_i} \exp(\beta_1 x_{j1} + \ldots + \beta_p x_{jp})}, \tag{2.9}$$

to estimate the regression parameters. Here, $\mathcal{R}_i$ is the risk set at t_i, the collection of all individuals still at risk at time t_i. E_i is an indicator, indicating whether t_i is an event time or only a censored survival time, therefor the product is over all event times.

A bit more formal argument for (2.9) is as follows; if α_0 was known we, know from (2.7) that the likelihood would look like

$$\begin{aligned}
L' = & \prod_i \{\alpha_0(t_i)\exp(\beta x_i)\}^{1_{E_i=1}} \exp\left\{-\int_0^{t_i} \alpha_0(s)\exp(\beta x_i)ds\right\} \\
= & \prod_i \left[\alpha_0(t_i)^{1_{E_i=1}} \sum_{j\in\mathcal{R}_i}\{\exp(\beta x_j)\}\exp\left\{-\int_0^{t_i}\alpha_0(s)\exp(\beta x_i)ds\right\}\right] \times \\
& \prod_i \frac{\exp(\beta x_i)^{1_{E_i=1}}}{\sum_{j\in\mathcal{R}_i}\exp(\beta x_j)} \\
= & \prod_i \left[\alpha_0(t_i)^{1_{E_i=1}} \sum_{j\in\mathcal{R}_i}\{\exp(\beta x_j)\}\exp\left\{-\int_0^{t_i}\alpha_0(s)\exp(\beta x_i)ds\right\}\right] \times \\
& \prod_{E_i=1} \frac{\exp(\beta x_i)}{\sum_{j\in\mathcal{R}_i}\exp(\beta x_j)}
\end{aligned} \tag{2.10}$$

The further reasoning is that the expression in brackets doesn't provide much information about β when α_0 is unknown and Cox [8] suggested that if a full likelihood can be written as

$$\prod_{i=1}^{m} f_{X_i|X^{(i-1)}S^{(i-1)}}(X_i|X^{(i-1)}S^{(i-1)};\theta)\prod_{i=1}^{m} f_{S_i|X^{(i)}S^{(i-1)}}(S_i|X^{(i)}S^{(i-1)};\theta)$$

then the second product is a partial likelihood. Here the $X^{(j)} = (X_1,\dots,X_j)$, $S^{(j)} = (S_1,\dots,S_j)$ and $f_{X_i|X^{(i-1)}S^{(i-1)}}(X_i|X^{(i-1)}S^{(i-1)};\theta)$ is the conditional distribution of X_i given $X^{(i-1)}$ and $S^{(i-1)}$ and similar for $f_{S_i|X^{(i)}S^{(i-1)}}$. Generally X_i and S_i are random variables, or even random vectors. If we let X_i specify the censoring in $[t_{i-1}, t_i)$ plus the information that the event happened for the first time at t_i and let S_i specify the particular individual that experience the event at t_i, then the likelihood of the sequence $(X_1, S_1, X_2, S_2, \dots, X_m, S_m)$ is exactly (2.10) and the second product is the partial likelihood.

The partial likelihood is not a full likelihood because we completely disregard the baseline, but never the less it shares the same properties as a full likelihood. This means that

$$E[U(\theta)] = 0$$
and
$$\text{Var}(U(\theta)) = E(I(\theta))$$

here $U(\theta) = \frac{\partial l(\theta)}{\partial \theta}$ is the score statistic and $I(\theta) = -\frac{\partial U(\theta)}{\partial \theta}$ is the observed information matrix.

The Cox standard proportional hazard model rests on two assumptions; log-linear effects of numerical covariates $\log(\alpha(t|x;\beta)) = \alpha_0(t) + x\beta$ and proportional hazards. The assumption of log-linear effects imply that one unit increase in a numeric covariate should have the same effect on log scale independent of the value of that covariate and of all other covariates. The proportional hazards assumption means that the hazard ratio is independent of time and if x_1 and x_2 are equal except for the i-th component the hazard ratio is $\exp(\beta)$.

The Cox-likelihood is based on knowing all covariates for the entire cohort. To collect covariate information for all individuals in a cohort can both be time consuming and very expensive. Especially in epidemiologic studies of rare diseases this can be a problem because one need to follow up a large group of people in order to get enough death's to obtain precise estimates. As an alternative one can instead use methods which is based on collecting information from the individuals who experienced the event and a subgroup of individuals who did not experience it. Two such designs are nested case-control (NCC) and case-cohort(CC) which are two major cost-efficient sampling schemes.

2.2.1 Parametric survival models

The Cox proportional hazards model $\alpha(t|x_i;\beta,\psi) = \alpha_0(t|\psi)\exp(\beta x_i)$ is a semi-parametric regression model, because of the unspecified baseline hazard. But it is of course possible to have a parametric specification of baseline as well, then the model would be fully parametric and the likelihood could be optimized directly. From (2.7) we know that the likelihood would be

$$L(\psi,\beta) = \prod_i \{\alpha_0(t_i|\psi)\exp(\beta x_i)\}^{1_{E_i=1}} \exp\left\{-\int_0^{t_i} \alpha_0(s|\psi)\exp(\beta x_i)ds\right\} \tag{2.11}$$

where ψ is the parameters describing the baseline hazard and β is the regression paramters.

There are different ways of specifying the baseline, but since there is a close connection between the distribution of the survival times and the hazard rate we can get some reasonable expressions for baseline when the survival times follows the most common distributions, which is the exponential and Weibull. We know that the hazard of a proportional hazard model looks like $\alpha(t|x_i;\beta,\psi) = \alpha_0(t|\psi)\exp(\beta x_i)$, so by assuming some structure on $\alpha_0(t|\psi)$ one can get hold of the distribution of T. A baseline on the form $\lambda\nu t^{\nu-1}$ result in a Weibull distribution for the survival times. A special case is when $\nu = 1$, which result in a constant baseline and an exponential distribution for the survival times.

Constant baseline, exponential survival times

If one assumes a constant baseline λ, then the hazard can be written as

$$\alpha(t|x_i;\beta,\lambda) = \lambda\exp(\beta x_i)$$

and the cumulative hazard as

$$A(t|x_i;\beta,\lambda) = \int_0^t \lambda\exp(\beta x_i)ds = \lambda\exp(\beta x_i)t$$

the survival function is then

$$S(t|x_i;\beta,\lambda) = \exp(-A(t|x_i;\beta,\lambda)) = \exp(-\lambda\exp(\beta x_i)t).$$

And then since we know that $f(t|x_i;\beta,\lambda) = \alpha(t|x_i;\beta,\lambda)S(t|x_i;\beta,\lambda)$

$$f(t|x_i;\beta,\lambda) = \lambda\exp(\beta x_i)\exp(-\lambda\exp(\beta x_i)t)$$

As we see, this is an exponential distribution with parameter $\lambda\exp(\beta x_i)$

Weibull proportional hazard model

The exponential distribution assumes that the baseline hazard is constant over time, this is often not the case in practice. Then, for instance a Weibull distribution can be used. The hazard and cumulative hazard is given by

$$\begin{aligned}\alpha(t|x_i;\beta,\psi) &= \lambda^\nu\nu t^{\nu-1}\exp(\beta x_i)\\ A(t|x_i;\beta,\psi) &= (\lambda t)^\nu\exp(\beta x_i),\end{aligned}$$

then the survival function and density can be written as

$$\begin{aligned}S(t|x_i;\beta,\psi) &= \exp(-(\lambda t)^\nu\exp(\beta x_i))\\ f(t|x_i;\beta,\psi) &= \lambda^\nu\exp(\beta x_i)\nu t^{\nu-1}\exp(-(\lambda t)^\nu\exp(\beta x_i)).\end{aligned}$$

where $\psi = (\lambda,\nu)$. This is Weibull$(\lambda\exp(\beta x_i),\nu)$ distributed. The scale parameter $\lambda\exp(\beta x_i)$ vary with coefficients and covariates, while the shape parameter ν is fixed. For $\nu = 1$ we get back to the exponential distribution, while for $\nu = 2$ we get a linear increasing baseline $2\lambda t$. Generally, $\nu > 1$ results in an increasing baseline, while $\nu < 1$ gives us a decreasing baseline.

Piecewise constant baseline

Another possibility for baseline hazard is piecewise constant

$$\sum_{m=1}^{M} \psi_m I_m(t)$$

where $I_m(t)$ is the indicator function for the m-th interval and $\psi = (\psi_1, \ldots, \psi_M)$ where ψ_m is the constant basline value for the m-th interval. This result in a hazard on the form

$$\alpha(t|x_i; \beta, \psi) = \sum_{m=1}^{M} \psi_m I_m(t) \exp\{\beta x_i\}. \tag{2.12}$$

The piecewise constant baseline doesn't result in a nice distribution for the survival times, but it is nice anyhow because the likelihood with this baseline is proportional to a Poisson likelihood. Therefor the parameter estimates are easily obtained with software for Poisson regression, for instance the `glm` package in **R** [17].

2.3 Competing risks

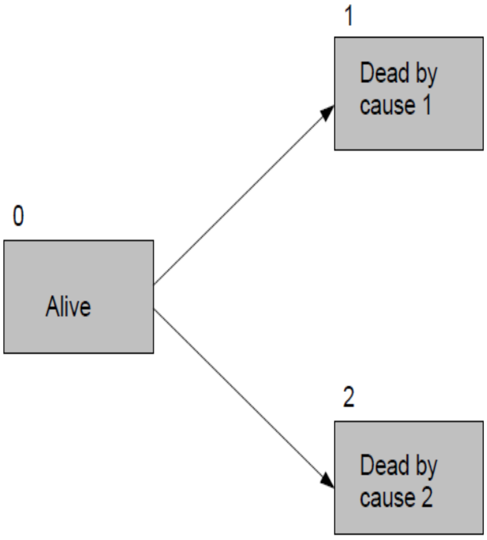

Figure 2.1: Model for K=2 competing causes of death

If there are more than one event that can occur, for instance death from different kind of cancer types and we are interested in the cause-specific mortality, then we have a competing risk situation, we have two or more causes of death that are "competing". Each individual can be in $1 + K$ different states, where K is the number of causes of death. The first state correspond to being alive while the K last states correspond to have died by the k-th cause. The competing risks can then be modeled by a Markov Chain, with one transient state corresponding to being alive and K absorbing states corresponding to death from one of the K causes.

The concept of hazard rate may also be generalized to competing risks, where α_k denotes the instantaneous risk of dying from cause k.

2.3.1 Estimation

The Cox-likelihood for competing risks can be outlined similar to the usual Cox-likelihood, if α_k was known for all inidividuals the likelihood would be

$$L(\beta_1', \ldots, \beta_K') = \prod_i \Bigg[\prod_{k=1}^{K} \{\alpha_{0k}(t_i) \exp(\beta_k' x_i)\}^{1_{E_i=k}} \\ \exp\left\{ -\sum_{k=1}^{K} \int_0^{t_i} \alpha_{0k}(s) \exp(\beta_k' x_i) ds \right\} \Bigg]$$

where β_k' is the regression parameters corresponding to the k-th endpoint. $1_{E_i=k}$ is again an indicator function that is one if $E_i = k$ and 0 otherwise. By going through the same argument as we did for the usual Cox-likelihood the partial likelihood can be written as

$$\prod_{k=1}^{K} \prod_{E_i=k} \frac{\exp(\beta_k' x_i)}{\sum_{j \in \mathcal{R}_i} \exp(\beta_k' x_j)} = \prod_{k=1}^{K} L_k(\beta_k')$$

where each L_k is itself a Cox-likelihood, the notation $\prod_{E_i=k}$ means the product over all i where $E_i = k$, this means the product over all individuals that experienced endpoint k . We see that when we estimate β_k' all products except the k-th are constants, therefor the information matrix will be a block diagonal matrix and one Cox-regression per endpoint can be done.

With a parametric proportional hazard model on the other hand, the likelihood would look like

$$L(\psi_1', \ldots, \psi_K', \beta_1', \ldots, \beta_K') = \prod_i \Bigg[\prod_{k=1}^{K} \{\alpha_{0k}(t_i|\psi_k') \exp(\beta_k' x_i)\}^{1_{E_i=k}} \\ \exp\left\{ -\sum_{k=1}^{K} \int_0^{t_i} \alpha_{0k}(s|\psi_k') \exp(\beta_k' x_i) ds \right\} \Bigg]$$

where ψ_k' is a vector of parameters describing baseline. Since it is reasonable to assume that there may be a different baseline connected to each event type, we have $(\psi_1' \ldots \psi_K')$. We also see that if $K = 1$ then the likelihood reduces to (2.11), this likelihood is in fact a direct generalization of (2.11).

2.4 Nested case-control design

As mentioned above, nested case-control together with case-cohort sampling are the two most common cost-efficient sampling schemes. In a NCC design one samples $m-1$ controls without replacement from the $n(t)-1$ non-failing individuals from the risk set $\mathcal{R}(t)$, at each event time t, and compare the failing individual with these controls. Usually one sample between 1 and 10 controls per case.

The sampling at each event time is done independently, which means that a member of the cohort can serve as a control for more than one case, and a control can later end up as a case.

Both the NCC design and the CC design has later been modified to allow for stratified sampling [3, 14]. This can be useful if there exists a surrogate measure for the covariate of main interest for all members of the cohort. This measure is then used to classify the individuals into sampling strata.

2.4.1 Estimation in a NCC-design

The $m-1$ controls together with the failing individuals i is denoted $\tilde{\mathcal{R}}_i$ and is the sampled risk set at time t_i. In order to estimate β one maximize a partial likelihood similar to the Cox-likelihood [27]

$$L(\beta_1,\ldots,\beta_p) = \prod_{E_i=1} \frac{\exp(\beta_1 x_{i1} + \ldots + \beta_p x_{ip})}{\sum_{j\in\tilde{\mathcal{R}}_i} \exp(\beta_1 x_{j1} + \ldots + \beta_p x_{jp})}, \tag{2.13}$$

the only difference is that the summation in the denominator is only over the sampled risk set $\tilde{\mathcal{R}}_i$ and not over the entire risk set, actually the Cox-likelihood is a special case of this likelihood where the entire cohort is sampled with probability 1.

(2.13) can be seen as a likelihood for a stratified Cox-regression, by treating the label for the sampled risk sets as a stratification variable. This means that standard software in for instance **R** can be used for estimation. Inference can be based on usual large sample theory for likelihoods, which means that the estimators is approximately normally distributed and their variance can be found in the same way as for usual maximum likelihood estimators, namely as the inverse of the information matrix.

2.5 Case-cohort design

Case-cohort design is another important sampling design. In a CC design one samples a subcohort of size $m-1$ from the full cohort at the outset of the study. In contrast to the NCC design, the sampled subcohort is used as a comparison at all event times.

2.5.1 Estimation in a CC-design

There have been proposed different methods for estimating the regression coefficients. Prentice [16] suggested

$$L(\beta_1, \ldots \beta_p) = \prod_{E_i=1} \frac{\exp(\beta_1 x_{i1} + \ldots + \beta_p x_{ip})}{\sum_{j \in \mathcal{S}_i} \exp(\beta_1 x_{j1} + \ldots + \beta_p x_{jp})}, \tag{2.14}$$

where $\mathcal{S}_i$ is the sampled subcohort together with the case at t_i. This likelihood is very similar to both the likelihood for a full cohort analysis and the NCC likelihood. But there is one important difference, the subcohort is used over and over again therefore this is not a partial likelihood. But anyhow, it can be showed that the estimator is approximately normally distributed. Since the likelihood isn't a partial likelihood the estimation of standard errors become more complicated. There has however been proposed different methods for estimating $\text{Var}(\hat{\beta})$. Self and Prentice [24] proposed an asymptotically consistent estimator, another is Barlow's robust variance estimator [2]. By 1. order Taylor expansion

$$\text{Var}(\hat{\beta}) = I^{-1}(\hat{\beta})\text{Var}(U(\hat{\beta}))I^{-1}(\hat{\beta})$$

where $I^{-1}(\hat{\beta})$ is the inverse of the information matrix. Let $U_i(t)$ be the overall score of individual i at time t, Barlow suggested using $\hat{V}(\hat{\beta}) = 1/n \sum_i \hat{U}_i(t)\hat{U}_i(t)^T$ as a variance estimate of the overall score and this result in the sandwich estimator

$$\text{Var}(\hat{\beta}) = I(\hat{\beta})^{-1}\hat{V}(\hat{\beta})I(\hat{\beta})^{-1} = \frac{1}{n}\sum_i \Delta_i\hat{\beta}(\Delta_i\hat{\beta})^{\text{T}}$$

where $\Delta_i\hat{\beta} = \hat{\beta} - \hat{\beta}_{(i)} = I^{-1}(\hat{\beta})\hat{U}_i(t)^1$, is the change in $\hat{\beta}$ when the i-th observation is deleted.

In the previous likelihood the cases was only used at the time they failed, but by using a weighted partial likelihood [12]

$$L(\beta_1, \ldots, \beta_p) = \prod_{E_i=1} \frac{\exp(\beta_1 x_{i1} + \ldots + \beta_p x_{ip})}{\sum_{j \in \tilde{\mathcal{S}}_i} \exp(\beta_1 x_{j1} + \ldots + \beta_p x_{jp}) w_j} \tag{2.15}$$

they can be included whenever they are at risk. Here $\tilde{\mathcal{S}}_i$ is the subcohort together with all the cases still at risk at t_i. The weights w_j are 1 for all cases and $1/p_j$, where p_j is inclusion probability for the controls. For different methods of estimating the inclusion probability see next section.

The variance estimate can also under (2.15) be corrected by using robust variances.

[1] $\hat{\beta} - \hat{\beta}_{(i)} = I^{-1}(\hat{\beta})\hat{U}_i(t)$ by a 1.order Taylor

2.6 Weighted partial likelihoods for NCC design

The traditional NCC design is based on comparing cases with their sampled controls. This is not the most efficient way of using the information in the controls since they are only used once. Another way of doing it is by using a weighted partial likelihood on the form of (2.15) [20]. But for NCC design, $\tilde{S}$ isn't sampled at the outset of the study, but rather the collection of all sampled risk sets at every event time. From now on this collection of sampled risk sets will also be called subcohort even though it is not a random sample from the cohort.

Using unit weights for both cases and controls will result in biased estimates because the cases are included in the subcohort with probability 1, whereas the controls are sampled from the full cohort and is included in the subcohort with a probability (much) smaller than one. One usually chooses unit weights for the cases and the inverse of the probability of being sampled for the controls. Samuelsen's proposal for the p_j's is

$$p_j = \begin{cases} 1 & \text{cases} \\ 1 - \prod_{t_i < t_j} \left\{ 1 - \frac{m-1}{n(t_i)-1} \right\} & \text{controls} \end{cases}$$

this follows because the probability of being sampled at time t_i is $\frac{m-1}{n(t_i)-1}$, we also see that this is 1 - a "Kaplan-Meier like" estimator.

The estimation is straight forward by maximizing the weighted partial likelihood, but since the controls enter the likelihood at all event times whenever they are at risk, the estimation of the variance is not so straightforward. One possibility is using the robust variance estimator [2] based on influence terms described above. Samuelsen et al. [22] showed that this estimator can be somewhat too conservative and have proposed another possibility [20] which is less conservative.

There are other possibilities for estimating the inclusion probability than the one proposed by Samuelsen. One possibility is by using logistic GAM. Let V_i be the indicator that individual i is sampled to the risk set. Then we may model

$$\mathbf{E}(V_i|\tilde{T}_i) = \frac{\exp(\alpha + f(\tilde{T}_i))}{1 + \exp(\alpha + f(\tilde{T}_i))}$$

which means that we regress an indicator of being included in the subcohort, V_i on the censored survival times and use the fitted values from the regression as inclusion probabilities. Here $\tilde{T}$ is either the actual survival time or the censored survival time and $f(\tilde{T})$ is some smooth function of $\tilde{T}$.

It is also possible to use ordinary logistic regression, where for instance $f(\tilde{T}) = \eta\tilde{T}$, which will be a special case of logistic GAM.

A fourth possibility is a method, called local averaging, first proposed by Chen [6] for generalized case-cohort designs, which is a class of sampling designs that includes NCC and CC. This method involves choosing a partition for the time axis and calculate weights for censored individuals with exit times in each interval. Let $0 = t_0 < t_1 < \ldots < t_k = t$ be the partition, where t is the upper time limit for the study, and let $\tilde{\mathcal{S}}$ denote the collection of all sampled controls, then

$$w_{t_{j-1},t_j} = \frac{\sum_{i=1}^{n} I(t_{j-1} < \tilde{T}_i \leq t_j, E_i = 0)}{\sum_{i=1}^{n} I(t_{j-1} < \tilde{T}_i \leq t_j, E_i = 0, i \in \tilde{\mathcal{S}})},$$

where $i \in \tilde{\mathcal{S}}$ means that individual i has been selected as a control and $E_i = 0$ means that subject i isn't a case. Then the numerator counts the number of people censored in $(t_{j-1}, t_j]$ and the denominator counts how many of them who where sampled as controls. Individual i is then assigned weight w_{t_{j-1},t_j} if censored in $(t_{j-1}, t_j]$ and 1 if the individual is a case. This method can also handle sampling of cases by making separate partitions of the time axis for cases and controls. The number of intervals one should choose is somewhat arbitrary, there are problems connected to both choosing too many, and too few. If the time intervals are too narrow, the number of people censored in each interval will be small, and then it might happen that nobody censored in this interval is chosen as a control, which results in a non-defined weight. On the other hand if the intervals are too wide, the estimated weights wouldn't follow the "true" weights in a good way. I have chosen to use ten intervals in the simulation experiment in chapter 3, and from Figure 2.2, which shows an example of how the inclusion probabilities could look like with a cohort of size 1000 and about 10% cases, this seems like a pretty reasonable choice.

2.6.1 Multiple outcomes and NCC

The competing risk (multiple outcomes) setup can be very useful in connection with NCC because sometimes you can find yourself in a situation where one analysis has been done with one outcome in mind and then maybe later one wants to investigate a different outcome in the same cohort. If it is natural to use the same covariates, then it seems appealing to be able to use the information already gathered for the cases and the controls in the first analysis, in the second analysis. This is something that is not possible with the traditional way of analyzing NCC data since one only compare the cases with the sampled controls for that particular case. But if one chooses to use a weighted partial likelihood this can be made possible. One still uses a likelihood on the form (2.15), but now the risk set $\tilde{\mathcal{S}}'$ includes the cases and the sampled controls for the second outcome, together with the cases and the controls for the first outcome. This means that the hole control set at t_i consists of the controls for both endpoints together with the cases of the first endpoint who are still at risk at t_i. The weights that goes into

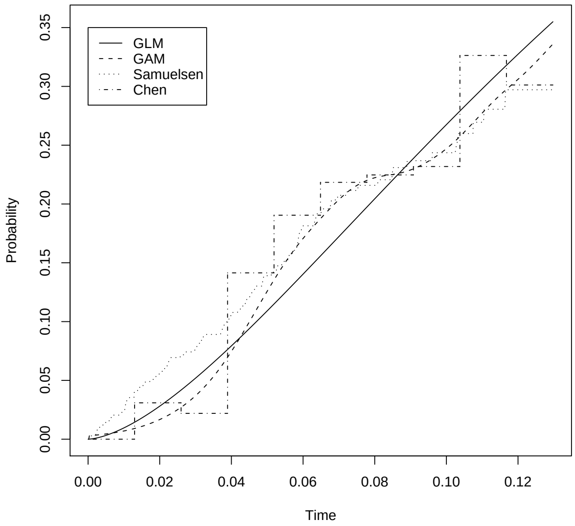

Figure 2.2: Inclusion probabilities

the likelihood are 1 for cases from both outcomes and the inverse of the probability of being sampled for all the controls. The sampling probability can be estimated with the same methods as above.

If both outcomes are fairly common or if we have a large m there is not so much to gain by utilizing the information in the first analysis when doing the second analysis. But if the second outcome is rather rare, or we have a small m, the number of individuals in $\tilde{\mathcal{S}}_i'$ at t_i will increase drastically when using the cases and the controls from the first analysis. For example let's say we have a cohort of size 1000, where about 10% dies from the first disease and only 3% dies from the second disease, and we only sample one control per case. Then for the second disease we have 30 controls together with 30 cases at start of study, but if we also utilize the information in the first analysis, we still have 30 cases, but now we have $30 + 200$ controls to compare with, which on average gives us almost 8 controls per case.

2.7 A full likelihood for NCC data

As noted earlier, one would like to utilize the information in the subcohort better than what can be done with the traditional NCC analysis, and one would also like to use information gathered in one analysis, in another analysis done in the same cohort. But not just that, often there are some covariates that are obtained for the entire cohort, this can be easy obtainable information like for instance age, gender etc. Saarela et al. [19] have proposed a full likelihood for partially observed covariate data which is applicable when there are K different outcomes and it also utilize the information known outside the subcohort.

2.7.1 Notation

We will first introduce some useful notation. Let $\mathcal{C} = \{1, \dots, n\}$ denote the cohort consisting of n subjects who are followed up for for the incidence of K different diseases. We account for the possibility that some covariates are collected for all individuals in the cohort $i \in \mathcal{C}$, this can be easy collectable information such as age, gender etc. those covariates are denoted x_i. We assume a competing risk situation, therefor all individuals enters the cohort healthy and is followed up until the first event of interest or right censoring because of death of other causes, loss of follow-up or end of study. Let $\tilde{T}_i$ be the observed time for individual i, this time is either the actual survival time or the censored survival time depending on whether or not the individual experienced the event. Further, let E_i be an indicator, indicating which event individual i experienced. E_i will take values in $0, \dots, K$, where 0 is referring to censoring at $\tilde{T}_i$. An individual is a case of type k if the individual experience the event k first and a group of such people is denoted $\mathcal{E}_k = \{i \in \mathcal{C} : E_i = k\}$.

Because of the study design, some or all covariates are only collected for the cases and their sampled controls, those covariates are denoted Z_i. Let O_i be an indicator that Z_i is collected for individual i, meaning that $O_i = 1$ if individual i is a member of the subcohort or a case, and $O_i = 0$ if not. Further, let $\mathcal{O} = \{i \in \mathcal{C} : O_i = 1\}$ represent the set of all cases and controls. As mentioned before, in a typical NCC study the the covariates are collected from the cases of one particular endpoint $\mathcal{E}_k$, and a group of time-matched controls S_k. For each time $\tilde{T}_i$, when an event of type k happens, a control set $S_{k,i}$ is sampled without replacement from $\mathcal{R}_i \backslash \{i\}$, where $\mathcal{R}_i$ is the risk set at time $\tilde{T}_i$.

2.7.2 The likelihood

Saarela et al. have proposed a full likelihood for survival data with a nested case-control design. First let $\theta = (\theta_1, \dots, \theta_K)$, where $\theta_k = (\psi_k, \beta_k)$ is a vector of parameters characterizing the hazard function $\alpha_k(t_i | Z_i, x_i; \theta_k)$ for individual i specific to event type k, $k = 1, \dots, K$. ψ_k include the parameters connected to baseline hazard, while β_k is the regression coefficients. Since Z is not fully observed we model it through a parametric distribution with μ being the parameters characterizing it. Therefor Z is considered to be stochastic and this is indicated by the capital Z, while x is considered to be known and is denoted with little x. $\tilde{Z}$ is the partially observed covariate vector, which means that $\tilde{Z}$ is of same dimension as Z and $\tilde{Z} = Z$ for $i \in \mathcal{O}$

and is unobserved for $i \in \mathcal{C} \backslash \mathcal{O}$. The full likelihood is

$$L(\theta, \mu) = p(\tilde{T}, E, O, \tilde{Z} | x, \theta, \mu)$$

where $p(\tilde{T}, E, O, \tilde{Z} | x, \theta, \mu)$ is the joint distribution of $(\tilde{T}, E, O, \tilde{Z})$ given (x, θ, μ). By two assumptions:

1. Random vectors $(\tilde{T}_i, E_i, Z_i, X_i)$, $i \in \mathcal{C}$ are independent
2. The conditional distribution of O, $p(O|\tilde{T}, E, Z, x; \theta, \mu)$ only depend on data observed for all $i \in \mathcal{C}$, which means that O is independent of Z

the likelihood can be written as a product, and when Z is discrete it looks like

$$\begin{aligned} L(\theta, \mu) \propto & \prod_{i \in \mathcal{O}} p(\tilde{T}_i, E_i | Z_i, x_i; \theta) p(Z_i | x_i; \mu) \\ & \times \prod_{i \in \mathcal{C} \backslash \mathcal{O}} \sum_{z_i} p(\tilde{T}_i, E_i | z_i, x_i; \theta) p(Z_i = z_i | x_i; \mu). \end{aligned} \tag{2.16}$$

We see that the likelihood is made up of two parts, this is due to the fact that we don't have the same information about all individuals. For cases and controls, Z is known, but for individuals outside the subcohort Z is unknown. Therefor we model it by $p(Z_i|x_i; \mu)$ and then "integrate" it out. The proportionality sign is due to disregarding the sampling distribution.

The likelihood expression for $(\tilde{T}_i, E_i)$ can be defined in terms of the outcome specific hazard

$$\begin{aligned} & p(\tilde{T}_i, E_i | Z_i, x_i; \theta) \\ & \propto \prod_{k=1}^{K} [\alpha_k(\tilde{T}_i | Z_i, x_i; \theta_k)]^{1\{E_i = k\}} \exp\left\{ - \int_0^{\tilde{T}_i} \sum_{k=1}^{K} \alpha_k(t | Z_i, x_i; \theta_k) dt \right\}. \end{aligned} \tag{2.17}$$

In the likelihood expression (2.16) we implicitly assume only one missing covariate and that this covariate is discrete. Define Z_{ij} to be the j-th possible missing covariate for the i-th individual, where $j = 1, \ldots, p$, then assume that $Z_{i,1}, \ldots, Z_{i,q}$ are continuous and $Z_{i,q+1}, \ldots, Z_{i,p}$ are discrete, a more general expression for the full likelihood can then be written as

$$\begin{aligned} L(\theta, \mu) \propto & \prod_{i \in \mathcal{O}} p(\tilde{T}_i, E_i | Z_{i,1:p}, x_i; \theta) p(Z_{i,1:p} | x_i; \mu) \\ & \times \prod_{i \in \mathcal{C} \backslash \mathcal{O}} \int_{z_{i,1:q}} \sum_{z_{i,(q+1):p}} \Big[p(\tilde{T}_i, E_i | z_{i,1:p}, x_i; \theta) p(Z_{i,1:p} = z_{i,1:p} | x_i; \mu) dz_{i,1:q} \Big] \end{aligned} \tag{2.18}$$

If we want to directely optimize this likelihood we need a parametric specification of baseline. Apart from this we also need to specify the distribution of the partly observed covariates. In (2.16) we have one possible missing covariate implicitly assumed to be discrete, but (2.18) shows that this, at least in theory, will work for more than one missing covariate and that these can be both discrete and continuous.

The drawbacks of this method is first of all extra modeling assumptions, both a parametric baseline, but also realistic distributions of the partially observed covariates needs to be decided upon. The extra modeling assumptions result in more parameters which means more uncertainty and it also increase the risk of model miss specification. Another drawback is computational time, at least if you need to do Monte Carlo approximation then it takes quite a long time to optimize the likelihood even with very good starting values.

2.7.3 Approximations to the likelihood

As we can see from (2.18) the likelihood can be hard to evaluate, and then especially the integral. A way around this could be to use a Monte Carlo approximation to the integral, this is briefly mentioned in the appendix of [19]. If Monte Carlo integration is applied to the integral in (2.18) the likelihood would look like

$$\begin{aligned} L(\theta, \mu) \propto & \prod_{i \in \mathcal{O}} p(\tilde{T}_i, E_i | Z_{i,1:p}, x_i; \theta) p(Z_{i,1:p} | x_i; \mu) \\ & \times \prod_{i \in \mathcal{C} \setminus \mathcal{O}} \frac{1}{M} \sum_{m=1}^{M} p(\tilde{T}_i, E_i | z_{i,1:p,m}, x_i; \theta) \end{aligned}$$

where the $z_{i,1:p,m}$ are drawn from $p(Z_{i,1:p}|x_i;\mu)$. Monte Carlo estimates is unbiased and consistent (as long as $p(\tilde{T}_i, E_i|z_{i,1:p,m}, x_i; \theta)$ has finite variance).

When the covariates are independent we can assume a (simple) one-dimensional distribution for each covariate, and the sampling step would be easy. When we no longer can assume independence, the sampling step may be complicated because the distribution $p(Z_{i,1:p}|x_i;\mu)$ may be non-standard. This can of course be the case with independent covariates as well, but I think it's easier to find reasonable distributions for each covariate alone, that are for instance implemented in **R**. If we find ourself in a situation where it's too hard or even impossible to sample from $p(Z_{i,1:p}|x_i;\mu)$ one solution is to use importance sampling [18]

$$\begin{aligned} & \int_{z_{i,1:p}} p(\tilde{T}_i, E_i | z_{i,1:p}, x_i; \theta) p(z_{i,1:p} | x_i; \mu) dz_{i,1:p} \\ & \approx \frac{1}{M} \sum_{m=1}^{M} \frac{p(\tilde{T}_i, E_i | z_{i,1:p,m}, x_i; \theta) p(z_{i,1:p,m} | x_i; \mu)}{g(z_{i,1:p,m})} \end{aligned}$$

Then $z_{i,1:p}$ is sampled from $g(Z_{i,1:p})$ which we can decide ourself, it should be as similar as possible to $p(Z_{i,1:p}|x_i;\mu)$, but we can for instance sample each covariate independent of the others from their marginal distribution, given that the marginal distributions is easy to sample from.

Another problem with the Monte Carlo approximation is that each new value of μ in the optimization routine would require a new sequence of $z_{i,1:p}$'s since $p(Z_{i,1:p}|x_i;\mu)$ depend on μ, this results in a jagged likelihood function that is hard to optimize. A smart trick is then to use importance sampling and choose $g(Z_{i,1:p})$ to be $g(Z_{i,1:p}|x_i;\mu_0)$, where μ_0 is given. Then the same sequence of simulated values can be used in every step of the optimization and the likelihood function will also be smooth and therefore much easier to optimize.

Another alternative is of course Markov Chain Monte Carlo [18], where the idea is to generate $z_{1:p}^{(1)}, z_{1:p}^{(2)}, \ldots$ in such a way that the sequence converge in distribution to $p(Z_{i,1:p}|x_i;\mu)$.

2.8 Two phase design and calibration

2.8.1 Two phase stratified sampling

Assume that we have N subjects in the cohort, this is called the phase one sample and can be treated as a random sample from an infinite population. The cohort is stratified into M different strata on the basis of information available for everyone, with N_m subjects in stratum m. The cases could make up an additional stratum, and in a competing risk setting K different types of cases would constitute K additional strata. Then n_m subjects are sampled at random without replacement from the m-th stratum, the phase two subjects and the total number of sampled subjects are then $n = n_1 + \ldots + n_{M+K}$. Each subject has an associated sampling weight N_m/n_m, the contribution from a sampled subject is then up-weighted so that the total contribution from a stratum is representative of the total contribution assuming all cohort members from that stratum had been analyzed.

Both CC- and NCC-designs can be considered as two-phase sampling designs, at phase 1 we sample N individuals from an infinite population, those individuals constitute the cohort. At this phase we gather easy collectable information from all N individuals. At phase 2 we use random sampling on the cohort to obtain the subcohort, however the sampling plan in an NCC is somewhat more complicated than stratified sampling since sampled individuals also are matched on time. Then additional information is collected from cases and sampled controls.

The variance of parameter estimates in a two phase sampling design can be divided into two parts, see B.2, corresponding to each phase of the sampling. The first part is the cohort variance, the variance we would get if all data from the entire cohort was known and this represent the phase one variance. The second part is the variability coming from the fact that some covariates are only observed at phase two, this is the sampling variance.

Breslow et al. [4, 5] wants to minimize the phase two variance in a case-cohort study. This variance is also the normalized design based variance of the standard Horvitz-Thompson estimator for an unknown finite population total, namely the total of the efficient influence function (IF) contribution for all N phase one subjects. For the phase two subjects the IF contribution may be approximated from the observed x_i by delta-betas [26] page 155.

The Horwitz-Thompson estimator is a general estimator for a population total. If $\tau = \sum_{i=1}^{n} y_i$ is a population total and let π_i be the probability that y_i is included in the sample (subcohort), then the estimator is

$$\hat{\tau} = \sum_{i=1}^{s} \frac{y_i}{\pi_i} = \sum_{i=1}^{N} \frac{\mathcal{O}_i}{\pi_i} y_i$$

where s is size of the sample and $\mathcal{O}_i$ indicating whether or not individual y_i is included in the sample. If we apply this to our case-cohort situation, we would base estimation on

$$\sum_{i=1}^{N} \frac{\mathcal{O}_i}{\pi_i} \dot{l}(\beta) \qquad \dot{l}(\beta) = \frac{\partial \log L(\beta)}{\partial \beta}, \text{ score for } \beta$$

this weighing method is related to the WPL-method described earlier.

When additional information exists for phase one subjects, the Horvitz-Thompson estimator is inefficient. In this case, when auxiliary variables correlated with the efficient influence function exists for all individuals in the subcohort. To approximate the optimum choice of auxiliary variables Breslow et al. suggests the five step "plug-in" method of Kulich and Lin [15].

The five step procedure

1. Use a weighted regression to predict the partially missing covariates from information known for the entire cohort. According to [5] there has to be done one regression per missing covariate, but alternatively one multivariate regression could be done.

2. Impute the predicted values for the missing covariate for all cohort members, variables already known for everyone are used as they are.

3. Fit a Cox-model to the entire cohort by using the imputed values for the partially missing covariates and the known values for the other variables. Determine the imputed delta-beta contribution $\Delta_i\hat{\beta}$ [2] for each cohort member.

[2] a 1.order approximation is return by **R**

4. Use the imputed delta-betas as auxiliary variables in calibration or estimation of the weights, (see below).

5. Finally estimate β by a weighted Cox-regression of the phase two data.

Where $\Delta_i\hat{\beta} = \hat{\beta} - \hat{\beta}_{(i)}$ and $\hat{\beta}_{(i)}$ is the estimate of β obtained without including observation i in the estimation.

2.8.2 Calibration

Calibrated weights $w_i = g_i d_i$, are weights that are as close as possible to the population based weights, but at the same time respecting a set of constraints. Let $\pi_i = Pr(i \in \mathcal{O})$ where $\mathcal{O}$ is the collection of all cases and sampled controls, while $\mathcal{C}$ again denote the cohort. Let $x_i = (x_{i,1}, \dots, x_{i,p})$ be the imputed delta-betas for observation i, and further let $x_{\text{tot}} = \sum_{\mathcal{C}} x_i$, which is assumed to be known. We want the calibrated weights w_i to be as close as possible to $d_i = 1/\pi_i$ while respecting the calibration equation

$$\hat{x}_{\text{tot}} = \sum_{\mathcal{O}} w_i x_i = x_{\text{tot}}$$

which means that we want the population total of the auxiliary variables to be estimated exact. The term "as close as" requires a measure of distance $G_i(w, d)$ and these distance measures should share some basic features. For element i for fixed $d > 0$, $G_i(w, d)$ is nonnegative, differential with respect to w, strictly convex and defined on an interval containing d such that $G_i(d, d) = 0$. There are different suggestions concerning these distant measures, one is something reminiscent of Pearson chi-square statistic

$$G_i(w, d) = (w_i - d_i)^2/2d_i \tag{2.19}$$

If we let λ be a vector of Lagrange multipliers, then the problem is to minimize

$$f = \sum_{\mathcal{O}} \frac{(g_i d_i - d_i)^2}{2d_i} + \sum_{\mathcal{O}} \lambda g_i d_i x_i - \lambda x_{\text{tot}},$$

for which the result is

$$\begin{aligned} g_i &= 1 - \hat{\lambda}^T x_i \\ \hat{\lambda} &= \left(\sum_{\mathcal{O}} d_i x_i x_i^T \right)^{-1} \left(\sum_{\mathcal{O}} d_i x_i - x_{\text{tot}} \right). \end{aligned}$$

This results in the generalized regression estimator (GREG) [25], but g_i can be negative, which will result in negative weights. Poisson deviance is another distant measure suggestion which guarantees that the weights are always positive.

$$G_i'(w,d) = w_i \log(w_i/d_i) - w_i + d_i$$

now the minimization problem is

$$f = \sum_{\mathcal{O}} [g_i d_i \log(g_i) - g_i d_i + d_i] + \sum_{\mathcal{O}} \lambda g_i d_i x_i - \lambda x_{\text{tot}}$$

which result in

$$\begin{aligned} g_i &= \exp(-\lambda^T x_i) \\ \sum_{\mathcal{O}} \exp(\lambda^T x_i) d_i x_i &= x_{\text{tot}} \end{aligned}$$

where the last equation can't be solved explicitly for λ. This is known as the (generalized) raking estimator [10], as we can see g_i can't be negative, and therefor the calibrated weights with Poisson deviance as distance measure will always be positive. Even though different distance measures results in different weights and thereby different estimators, Deville and Särndal [11] have shown that all estimators are asymptotical equal to the GREG estimator, actually by a 2. order Taylor expansion at $w = d$, Pearsons chi-squared statistic is equal to Poisson deviance, $G'(w,d) \approx G(w,d)$

$$\begin{aligned} G'(w,d) &\approx G'(d,d) + \frac{\partial G'(d,d)}{\partial w}(w-d) + \frac{\partial^2 G'(d,d)}{\partial w^2}\frac{(w-d)^2}{2!} \\ &= 0 + 0 + \frac{(w-d)^2}{2d} = G(w,d) \end{aligned}$$

For more distance measures and discussion of their properties see [10, 11].

2.8.3 The five step procedure and calibration with competing risk and NCC sampling

A thread through this thesis has been NCC sampling and multiple outcomes, to my knowledge the calibration method hasn't been generalized to this setting and this is an attempt at doing so.

The five step procedure with multiple outcomes

1. Use a weighted regression to predict the partially missing covariates from information known for the entire cohort, either one regression per endpoint or one multivariate regression

2. Impute the predicted values for the missing covariates for all cohort members, variables already known for everyone are used as they are.

3. Fit one Cox-model per endpoint to the entire cohort by using the imputed values for the partially missing covariates and the known values for the other variables. Determine the imputed delta-beta contribution for each cohort member for each endpoint.

4. Use the imputed delta-betas as auxiliary variables in calibration or estimation of the weights, there will be one vector of weights per endpoint.

5. Finally estimate β belonging to each endpoint by a weighted Cox-regression of the phase two data.

With CC-sampling the weights used in the the weighted regression in 1. was the number of individuals in each stratum divided by the number of sampled individuals in each stratum. When we now have a NCC-design, the natural choice of weights is the inverse of the sampling probability, estimated with one of the techniques described in section 2.6.

Calibration

Again let $\pi_i = Pr(i \in \mathcal{O})$, and $x_{i,k} = (x_{i,1,k}, \ldots, x_{i,p,k})$ be the delta-beta vector for individual i corresponding to the k-th endpoint. We then have $x_{\text{tot},k} = \sum_{i \in \mathcal{C}} x_{i,k}$ and a $n \times K$ matrix of population based weights and also a matrix of calibrated weights. We get a set of calibration equations

$$\hat{x}_{\text{tot},k} = \sum_{i \in \mathcal{O}} w_i x_{i,k} = x_{\text{tot},k}$$

and we suggest that the i-th column of the weight matrix can be calibrated by using the i-th calibration equation.

2.9 Scheike's likelihood

Scheike and Juul [23] have also proposed a maximum likelihood estimator for NCC data. Their likelihood is deep down the same as Saarela's, but the reasoning is somewhat different. Scheike recognize that the likelihood contribution can be divided into three parts; contributions from cases, from controls and contributions from individuals outside the subcohort, but when put together result in the same likelihood as Saarela's.

Saarela et al. model the baseline hazard and the distribution of the partially observed covariates and is therefore able to directly optimize the likelihood, but Scheike choose a different way. First instead of using a parametric specification of the distribution of Z, they choose to use a non-parametric specification through strata defined by some or all covariates known for the entire cohort. Secondly they choose to keep the baseline unspecified, as in a usual Cox-likelihood. Because of this the likelihood can't be directly optimized and instead the EM-algorithm [18] is used and the standard errors are obtained by EM-aided differentiation.

This likelihood, as Saarela's likelihood, rest on the assumption that the censoring only may depend on X, but also that the survival times and censoring times are independent.

Scheike points out that the likelihood is equal for both NCC and CC, then the same program can be used in both settings and the efficiency is the same if data are of equivalent size. But I think the same holds for Saarela's likelihood, Kulathinal and Arjas [13] have proposed a likelihood for case-cohort data, they take a Bayesian standing and use data augmentation, but the likelihood part is still the same as Saarela's for NCC data.

Chapter 3

Simulations

To get a feeling with the properties of estimators simulation can be a useful tool. You can compare efficiency between estimators and maybe just as important, see if the estimators adds bias.

In order to simulate survival data, and nested case-control data in particular, one first has to decide upon what kind of baseline one wants to use, and thereby what kind of distribution the survival times are coming from. Secondly one also need to determine the censoring scheme.

In practice, one both draws a censoring time C and a survival time T for every individual in the cohort. If the censoring time is smaller than the survival time, then that individual is censored at C, else that individual dies at T. For competing risks with two endpoints, two survival times T_1 and T_2 are drawn for every individual and if $T_1 = \min(T_1, T_2, C)$, the individual experience event 1 and corresponding for event 2 and censoring. The number of individuals who dies can then be regulated by changing the parameters in the survival distribution or the censoring distribution.

3.1 Censoring schemes

There are different strategies concerning how the censoring should be carried out, censoring at a given time, random censoring or something in between the two of them. All of them mimicking different, more or less, real situations.

Censoring at a fixed time

This censoring scheme will mimic a study where all individuals enter the study at the same time. There are no loss of followup and at a fixed time, all individuals still under risk will be censored.

Random censoring

This censoring scheme, is in a way, the opposite of censoring at a given time. The censoring times are random, they are for instance drawn from a uniform or an exponential distribution. It mimics a study where not all individuals enter the study at the same time and there may also be loss of follow-up. In a way, this is a more relevant censoring scheme since, in practice there will almost always be loss of follow-up to some extent.

”Random-fixed” censoring

This scheme is something in between the two schemes above. The censoring time for some of the individuals are for instance drawn from a uniform distribution, and the rest are censored at the maximum value of the distribution. This mimics a study, where all individuals enter the study at the same time, but there are loss of follow-up to some extent, how much depend on how the censoring distribution is defined.

3.2 Simulation with one binary covariate

The simulation is done on a cohort of size 1000 and are run 1000 times. We have two different events, a relative common one, $E_i = 1$, that about 10% of the individuals experience and more rare event, $E_i = 2$, that only about 3% of the cohort experience. We sample one control per case, which means that the subcohorts for the two single endpoint models consist of approximately 200 and 60 individuals respectively, at start of study, while the subcohort for the multiple endpoint model consists of approximately 260 individuals at start of study.

All in all we do seven analysis' on these data, first a normal Cox-regression and a traditional NCC-analysis. Then four analysis based on the weighted partial likelihood (2.15), with the four different weighting schemes described in section 2.6. Finally we also do the analysis based on the full likelihood explained in chapter 2.7.2, where a Weibull baseline is chosen. The real distribution for the survival times is the exponential, but since the exponential distribution is a special case of Weibull(λ, ν) when $\nu = 1$, this should not be a problem. The analysis based on the weighted likelihoods and the full likelihood is done both with a multiple endpoint model and with two separate single endpoint models.

In order to structure the discussion about the simulation result I have made a list of discussion points:

1. Bias, $\frac{\text{bias}}{\text{empirical standard deviation}} > \frac{1}{3}$ is considered to be a serious bias
2. Empirical standard deviation compared to model based (robust) standard deviation
3. The performance of the WPL-models
4. Full likelihood compared to WPL
5. Traditional NCC compared to the alternatives
6. Efficiency improvements with multiple outcome models compared to single outcome models

3.2.1 Random cencoring

The simulation model is

$$\begin{aligned} Z &\sim Bin(1, 0.5) \\ T_1 &\sim Exp(\lambda_1 \exp(Z\beta_1)) \\ T_2 &\sim Exp(\lambda_2 \exp(Z\beta_2)) \\ C &\sim U[0, 0.13] \\ \tilde{T} &= \min(T_1, T_2, C) \end{aligned}$$

given (Z, T_1, T_2) and C are independent. β_1 and β_2 are regression coefficients connected to endpoint 1 and 2. An individual is a case of type 1, $E_i = 1$, if T_1 is smaller than T_2 and the C, and similar with cases of type 2, $E_i = 2$, and censored individuals, $E_i = 0$. By altering λ, the expectation of the event times changes and thereby the number of cases in the cohort can be decided.

Results

Table 3.1 shows the results of this experiment for $\beta_1 = \beta_2 = 0$ and $\beta_1 = \beta_2 = 1$, the result of the entire simulation experiment can be found in Table A.1 and A.2 in the appendix.

1. For the common endpoint β is estimated without any noticeable bias. The same is true for the rare endpoint when $\beta = 0$, when $\beta = 1$ there is a small bias, but compared to the standard deviation this is not important. We see that the full Cox-regression suffer from this as well and it's probably due to small sample sizes.
2. Both the empirical variance and the estimated model based variance, the robust model based variance for the WPL-models, is reported. The robust variance can in some cases be too conservative, but for

Table 3.1: Simulation, one covariate

			Cause 1: 10% cases				Cause2: 3% cases			
		Model/	Mean	Mean	Emp.		Mean	Mean	Emp.	
β_1, β_2	Method	weights	est.	est. sd	sd	Eff.	est.	est. sd	sd	Eff.
0.00	Cohort	Cox	0.008	0.202	0.202		-0.011	0.378	0.376	
	Trad. NCC	Strat. Cox	0.004	0.288	0.290	0.480	-0.014	0.558	0.578	0.423
	WPL	Samuelsen	0.010	0.259	0.255	0.621	-0.009	0.413	0.410	0.841
	WPL	GAM	0.010	0.260	0.256	0.616	-0.009	0.414	0.411	0.837
	WPL	Logistic	0.010	0.248	0.256	0.616	-0.009	0.370	0.411	0.837
	WPL	Chen	0.010	0.250	0.261	0.593	-0.009	0.369	0.414	0.825
	MLE	Weibull	0.009	0.258	0.255	0.621	-0.009	0.411	0.406	0.858
1.00	Cohort	Cox	1.015	0.226	0.227		1.051	0.422	0.495	
	Trad. NCC	Strat. Cox	1.000	0.323	0.315	0.519	1.015	0.598	0.562	0.776
	WPL	Samuelsen	1.006	0.278	0.276	0.676	1.041	0.456	0.525	0.890
	WPL	GAM	1.011	0.280	0.280	0.657	1.046	0.457	0.528	0.879
	WPL	Logistic	1.011	0.276	0.280	0.657	1.046	0.454	0.528	0.879
	WPL	Chen	1.011	0.279	0.282	0.648	1.046	0.457	0.528	0.879
	MLE	Weibull	1.004	0.277	0.274	0.686	1.041	0.452	0.524	0.892

WPL = weighted partial likelihood estimation, MLE = maximum likelihood estimation

Number of times the likelihood is flat for $\beta = (0,1)$ *is* $(0,2)$

the common endpoint the variances seems to be in good agreement. For the rare endpoint, the differences are a bit bigger, but the model based variance isn't consistently bigger than the empirical variance, and the differences isn't that big for any of the models in any of the simulations. Therefore, I think that the robust variance estimator does the job, at least in this situation.

3. All WPL-models are very alike both regarding bias and efficiency. But if we look at the small differences, Samuelsen's weights take a narrow victory. Chen's weights is the least efficient method while GAM and logistic perform equally.

4. Saarela et al. reported a marginal efficiency gain by using their likelihood compared to WPL, we see that the differences in variance between the WPL-models and the full likelihood in our simulation are almost non-existing and compared to the extra complexity both regarding modeling and implementation it is not worth it.

5. There is quite a lot to gain by doing something else than the traditional analysis The most pronounced difference is for $\beta = 0$ with the rare endpoint where the efficiency with the full likelihood is twice as big as with the traditional NCC analysis.

6. Table A.1 and A.2 shows the complete result from the simulation experiment, the numbers in brackets are the result when two single endpoint models is used instead of one multiple endpoint model. We see that there are efficiency improvements with the multiple endpoint model compared to single endpoint models and obviously the improvements are biggest for the rare endpoint since the increase in number of controls per case is much larger there. Saarela et al. noticed only a small variance reduction by using the multiple endpoint model compared to the single endpoint model. We see, especially for the rare endpoint, that there is a considerably efficiency gain by using the multiple endpoint model. I think part of the reason for Saarela et al.'s small efficiency gain is that they have 5% cases of the endpoint their looking at and 9% cases of the other endpoint, out of a cohort of size 3815. This means that they actually have a quite big subcohort without the extra cases and controls from the other endpoint and therefore the efficiency gain is more moderate than what we get in this simulation study.

Chen [6] states that his local averaging method is superior to the typical inclusion probability methods, this doesn't seem to be the case in this simulation study. We actually see that it's mainly the least efficient model. It might have something to do with the size of the subcohort, his full cohort consists of 1000 individuals and it seems like there are about 10%

cases, which is what we have for the common endpoint, but he sample three controls per case, whereas we only sample one, which might explain the differences. But if this is the reason, then it seems like in order for the local averaging method to be more efficient that the inclusion probability of for instance Samuelsen it needs to have more than one control per case. Another possible explanation for why the local averaging method isn't as good as the other weights might be that we haven't chosen the "right" number of partitions of the time axis. This should of course be tested, but since the focus here is on the full likelihood, and how it does compared to the other methods, we haven't tried to partition the time axis differently.

3.2.2 Fixed censoring

The simulation model is:

$$\begin{aligned} Z &\sim Bin(1, 0.5) \\ T_1 &\sim Exp(\lambda_1 \exp(Z\beta_1)) \\ T_2 &\sim Exp(\lambda_2 \exp(Z\beta_2)) \\ C &= 0.13 \\ \tilde{T} &= \min(T_1, T_2, C) \end{aligned}$$

Results

Table 3.2 shows the results for $\beta_1 = \beta_2 = 0$ and $\beta_1 = \beta_2 = 1$ when we apply fixed censoring instead of random censoring.

1. When $\beta = 0$ the estimates are very accurate both for the common and the rare endpoint. When $\beta = 1$ the estimates are a bit more skewed at least for the rare endpoint, but compared to standard deviation this isn't important.

2. The empirical standard deviations and the model based (robust) standard deviations are mostly in good agreement also here, but the model based standard deviations are actually a bit smaller than the empirical standard deviations.

3. Due to the fixed sampling regardless of which weighing method used the weights are the same for all individuals. Logistic regression, GAM and Chen result in the same weights, while Samuelsen's is a bit different, therefore only weights from logistic regression and Samuelsen's weights are used. Since the weights are almost equal we would expect the results to be almost identical as well, and they are, both the estimates and the empirical and robust model based standard deviations are almost identical for both endpoints and both β-values.

Table 3.2: One covariate, fixed censoring

			Cause 1: 10% cases				Cause 2: 3% cases			
β_1, β_2	Method	Model/ weights	Mean est.	Mean est. sd	Emp. sd	Efficiency	Mean est.	Mean est. sd	Emp. sd	Efficiency
0.00	Cohort	Cox	0.005	0.202	0.208		0.008	0.377	0.384	
	Trad. NCC	Strat. Cox	-0.013	0.289	0.297	0.490	0.019	0.553	0.582	0.435
	WPL	Samuelsen	-0.004	0.261	0.270	0.593	0.001	0.411	0.425	0.816
				(0.278)	(0.284)	(0.536)		(0.507)	(0.524)	(0.535)
	WPL	Logistic	-0.004	0.261	0.270	0.593	0.001	0.387	0.425	0.816
				(0.277)	(0.284)	(0.536)		(0.480)	(0.524)	(0.535)
	MLE	Weibull	-0.004	0.261	0.270	0.593	-0.001	0.412	0.425	0.816
				(0.283)	(0.289)	(0.518)		(0.528)	(0.543)	(0.500)
1.00	Cohort	Cox	1.024	0.226	0.236		1.077	0.430	0.455	
	Trad. NCC	Strat. Cox	1.044	0.327	0.338	0.488	1.078	0.554	0.566	0.646
	WPL	Samuelsen	1.026	0.280	0.290	0.662	1.080	0.461	0.474	0.921
				(0.296)	(0.309)	(0.583)		(0.548)	(0.560)	(0.660)
	WPL	Logistic	1.027	0.302	0.290	0.662	1.080	0.486	0.474	0.921
				(0.319)	(0.309)	(0.583)		(0.555)	(0.567)	(0.644)
	MLE	Weibull	1.028	0.280	0.290	0.662	1.084	0.461	0.484	0.884
				(0.300)	(0.314)	(0.565)		(0.567)	(0.586)	(0.603)

The numbers in brackets are the result if one only uses the controls sampled for that particular outcome.
WPL = weighted partial likelihood estimation, MLE = maximum likelihood estimation
Number of times the likelihood is flat for $\beta = (0,1)$ *is* $(1,18)$

4. The full likelihood is not more efficient than the WPL-models. The efficiency of the full likelihood when using multiple outcomes is the same as the WPL-models, but it has somewhat lower efficiency if we are using two single endpoint models. Which means that if we had been in a situation with only one outcome, then the full likelihood would actually have been less efficient than WPL.

5. The estimates from the traditional NCC is somewhat more skewed than the estimates from the rest of the methods, but compared to the standard deviation this isn't important. The efficiency with the traditional NCC compared to doing something else is about the same as what we had with random censoring.

6. The efficiency gain with a multiple endpoint model compared to two single endpoint models is perhaps a bit lower with fixed censoring than with random censoring for $\beta = 0$. For $\beta = 1$ on the other hand it looks like it's slightly higher, we saw that with random censoring there was nothing to gain for the common endpoint, here we see a small efficiency gain for both endpoints.

3.3 Simulation with two covariates

The experiment done above has a really easy setting with only one covariate. A usual setting for NCC-designs is that there are some information available for every individual in the cohort and some information only available for the cases and controls. Even though the WPL-methods fails to utilize information known for everybody in the cohort it is possible to indirectly include the extra information by using it in the estimation of the weights. The full likelihood on the other hand can easily handle both partially and fully observed covariates. If it can estimate the fully observed covariates or even the partially observed covariate as good, or almost as good as the the method based on the entire cohort this would be a big plus.

Again the simulation is done on a cohort of size 1000 and are run 1000 times. There are one common $E_i = 1$ and one rare $E_i = 2$ endpoint and one control per case is sampled.

When we now have two covariates we can test the calibrated weights as well. The calibration function used is unbounded raking, which means that Poisson deviance is used as distant measure. We stratify according to status, which means that we have two strata with cases and a third strata with all the controls. We have chosen weights estimated with GAM to be the weights that goes into the logistic regression that is used to estimate the partially observed covariate.

The discussion points are:

1. Bias, $\frac{\text{bias}}{\text{empirical standard deviation}} > \frac{1}{3}$ is considered to be a serious bias
2. Empirical standard deviation compared to model based (robust) standard deviation
3. The performance of the WPL-models
4. Full likelihood compared to WPL
5. Traditional NCC compared to the alternatives
6. Calibrated weights

3.3.1 Two independent covariates

The simulation model used her is:

$$\begin{aligned}
X &\sim U[0,1] \\
Z &\sim Bin(1, 0.5) \\
T_1 &\sim Exp(\lambda_1 \exp(X\beta_{X_1} + Z\beta_{Z_1})) \\
T_2 &\sim Exp(\lambda_2 \exp(X\beta_{X_2} + Z\beta_{Z_2})) \\
C &\sim U[0, 0.13] \\
\tilde{T} &= \min(T_1, T_2, C)
\end{aligned}$$

Given (Z, X, T_1, T_2) and C are independent, Z is only known for cases and controls while X is known for the entire cohort. Part of the simulation result can be found in Table 3.3 and the full result can be found in the appendix, Table A.3.

Results

1. Most of the estimates, especially when $\beta = 1$ have some bias, the biggest being traditional NCC where $\hat{\beta}_{X_2} = 1.108$, but because of the big standard deviation not even that bias is actually important.

2. It is only small differences between the empirical standard deviations and the (robust) model based standard deviations, but anyhow these differences is mainly opposite of what one would expect. At least with WPL models we would expect that the robust standard deviation to be bigger than the empirical, but it is mainly the other way around.

3. Table A.3 include the result from all four WPL models and can be found in the appendix. We see that the differences between the weighted partial likelihoods is still almost non-existing, but perhaps again with a slightly advantage for Samuelsen's weights for both covariates.

Table 3.3: Simulation, two independent covariates

				Cause 1: 10% cases				Cause 2: 3% cases			
			Model/	Mean	Mean	Emp.		Mean	Mean	Emp.	
β_1, β_2	Cov.	Method	weights	est.	est. sd	sd	Eff.	est.	est. sd	sd	Eff.
0.00	z	Cohort	Cox	-0.021	0.203	0.213		-0.015	0.374	0.375	
	x			-0.006	0.348	0.355		-0.014	0.638	0.606	
	z	Trad. NCC	Strat. Cox	-0.019	0.292	0.314	0.460	0.001	0.572	0.614	0.373
	x			-0.001	0.506	0.538	0.435	0.005	0.988	1.048	0.334
	z	WPL	Samuelsen	-0.014	0.262	0.273	0.609	-0.010	0.415	0.416	0.813
	x			-0.005	0.453	0.465	0.583	-0.070	0.709	0.681	0.792
	z	MLE	Weibull	-0.015	0.258	0.282	0.571	-0.008	0.407	0.410	0.837
	x			-0.006	0.348	0.352	1.017	-0.012	0.639	0.608	0.993
	z	WPL	Calibrated	-0.017	0.276	0.283	0.566	-0.011	0.430	0.424	0.782
	x			-0.003	0.373	0.381	0.868	-0.014	0.664	0.634	0.914
1.00	z	Cohort	Cox	1.023	0.227	0.227		1.038	0.423	0.435	
	x			1.004	0.352	0.363		1.002	0.649	0.675	
	z	Trad. NCC	Strat. Cox	1.064	0.337	0.341	0.443	1.082	0.662	0.675	0.415
	x			1.049	0.554	0.582	0.389	1.108	1.083	1.202	0.315
	z	WPL	Samuelsen	1.039	0.283	0.283	0.643	1.049	0.460	0.467	0.868
	x			1.032	0.473	0.497	0.533	1.032	0.730	0.763	0.783
	z	MLE	Weibull	1.041	0.277	0.276	0.676	1.057	0.452	0.458	0.902
	x			1.050	0.358	0.367	0.978	1.047	0.652	0.679	0.988
	z	WPL	Calibrated	1.074	0.297	0.298	0.606	1.082	0.477	0.471	0.823
	x			1.019	0.405	0.418	0.754	1.013	0.691	0.708	0.909

WPL = weighted partial likelihood estimation, MLE = maximum likelihood estimation
Number of times the likelihood is flat for $\beta = (0, 1)$ is $(0, 15)$
x is observed for the entire cohort, z only observed for the cases and controls
Distance measure for the calibrated weights are Poisson deviance

4. The full likelihood is superior in estimating the fully observed covariate in this simulation. It is, as efficient as Cox-regression on the full likelihood, which in a way is quite natural since it's a ML-method and it uses information for every member of the cohort in the estimation. But when estimating the coefficient connected to the partially observed covariate, the variance difference between the WPL-method and the full likelihood isn't that big. Saarela et al. proclaim however that efficiency gains also can be achieved for the partially observed covariate when it's correlated with a fully observed covariate. In the next section we see that this is the case.

5. Again there is most to gain by doing something else than the traditional NCC with the rare endpoint and then of course especially when estimating X with the full likelihood. Apart from with the full likelihood there isn't any difference between the efficiency in estimating X and Z with the other models compared to the traditional NCC. This is quite natural since non of the other models actually use the extra information in knowing X for the entire cohort.

6. The calibrated weights doesn't improve the efficiency when estimating the partial observed covariate, but for the fully observed covariat there are some improvements compared to the other weights. For instance, the efficiency increase from about 0.6 to 0.868 for $\beta_{X_1} = 0$ with 10% cases.

3.3.2 Two dependent covariates

Saarela proclaim that efficiency improvements also can be obtained for the partially observed covariate if it is correlated with the fully observed covariate and it is also natural to believe that further efficiency improvements can be obtained with the calibrated weights when X and Z are correlated. The simulation model used in this case is:

$$\begin{aligned} X &\sim U[0,1] \\ Z|X &\sim Bin(1,X) \\ T_1 &\sim Exp(\lambda_1 \exp(X\beta_{X_1} + Z\beta_{Z_1})) \\ T_2 &\sim Exp(\lambda_2 \exp(X\beta_{X_2} + Z\beta_{Z_2})) \\ C &\sim U[0,0.13] \\ \tilde{T} &= \min(T_1,T_2,C) \end{aligned}$$

Again given that (Z,X,T_1,T_2) and C are independent, Z is still only known for cases and controls while X is known for the entire cohort and from the simulations we have that $\mathrm{Cor}(X,Z) = 0.577$. Part of the result can be found in Table 3.4, the complete result can be found in the appendix, Table A.4.

Table 3.4: Simulation, two dependent covariates

				Cause 1: 10% cases				Cause 2: 3% cases			
			Model/	Mean	Mean	Emp.		Mean	Mean	Emp.	
β_1, β_2	Cov.	Method	weights	est.	est. sd	sd	Eff.	est.	est. sd	sd	Eff.
0.00	z	Cohort	Cox	-0.006	0.243	0.237		-0.006	0.447	0.433	
	x			-0.023	0.425	0.428		-0.037	0.779	0.777	
	z	Trad. NCC	Strat. Cox	-0.006	0.353	0.366	0.419	-0.001	0.649	0.743	0.340
	x			-0.017	0.618	0.656	0.426	0.002	1.223	1.348	0.332
	z	WPL	Samuelsen	-0.006	0.317	0.329	0.519	-0.008	0.498	0.490	0.781
	x			-0.005	0.556	0.580	0.545	-0.020	0.874	0.907	0.734
	z	MLE	Weibull	-0.041	0.260	0.249	0.906	-0.042	0.460	0.445	0.947
	x			0.012	0.438	0.439	0.951	-0.002	0.792	0.797	0.950
	z	WPL	Calibrated	-0.006	0.332	0.340	0.486	-0.007	0.514	0.502	0.744
	x			-0.026	0.503	0.506	0.715	-0.035	0.852	0.852	0.832
1.00	z	Cohort	Cox	1.009	0.282	0.274		1.050	0.534	0.558	
	x			0.994	0.449	0.458		1.031	0.834	0.865	
	z	Trad. NCC	Strat. Cox	1.058	0.423	0.436	0.395	1.090	0.846	0.865	0.416
	x			1.000	0.714	0.729	0.395	1.074	1.445	1.585	0.298
	z	WPL	Samuelsen	1.026	0.351	0.344	0.634	1.067	0.577	0.612	0.831
	x			0.992	0.608	0.632	0.525	1.026	0.939	0.967	0.800
	z	MLE	Weibull	0.985	0.292	0.276	0.986	1.028	0.535	0.552	1.022
	x			1.049	0.454	0.456	1.009	1.081	0.830	0.851	1.033
	z	WPL	Calibrated	1.053	0.365	0.357	0.589	1.100	0.593	0.625	0.797
	x			0.952	0.537	0.536	0.730	0.981	0.898	0.952	0.826

WPL = weighted partial likelihood estimation, MLE = maximum likelihood estimation
Number of times the likelihood is flat for $\beta = (0, 1)$ is $(4, 42)$
x is observed for the entire cohort, z only observed for the cases and controls
Distance measure for the calibrated weights are Poisson deviance

Except from point 5 and 6, the result from this simulation is comparable with the previous simualtion.

We would expect that the calibration method is able to reduce the variance even further when X and Z is correlated since the auxiliary variables is based on a logistic regression on Z with X as explanatory variable, but this is not the case. The efficiency compared to the other models are about the same as in the previous simulation.

The full likelihood on the other hand is able to estimate both covariates with very high efficiency, and in this simualtion we see that it's actually manage to estimate both $\beta_{X_2} = 1$ and $\beta_{Z_2} = 1$ with higher efficiency than Cox-regression on the full cohort. This means that it is actually more efficient to use Saarela's likelihood on NCC-data than Cox-regression on the full data when estimating the partially observed covariate. It almost seems like a contradiction that you are able to decrease the variance with less data, but I presume that this is not what usually would happen and the reason for it is that the full likelihood take advantage of the dependencies between the fully observed and the partially observed covariate.

And finally, is there anything to gain by using the (much) more complicated ML-method? The answer to that is yes with one reservation; we need to have some covariates known for the entire cohort. We have see that the ML-model are able to estimate the parameters related to covariates known for the entire cohort almost as accurate as the Cox-regression on the full cohort. And when the fully observed covariate is correlated with the partly observed covariate it also manages to estimate that one almost as accurate as the full Cox-regression. This is a big plus for the full likelihood. But when there is no extra information available in the cohort, the full likelihood is only marginally better, thereof the reservation. In order to want to use the (much) more complicated ML-method it should be superior compared to the WPL-methods because it requires more modeling assumptions, and also because it requires evaluations of possibly many potentially complicated integrals and sums. Therefore a standard analysis using the WPL-model can turn it to a relatively time consuming analysis both regarding computer time and "thinking" time using the ML-method. The preliminary conclusion is then that when there is information available for the entire cohort and the parameters connected to this information is important, or the information known for the entire cohort is correlated with the partly observed covariate then the full likelihood should be used if possible.

3.4 Including more information in the estimation of inclusion probabilities

As mentioned earlier, WPL-methods doesn't itself incorporate the additional information when one or more covariates x, are known for the entire cohort.

One way of using this extra information is through the estimation of weights, where we can let the inclusion probabilities depend on both the survival time and x. With logistic GAM this is straight foreward. We model

$$\mathbf{E}(V_i|\tilde{T}_i) = \frac{\exp(\alpha + f(\tilde{T}) + f(x))}{1 + \exp(\alpha + f(\tilde{T}) + f(x))}$$

Here we have smoothed on x, this is not necessary, but simulation (not displayed) showed that it didn't seem to matter if we used $f(x)$ or ηx, where η is a regression coefficient.

Samuelsen's weights can make use of the information by dividing x into intervals and estimate different sampling probabilities on the basis of which interval the observation falls into. For instance we can divide x into two parts and calculate the sampling probabilities as follows

$$p_j = \begin{cases} 1 & \text{cases} \\ 1 - \prod_{t_i < t_j} \left\{ 1 - \frac{m-1}{n_1(t_i)-1} \right\} & \text{controls with } x_j < 0.5 \\ 1 - \prod_{t_i < t_j} \left\{ 1 - \frac{m-1}{n_2(t_i)-1} \right\} & \text{controls with } x_j \geq 0.5 \end{cases}$$

We assume her that x is one-dimensional and uniform over [0,1], $n_1(t_i)$ is the number of individuals under risk at t_i with $x < 0.5$, similarly $n_2(t_i)$ is number individuals under risk with $x \geq 0.5$.

Table 3.5 og 3.6 shows the efficiency of Samuelsen's and GAM-weights when using x in the estimation compared to not doing so. The estimates and variances are not shown, but the estimates when using x in the estimation are almost unbiased. Samuelsen's weights shows no consistent improvements when x is included in the estimation, actually the variance is mainly larger. It's a different story with GAM-weights, when estimating the partially observed covariate, there is no efficiency gain, but when estimating the fully observed covariate, the efficiency increase quite a lot. The biggest gain is with two independent covariates with $\beta = 0$ where the efficiency increases from 0.570 to 0.920 when x is used in the estimation.

Table 3.5: Efficiency, two independent covariates

		Cause 1: 10% cases				Cause 2: 3% cases			
		Without x		With x		Without x		With x	
β_1, β_2	Weights	z	x	z	x	z	x	z	x
0.00	Samuelsen	0.579	0.578	0.584	0.560	0.860	0.803	0.796	0.820
	GAM	0.567	0.570	0.616	0.920	0.852	0.800	0.803	0.945
1.00	Samuelsen	0.616	0.541	0.610	0.503	0.849	0.827	0.840	0.779
	GAM	0.607	0.538	0.637	0.782	0.846	0.822	0.840	0.898

x is observed for all individuals, z is only observed for cases and controls
Without x - weights estimated without including x, With x - x included in the estimation

Table 3.6: Efficiency, two dependent covariates

		Cause 1: 10% cases				Cause 2: 3% cases			
		Without x		With x		Without x		With x	
β_1, β_2	Weights	z	x	z	x	z	x	z	x
0.00	Samuelsen	0.579	0.596	0.559	0.561	0.766	0.793	0.815	0.780
	GAM	0.575	0.588	0.576	0.744	0.766	0.786	0.827	0.845
1.00	Samuelsen	0.647	0.568	0.574	0.487	0.855	0.775	0.889	0.716
	GAM	0.644	0.560	0.593	0.649	0.855	0.765	0.899	0.857

x is observed for all individuals, z is only observed for cases and controls
Without x - weights estimated without including x, With x - x included in the estimation

3.5 Comparison between the accelerated failure time model and Saarela's likelihood

We have seen through the simulations that the full likelihood is much more efficient than the weighted partial likelihoods, at least when we have one covariate known for the entire cohort, and if even the covariates are correlated, then it is as efficient as Cox-regression on the entire cohort. It could be interesting to find out how much of the efficiency gain that is due to the parametric specification of the baseline. One way of doing this is through an accelerated failure time (AFT) model

$$\log(T) = -Z^T\beta + \varepsilon$$

where ε follows some unspecified distribution. When $T \sim$ Weibull(ν, λ), then the distribution of $\exp(\varepsilon)$ is $\nu\lambda^\nu(t\exp(Z^T\beta))^{\nu-1}$, which results in a hazard on the form

$$\begin{aligned}\alpha(t) &= \nu\lambda^\nu(t\exp(Z^T\beta))^{\nu-1}\exp(Z^T\beta)\\ &= \nu\lambda^\nu t^{\nu-1}\exp(\nu Z^T\beta)\\ &= \nu\lambda^\nu t^{\nu-1}\exp(Z^T\Gamma)\\ &= \alpha_0(t)\exp(Z^T\Gamma)\end{aligned}$$

where $\Gamma = \nu\beta$ and $\alpha_0(t) = \nu\lambda^\nu t^{\nu-1}$. This is a proportional hazard model, the reason for why we use the AFT-model is that this is implemented in **R**, generally parametric proportional hazard models isn't. We can then fit the AFT model, and compare the variance from this model with the variance from the weighted partial likelihood models.

Table 3.7 shows the efficiency of the accelerated failure time model compared to the full likelihood and weighted partial likelihood with GAM weights. The simulation models are the same as in 3.3.1 and 3.3.2.

The point with this comparison is that both the full likelihood and the AFT model specify baseline hazard, and if much of the efficiency gain of the full likelihood is due to this, then the AFT model should have similar efficiency that the full likelihood. But if the efficiency gain is mostly due to the fact that all information known is used, then the efficiency of the AFT model should be similar to the weighted partial likelihood. And we see that it is the latter that actually is the case. This means that since the full likelihood is able to use all information available, it is able to decrease the variance.

Table 3.7: Efficiency compared to Cox-regression on full cohort

		Independent covariates				Dependent covariates			
		10% cases		3% cases		10% cases		3% cases	
β_1, β_2	Weights	z	x	z	x	z	x	z	x
0.00	AFT	0.575	0.561	0.825	0.852	0.624	0.600	0.836	0.837
	Full lik.	0.585	1.000	0.888	1.003	0.932	0.989	1.003	0.951
	GAM	0.575	0.557	0.825	0.836	0.614	0.590	0.824	0.830
1.00	AFT	0.597	0.577	0.903	0.818	0.630	0.537	0.937	0.808
	Full lik.	0.675	1.000	0.999	1.009	1.003	1.028	1.015	1.021
	GAM	0.592	0.574	0.884	0.810	0.619	0.532	0.934	0.787

x is observed for all individuals, z is only observed for cases and controls

3.6 Comparison between Scheike and Juuls MLE and the weighted partial likelihoods

Scheike and Juul [23] describes a maximum likelihood estimator in a Cox's regression model and it could be interesting to compare their likelihood to the weighted partial likelihoods. We have done this by doing the same simulation experiment as they have done only with the WPL-methods, and we have also run Scheike's code for the maximum likelihood estimator.

The simulation model is:

$$\begin{aligned} Z_1 &\sim N(0,1) \\ Z_2 &\sim N(0,1) \\ T &\sim Exp(\lambda \exp(Z_1\beta_1 + Z_2\beta_2)) \\ C &= 15 \\ \tilde{T} &= \min(T, C) \end{aligned}$$

The simulation is done with $\beta_1 = (0, 0.5, 1)$, $\beta_2 = (0, -0.5, -1)$ and baseline $= (0.004, 0.008, 0.016)$ and the results can be found in Table 3.8. Since the four weighing methods gives very similar results[1], we only include one of them. Scheike and Juul's simulation result is found in [23], but for easier comparison, part of their result is given in Table 3.9.

[1] With baseline $= 0.016$ and $\beta = -1$ weights from GAM and logistic regression is a bit more skewed than the result from using Samuelsens weights

Table 3.8: Simulation Scheike and Juul

			Covariate 1			Covariate 2		
	Av.	Model/	Mean	Mean	Emp.	Mean	Mean	Emp.
$\alpha_0(t)$	cases	weights	est.	est. sd	sd	est.	est. sd	sd
0.004	233	Cohort	0.00	0.07	0.07	0.00	0.07	0.07
		Trad. NCC	0.00	0.08	0.08 (1.27)	0.00	0.08	0.08 (1.20)
		Samuelsen	0.00	0.08	0.08 (1.24)	0.00	0.08	0.08 (1.18)
	234	Scheike	0.00	0.08	0.08 (1.22)	0.00	0.08	0.08 (1.20)
0.008	452	Cohort	0.00	0.05	0.04	0.00	0.05	0.05
		Trad. NCC	0.00	0.06	0.05 (1.23)	0.00	0.06	0.06 (1.26)
		Samuelsen	0.00	0.06	0.05 (1.21)	0.00	0.06	0.06 (1.21)
	453	Scheike	0.00	0.05	0.05 (1.15)	0.00	0.06	0.05 (1.13)
0.016	852	Cohort	0.00	0.03	0.03	0.00	0.04	0.03
		Trad. NCC	0.00	0.04	0.04 (1.24)	0.00	0.04	0.04 (1.26)
		Samuelsen	0.00	0.04	0.04 (1.18)	0.00	0.04	0.04 (1.18)
	854	Scheike	0.00	0.04	0.04 (1.14)	0.00	0.04	0.04 (1.09)
0.004	283	Cohort	0.50	0.06	0.06	-0.50	0.06	0.06
		Trad. NCC	0.50	0.08	0.08 (1.42)	-0.51	0.09	0.09 (1.37)
		Samuelsen	0.50	0.08	0.08 (1.37)	-0.51	0.08	0.08 (1.33)
	292	Scheike	0.48	0.07	0.07 (1.30)	-0.49	0.07	0.08 (1.25)
0.008	553	Cohort	0.50	0.04	0.04	-0.50	0.04	0.04
		Trad. NCC	0.50	0.06	0.06 (1.36)	-0.50	0.06	0.06 (1.36)
		Samuelsen	0.50	0.06	0.06 (1.27)	-0.50	0.06	0.06 (1.27)
	551	Scheike	0.47	0.05	0.05 (1.23)	-0.47	0.05	0.05 (1.21)
0.016	965	Cohort	0.50	0.03	0.03	-0.50	0.03	0.03
		Trad. NCC	0.50	0.05	0.05 (1.39)	-0.50	0.04	0.04 (1.34)
		Samuelsen	0.50	0.04	0.04 (1.24)	-0.50	0.04	0.04 (1.19)
	991	Scheike	0.46	0.04	0.04 (1.15)	-0.46	0.04	0.04 (1.15)
0.004	459	Cohort	1.00	0.05	0.05	-1.01	0.05	0.05
		Trad. NCC	1.01	0.09	0.09 (1.86)	-1.01	0.09	0.09 (1.84)
		Samuelsen	1.01	0.07	0.07 (1.40)	-1.01	0.07	0.07 (1.43)
	491	Scheike	0.94	0.06	0.07 (1.32)	-0.94	0.06	0.07 (1.27)
0.008	782	Cohort	1.00	0.04	0.04	-1.00	0.04	0.04
		Trad. NCC	1.00	0.07	0.07 (1.77)	-1.00	0.07	0.07 (1.70)
		Samuelsen	1.00	0.05	0.05 (1.26)	-1.00	0.05	0.05 (1.23)
	800	Scheike	0.94	0.05	0.05 (1.29)	-0.93	0.05	0.05 (1.28)
0.016	1198	Cohort	1.00	0.03	0.03	-1.00	0.03	0.03
		Trad. NCC	1.00	0.06	0.06 (1.67)	-1.00	0.05	0.06 (1.72)
		Samuelsen	1.01	0.04	0.04 (1.15)	-1.00	0.04	0.04 (1.16)
	1248	Scheike	0.95	0.04	0.04 (1.28)	-0.94	0.04	0.04 (1.21)

Number in brackets are $\sqrt{efficiency}$, standard deviation divided by cohort standard deviation

Table 3.9: A subset of Table 1 in [23]

		Covariate 1				Covariate 2			
	Av.	Mean	Mean	Emp.		Mean	Mean	Emp.	
$\alpha_0(t)$	cases	est.	est. sd	sd	$\sqrt{\text{Eff.}}$	est.	est. sd	sd	$\sqrt{\text{Eff.}}$
0.004	234	0.00	0.08	0.08	1.21	0.00	0.08	0.07	1.23
0.008	452	0.00	0.06	0.06	1.16	0.00	0.06	0.06	1.24
0.016	852	0.00	0.05	0.04	1.30	0.00	0.04	0.04	1.26
0.004	288	0.52	0.08	0.08	1.52	-0.52	0.07	0.08	1.44
0.008	550	0.54	0.06	0.06	1.31	-0.54	0.06	0.06	1.35
0.016	986	0.55	0.04	0.04	1.21	-0.55	0.04	0.04	1.28
0.004	196	1.05	0.10	0.10	1.25	-1.05	0.09	0.10	1.25
0.008	475	1.04	0.06	0.06	1.26	-1.05	0.07	0.06	1.32
0.016	1232	1.03	0.04	0.04	1.21	-1.03	0.03	0.04	1.12

We see that the estimates of $\beta \neq 0$ is a bit skewed when estimated with Scheike's method. Table 3.9 shows that when the true value of β is (-0.5,0.5), the absolute value of the estimate is too high and the same goes when the true value of β is (-1,1). The strange thing is that when we use Scheike's code, the bias goes the other way.

The empirical variance and the EM-based variance is in good agreement here and the standard deviation of Scheike's likelihood and the weighted likelihoods is mainly equal to the second decimal place, but with a slight advantage of Scheike's likelihood in some cases. But when we look at the square root of the efficiency we see that Scheike's likelihood has somewhat smaller standard deviation at least for $\beta = 0$ and $\beta = (-0.5, 0.5)$.

Another thing is that there is something strange with Scheike and Juul's simulation when $\beta = 1$ and baseline equals 0.004 and 0.008. The number of cases we get in our simulation experiment and the number of cases they reported is not comparable. While we get $(491, 800)$ they report $(196, 475)$. 196 is less cases that they, and we, got when $\beta = 0$. It's obvious that when β increase the number of cases will increase. But apart from that the numbers are comparable, therefore it's probably just a typing error or perhaps they have used a slightly different simulation setup than described.

3.7 Simulation problems

Traditional nested case-control

I have encountered some different simulation problems, the first problem is only a problem for the traditional nested case-control analysis, where the optimization of the likelihood doesn't converge. In some of the simulations we have only one binary covariate, and there is only one control per case. If there are zero strata where the case and the control have opposite covariate values, then the likelihood will be flat (equal one half), and a maximum doesn't exist. If the likelihood contribution from one individual is

$$L_i = \frac{\exp(\beta x_i)}{\exp(\beta x_i) + \exp(\beta x_{i'})}$$

where $x_{i'}$ is the covariate for the control. Then

$$L_i = \begin{cases} \frac{1}{2} & \text{if } \; x_i = x_{i'} = 0 \\ \frac{\exp(\beta)}{2\exp(\beta)} = \frac{1}{2} & \text{if } \; x_i = x_{i'} = 1 \end{cases}$$

Empty risk set

The other problem is a problem that also can happen in a real data set, namely a problem concerning an empty risk set. For each event time, one sample m controls from the risk set at that time, but if the risk set is empty, all individuals are either censored or dead, then there are nobody to sample. This will mainly happen if the censoring distribution has a "heavy tail", since it is then possible that the longest survival time is bigger than the longest censoring time.

Our "solution" to this and the previous problem, is to exclude those simulations where it happened. This may seem non-optimal and the easies way out, but as long as the problems are as rare as they are, it wouldn't be a lot to gain to try to use more sophisticated methods in order to fix the problems.

Failing calibration

The calibration may sometimes fail to match the population totals. One way of dealing with this is to force convergence and then exclude the simulations where it happened.

AFT-model

The `survReg` routine in **R** that is used to fit the AFT-model fail from time to time, we have therefor used weighted poisson regression, where GAM-weights are used, instead. It is possible to use Poisson regression since when the survival times are exponential distributed then the AFT-likelihood is on the same form as a Poisson-likelihood.

Chapter 4

Application to data

So far in this thesis we have only tried out the different methods on simulated data with only one or at most two covariates. This is a natural start when we want to compare methods, because the data are easy to handle and since the "true" parameter values are determined on before hand, it's easy to check whether the results are likely or not. On the other hand these simulated settings are too simple to be of any real interest. Therefore we have also tried out the different methods on a real data set.

4.1 Data

The data set [21] consists of all children born in Norway between 1967 and 1989 who survived their first year and had a gestational age ≥ 16 weeks. The cohort consisted of 1,270,016 subjects and the children were followed to death or to the end of 1991. This can be considered as a situation with censoring at a fixed time, in the sense of not being any randomness in the censoring, since there is no loss of follow-up except for a few subjects who moved abroad. For reasons explained later we only used follow-up time ≤ 10 years. We also excluded subjects with missing covariates or covariates that were obviously wrongly coded, there were 83,361 such subjects and the number of individuals left in the cohort was then 1,186,655. Because of time considerations, we had to reduce the cohort further, therefore we only used first born boys, and ended up with a cohort of size 254,572.

The original use of these data was analysis on the entire cohort where they looked at how gestational age and other covariates influenced childhood mortality. The analysis was cause specific, but we are going to limit ourselves to two causes; death of cancer and death of all other causes. Since the topic of this thesis has been nested case-control studies with multiple outcomes, we are going to do synthetic case-control studies where we take the cohort to be the 254,572 subjects.

Childhood mortality is luckily small in Norway, out of 254,572 subjects

868 died, 125 of the children died of cancer and 743 of the children died of other causes. If we choose $m = 1$ control per case, then the subcohort belonging to endpoint 2 would consist of 1,486 subjects at start of study while the subcohort belonging to endpoint 1 would only consist of 250 subjects. But when we also use cases and controls from endpoint 2 in the analysis of endpoint 1 the subcohort increases to 1,736 which result in about 13 controls per case on average.

Figure 4.1 displays the baseline hazards for endpoint 1 and 2, both with follow-up time until 10 years (bottom) and follow-up until death or censoring (top). We see from this that with follow-up until death or censoring, a Weibull baseline wouldn't fit very good especially for endpoint 2. If we on the other hand only use follow-up time until the age of 10, then we see that it looks much more reasonable to use a Weibull baseline. With cancer endpoint it looks like the baseline hazard is almost constant for a long time and then it starts to decrease a bit, while for endpoint 2 baseline hazard is decreasing all the time.

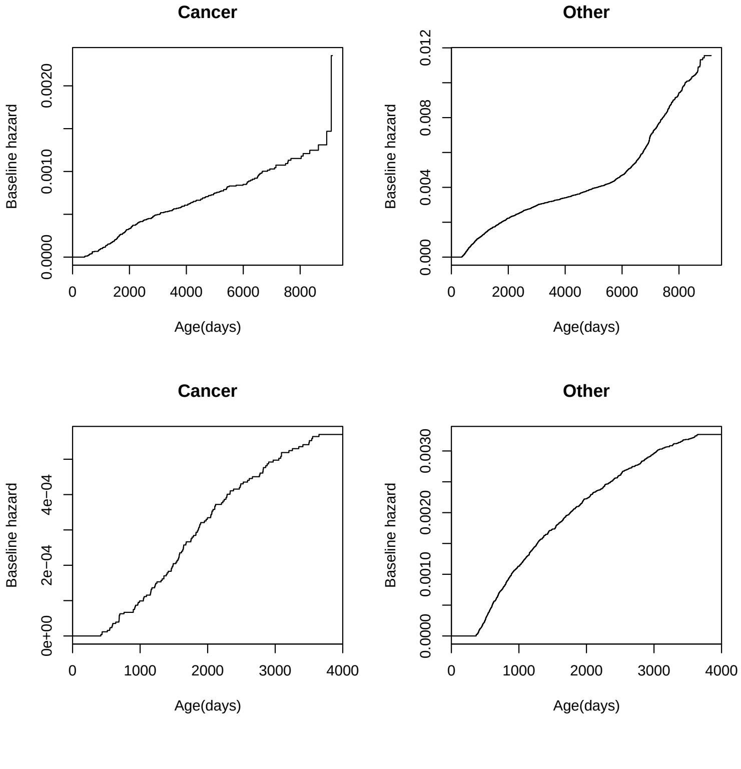

Figure 4.1: Baseline hazard for cancer deaths and deaths from all other causes, top: follow-up until death or censoring, bottom: follow-up until 10 years old.

4.2 Methods

The methods we have tried out on simulated data are weighted partial likelihoods, with four different weights; Samuelsen's, estimated with logistic GAM, estimated with logistic regression and local averaging. Since logistic regression is a special case of logistic GAM and the local averaging method consistently had higher variance than the other methods, only GAM and Samuelsen's weights are used. Further we have also tried out the full likelihood of Saarela et al. and the calibration method of Breslow.

The covariates included are:
x_1 = gestational age in days
x_2 = birth weight in kilos

4.2.1 Weighted partial likelihoods

I have chosen to only try out Samuelsen's Kaplan-Meier type of weights

$$p_j = \begin{cases} 1 & \text{cases} \\ 1 - \prod_{t_i < t_j} \left\{ 1 - \frac{m-1}{n(t_i)-1} \right\} & \text{controls} \end{cases}$$

and the weights estimated with logistic GAM,

$$\mathbf{E}(V_i|\tilde{T}_i) = \frac{\exp(\alpha + f(\tilde{T}_i))}{1 + \exp(\alpha + f(\tilde{T}_i))}$$

The estimation is then exactly as it was with the simulated data in chapter 3.

4.2.2 Full likelihood

Two partially unknown binary covariates

The first analysis is based on both gestational age x_1 and birth weight x_2 being unknown for individuals outside the subcohort and both of them are used as binary covariates.

$$Z_1 = \begin{cases} 0 & \text{if } x_1 \leq 37 \text{ weeks} \\ 1 & \text{else} \end{cases}$$

$$Z_2 = \begin{cases} 0 & \text{if } x_2 \leq 3 \text{ kg} \\ 1 & \text{else} \end{cases}$$

The joint distribution of Z_1 and Z_2 is then

$$p(Z_1, Z_2|\mu) = \mu_1^{Z_1 Z_2} \mu_2^{(1-Z_1)Z_2} \mu_3^{Z_1(1-Z_2)} (1 - \mu_1 - \mu_2 - \mu_3)^{(1-Z_1)(1-Z_2)}$$

where

$$\begin{aligned}\mu_1 &= P(Z_1 = 1, Z_2 = 1)\\ \mu_2 &= P(Z_1 = 0, Z_2 = 1)\\ \mu_3 &= P(Z_1 = 1, Z_2 = 0),\end{aligned}$$

is the probability of the different combinations of Z_1 and Z_2.

One binary partially known covariate and one fully observed numerical covariate

The second thing I have tried out is gestational age, the known x, being numerical and known for the entire cohort, while birth weight, Z is binary and only partially known. Then the distribution of Z given x is $p(Z_i|x_i;\mu) = \mu^{Z_i}(1-\mu)^{1-Z_i}$, but it needs some special attention since it is natural to believe that birth weight depend on gestational age. The natural way of model this is through a logistic regression where we have chosen a probit-link

$$g(\mu) = \Phi^{-1}(\mu) = \xi_0 + \xi_1 x$$

where Φ is the cumulative standard Normal probability function. A probit-model is a natural choice since the underlying birth weight is approximately normally distributed.

One numerically covariate

If we now use birth weight in the original coding in kilo, the full likelihood looks like

$$\begin{aligned}L(\beta,\mu) \propto &\prod_{i\in\mathcal{O}} p(T_i, E_i|Z_i, x_i;\beta)p(Z_i|x_i;\mu)\\ &\times \prod_{i\in\mathcal{C}\setminus\mathcal{O}} \int_{z_i} p(T_i, E_i|z_i, x_i;\beta)p(Z_i = z_i|x_i;\mu)dz_i\end{aligned}$$

and if we model Z as

$$Z = \tau_0 + \tau_1 x + \varepsilon, \quad \varepsilon \sim N(0,\sigma^2)$$

which means that $Z \sim N(\tau_0 + \tau_1 x, \sigma^2)$ then the last part of the likelihood above

$$\prod_{i \in \mathcal{C} \setminus \mathcal{O}} \Bigg[\int_{z_i} \Bigg(\prod_{k=1}^{K} \{ \lambda_k^{\nu_k} \nu_k t^{\nu_k - 1} \exp(\gamma_k x_i + \eta_k z_i) \} \\ \times \exp \left\{ \sum_{k=1}^{K} -(\lambda_k t)^{\nu_k} \exp(\gamma_k x_i + \eta_k z_i) \right\} \\ \times \frac{1}{\sqrt{2\pi}\sigma} \exp \left\{ -\frac{1}{2\sigma^2} (z_i - \tau_0 - \tau_1 x)^2 \right\} \Bigg) dz_i \Bigg] \tag{4.1}$$

includes a quite complicated integral. It might be that it's possible to solve it analytically, but in order to try out the Monte Carlo approach I haven't tried that. By using Monte Carlo integration on (4.1) it becomes

$$\prod_{i \in \mathcal{C} \setminus \mathcal{O}} \Bigg[\frac{1}{M} \sum_{z_{i,m}} \Bigg(\prod_{k=1}^{K} \{ \lambda_k^{\nu_k} \nu_k t^{\nu_k - 1} \exp(\gamma_k x_i + \eta_k z_{i,m}) \} \\ \times \exp \left\{ \sum_{k=1}^{K} -(\lambda_k t)^{\nu_k} \exp(\gamma_k x_i + \eta_k z_{i,m}) \right\} \Bigg) \Bigg] \tag{4.2}$$

Where $z_{i,m} \sim N(\tau_0 + \tau_1 x_i, \sigma^2)$ and $m = 1 \ldots M$, we have chosen $M = 100$.

Since $\mu = (\tau_0, \tau_1, \sigma)$ in $p(Z_i = z_i | x_i; \mu)$ are parameters that needs to be estimated as well, we run into a problem. One solution is importance sampling; instead of drawing $z_{i,m}$ from $N(\tau_0 + \tau_1 x_i, \sigma^2)$ we draw it from $N(\tau_0' + \tau_1' x_i, \sigma'^2)$ where τ_0', τ_1' and σ' are decided on beforehand. And we correct for drawing from the wrong distribution by multiplying (4.2) by the importance weights

$$\frac{p(Z_i = z_i | x_i; \tau_0, \tau_1, \sigma)}{p(Z_i = z_i | x_i; \tau_0', \tau_1', \sigma')}$$

When we look at the importance weights it's obvious that τ_0', τ_1' and σ' should be as close as possible to τ_0, τ_1 and σ. In order to get that we did a linear regression on the full data and put $\tau_0' = \hat{\tau}_{0,\text{MLE}}, \tau_1' = \hat{\tau}_{1,\text{MLE}}$ and $\sigma' = \hat{\sigma}_{\text{MLE}}$. (In a real situation the full data wouldn't be known, then it's possible to do the regression only on the NCC-data.)

Problems

The likelihood is hard to optimize, the optimization algorithm requires good starting values and you have to be careful with parameters that can't be negative. I found out that in order to make the program more robust it is smart to take the logarithm of the starting values for those parameters and then exponentiate them again in the likelihood program. The coefficients from the Cox-regression is a good choice of starting values, but the transformed coefficients from the accelerated failure time model might be even better since the estimates from the full likelihood on the full cohort and the AFT-estimates should be equal.

Even with a very strict convergence criterion the estimates doesn't turn out exactly the same with different starting values, which is a bit worrying. With a strict convergence criterion the variances are very similar with different starting values, which is reassuring. The empirical variances on the other hand gets smaller as the strictness of the convergence criterion increase. This is in a way natural, but the empirical variance also differ with starting values, it's much smaller when AFT-starting values are chosen. This lead me to the thought that the empirical variance from the full likelihood is not comparable to the empirical variance of the WPL models. With fully observed covariates the empirical variance should be close to 0 because we use the full data every time. With partially observed covariates it is somewhat different, with WPL we sample different controls each time, and there will naturally be variation in the estimates. With the full likelihood we also sample controls, but we model the covariates that are not observed. If the distribution we have chosen describe the non-observed covariates in a good way, I think the variation due to the sampling then should be minimal.

4.2.3 Calibration

The calibration requires some information to be fully known, since we have to predict the partially observed covariate on the background on information known for everybody, it is only done in the setting where birth weight is known for everybody, while gestational age is only known for sampled individuals. Then the calibration is done as described in section 2.8.3 and the calibration function used is unbounded raking. As you will see from Table 4.3 and 4.4, this doesn't work very good.

4.3 Results

In order to structure the discussions a bit, I will also here use a list of discussion points:

1. Saarela's likelihood on cohort data compared to Cox regression on cohort data
2. Estimates, bias, $\frac{\text{bias}}{(SE_0^2+S^2)^{\frac{1}{2}}} > \frac{1}{3}$ is considered to be serious
3. Comparison of standard errors, efficiency
4. Standard error compared to $(SE_0^2 + S^2)^{\frac{1}{2}}$

Here SE_0 is the standard error from Cox-regression on the full cohort and S is the empirical variance. From now on we are going to call $(SE_0^2 + S^2)^{\frac{1}{2}}$ simulation based standard deviation. The reason for why this quantity is interesting is that the variance of the estimates from the weighted partial likelihoods can be divided into two parts (see B.2), one reflecting the variance we would have if all covariates for the hole cohort was known, and the other part reflecting the sampling variability. It is natural to assume that the same apply to the full likelihood and the traditional NCC likelihood. But with the full likelihood it is natural to use SE_0' which is standard error from Saarela's likelihood on the entire cohort. The estimated variance should then be approximately equal to the simulation based standard deviation if the estimators are right, which means that the empirical variances show how much extra variance that is added due to the sampling.

4.3.1 Two partially unknown binary covariates

Table 4.1: Comparison cancer endpoint, two unknown binary covariates

	Method	Model/weights	$\hat{\beta}_1$	$\hat{\beta}_2$
Estimate	Cohort	Cox	0.6812	0.3078
	Cohort MLE	Weibull	0.6772	0.3213
	Trad. NCC	Strat. Cox	0.6280	0.2698
	WPL	Samuelsen	0.6809	0.2923
	WPL	GAM	0.6820	0.2918
	MLE	Weibull	0.6976	0.3032
Standard error	Cohort	Cox	0.6067	0.3042
	Cohort MLE	Weibull	0.6061	0.3041
	Trad. NCC	Strat. Cox	0.7787	0.4195
	WPL	Samuelsen	0.6276	0.3212
	WPL	GAM	0.6277	0.3213
	MLE	Weibull	0.6218	0.3181
Empirical standard deviation	Trad.NCC	Strat.Cox	0.4986	0.2656
	WPL	Samuelsen	0.1576	0.1005
	WPL	GAM	0.1565	0.1011
	MLE	Weibull	0.1488	0.0977
Efficiency	Trad.NCC	Strat.Cox	0.6071	0.5257
	WPL	Samuelsen	0.9345	0.8969
	WPL	GAM	0.9343	0.8966
	MLE	Weibull	0.9519	0.9145
$(SE_0^2 + S^2)^{\frac{1}{2}}$	Trad.NCC	Strat.Cox	0.7853	0.4038
	WPL	Samuelsen	0.6269	0.3204
	WPL	GAM	0.6266	0.3205
$(SE_{0'}^2 + S^2)^{\frac{1}{2}}$	MLE	Weibull	0.6241	0.3194

β_1 - gestational age, β_2 - birth weight, MLE - Maximum likelihood
WPL - weighted partial likelihood, Cohort MLE - Saarela's likelihood on cohortdata
SE_0 - standard error from Cox on cohort, S - empirical standard deviation
$SE_{0'}$ - standard error from MLE on cohort

Here both gestational age and birth weight are used as binary covariates, which means that $Z_1 = 1$ if gestational age is above 37 weeks and $Z_2 = 1$ if birth weight is above 3 kg. Table 4.1 and 4.2 shows the result of sampling controls and doing the analysis 200 times.

1. There are some differences between the Cox-regression on the full cohort and Saarela's likelihood on the full cohort. This is of course natural since they are based on a bit different assumptions, but it makes it difficult to talk about bias in the estimates from the full likelihood, because it is not quite obvious what it should be compared to. But even though the estimates aren't exactly the same they are very alike, and the differences wouldn't make any differences in practice. Also the standard deviations of the two models are almost the same, with other deaths endpoint they actually are the same to at least the third decimal place.

Table 4.2: Comparison other deaths, two unknown binary covariates

	Method	Model/weights	$\hat{\beta}_1$	$\hat{\beta}_2$
Estimate	Cohort	Cox	-0.1999	-0.3973
	Cohort MLE	Weibull	-0.2042	-0.3866
	Trad. NCC	Strat. Cox	-0.2275	-0.4117
	WPL	Samuelsen	-0.1970	-0.4128
	WPL	GAM	-0.1964	-0.4134
	MLE	Weibull	-0.2234	-0.4111
Standard error	Cohort	Cox	0.1459	0.0970
	Cohort MLE	Weibull	0.1459	0.0970
	Trad. NCC	Strat. Cox	0.2265	0.1460
	WPL	Samuelsen	0.2235	0.1440
	WPL	GAM	0.2237	0.1441
	MLE	Weibull	0.2150	0.1388
Empirical standard deviation	Trad.NCC	Strat.Cox	0.1666	0.1125
	WPL	Samuelsen	0.1552	0.1017
	WPL	GAM	0.1538	0.1022
	MLE	Weibull	0.1501	0.0975
Efficiency	Trad.NCC	Strat.Cox	0.4151	0.4414
	WPL	Samuelsen	0.4265	0.4541
	WPL	GAM	0.4256	0.4532
	MLE	Weibull	0.4609	0.4885
$(SE_0^2+S^2)^{\frac{1}{2}}$	Trad.NCC	Strat.Cox	0.2215	0.1485
	WPL	Samuelsen	0.2130	0.1406
	WPL	GAM	0.2120	0.1410
$(SE_{0'}^2+S^2)^{\frac{1}{2}}$	MLE	Weibull	0.2093	0.1375

β_1 - gestational age, β_2 - birth weight, MLE - Maximum likelihood
WPL - weighted partial likelihood, Cohort MLE - Saarela's likelihood on cohortdata
SE_0 - standard error from Cox on cohort, S - empirical standard deviation
$SE_{0'}$ - standard error from MLE on cohort

2. If we first look at cancer endpoint, all estimates are very alike, except the estimates from the traditional NCC which are a bit smaller than

the rest. But this bias isn't really important compared to the standard error.

If we then look at other deaths endpoint, β_1 from the full likelihood and the traditional NCC is very similar, while the WPL-estimates are a bit smaller than the rest. With β_2 there is a difference between the estimates from the full cohort and the NCC estimates. But again these differences are not important compared to the standard deviations.

3. The standard errors are as expected, with cancer endpoint there are very little to gain by using the full likelihood because both covariates are only partially observed. But by using WPL instead of traditional NCC the efficiency increase with over 50% and 70% for β_1 and β_2 respectively, this is due to the high number of controls. With other deaths endpoint on the other hand, there are very little to gain by choosing WPL instead of the traditional NCC and the full likelihood only improve the efficiency a little bit.

4. The last rows in Table 4.1 and 4.2 shows the simulation based standard deviation. First of all, the empirical variances are only based on 200 estimates, which probably is a bit too few. Anyway it looks quite reasonable, perhaps we see a tendency that WPL overestimate the variance a bit, at least with other deaths endpoint. But this is only minor differences, the overall impression is that the variances is estimated very accurate.

4.3.2 One partially unknown and one fully known covariate

Table 4.3 and 4.4[1] shows the result from the analysis when we again are using gestational age and birth weight as covariates. Gestational age is used in days and is assumed to be known for everybody, while birth weight only is known for cases and controls and is therefore used as a binary covariate. The estimates are the average of 200 analysis where new controls are sampled each time. β_1 is the estimate of gestational age, while β_2 correspond to the birth weight indicator.

1. It is reassuring to see that the estimates from the full likelihood and the Cox-regression on the entire cohort are very similar also in this case. The same can be said about the standard deviation, but it is slightly smaller with the full likelihood, than with Cox-regression.

2. First of all, we see that the estimates obtained with calibrated weights, at least for the partially observed covariate is quite different from the

[1] The efficiency of the calibration method isn't calculated. It doesn't really make sence to look at it because of the biased estimates and because the estimates of the standard deviations are too small.

Table 4.3: Comparison cancer endpoint, one partially unknown binary covariate and one fully known numerical covariate

	Method	Model/weights	$\hat{\beta}_1$	$\hat{\beta}_2$
Estimate	Cohort	Cox	-0.0033	0.4813
	Cohort MLE	Weibull	-0.0035	0.4960
	Trad. NCC	Strat. Cox	-0.0036	0.4342
	WPL	Samuelsen	-0.0033	0.4652
	WPL	GAM	-0.0032	0.4645
	WPL	Calibration	-0.0030	0.5905
	MLE	Weibull	-0.0035	0.4969
Standard deviation	Cohort	Cox	0.0070	0.3141
	Cohort MLE	Weibull	0.0067	0.3136
	Trad. NCC	Strat. Cox	0.0106	0.4256
	WPL	Samuelsen	0.0073	0.3343
	WPL	GAM	0.0073	0.3343
	WPL	Calibration	0.0031	0.3123
	MLE	Weibull	0.0069	0.3304
Empirical standard deviation	Trad.NCC	Strat.Cox	0.0078	0.2590
	WPL	Samuelsen	0.0027	0.1111
	WPL	GAM	0.0027	0.1116
	WPL	Calibration	0.0019	0.0740
	MLE	Weibull	$8.0 \cdot 10^{-5}$	0.0055
Efficiency	Trad.NCC	Strat.Cox	0.4417	0.5445
	WPL	Samuelsen	0.9346	0.8827
	WPL	GAM	0.9340	0.8823
	WPL	Calibration	—	—
	MLE	Weibull	1.0302	0.9037
$(SE_0^2 + S^2)^{\frac{1}{2}}$	Trad.NCC	Strat.Cox	0.0105	0.4071
	WPL	Samuelsen	0.0075	0.3331
	WPL	GAM	0.0075	0.3333
	WPL	Calibration	0.0073	0.3227
$(SE_{0'}^2 + S^2)^{\frac{1}{2}}$	MLE	Weibull	0.0067	0.3137

β_1 - gestational age, β_2 - birth weight, MLE - Maximum likelihood
WPL - weighted partial likelihood, Cohort MLE - Saarela's likelihood on cohortdata
SE_0 - standard error from Cox on cohort, S - empirical standard deviation
$SE_{0'}$ - standard error from MLE on cohort

rest of the estimates. Bias divided by standard deviation is 0.338 and 1.127 for cancer and other deaths endpoint respectively. This means that the estimate for the partially observed covariate for cancer endpoint is borderline biased, while for other deaths endpoint it is seriously biased.

If we then look at the other parameter estimates we see that both birth weight and gestational age has opposite effects on death from cancer and death of other causes. We also see that WPL adds some bias to β_2 for both endpoints, while Saarela's likelihood is in better agreement with the cohort analysis. On the other hand, β_1 is estimated very accurate with the WPL-models while Saarela's likelihood result in a bit different estimates, especially for other deaths endpoint. If this difference was only due to the parametric specification of baseline, then the estimates should be similar to the estimate from the full likelihood on the full data, but this is not the case for other deaths endpoint.

The last thing to notice is that while Table 4.3 and 4.4 showed that gestational age and birth weight had opposite effects of each other and opposite effect on cancer deaths and death of other causes, both coefficients were positive for cancer deaths and negative for death of other causes when both gestational age and birth weight was used as binary covariates. This has probably to do with the fact that gestational age is not significant, and there are probably some interactions between gestational age and birth weight that are not included, that can influence the estimates.

Table 4.4: Comparison other deaths, one partially unknown binary covariate and one fully known numerical covariate

	Method	Model/weights	$\hat{\beta}_1$	$\hat{\beta}_2$
Estimate	Cohort	Cox	$2.9 \cdot 10^{-4}$	-0.4513
	Cohort MLE	Weibull	$3.6 \cdot 10^{-5}$	-0.4380
	Trad. NCC	Strat. Cox	$1.9 \cdot 10^{-4}$	-0.4682
	WPL	Samuelsen	$2.7 \cdot 10^{-4}$	-0.4698
	WPL	GAM	$2.7 \cdot 10^{-4}$	-0.4703
	WPL	Calibration	$-1.1 \cdot 10^{-4}$	-0.3186
	MLE	Weibull	$3.2 \cdot 10^{-8}$	-0.4406
Standard error	Cohort	Cox	0.0027	0.0977
	Cohort MLE	Weibull	0.0026	0.0976
	Trad. NCC	Strat. Cox	0.0038	0.1489
	WPL	Samuelsen	0.0041	0.1440
	WPL	GAM	0.0041	0.1442
	WPL	Calibration	0.0019	0.0937
	MLE	Weibull	0.0028	0.1402
Empirical standard deviation	Trad.NCC	Strat.Cox	0.0026	0.1257
	WPL	Samuelsen	0.0026	0.1106
	WPL	GAM	0.0027	0.1112
	WPL	Calibration	0.0016	0.0657
	MLE	Weibull	$7.5 \cdot 10^{-5}$	0.0140
Efficiency	Trad.NCC	Strat.Cox	0.5087	0.4304
	WPL	Samuelsen	0.4397	0.4601
	WPL	GAM	0.4388	0.4591
	WPL	Calibration	—	—
	MLE	Weibull	0.9283	0.4859
$(SE_0^2 + S^2)^{\frac{1}{2}}$	Trad.NCC	Strat.Cox	0.0037	0.1592
	WPL	Samuelsen	0.0038	0.1476
	WPL	GAM	0.0038	0.1485
	WPL	Calibration	0.0031	0.1177
$(SE_{0'}^2 + S^2)^{\frac{1}{2}}$	MLE	Weibull	0.0027	0.0987

β_1 - gestational age, β_2 - birth weight, MLE - Maximum likelihood
WPL - weighted partial likelihood, Cohort MLE - Saarela's likelihood on cohortdata
SE_0 - standard error from Cox on cohort, S - empirical standard deviation
$SE_{0'}$ - standard error from MLE on cohort

3. The first thing to notice about the standard deviations is that the variance of β_1 with cancer endpoint is lower with Saarela's likelihood than it is with Cox-regression on the hole cohort. This can seem a bit strange at first since the cohort analysis use the hole data set while MLE is done in nested case-control setting, but there are two things to remember. Firstly gestational age is known for the entire cohort, and Saarela's likelihood utilize this, secondly it might be that there are some information in the baseline. Cox-regression disregard the baseline, while with Saarela's likelihood we model it, but the differences are very small.

   The second thing to notice is that the standard deviation from the calibration method of the fully observed covariate is very small. Actually it is much smaller than Cox regression on the full cohort, this shouldn't happen and it has to be something wrong with the variance estimation.

   Further we see that the variances from the other WPL models with cancer endpoint are only marginally bigger than the variances from the full cohort, this is probably due to the big number of controls, roughly about 13 per case. For β_2 the efficiency is a bit lower, but the differences between WPL and full likelihood is still very small. With other deaths on the other hand, the traditional nested case-control analysis is slightly more efficient than WPL when estimating β_1, but the traditional NCC estimate is also a bit smaller than estimates from WPL, which is probably the reason for that. The full likelihood is almost as efficient as the Cox-regression on the cohort. With β_2 it is a different story, all methods have low efficiency, but with a slight advantage for Saarela's likelihood.

4. Again empirical standard deviation should reflect the extra variance added to the cohort variance and the simulation based standard deviation should be about the same as the (robust) model based standard deviation. We see that this is not the case with the calibrated weights, the model based standard deviation is much smaller. We also see that if we can trust the simulation based standard deviation as an estimate of the standard deviation then the calibration method actually increase the efficiency a bit compared to the other WPL models, at least with other deaths endpoint, but this isn't really that interesting since the estimates are as biased as they are. Apart from with the calibration model it seems that the numbers are in quite good agreement, perhaps except for β_2 from the full likelihood with other deaths endpoint, where there is a difference.

From the discussion above we see that the calibration method isn't really working that well. The estimates are skewed, especially those connected to the partially observed covariate and the standard deviations are too small. We can see this both from the fact that they are smaller than the standard deviation from Cox-regression on the full cohort, but also from the fact that the simulation based standard deviations are (much) bigger than the estimated standard deviations. We also tried to use $\tilde{T}$ in addition to the fully observed covariate as explanatory variable in the regression used to predict the partially observed covariate (results now shown), but this didn't help anything either.

One possible reason for why the calibration approach doesn't work that well is that it's designed for a case-cohort setting and it is not directly transferable to a nested case-control setting with multiple outcomes. Another thing is that the in the `survey` package, that is used to do this calibration, there are a lot of options and choices, it might be that we are not using the methods quite right in the sense of now being in a NCC-situation with multiple outcomes.

4.3.3 Monte Carlo approach

In order to try out the Monte Carlo approximation to the likelihood we now have one numerical partially observed covariate, birth weight in kilo. We have only one covariate in order to try it out in a very simple situation. This means that Z no longer is modeled through a linear regression, but rather as $N(\tau, \sigma^2)$. The numbers are based on sampling controls 50 times, this is too few, but it takes about 2 and a half hour to optimize the full likelihood once, and it took about a week to run the hole program. The result can be found in Table 4.5, where β_1 is the estimate of birth weight with cancer endpoint while β_2 is the estimate corresponding to other deaths endpoint.

We see that the estimates are in quite good agreement, perhaps β_2 from Saarela's likelihood is a bit different from the other estimates, but compared to the standard deviation this is not important. We further see that the standard deviation of β_1 with Saarela's likelihood is actually smaller than the same likelihood on the full cohort, this is probably due to the fact that the standard deviation doesn't take into account the extra uncertainty in the Monte Carlo approximation. If we look at $(SE_{0'}^2 + S^2)^{\frac{1}{2}}$, we see that it is somewhat bigger than the cohort variance, but I have a feeling that the standard deviation is even bigger. This is because so far we have seen that the full likelihood is only marginally better than WPL models when we only have partially observed covariates and therefore it is unlikely that the variance is so close to the variance from Cox regression on the full cohort now.

Table 4.5: One partially observed numerical covariate

	Method	Model/weights	$\hat{\beta}_1$	$\hat{\beta}_2$
Estimate	Cohort	Cox	0.2052	-0.4294
	Cohort MLE	Weibull	0.2192	-0.4224
	Trad. NCC	Strat. Cox	0.2215	-0.4061
	WPL	Samuelsen	0.2148	-0.4282
	WPL	GAM	0.2142	-0.4283
	MLE	Weibull	0.1928	-0.4572
Standard error	Cohort	Cox	0.1712	0.0643
	Cohort MLE	Weibull	0.1718	0.0646
	Trad. NCC	Strat. Cox	0.2534	0.0956
	WPL	Samuelsen	0.1763	0.0977
	WPL	GAM	0.1763	0.0978
	MLE	Weibull	0.1668	0.0684
Empirical standard deviation	Trad.NCC	Strat.Cox	0.1920	0.0697
	WPL	Samuelsen	0.0657	0.0684
	WPL	GAM	0.0642	0.0673
	MLE	Weibull	$4.1 \cdot 10^{-4}$	$3.5 \cdot 10^{-4}$
Efficiency	Trad.NCC	Strat.Cox	0.4563	0.4527
	WPL	Samuelsen	0.9432	0.4330
	WPL	GAM	0.9435	0.4321
	MLE	Weibull	1.0533	0.8847
$(SE_0^2 + S^2)^{\frac{1}{2}}$	Trad.NCC	Strat.Cox	0.2572	0.0948
	WPL	Samuelsen	0.1834	0.0939
	WPL	GAM	0.1828	0.0931
$(SE_{0'}^2 + S^2)^{\frac{1}{2}}$	MLE	Weibull	0.1719	0.0646

β_1 - cancer endpoint, β_2 - other deaths endpoint
MLE - Maximum likelihood, WPL - weighted partial likelihood
Cohort MLE - Saarela's likelihood on cohort data
SE_0 - standard error from Cox on cohort, S - empirical standard deviation
$SE_{0'}$ - standard error from MLE on cohort, number in brackets - $(SE_0/S)^2$

4.3.4 Summing up

We have seen much of the same things we saw in the simulations; there are very little to gain by using Saarela's likelihood in stead of WPL when both covariates are only partially known. When we went a bit further and let one covariate being known for the entire cohort, we got efficiency improvements with Saarela's likelihood for the fully observed covariate, but not for the partially observed covariate. We also tried out the Monte Carlo approach, this worked quite reasonable, both regarding the estimates and the size of the standard error, but the standard errors are probably somewhat bigger than reported, since the extra variance from the Monte Carlo approximation is not taken into account.

We also saw that with cancer endpoint the estimates from both WPL and the full likelihood is close to being efficient, this of course due to the high number of controls.

Another thing we noticed was that the calibration doesn't really work, the estimates of the partially observed covariate are biased and the variance estimates are too small.

Chapter 5

Discussion

5.1 Summary

The topic of this thesis has been estimators in a nested case-control design with multiple outcomes, and in particular ways of obtaining more accurate estimates in one analysis when cases and controls from another analysis can be used as additional controls. We have both tried out maximum likelihood approaches and a weighted partial likelihood approach, with different suggestions concerning the estimation of the weights. This has been done both through simulations and on real data.

The simulation showed that the choice of weights in the weighted partial likelihood is really not important, all weights except the calibrated showed very similar behaviour. Because of that only GAM and Samuelsen's weights was tried out on real data and the differences was only marginally also there. The other type of weights was the calibrated ones, the approach is really aimed at case-cohort studies with single outcomes. The setting in this thesis is therefore a bit outside the framework of Breslow et al. and we are still not certain whether this untraditional use of theory really work or how to use it to make it work the best.

The simulation also showed that the full likelihood managed to estimate fully known covariates as efficient or almost as efficient as Cox-regression on the full cohort. And if partially known covariates was correlated with fully known covariates it also managed to estimate those as efficient as the cohort analysis. The analysis on birth registry data showed much of the same tendency, but the differences between WPL and full likelihood was less pronounced due to the high number of controls. Since the full likelihood needs a specification of baseline, we also tried to find out how much of the efficiency gain that was due to that. By fitting an accelerated failure time model, that also specify baseline, to the data, we could compare the full likelihood with AFT and WPL. This showed that the biggest efficiency improvements stems from the fact that we are using data in a more efficient way. Since the efficiency gain of the full likelihood was mostly due to more efficient use of data, we also tried to use a fully known covariate in estimation of GAM-weights and Samuelsen's weights. Efficiency improvements was obtained with GAM-weights, but with Samuelsen's weights the standard error slightly increased when a fully observed covariate was included in the estimation.

Scheike's maximum likelihood approach was also tried out on simulated data. The result was that Scheike's likelihood was somewhat more efficient, but at the same time the estimates was a bit more skewed than WPL. We also wanted to try it out on birth registry data, but the program crashed and we haven't been able to figure out why.

5.2 Conclusion

The salient goal of this thesis has been to find out if and when the full likelihood of Saarela et al. is better than the partial likelihood with inverse probability weighing. The conclusion reached is that first of all, the more controls there is the less it is to gain by using more sophisticated methods, which is quite natural. Second of all, if there are covariates known for the entire cohort the full likelihood are able to estimate those almost as efficient as Cox on full cohort. Thirdly, if partly observed covariates are correlated with fully observed covariates, the full likelihood are also able to estimate those about as accurate as the Cox-regression on the full cohort. The only situation where there is nothing to gain by using Saarela's likelihood is when no covariates are known for the entire cohort. We have also found out that most of the variance reduction is due to more efficient use of data and not because of the parametric specification of baseline. There are however some drawbacks; I have experienced that the likelihood is hard to optimize and it rests on more modeling assumptions and therefore the number of parameters that needs to be estimated will increase. The extra variance connected to this is not taken into account, this is not unique for the full likelihood, but with more modeling assumptions I imagine that the "forgotten" variance will be higher.

The WPL models, at least with GAM-weights are also able to increase efficiency when fully observed covariates are included in the estimation of weights. This only apply for the fully observed covariates and dependent covariates doesn't improve the efficiency for the partly observed covariates.

When it comes to the calibration method we haven't really reached a conclusion. It works fine in the simulation experiment, but when we tried it out on the birth registry data something strange happened, the estimates was really skewed and the variance estimates was really too small. Therefore it needs more testing before we are able to say anything.

5.3 Further research

There are a number of things we would have liked to try out, but because of time limitation we haven't been able to. First, how critical are the extra assumptions in Saarela's likelihood? It would be interesting to find out how the estimates would behave if the distribution of the partially observed covariates was wrong and/or a wrong specification of baseline was assumed. It would also be nice to know how much extra variance is added to the usual variance when a Monte Carlo approximation is used in the likelihood.

Chen have stated that his local averaging method is superior compared to the usual inclusion probabilities. In our simulations we saw that Chen's method was mainly the least efficient of the WPL methods and the most obvious reason for that is that we have chosen the wrong number of partitions of the time axis. It could then be interesting to find out how much efficiency improvements Chen's weights could give by altering the partition and trying to find the optimal intervals.

Another thing we tried out was Breslow's calibration method, with simulated data it seems like it works fine, but with the birth registry data the standard deviation is really too small and the estimates are biased. Therefore more research is needed in order to generalize the theory. Another thing that perhaps could be useful is to base the regression used to predict the partially observed covariate on $\tilde{T}$, then the calibration wouldn't depend on some information being known for the entire cohort.

The last thing that could have been done is more testing with birth registry data; we could have included more covariates, both fully and partially observed and we could have included fully observed covariates in the estimation of the weights.

Appendix A

More simulation results

In chapter 3 we have only included simulation results when $\beta = 0$ and $\beta = 1$ and just the results from the multiple endpoint models and not the single endpoint models. In the simulations with two covariates we have only included the results with Samuelsens weights from the weighted likelihood models. Here are the results with one covariate when $\beta = 0.3$ and $\beta = 0.6$ and also the results when only the original controls are used. The last tables are the results with two covariates with all the four different weighting schemes included.

Table A.1: One covariate, random censoring, simulation I

			Cause 1: 10% cases				Cause 2: 3% cases			
		Model/	Mean	Mean	Emp.		Mean	Mean	Emp.	
β_1, β_2	Method	weights	est.	est. sd	sd	Efficiency	est.	est. sd	sd	Efficiency
0.00	Cohort	Cox	0.008	0.202	0.202		-0.011	0.378	0.376	
	Trad. NCC	Strat. Cox	0.004	0.288	0.290	0.480	-0.014	0.558	0.578	0.423
	WPL	Samuelsen	0.010	0.259	0.255	0.621	-0.009	0.413	0.410	0.841
				(0.277)	(0.275)	(0.534)		(0.507)	(0.516)	(0.531)
	WPL	GAM	0.010	0.260	0.256	0.616	-0.009	0.414	0.411	0.837
				(0.278)	(0.276)	(0.530)		(0.509)	(0.521)	(0.521)
	WPL	Logistic	0.010	0.248	0.256	0.616	-0.009	0.370	0.411	0.837
				(0.267)	(0.275)	(0.534)		(0.477)	(0.518)	(0.527)
	WPL	Chen	0.010	0.250	0.261	0.593	-0.009	0.369	0.414	0.825
				(0.272)	(0.282)	(0.508)		(0.471)	(0.530)	(0.503)
	MLE	Weibull	0.009	0.258	0.255	0.621	-0.009	0.411	0.406	0.858
				(0.281)	(0.279)	(0.512)		(0.527)	(0.530)	(0.503)
0.30	Cohort	Cox	0.305	0.203	0.207		0.297	0.375	0.363	
	Trad. NCC	Strat. Cox	0.312	0.291	0.299	0.479	0.318	0.554	0.566	0.411
	WPL	Samuelsen	0.307	0.259	0.259	0.639	0.300	0.409	0.407	0.795
				(0.277)	(0.279)	(0.550)		(0.499)	(0.510)	(0.507)
	WPL	GAM	0.305	0.260	0.260	0.634	0.298	0.410	0.408	0.792
				(0.279)	(0.279)	(0.550)		(0.502)	(0.513)	(0.501)
	WPL	Logistic	0.306	0.285	0.260	0.634	0.300	0.443	0.408	0.792
				(0.305)	(0.279)	(0.550)		(0.555)	(0.511)	(0.505)
	WPL	Chen	0.305	0.293	0.264	0.615	0.298	0.453	0.410	0.784
				(0.309)	(0.284)	(0.531)		(0.606)	(0.520)	(0.487)
	MLE	Weibull	0.306	0.258	0.259	0.639	0.298	0.407	0.405	0.803
				(0.282)	(0.284)	(0.531)		(0.520)	(0.527)	(0.474)

The numbers in brackets are the result if one only uses the controls sampled for that particular outcome.
WPL = weighted partial likelihood estimation, MLE = maximum likelihood estimation
Number of times the likelihood is flat for $\beta = (0, 0.3)$ *is* $(0, 0)$

Table A.2: One covariate, random censoring, simulation II

			Cause 1: 10% cases				Cause 2: 3% cases			
β_1, β_2	Method	Model/ weights	Mean est.	Mean est. sd	Emp. sd	Efficiency	Mean est.	Mean est. sd	Emp. sd	Efficiency
0.60	Cohort	Cox	0.613	0.211	0.220		0.630	0.390	0.323	
	Trad. NCC	Strat. Cox	0.635	0.302	0.296	0.552	0.634	0.572	0.585	0.305
	WPL	Samuelsen	0.632	0.266	0.270	0.664	0.639	0.425	0.396	0.766
				(0.285)	(0.270)	(0.564)		(0.514)	(0.510)	(0.401)
	WPL	GAM	0.621	0.268	0.270	0.664	0.641	0.426	0.396	0.766
				(0.287)	(0.271)	(0.659)		(0.518)	(0.519)	(0.387)
	WPL	Logistic	0.621	0.274	0.270	0.664	0.641	0.440	0.396	0.766
				(0.295)	(0.271)	(0.659)		(0.523)	(0.519)	(0.387)
	WPL	Chen	0.618	0.276	0.268	0.674	0.637	0.440	0.370	0.762
				(0.298)	(0.269)	(0.669)		(0.527)	(0.527)	(0.376)
	MLE	Weibull	0.620	0.265	0.269	0.669	0.637	0.422	0.367	0.775
				(0.289)	(0.275)	(0.640)		(0.534)	(0.550)	(0.345)
1.00	Cohort	Cox	1.015	0.226	0.227		1.051	0.422	0.495	
	Trad. NCC	Strat. Cox	1.000	0.323	0.315	0.519	1.015	0.598	0.562	0.776
	WPL	Samuelsen	1.006	0.278	0.276	0.676	1.041	0.456	0.525	0.890
				(0.297)	(0.276)	(0.676)		(0.539)	(0.559)	(0.784)
	WPL	GAM	1.011	0.280	0.280	0.657	1.046	0.457	0.528	0.879
				(0.298)	(0.278)	(0.667)		(0.541)	(0.565)	(0.768)
	WPL	Logistic	1.011	0.276	0.280	0.657	1.046	0.454	0.528	0.879
				(0.297)	(0.278)	(0.667)		(0.514)	(0.565)	(0.768)
	WPL	Chen	1.011	0.279	0.282	0.648	1.046	0.457	0.528	0.879
				(0.300)	(0.276)	(0.676)		(0.516)	(0.584)	(0.718)
	MLE	Weibull	1.004	0.277	0.274	0.686	1.041	0.452	0.524	0.892
				(0.301)	(0.278)	(0.667)		(0.555)	(0.568)	(0.759)

The numbers in brackets are the result if one only uses the controls sampled for that particular outcome.
WPL = weighted partial likelihood estimation, MLE = maximum likelihood estimation
Number of times the likelihood is flat for $\beta = (0.6, 1)$ *is* $(1, 2)$

Table A.3: Simulation, two independent covariates

				Cause 1: 10% cases				cause 2: 3% cases			
			Model/	Mean	Mean	Emp.		Mean	Mean	Emp.	
β_1, β_2	Cov.	Method	weights	est.	est. sd	sd	Eff.	est.	est. sd	sd	Eff.
0.00	z	Cohort	Cox	-0.021	0.203	0.213		-0.015	0.374	0.375	
	x			-0.006	0.348	0.355		-0.014	0.638	0.606	
	z	Trad. NCC	Strat. Cox	-0.019	0.292	0.314	0.460	0.001	0.572	0.614	0.373
	x			-0.001	0.506	0.538	0.435	0.005	0.988	1.048	0.334
	z	WPL	Samuelsen	-0.014	0.262	0.273	0.609	-0.010	0.415	0.416	0.813
	x			-0.005	0.453	0.465	0.583	-0.070	0.709	0.681	0.792
	z	WPL	GAM	-0.012	0.263	0.274	0.604	-0.008	0.415	0.417	0.809
	x			-0.004	0.455	0.470	0.571	-0.006	0.711	0.683	0.787
	z	WPL	Logistic	-0.013	0.258	0.273	0.609	-0.009	0.387	0.416	0.813
	x			-0.005	0.434	0.467	0.578	-0.006	0.646	0.683	0.787
	z	WPL	Chen	-0.014	0.262	0.279	0.583	-0.010	0.387	0.421	0.793
	x			-0.005	0.442	0.452	0.617	-0.008	0.661	0.688	0.776
	z	MLE	Weibull	-0.015	0.258	0.282	0.571	-0.008	0.407	0.410	0.837
	x			-0.006	0.348	0.352	1.017	-0.012	0.639	0.608	0.993
	z	WPL	Calibrated	-0.017	0.276	0.283	0.566	-0.011	0.430	0.424	0.782
	x			-0.003	0.373	0.381	0.868	-0.014	0.664	0.634	0.914
1.00	z	Cohort	Cox	1.023	0.227	0.227		1.038	0.423	0.435	
	x			1.004	0.352	0.363		1.002	0.649	0.675	
	z	Trad. NCC	Strat. Cox	1.064	0.337	0.341	0.443	1.082	0.662	0.675	0.415
	x			1.049	0.554	0.582	0.389	1.108	1.083	1.202	0.315
	z	WPL	Samuelsen	1.039	0.283	0.283	0.643	1.049	0.460	0.467	0.868
	x			1.032	0.473	0.497	0.533	1.032	0.730	0.763	0.783
	z	WPL	GAM	1.038	0.284	0.283	0.643	1.048	0.461	0.469	0.860
	x			1.031	0.476	0.500	0.527	1.031	0.732	0.761	0.787
	z	WPL	Logistic	1.039	0.289	0.283	0.643	1.050	0.439	0.468	0.864
	x			1.033	0.425	0.499	0.529	1.033	0.809	0.764	0.781
	z	WPL	Chen	1.039	0.290	0.287	0.626	1.049	0.439	0.470	0.857
	x			1.032	0.430	0.510	0.507	1.033	0.822	0.766	0.777
	z	MLE	Weibull	1.041	0.277	0.276	0.676	1.057	0.452	0.458	0.902
	x			1.050	0.358	0.367	0.978	1.047	0.652	0.679	0.988
	z	WPL	Calibrated	1.074	0.297	0.298	0.606	1.082	0.477	0.471	0.823
	x			1.019	0.405	0.418	0.754	1.013	0.691	0.708	0.909

WPL = weighted partial likelihood estimation, MLE = maximum likelihood estimation
Number of times the likelihood is flat for $\beta = (0, 1)$ is $(0, 15)$
x is observed for the entire cohort, z only observed for the cases and controls
Distance measure for the calibrated weights are Poisson deviance

Table A.4: Simulation, two dependent covariates

				Cause 1: 10% cases				Cause 2: 3% cases			
			Model/	Mean	Mean	Emp.		Mean	Mean	Emp.	
β_1, β_2	Cov.	Method	weights	est.	est. sd	sd	Eff.	est.	est. sd	sd	Eff.
0.00	z	Cohort	Cox	-0.006	0.243	0.237		-0.006	0.447	0.433	
	x			-0.023	0.425	0.428		-0.037	0.779	0.777	
	z	Trad. NCC	Strat. Cox	-0.006	0.353	0.366	0.419	-0.001	0.649	0.743	0.340
	x			-0.017	0.618	0.656	0.426	0.002	1.223	1.348	0.332
	z	WPL	Samuelsen	-0.006	0.317	0.329	0.519	-0.008	0.498	0.490	0.781
	x			-0.005	0.556	0.580	0.545	-0.020	0.874	0.907	0.734
	z	WPL	GAM	-0.006	0.318	0.331	0.513	-0.007	0.499	0.492	0.775
	x			-0.004	0.559	0.585	0.535	-0.018	0.876	0.911	0.727
	z	WPL	Logistic	-0.005	0.330	0.330	0.516	0.007	0.471	0.491	0.778
	x			-0.006	0.569	0.580	0.545	-0.021	0.870	0.909	0.731
	z	WPL	Chen	-0.005	0.339	0.338	0.492	-0.007	0.476	0.495	0.765
	x			-0.009	0.579	0.599	0.511	-0.022	0.865	0.920	0.713
	z	MLE	Weibull	-0.041	0.260	0.249	0.906	-0.042	0.460	0.445	0.947
	x			0.012	0.438	0.439	0.951	-0.002	0.792	0.797	0.950
	z	WPL	Calibrated	-0.006	0.332	0.340	0.486	-0.007	0.514	0.502	0.744
	x			-0.026	0.503	0.506	0.715	-0.035	0.852	0.852	0.832
1.00	z	Cohort	Cox	1.009	0.282	0.274		1.050	0.534	0.558	
	x			0.994	0.449	0.458		1.031	0.834	0.865	
	z	Trad. NCC	Strat. Cox	1.058	0.423	0.436	0.395	1.090	0.846	0.865	0.416
	x			1.000	0.714	0.729	0.395	1.074	1.445	1.585	0.298
	z	WPL	Samuelsen	1.026	0.351	0.344	0.634	1.067	0.577	0.612	0.831
	x			0.992	0.608	0.632	0.525	1.026	0.939	0.967	0.800
	z	WPL	GAM	1.024	0.353	0.348	0.620	1.065	0.578	0.614	0.826
	x			0.991	0.611	0.638	0.515	1.024	0.942	0.970	0.759
	z	WPL	Logistic	1.025	0.365	0.345	0.631	1.067	0.541	0.613	0.829
	x			0.992	0.557	0.634	0.522	1.027	0.732	0.968	0.799
	z	WPL	Chen	1.027	0.372	0.386	0.504	1.069	0.551	0.616	0.821
	x			0.989	0.567	0.699	0.429	1.022	0.737	0.982	0.776
	z	MLE	Weibull	0.985	0.292	0.276	0.986	1.028	0.535	0.552	1.022
	x			1.049	0.454	0.456	1.009	1.081	0.830	0.851	1.033
	z	WPL	Calibrated	1.053	0.365	0.357	0.589	1.100	0.593	0.625	0.797
	x			0.952	0.537	0.536	0.730	0.981	0.898	0.952	0.826

WPL = weighted partial likelihood estimation, MLE = maximum likelihood estimation
Number of times the likelihood is flat for $\beta = (0, 1)$ is $(4, 42)$
x is observed for the entire cohort, z only observed for the cases and controls
Distance measure for the calibrated weights are Poisson deviance

Appendix B

Theoretical derivations

B.1 The full likelihood - how it looks like

The general expression for the full likelihood is

$$\begin{aligned} L(\theta,\mu) \propto & \prod_{i\in\mathcal{O}} p(T_i, E_i | Z_{i,1:p}, X_i;\theta) p(Z_{i,1:p}|X_i;\mu) \\ & \times \prod_{i\in\mathcal{C}\setminus\mathcal{O}} \int_{z_{i,1:q}} \sum_{z_{i,(q+1):p}} [p(T_i, E_i | z_{i,1:p}, X_i;\theta) p(Z_{i,1:p} = z_{i,1:p}|X_i;\mu) dz_{i,1:q}] \end{aligned} \tag{B.1}$$

and the likelihood expression for (T_i, E_i) can be defined by the outcome specific hazard

$$\begin{aligned} & p(T_i, E_i | Z_i, X_i; \theta) \\ & \quad \propto \prod_{k=1}^{K} [\alpha_k(T_i|Z_i, X_i;\theta_k)]^{1\{E_i=k\}} \exp\left\{ -\int_0^{T_i} \sum_{k=1}^{K} \alpha_k(t|Z_i, X_i;\theta_k) dt \right\}. \end{aligned}$$

Put together this is

$$\begin{aligned} L(\theta,\mu) \propto & \prod_{i\in\mathcal{O}} \left[\prod_{k=1}^{K} [\alpha_k(T_i|Z_{i,1:p}, X_i;\theta_k)]^{1\{E_i=k\}} \right. \\ & \quad \left. \exp\left\{ -\int_0^{T_i} \sum_{k=1}^{K} \alpha_k(t|Z_{i,1:p}, X_i;\theta_k) dt \right\} p(Z_{i,1:p}|X_i;\mu) \right] \\ & \quad \times \prod_{i\in\mathcal{C}\setminus\mathcal{O}} \int_{z_{i,1:q}} \sum_{z_{i,(q+1):p}} \left[\prod_{k=1}^{K} [\alpha_k(T_i|Z_{i,1:p}, X_i;\theta_k)]^{1\{E_i=k\}} \right. \\ & \quad \left. \exp\left\{ -\int_0^{T_i} \sum_{k=1}^{K} \alpha_k(t|Z_{i,1:p}, X_i;\theta_k) dt \right\} p(Z_{i,1:p} = z_{i,1:p}|X_i;\mu) dz_{i,1:q} \right] \end{aligned} \tag{B.2}$$

We assume that the survival times are Weibull distributed, which means that the cause-specific hazard function is

$$\alpha_k(t|Z_i, X_i; \theta_k) = \lambda_k^{\nu_k} \nu_k t^{\nu_k - 1} \exp(\gamma_k Z_i + \eta_k X_i)$$

we also need

$$A(t|Z_i, X_i; \theta_k) = \frac{(\lambda_k t_k)^{\nu_k}}{\nu_k} \exp(\gamma_k Z_i + \eta_k X_i).$$

And at last we need to assume a probability distribution for the partially observed covariate, her we assume that $Z \sim$ Bernoulli(μ), which means that

$$p(Z_i|\mu) = \mu^{Z_i}(1-\mu)^{1-Z_i}$$

One covariate

If $Z \sim Bin(1, \mu)$, then the likelihood is

$$\begin{aligned} L(\theta, \mu) \propto \prod_{i \in \mathcal{O}} & \left[\prod_{k=1}^{K} \left[\lambda_k^{\nu_k} \nu_k t^{\nu_k - 1} \exp(\gamma_k Z_i) \right]^{I(E_i = k)} \right. \\ & \left. \times \exp\left\{ \sum_{k=1}^{K} -(\lambda_k t)^{\nu_k} \exp(\gamma_k Z_i) \right\} \mu^{Z_i}(1-\mu)^{1-Z_i} \right] \\ \times \prod_{i \in \mathcal{C} \setminus \mathcal{O}} & \left[\sum_{z_i} \left(\prod_{k=1}^{K} \lambda_k^{\nu_k} \nu_k t^{\nu_k - 1} \exp(\gamma_k z_i) \right. \right. \\ & \left. \left. \times \exp\left\{ \sum_{k=1}^{K} -(\lambda_k t)^{\nu_k} \exp(\gamma_k Z_i) \right\} \mu^{z_i}(1-\mu)^{1-z_i} \right) \right] \end{aligned}$$

Two independent covariates

The likelihood with two covariates, one known for the entire cohort and one only known for the cases and the controls that is the same as above.

$$\begin{aligned} L(\theta, \mu) \propto \prod_{i \in \mathcal{O}} & \left[\prod_{k=1}^{K} \left[\lambda_k^{\nu_k} \nu_k t^{\nu_k - 1} \exp(\gamma_k Z_i + \eta_k X_i) \right]^{I(E_i = k)} \right. \\ & \left. \times \exp\left\{ \sum_{k=1}^{K} -(\lambda_k t)^{\nu_k} \exp(\gamma_k Z_i + \eta_k X_i) \right\} \mu^{Z_i}(1-\mu)^{1-Z_i} \right] \\ \times \prod_{i \in \mathcal{C} \setminus \mathcal{O}} & \left[\sum_{z_i} \left(\prod_{k=1}^{K} \lambda_k^{\nu_k} \nu_k t^{\nu_k - 1} \exp(\gamma_k z_i + \eta_k X_i) \right. \right. \\ & \left. \left. \times \exp\left\{ \sum_{k=1}^{K} -(\lambda_k t)^{\nu_k} \exp(\gamma_k Z_i + \eta_k X_i) \right\} \mu^{z_i}(1-\mu)^{1-z_i} \right) \right] \end{aligned}$$

Two dependent covariates

Here the distribution of Z is dependent of X. $p(Z_i|X_i) = X_i^{Z_i}(1-X_i)^{1-Z_i}$, where $X \sim$ runif$[0,1]$ and known for the entire cohort, then the likelihood expression is

$$\begin{aligned} L(\theta,\mu) \propto \prod_{i\in\mathcal{O}} &\left[\prod_{k=1}^{K} \left[\lambda_k^{\nu_k}\nu_k t^{\nu_k-1}\exp(\gamma_k Z_i + \eta_k X_i)\right]^{I(E_i=k)} \right. \\ &\left. \times \exp\left\{\sum_{k=1}^{K} -(\lambda_k t)^{\nu_k}\exp(\gamma_k Z_i + \eta_k X_i)\right\} X_i^{Z_i}(1-X_i)^{1-Z_i}\right] \\ \times \prod_{i\in\mathcal{C}\setminus\mathcal{O}} &\left[\sum_{z_i}\left(\prod_{k=1}^{K}\lambda_k^{\nu_k}\nu_k t^{\nu_k-1}\exp(\gamma_k z_i + \eta_k X_i)\right.\right. \\ &\left.\left. \times \exp\left\{\sum_{k=1}^{K} -(\lambda_k t)^{\nu_k}\exp(\gamma_k Z_i + \eta_k X_i)\right\}\ X_i^{z_i}(1-X_i)^{1-z_i}\right)\right] \end{aligned}$$

B.2 Two phase variance

The variability of a two-phase design can be divided into two parts belonging to each phase of the sampling design. This can be seen by looking at the phase-two score function. First, the score for the phase one sampling is

$$\begin{aligned} U &= \sum_{E_i=1}\left\{X_i - \frac{\sum_{j\in\mathcal{R}_i} X_j\exp(X_j\beta)}{\sum_{j\in\mathcal{R}_i}\exp(X_j\beta)}\right\} \\ &= \sum_{E_i=1}\left\{X_i - \frac{S^{(1)}}{S^{(0)}}\right\} \end{aligned}$$

where

$$\begin{aligned} S^{(0)} &= \sum_{j\in\mathcal{R}_i}\exp(X_j\beta) \\ S^{(1)} &= \sum_{j\in\mathcal{R}_i} X_j\exp(X_j\beta) \end{aligned}$$

Analogous to this the score function for phase-two data is

$$\tilde{U} = \sum_{E_i=1}\left\{X_i - \frac{\tilde{S}^{(1)}}{\tilde{S}^{(0)}}\right\}$$

where

$$
\begin{aligned}
\tilde{S}^{(0)} &= \sum_{j \in \mathcal{R}_i} \frac{\mathcal{O}_j}{\pi_j} \exp(X_j \beta) \\
\tilde{S}^{(1)} &= \sum_{j \in \mathcal{R}_i} \frac{\mathcal{O}_j}{\pi_j} X_j \exp(X_j \beta)
\end{aligned}
$$

If we now write $\tilde{U}$ as

$$
\begin{aligned}
\tilde{U} &= U + (\tilde{U} - U) \\
&= U + \sum_{E_i=1} \left(X_i - \frac{\tilde{S}^{(1)}}{\tilde{S}^{(0)}} \right) - \sum_{E_i=1} \left(X_i - \frac{S^{(1)}}{S^{(0)}} \right) \\
&= U + \sum_{E_i=1} \left(\frac{S^{(1)}}{S^{(0)}} - \frac{\tilde{S}^{(1)}}{\tilde{S}^{(0)}} \right) \\
&= U + \sum_{E_i=1} \left(\frac{S^{(1)}}{S^{(0)}} - \frac{\tilde{S}^{(1)}}{\tilde{S}^{(0)}} - \frac{\tilde{S}^{(1)}}{S^{(0)}} + \frac{\tilde{S}^{(1)}}{S^{(0)}} \right) \\
&= U + \sum_{E_i=1} \left(\frac{S^{(1)}}{S^{(0)}} - \frac{\tilde{S}^{(1)} S^{(0)}}{\tilde{S}^{(0)} S^{(0)}} - \frac{\tilde{S}^{(1)}}{S^{(0)}} + \frac{\tilde{S}^{(1)} \tilde{S}^{(0)}}{S^{(0)} \tilde{S}^{(0)}} \right) \\
&= U + \sum_{E_i=1} \sum_{j \in \mathcal{R}_i} \left(1 - \frac{\mathcal{O}_j}{\pi_j} \right) \underbrace{\left(X_j - \frac{\tilde{S}^{(1)}}{\tilde{S}^{(0)}} \right) \frac{\exp(X_j \beta)}{S^{(0)}}}_{\tilde{V}_j} \\
&= U + \sum_{E_i=1} \sum_{j \in \mathcal{R}_i} \left(1 - \frac{\mathcal{O}_j}{\pi_j} \right) \tilde{V}_j \\
&= U + \sum_{E_i=1} \sum_{j \in \mathcal{R}_i} \left(1 - \frac{\mathcal{O}_j}{\pi_j} \right) V_j + o_p(1) \qquad \text{(B.3)}
\end{aligned}
$$

where

$$
V_j = \sum_{E_i=1} \sum_{j \in \mathcal{R}_i} \left(X_j - \frac{s^{(1)}}{s^{(0)}} \right) \frac{\exp(X_j \beta)}{S^{(0)}}
$$

and we assume that $\frac{1}{n} S^{(0)} \rightarrow s^{(0)}$ and $\frac{1}{n} S^{(1)} \rightarrow s^{(1)}$ where n is the cohort size and $s^{(0)}$ and $s^{(1)}$ is non-random. We want to find the variance of the two main terms in (B.3)

$$\begin{aligned}
&\text{Cov}(U, \sum_{E_i=1}\sum_{j\in\mathcal{R}_i}\left(1-\frac{\mathcal{O}_j}{\pi_j}\right)V_j) \\
&\quad= E\left[U\sum_{E_i=1}\sum_{j\in\mathcal{R}_i}\left(1-\frac{\mathcal{O}_j}{\pi_j}\right)V_j\right] - \underbrace{E[U]}_{0}E\left[\sum_{E_i=1}\sum_{j\in\mathcal{R}_i}\left(1-\frac{\mathcal{O}_j}{\pi_j}\right)V_j\right] \\
&\quad= E\left[E\left[U\sum_{E_i=1}\sum_{j\in\mathcal{R}_i}\left(1-\frac{\mathcal{O}_j}{\pi_j}\right)V_j\middle|\mathcal{F}\right]\right] \\
&\quad= E\left[U\sum_{E_i=1}\sum_{j\in\mathcal{R}_i}\underbrace{E\left[\left(1-\frac{\mathcal{O}_j}{\pi_j}\right)V_j\middle|\mathcal{F}\right]}_{0}\right] = 0
\end{aligned}$$

where $\mathcal{F}$ is the cohort history. This means that U and $\sum_{E_i=1}\sum_{j\in\mathcal{R}_i}\left(1-\frac{\mathcal{O}_j}{\pi_j}\right)V_j$ are asymptotical uncorrelated and the variance is the sum of two components

$$\text{Var}(\tilde{U}) = \Sigma + \Gamma.$$

Here Σ correspond to the phase one variance, and is also the usual score variance when data for the complete cohort is known and Γ correspond to the phase two sampling, and is the additional variance one get when all covariates are known only for a sample of the cohort. This also means that variance of the estimates can be divided into two parts. By 1.order Taylor expansion $\tilde{\beta} \approx \Sigma^{-1}\tilde{U} + \beta$ and

$$\begin{aligned}
\text{Var}(\tilde{\beta}) &\approx \Sigma^{-1}(\Sigma+\Gamma)\Sigma^{-1} \\
&= \Sigma^{-1} + \Sigma^{-1}\Gamma\Sigma^{-1}
\end{aligned}$$

B.3 Monte Carlo integration and importance sampling

Monte Carlo integration

Assume we have

$$I = \int_x h(x)f(x)dx$$

where $f(x)$ is a density then

$$I = E_f[h(x)]$$

The Monte Carlo estimate is

$$I \approx \bar{h}_M = \frac{1}{M} \sum_{m=1}^{M} h(x_m)$$

where $x_m \sim f(x)$. By strong law of large numbers

$$\bar{h}_M {\rightarrow} E_f[h(x)]$$

almost surely when $M \rightarrow \infty$.

Importance sampling

$$I = \int_x h(x) f(x) dx = \int_x \frac{h(x) f(x)}{g(x)} g(x) dx$$

where $g(x)$ is a density that should be similar to $f(x)$. Then

$$I = E_g \left[\frac{h(x) f(x)}{g(x)} \right] = E_g[h(x) w(x)] \approx \frac{1}{M} \sum_{m=1}^{M} h(x_m) w(x_m)$$

where $x_m \sim g(x)$. $w(x)$ can thought of as a way of correcting for the fact that x_m is drawn from the wrong distribution.

Appendix C

Code

In order to minimize pages with code, I have removed "unnecessary" things like saving the results and generating tables. Also I have a lot of similar code and only one of each have been included.

Simulation two correlated covariates

This is code from the simulation part, in particular the simulation with two correlated covariates.

```
###############################################################
##                                                           ##
##Simulation with two correlated covariates                  ##
##x2 ~ U[0,1]                                                ##
##x1 ~ Bin(n,1,x2)                                           ##
##Two endpoints: One common(10%), one rare(3%)               ##
##Analysis both done on two single endpoint models and on    ##
##one multiple endpoint model                                ##
##Exponential survival times                                 ##
##Random censoring:                                          ##
##C ~ U[0,0.13]                                              ##
##Cohort size: 1000                                          ##
##Simualtions done: 1000                                     ##
##                                                           ##
##Analysis:                                                  ##
##WPL: Weights: Samuelsen's, logistic, GAM, Chen, calibrated ##
##Saarelas likelihood                                        ##
##                                                           ##
###############################################################

library(survival)
library(gam)
library(survey)

##cohort size
n = 1000
##number of simulations
ant = 1000
##number of controls
m = 1
##survival time
```

```
T = 1:n
##censoring time
C = 1:n

NCC_inf = 0
ant.tom.risikomengde = 0
no_cases1 = 0
no_cases2 = 0

##Drawing covariates
x2 = runif(n,0,1)
x1 = rbinom(n,1,x2)

for(j in 1:ant) {
  print(j)
  ind.nr = 1:n
  tom_risikomengde = 0

  status = array(0,dim=n)
  ##Drawing event times from the exponential distribution
  ##and censoring times from the uniform distribution.
  b1 = 0.515
  b2 = 0.155
  bb = 1
  T1 = rexp(n,b1*exp(bb*(x1+x2)))
  T2 = rexp(n,b2*exp(bb*(x1+x2)))

  ##uniform censoring
  C = runif(n,0,0.13)

  test1 = (T1 == pmin(T1,T2,C))
  test2 = (T2 == pmin(T1,T2,C))
  status[which(test1==TRUE)] = 1
  status[which(test2==TRUE)] = 2
  T = pmin(C,T1,T2)
  no_cases1[j] = sum(status==1)
  no_cases2[j] = sum(status==2)

  ##Cohort
  cox_cohort1 = coxph(Surv(T,status==1)~x1+x2)
  cox_cohort2 = coxph(Surv(T,status==2)~x1+x2)


######################## NCC sampling ##########################
  sett = 1:n
  ## for cases k which endpoint it experience, for controls it is
  ##which endpoint it is sampled for
  k = (1:n)*0
  cohortdata = data.frame(ind.nr,x1,x2,T,status,sett,k)
  cases1 = cohortdata[(cohortdata$status==1),]
  cases2 = cohortdata[(cohortdata$status==2),]
  cases = rbind(cases1,cases2)
  NCCdata = data.frame()

  ##The same number of sets that people who dies
  sett = 1:length(cohortdata$status[which(cohortdata$status != 0)])

  dis01 = 0
  dis10 = 0
  for(i in 1:length(sett)) {

  ##Checks that there are m individuals still at risk
  under_risiko = dim(cohortdata[which(cohortdata$T > cases$T[i]),])[1]

    mm = min(m,under_risiko)
    if(under_risiko > 0)  {
      R = sample(cohortdata$ind.nr[cohortdata$T > cases$T[i]
       & cohortdata$ind.nr != cases$ind.nr[i]], mm, replace=F)

      ##Setting risk set and which endpoint the control is sampled for
      cohortdata[R[1],6] = i
      cohortdata[R[1],7] = cohortdata$status[cases[i,1]]

      ##Setting risk set and endpoint for the case
      cohortdata[cases[i,1],6] = i
      cohortdata[cases[i,1],7] = cohortdata$status[cases[i,1]]

      NCCdata = rbind(NCCdata,cohortdata[cases[i,1],],
        cohortdata[R[1],])
      ##Putting status = 0 on controls in case a later case has been
      ##sampled
      NCCdata$status[length(NCCdata$ind.nr)] = 0
    }
    else  {
      ##Setting risk set and k for the case
      cohortdata[cases[i,1],6] = i
      cohortdata[cases[i,1],7] = cohortdata$status[cases[i,1]]
      NCCdata = rbind(NCCdata,cohortdata[cases[i,1],])
      ant.tom.risikomengde = ant.tom.risikomengde + 1
      tom_risikomengde = 1
    }

    ##check for discordant pairs
    if(cohortdata$x1[cases[i,1]] == 0 && cohortdata$x1[R[1]] == 1 &&
     cohortdata$status[cases[i,1]]==2)  {
      dis01 = dis01 + 1
    }
    if(cohortdata$x1[cases[i,1]] == 1 && cohortdata$x1[R[1]] == 0 &&
     cohortdata$status[cases[i,1]]==2)  {
      dis10 = dis10 + 1
    }
  }
  if(tom_risikomengde == 0)  {
    cox_NCC1 = coxph(Surv(T,status==1)~x1+x2+strata(sett),
      subset=c(NCCdata$k==1),data=NCCdata)

    ##check for discordant pairs
    if(dis01 == 0 || dis10 == 0)  {
      NCC_inf = NCC_inf + 1
    }
    else  {
      cox_NCC2 = coxph(Surv(T,status==2)~x1+x2+strata(sett),
        subset=c(NCCdata$k==2),data=NCCdata)
    }
  }

################### partial likelihood with IPW ###############
  cohortdata = cohortdata[order(cohortdata$T),]
  cohortdata$brukt[1:n] = 0
  cohortdata$brukt[which(cohortdata$ind.nr %in% NCCdata$ind.nr)] = 1


  ##SO
  for(k in 1:n)  {
    failuretimes = cohortdata$T[which(cohortdata$status != 0)]
    pk = 1
  }

  ##Finds the number individuals under risk
  nfail = 1:length(failuretimes)
  for(k in 1:length(failuretimes)) {
    nfail[k] = length(cohortdata$T[which(cohortdata$T >=
      failuretimes[k])])
  }

  psample = rep(0,n)
  qsample = (1-m/(nfail[1]-1))
  for (k in (1:sum(status != 0))) if (nfail[k] > m) {
    if (k > 1) {
      qsample[k] = qsample[k-1]*(1-m/(nfail[k]-1))
    }
    llim = n-nfail[k]+1
    psample[llim:n] = 1-qsample[k]
  }
  pso = psample

  ##Cox-regression with SO-weights
  data=data.frame()
  data=cbind(ind.nr=rev(data$ind.nr),x1=rev(data$x1),x2=rev(data$x2),
    T=rev(data$T),status=rev(data$status),sett=rev(data$sett),
    brukt=rev(data$brukt),w=rev(data$w))
  data=as.data.frame(data)

  w = array(0,dim=n)
  w[which(cohortdata$k != 0 & cohortdata$status == 0)] =
    pso[which(cohortdata$k != 0 & cohortdata$status == 0)]
  w[which(cohortdata$status != 0)] = 1

  ##Making the data that only includes cases and controls
  data=rbind(cbind(cohortdata[which(w != 0),],w = w[which(w != 0)]))
  data=cbind(ind.nr=rev(data$ind.nr),x1=rev(data$x1),x2=rev(data$x2),
    T=rev(data$T),status=rev(data$status),sett=rev(data$sett),
      k=rev(data$k),brukt=rev(data$brukt),w=rev(data$w))
  data=as.data.frame(data)

  cox_SO1 = coxph(Surv(T,status==1)~x1+x2, weights=1/w,data=data,
    robust=TRUE)
  cox_SO2 = coxph(Surv(T,status==2)~x1+x2, weights=1/w,data=data,
    robust=TRUE)
  cox_SO1_fc = coxph(Surv(T,status==1)~x1+x2, weights=1/w,data=data,
    subset=c(data$k==1),robust=TRUE)
  cox_SO2_fc = coxph(Surv(T,status==2)~x1+x2, weights=1/w,data=data,
    subset=c(data$k==2),robust=TRUE)

  ##GAM
  pgam = gam(brukt~s(T),family=binomial,data=cohortdata,
    subset=cohortdata$status==0)
  pgam = pgam$fitted

  data=data.frame()
  w = array(0,dim=n)
  w[which(cohortdata$status==0)] = pgam
  w[which(cohortdata$status != 0)] = 1
  w[which(cohortdata$brukt == 0)] = 0
  wgam = w

```

```r
##Making the data that only includes cases and controls
data=rbind(cbind(cohortdata[which(w != 0),],w = w[which(w != 0)]))
data=cbind(ind.nr=rev(data$ind.nr),x1=rev(data$x1),x2=rev(data$x2),
  T=rev(data$T),status=rev(data$status),sett=rev(data$sett),
    k=rev(data$k),brukt=rev(data$brukt),w=rev(data$w))
data=as.data.frame(data)

cox_gam1 = coxph(Surv(T,status==1)~x1+x2,weights=1/data$w,data=data,
  robust=TRUE)
cox_gam2 = coxph(Surv(T,status==2)~x1+x2,weights=1/data$w,data=data,
  robust=TRUE)
cox_gam1_fc = coxph(Surv(T,status==1)~x1+x2,weights=1/data$w,
  data=data, subset=c(data$k==1),robust=TRUE)
cox_gam2_fc = coxph(Surv(T,status==2)~x1+x2,weights=1/data$w,
  data=data,subset=c(data$k==2),robust=TRUE)

##logistic
pglm = glm(brukt~log(T),data=cohortdata,family=binomial,
  subset=cohortdata$status==0)
pglm = pglm$fit

data=data.frame()
w = array(0,dim=n)
w[which(cohortdata$status==0)] = pglm
w[which(cohortdata$status != 0)] = 1
w[which(cohortdata$brukt == 0)] = 0

data = rbind(cbind(cohortdata[cohortdata$brukt==1,],
  w = w[cohortdata$brukt==1]))
data=cbind(ind.nr=rev(data$ind.nr),x1=rev(data$x1),x2=rev(data$x2),
  T=rev(data$T),status=rev(data$status),sett=rev(data$sett),
    k=rev(data$k),brukt=rev(data$brukt),w=rev(data$w))
data=as.data.frame(data)

cox_glm1 = coxph(Surv(T,status==1)~x1+x2,weights=1/data$w,data=data,
  robust=TRUE)
cox_glm2 = coxph(Surv(T,status==2)~x1+x2,weights=1/data$w,data=data,
  robust=TRUE)
cox_glm1_fc = coxph(Surv(T,status==1)~x1+x2,weights=1/data$w,
  subset=c(data$k==1),data=data,robust=TRUE)
cox_glm2_fc = coxph(Surv(T,status==2)~x1+x2,weights=1/data$w,
  subset=c(data$k==2),data=data,robust=TRUE)

##Chen
pchen = 1:10
partT = seq(0,max(cohortdata$T),length=11)
for(i in 1:(length(partT)-1))  {
  ne = sum(cohortdata$status == 0 & cohortdata$T >= partT[i] &
     cohortdata$T <= partT[i+1])
  te = sum(cohortdata$status == 0 & cohortdata$brukt != 0 &
    cohortdata$T >= partT[i] & cohortdata$T <= partT[i+1])
  pchen[i] = te/ne
}

data=data.frame()
##controls
for(i in 1:10)  {
  test = cohortdata$ind.nr[(cohortdata$T > partT[i]
    & cohortdata$T <= partT[i+1] & cohortdata$brukt ==1
    & cohortdata$status == 0)]
  if(sum(test) != 0)  {
    for(k in 1:length(test))  {
        w = pchen[i]
        data = rbind(data,cbind(cohortdata[which(test[k]==
          cohortdata$ind.nr),],w))
      }
    }
  }
  #cases
  for(i in 1:dim(cohortdata)[1]) {
    if(cohortdata$status[i] != 0) {
      w=1
      data=rbind(data,cbind(cohortdata[i,],w))
    }
  }
  cox_chen1 = coxph(Surv(T,status==1)~x1+x2,weights=1/data$w,
    data=data,robust=TRUE)
  cox_chen2 = coxph(Surv(T,status==2)~x1+x2,weights=1/data$w,
    data=data,robust=TRUE)
  cox_chen1_fc = coxph(Surv(T,status==1)~x1+x2,weights=1/data$w,
    subset=c(data$k==1),data=data,robust=TRUE)
  cox_chen2_fc = coxph(Surv(T,status==2)~x1+x2,weights=1/data$w,
    subset=c(data$k==2),data=data,robust=TRUE)


###################### full likelihood #########################
  kk = cohortdata$ind.nr %in% NCCdata$ind.nr

  ##Z have value for all individuals, but are actually only known for
  ##cases and controls
  Z = cohortdata$x1
  X = cohortdata$x2
  t = cohortdata$T
  status = cohortdata$status
  k = as.numeric(kk)

  lik = function(para) {
    g1 = para[1]
    g2 = para[2]
    e1 = para[3]
    e2 = para[4]
    a1 = para[5]
    a2 = para[6]
    b1 = para[7]
    b2 = para[8]
    z=c(0,1)
    lO=1
    lIO = 1
    lO = log((((a1^b1)*b1*(t^(b1-1))*exp(g1*Z+e1*X))^(I(status == 1))*
       ((a2^b2)*b2*(t^(b2-1))*exp(g2*Z+e2*X))^(I(status == 2))*
       exp(-(a1*t)^b1*exp(g1*Z+e1*X)-(a2*t)^b2*exp(g2*Z+e2*X))*
       X^Z*(1-X)^(1-Z))^(I(kk != 0)))

    lIO = log(((((a1^b1)*b1*(t^(b1-1))*exp(e1*X))^(I(status == 1))*
       ((a2^b2)*b2*(t^(b2-1))*exp(e2*X))^(I(status == 2))*
       exp(-(a1*t)^b1*exp(e1*X)-(a2*t)^b2*exp(e2*X))*(1-X))+
       (((a1^b1)*b1*(t^(b1-1))*exp(g1+e1*X))^(I(status == 1))*
       ((a2^b2)*b2*(t^(b2-1))*exp(g2+e2*X))^(I(status == 2))*
       exp(-(a1*t)^b1*exp(g1+e1*X)-(a2*t)^b2*exp(g2+e2*X))*X))^
       (I(kk == 0)))

    lO = sum(lO)
    lIO = sum(lIO)
    l=lO+lIO
```

```
}

lik1_fc = function(para) {
  g = para[1]
  e = para[2]
  a = para[3]
  b = para[4]

  lO = log((((a^b)*b*(t^(b-1))*exp(g*Z+e*X))^(I(status == 1))*
         exp(-(a*t)^b*exp(g*Z+e*X))*X^Z*(1-X)^(1-Z))^
         (I(cohortdata$k == 1 & kk != 0)))

  lIO = log(((((a^b)*b*(t^(b-1))*exp(e*X))^(I(status == 1))*
         exp(-(a*t)^b*exp(e*X))*(1-X))+
         (((a^b)*b*(t^(b-1))*exp(g+e*X))^(I(status == 1))*
         exp(-(a*t)^b*exp(g+e*X))*X))^(I(kk == 0)))
  lO = sum(lO)
  lIO = sum(lIO)
  l=lO+lIO
}

lik2_fc = function(para) {
  g = para[1]
  e = para[2]
  a = para[3]
  b = para[4]

  lO = log((((a^b)*b*(t^(b-1))*exp(g*Z+e*X))^(I(status == 2))*
         exp(-(a*t)^b*exp(g*Z+e*X))*X^Z*(1-X)^(1-Z))^
         (I(cohortdata$k == 2 & kk != 0)))

  lIO = log(((((a^b)*b*(t^(b-1))*exp(e*X))^(I(status == 2))*
            exp(-(a*t)^b*exp(e*X))*(1-X))+
            (((a^b)*b*(t^(b-1))*exp(g+e*X))^(I(status == 2))*
            exp(-(a*t)^b*exp(g+e*X))*X))^(I(kk == 0)))
  lO = sum(lO)
  lIO = sum(lIO)
  l=lO+lIO
}

minuslik = function(para) {
  -lik(para)
}

minuslik1_fc = function(para) {
  -lik1_fc(para)
}

minuslik2_fc = function(para) {
  -lik2_fc(para)
}

opt = optim(c(bb,bb,bb,bb,b1,b2,1,1), minuslik, hessian=T,
  method = ”BFGS”)
opt1_fc = optim(c(bb,bb,b1,1), minuslik1_fc, hessian=T,
  method = ”BFGS”)
opt2_fc = optim(c(bb,bb,b2,1), minuslik2_fc, hessian=T,
  method = ”BFGS”)

i = solve(opt$hessian)
i1_fc = solve(opt1_fc$hessian)
i2_fc = solve(opt2_fc$hessian)
```

```


############################ Breslow ############################
  ##Stratify according to status
  strat = (1:n)*0
  strat[cohortdata$status==0] = 1
  strat[cohortdata$status==1] = 2
  strat[cohortdata$status==2] = 3
  cohortdata$strat = strat

  in.sample = cohortdata$k!=0
  cohortdata$ww = wgam

  dsingle = svydesign(id=~ind.nr,weights=~ww,data=
    subset(cohortdata,in.sample))
  dtwophs = twophase(id=list(~ind.nr,~ind.nr),subset=~in.sample,
    data=cohortdata,strata=list(~strat,~strat))

  ##Predicting the partially known covariate
  pred2 = svyglm(x1~x2,weights=ww,design=dtwophs,family=quasibinomial,
    control=glm.control(maxit=100))
  cohortdata$imp.x1 = predict(pred2,type="response",newdata=
    cohortdata,se=F)

  ##Cox-regression with predicted values
  cox.imp1 = coxph(Surv(T,status==1)~imp.x1+x2,data=cohortdata)
  cox.imp2 = coxph(Surv(T,status==2)~imp.x1+x2,data=cohortdata)

  ##Obtaining dfbetas
  imp.dfb1 = resid(cox.imp1,type="dfbeta")+1
  imp.dfb2 = resid(cox.imp2,type="dfbeta")+1
  colnames(imp.dfb1) = paste("imp.dfb1",1:ncol(imp.dfb1),sep="")
  colnames(imp.dfb2) = paste("imp.dfb2",1:ncol(imp.dfb1),sep="")
  cohortdata.imp = cbind(cohortdata,imp.dfb1,imp.dfb2)
  dtwophs.imp = twophase(id=list(~ind.nr,~ind.nr),subset=~in.sample,
    data=cohortdata.imp,strata=list(~strat,~strat))

  ##Calibration
  dcalibr1 = calibrate(dtwophs.imp,phase=2, formula=
    make.formula(colnames(imp.dfb1)),calfun="raking",eps=0.00001,
      maxit=100,force=TRUE)
  dcalibr2 = calibrate(dtwophs.imp,phase=2, formula=
    make.formula(colnames(imp.dfb2)),calfun="raking",eps=0.00001,
      maxit=100,force=TRUE)

  ##Cox-regression with calibrated weights
  calibr11 = svycoxph(Surv(T,status==1)~x1+x2,design=dcalibr1)
  calibr21 = svycoxph(Surv(T,status==2)~x1+x2,design=dcalibr2)
}
```

Birth registry data, one fully known numerical covariate and one partially known binary covariate

This code is from the data analysis part, the evaluation of the full likelihood is done in C, see below.

```
#################################################################
##                                                             ##
##Birth registry data:                                         ##
##x1 - gestational age in days, known for entire cohort        ##
##x2 ~ birth weight(0 if bw < 3 kg, 1 if bw > 3kg)             ##
##Two endpoints: Cancer and other deaths                       ##
##                                                             ##
##Simualtions done: 200                                        ##
##                                                             ##
##Analysis:                                                    ##
##WPL: Weights: Samuelsen's, GAM, calibrated                   ##
##MLE: Saarelas likelihood evaluated in C                      ##
##      Scheike(doesn't work)                                  ##
##                                                             ##
#################################################################


library(survival)
library(survey)
library(gam)
library(nccMLE)

ant = 200

dyn.load("loglik_mfr_probit_gest.so")
l = 0
loglik.probit.gest = function(status,t,z5,z6,opti,k,l,n)  {
  .C("loglik_mfr_probit_gest", as.integer(status),as.double(t),
    as.integer(z5),as.integer(z6),as.double(opti),as.integer(k),
      as.double(l),as.integer(n))
}

mfrdata = read.table("MFRDOD.dat",header=T)
mfrdata = mfrdata[mfrdata$pari==0,]
mfrdata = mfrdata[mfrdata$pari!=99,]
mfrdata = mfrdata[mfrdata$gest>100,]
mfrdata = mfrdata[mfrdata$gest<315,]
mfrdata = mfrdata[mfrdata$vekt>450,]
mfrdata = mfrdata[mfrdata$levetid>364,]
mfrdata = mfrdata[mfrdata$gender==1,]


vekt.ind = as.numeric(I(mfrdata$vekt >= 3000))
mfrdata$vekt.ind = vekt.ind
ord.mfr = mfrdata[order(mfrdata$levetid),]

n = dim(ord.mfr)[1]
ord.mfr$sett = rep(0,n)
ord.mfr$k = rep(0,n)
ord.mfr$ind.nr = 1:n
m = 1

cases = ord.mfr[which(ord.mfr$DOD==1 & ord.mfr$levetid<3650),]
kreft = cases$Kreft
cases.kreft = cases[kreft==1,]
cases.andre = cases[kreft==0,]
n.kreft = dim(cases.kreft)[1]
n.andre = dim(cases.andre)[1]
n.cases = n.kreft+n.andre

ind.nr.cases = cases$ind.nr
ind.nr.kreft = cases$ind.nr[kreft==1]
ind.nr.andre = cases$ind.nr[kreft==0]

ord.mfr$status = rep(0,n)
ord.mfr$status[which(ord.mfr$Kreft==1 & ord.mfr$levetid < 3650)] = 1
ord.mfr$status[which(ord.mfr$DOD == 1 &
  ord.mfr$Kreft==0 & ord.mfr$levetid < 3650)] = 2

coxfit1=coxph(Surv(levetid,status==1)~gest+vekt.ind,data=ord.mfr)
coxfit2=coxph(Surv(levetid,status==2)~gest+vekt.ind,data=ord.mfr)

sur1=survreg(Surv((levetid-364),status==1)~gest+vekt.ind,
  control=list(maxiter=50),data=ord.mfr)
sur2=survreg(Surv((levetid-364),status==2)~gest+vekt.ind,
  control=list(maxiter=50),data=ord.mfr)

##Have the same number of risk sets that the number of individuals
##that dies
sett = 1:n.cases
for(j in 1:ant)  {
print(date())
print(j)
NCCdata = data.frame()

##Samples controls for cancer endpoint first
for(i in 1:n.kreft)  {
  R = sample((ord.mfr$ind.nr[ind.nr.kreft[i]]+1):n,m,replace=F)
  NCCdata = rbind(NCCdata,ord.mfr[cases.kreft$ind.nr[i],],
    ord.mfr[R[1:m],])
}

##Then sampling of controls  for other endpoint
for(i in 1:n.andre)  {
  R = sample((ord.mfr$ind.nr[ind.nr.andre[i]]+1):n,m,replace=F)
  NCCdata = rbind(NCCdata,ord.mfr[cases.andre$ind.nr[i],],
    ord.mfr[R[1:m],])
}

akreft = rep((m+1),n.kreft)
aandre = rep((m+1),n.andre)
settkreft = rep(1:n.kreft,akreft)
settandre = rep(1:n.andre,aandre)
sett = c(settkreft,settandre)
NCCdata$sett = sett

kkreft = rep(1,(n.kreft)*(m+1))
kandre = rep(2,(n.andre)*(m+1))
k = c(kkreft,kandre)
NCCdata$k=k
ord.mfr$k=rep(0,n)
ord.mfr$k[NCCdata$ind.nr] = NCCdata$k
ord.mfr$sett[NCCdata$ind.nr] = NCCdata$sett

cox.NCC1 = coxph(Surv(levetid,status==1)~gest+vekt.ind+strata(sett),
  subset=c(NCCdata$k==1),data=NCCdata)
cox.NCC2 = coxph(Surv(levetid,status==2)~gest+vekt.ind+strata(sett),
  subset=c(NCCdata$k==2),data=NCCdata)


############################ IPW ############################
##SO-vekter
failuretimes = cases$levetid

##Finds the number individuals at risk
nfail = 1:length(failuretimes)
for(k in 1:length(failuretimes)) {
  nfail[k] =  length(ord.mfr$levetid[which(ord.mfr$levetid >=
    failuretimes[k])])
}

psample = rep(0,n)
qsample = (1-m/(nfail[1]-1))
for (k in 1:n.cases) if (nfail[k] > m) {
    if (k > 1) {
      qsample[k] = qsample[k-1]*(1-m/(nfail[k]-1))
    }
    llim = n-nfail[k]+1
    psample[llim:n] = 1-qsample[k]
}
pso = psample

##Cox-regression with SO-weights
data=data.frame()
w = array(0,dim=n)
w[which(ord.mfr$k != 0 & ord.mfr$status == 0)] =
  pso[which(ord.mfr$k != 0 & ord.mfr$status == 0)]
w[which(ord.mfr$status != 0)] = 1

##Making the data that includes cases and controls
data=rbind(cbind(ord.mfr[which(w != 0),],w = w[which(w != 0)]))
data=cbind(ind.nr=rev(data$ind.nr),vekt.ind=rev(data$vekt.ind),
  gest=rev(data$gest),levetid=rev(data$levetid),status=
    rev(data$status), sett=rev(data$sett),w=rev(data$w))
data = data.frame(data)

cox.SO1 = coxph(Surv(levetid,status==1)~gest+data$vekt.ind,
  weights=1/w,data=data,robust=TRUE)
cox.SO2 = coxph(Surv(levetid,status==2)~gest+data$vekt.ind,
  weights=1/w, data=data,robust=TRUE)

beta.SO1[j,] = cox.SO1$coef
beta.SO2[j,] = cox.SO2$coef
se.robust.SO1[j,] = sqrt(diag(cox.SO1$var))
se.robust.SO2[j,] = sqrt(diag(cox.SO2$var))

##GAM
ord.mfr$brukt[1:n] = 0
ord.mfr$brukt[which(ord.mfr$ind.nr %in% NCCdata$ind.nr)] = 1
pgam = gam(brukt~s(levetid),family=binomial,data=ord.mfr,
  subset=ord.mfr$status==0)

pgam = pgam$fitted

data=data.frame()
w = array(0,dim=n)
w[which(ord.mfr$status==0)] = pgam
w[which(ord.mfr$status != 0)] = 1
w[which(ord.mfr$brukt == 0)] = 0
```

```


##Making the data that only includes cases and controls
data=rbind(cbind(ord.mfr[which(w != 0),],w = w[which(w != 0)]))
data=cbind(ind.nr=rev(data$ind.nr),vekt.ind=rev(data$vekt.ind),
  gest=rev(data$gest),levetid=rev(data$levetid),status=
    rev(data$status),sett=rev(data$sett),w=rev(data$w))
data = data.frame(data)

cox.gam1 = coxph(Surv(levetid,status==1)~gest+data$vekt.ind,
  weights=1/w,data=data,robust=TRUE)
cox.gam2 = coxph(Surv(levetid,status==2)~gest+data$vekt.ind,
  weights=1/w,data=data,robust=TRUE)


######################## Calibration ########################
##Breslow
strat = (1:n)*0
strat[ord.mfr$status==0] = 1
strat[ord.mfr$status==1] = 2
strat[ord.mfr$status==2] = 3
ord.mfr$strat = strat
in.sample = ord.mfr$k!=0

##using gam-weights
ord.mfr$ww = w
dsingle = svydesign(id=~ind.nr,weights=~ww,data=subset
  (ord.mfr,in.sample))
dtwophs = twophase(id=list(~ind.nr,~ind.nr),subset=~in.sample,
  data=ord.mfr,strata=list(~strat,~strat))

pred2 = svyglm(vekt.ind~gest,weights=ww,design=dtwophs,family=
  quasibinomial,control=glm.control(maxit=100))
ord.mfr$imp.ind.vekt = predict(pred2,type="response",newdata=ord.mfr,
  se=F)

cox.imp1 = coxph(Surv(levetid,status==1)~imp.ind.vekt+gest,
  data=ord.mfr)
cox.imp2 = coxph(Surv(levetid,status==2)~imp.ind.vekt+gest,
  data=ord.mfr)
imp.dfb1 = resid(cox.imp1,type="dfbeta")+1
imp.dfb2 = resid(cox.imp2,type="dfbeta")+1
colnames(imp.dfb1) = paste("imp.dfb1",1:ncol(imp.dfb1),sep="")
colnames(imp.dfb2) = paste("imp.dfb2",1:ncol(imp.dfb1),sep="")

ord.mfr.imp = cbind(ord.mfr,imp.dfb1,imp.dfb2)
dtwophs.imp = twophase(id=list(~ind.nr,~ind.nr),subset=~in.sample,
  data=ord.mfr.imp,strata=list(~strat,~strat))

dcalibr1 = calibrate(dtwophs.imp,phase=2, formula=make.formula
  (colnames(imp.dfb1)),calfun="raking",eps=0.00001,maxit=100)
dcalibr2 = calibrate(dtwophs.imp,phase=2, formula=make.formula
  (colnames(imp.dfb2)),calfun="raking",eps=0.00001,maxit=100)

calibr11 = svycoxph(Surv(levetid,status==1)~vekt.ind+gest,
  design=dcalibr1)
calibr21 = svycoxph(Surv(levetid,status==2)~vekt.ind+gest,
  design=dcalibr2)
```

```
############################### MLE ###############################
##Scheike
#utenfor = ord.mfr$levetid[ord.mfr$k==0]
#nno = length(utenfor)

#status2 = I(ord.mfr$status==1)
#em1 = em.ncc(cbind(ord.mfr$vekt.ind[ord.mfr$k!=0],
#   ord.mfr$gest[ord.mfr$k!=0]),ord.mfr$levetid[ord.mfr$k!=0],
#      status2,utenfor,nno,emvar=1,Nit = 100)
#status2 = I(ord.mfr$status==2)
#em2 = em.ncc(cbind(ord.mfr$vekt.ind[ord.mfr$k!=0],
#   ord.mfr$gest[ord.mfr$k!=0]),ord.mfr$levetid[ord.mfr$k!=0],
#      status2,utenfor,nno,emvar=1,Nit = 100)


##Saarela
status = ord.mfr$status
t = ord.mfr$levetid -364
z5 = ord.mfr$gest
z6 = ord.mfr$vekt.ind
k = ord.mfr$k
convergence=0*(1:ant)

wrapper = function(para)  {
  l=loglik.probit.gest(status,t,z5,z6,para,k,0,n)
  ll=as.double(l[7])
  ll = -ll
}

s.a1 = exp(-sur1$coef[1])
s.b1 = 1/sur1$scale
s.a2 = exp(-sur2$coef[1])
s.b2 = 1/sur2$scale

probit = glm(vekt.ind~gest,family=binomial(link=”probit”),
  data=ord.mfr)

scale1 = summary(sur1)$scale
scale2 = summary(sur2)$scale

start.par = c(sur1$coef[2]/-scale1,sur2$coef[2]/-scale2,
  sur1$coef[3]/-scale1,sur2$coef[3]/-scale2,log(s.a1),
    log(s.a2),log(s.b1),log(s.b2),probit$coef[1],
      probit$coef[2])

opt = optim(start.par, wrapper, hessian=T, method = ”BFGS”,control=
  list(reltol=10^(-15)))
convergence[j] = opt$conv
}

k = rep(1,n)
opt.full = optim(start.par,wrapper, hessian=T,method=”BFGS”,
  control=list(reltol=10^(-15)))
```

C-code for likelihood function

This is the likelihood function for the full likelihood called from the R-code above.

```
#include <R.h>
#include <Rmath.h>
/*equals func.*/
int I(int f,int a)  {
        if(f == a)
                return 1;
        else
                return 0;
}
/*unequals func.*/
int iI(int f,int a)  {
        if(f != a)
                return 1;
        else
                return 0;
}

double A1(double a, double b, double t)  {
  return pow(a*t,b);
}

/*Weibull baseline*/
double baseline1(double a, double b, double t)  {
  return pow(a,b)*b*pow(t,b-1);
}

double rr(int Z5, int Z6, double g5, double g6) {
  return exp(g5*Z5+g6*Z6);
}

double probit(int Z5, double beta0, double beta1)  {
  double mu = 0;
  double sigma = 1;

  int give_log = 0;
  int lower_tail = 1;
  return(pnorm(beta0+Z5*beta1,mu,sigma,lower_tail,give_log));
}


/*likelihood*/
double lik_mfr_probit_gest(int *status, double *t, int *Z5, int *Z6,
 double *para, int *kk, int *nn) {
  double g15 = para[0];
  double g25 = para[1];
  double g16 = para[2];
  double g26 = para[3];
  double a1 = exp(para[4]);
  double a2 = exp(para[5]);
  double b1 = exp(para[6]);
  double b2 = exp(para[7]);
  double beta0 = para[8];
  double beta1 = para[9];
  int n = nn[0];
  double lO[n];
  double lIO[n];
  double L = 0;


  for(int i = 0; i <n; i++) {
  /*Contribution from cases or controls*/
    lO[i] = log(pow(pow(baseline1(a1,b1,t[i])*rr(Z5[i],Z6[i],g15,g16),
            I(status[i],1))*
            pow(baseline1(a2,b2,t[i])*rr(Z5[i],Z6[i],g25,g26),
            I(status[i],2))*exp(-A1(a1,b1,t[i])*
            rr(Z5[i],Z6[i],g15,g16)-A1(a2,b2,t[i])*
            rr(Z5[i],Z6[i],g25,g26))*
            pow(probit(Z5[i],beta0,beta1),Z6[i])*
            pow(1-probit(Z5[i],beta0,beta1),1-Z6[i]),iI(kk[i],0)));

    /*Contribution from individuals outside the subcohort*/
    lIO[i] = log(pow((exp(-A1(a1,b1,t[i])*rr(Z5[i],0,g15,g16)-
             A1(a2,b2,t[i])*rr(Z5[i],0,g25,g26))*
             (1-probit(Z5[i],beta0,beta1)))+
             (exp(-A1(a1,b1,t[i])*rr(Z5[i],1,g15,g16)-
             A1(a2,b2,t[i])*rr(Z5[i],1,g25,g26))*
             probit(Z5[i],beta0,beta1)),I(kk[i],0)));
    L = L+lO[i]+lIO[i];
  }
  return L;
}

/*status - ind.status
  t - survival time
  Z5 - gestational age in days
  Z6 - weight at time of birt 0 if weight < 3kg, 1 if weight >= 3kg
  para - likelihood parameters
  k - indicator indicating whether or not Z1 is observed
  ll - likelihood value, should be 0 when the function is called
*/
void loglik_mfr_probit_gest(int *status, double *t, int *Z5, int *Z6,
  double *para, int *k, double *ll, int *n) {
  ll[0] = lik_mfr_probit_gest(status,t,Z5,Z6,para,k,n);
}
```

C-code for likelihood function with Monte Carlo approximation

```
#include <R.h>
#include <Rmath.h>
/*equals func.*/
int I(int f,int a) {
        if(f == a)
                return 1;
        else
                return 0;
}
/*unequals func.*/
int iI(int f,int a) {
        if(f != a)
                return 1;
        else
                return 0;
}

```

```
double cor(double Z, double mu, double sigma, double mu0,
  double sigma0) {
  return (dnorm(Z,mu,sigma,0)/dnorm(Z,mu0,sigma0,0));
}
/*likelihood with one partially observed covariat*/
double lik_mfr_crude_MC(int *status, double *t, double *Z,
  double *zsamp,double *para, int *kk, int *nn, double *mu00,
    double *sigma00) {
  double g1 = para[0];
  double g2 = para[1];
  double a1 = exp(para[2]);
  double a2 = exp(para[3]);
  double b1 = exp(para[4]);
  double b2 = exp(para[5]);
  double mu = para[6];
  double sigma = exp(para[7]);
  double mu0 = mu00[0];
  double sigma0 = sigma00[0];
  int n = nn[0];
  double lO[n];
  double lIO[n];
  double L = 0;
  int teller = 0;
  double lio = 0;
  for(int i = 0; i <n; i++) {
    if(kk[i] != 0) {
    /*Contribution from cases or controls*/
      lO[i] = log(pow(((pow(a1,b1))*b1*(pow(t[i],(b1-1)))*
      exp(g1*Z[i])),(I(status[i],1)))*pow(((pow(a2,b2))*b2*
      (pow(t[i],(b2-1)))*exp(g2*Z[i])),(I(status[i],2)))*
      exp(-pow((a1*t[i]),b1)*exp(g1*Z[i])-pow((a2*t[i]),b2)*
      exp(g2*Z[i]))*dnorm(Z[i],mu,sigma,0));
      L = L + lO[i];
    }
    else {
    /*Contribution from individuals outside the subcohort*/
      for(int j = 0; j < 100; j++) {
        lio = lio + log((exp(-pow((a1*t[i]),b1)*exp(zsamp[teller]*g1)-
        pow((a2*t[i]),b2)*exp(zsamp[teller]*g2)))*
        cor(zsamp[teller],mu,sigma,mu0,sigma0));
        teller = teller + 1;
      }
      lIO[i] = lio/100;
      lio = 0;
      L = L+lIO[i];
    }
  }
  return L;
}
/*status - ind.status
  t - survival time
  Z6 - weight at time of birt
  para - likelihood parameters
  k - indicator indicating whether or not Z1 is observed
  ll - likelihood value, should be 0 when the function is called
*/
void loglik_mfr_crude_MC(int *status, double *t, double *Z6,
                         double *zsamp, double *para, int *k,
                         double *ll, int *n, double *mu,
                         double *sigma) {
  ll[0] = lik_mfr_crude_MC(status,t,Z6,zsamp,para,k,n,mu,sigma);
}
```

Bibliography

[1] O. O. Aalen, Ø. Borgan, and H. K. Gjessing. *Survival and Event History Analysis.* Statistics for Biology and Health. Springer, first edition, 2008.

[2] W. E. Barlow. Robust variance estimation for the case-cohort design. *Biometrics*, 50(4):1064–1072, Dec 1994.

[3] Ø. Borgan, B. Langholz, S. O. Samuelsen, L. Goldstein, and J. Pogoda. Exposure stratified case-cohort designs. *Lifetime Data Analysis*, 6:39–58, Mar 2000.

[4] N. E. Breslow, T. Lumley, C. M. Ballantyne, L. E. Chambless, and M. Kulich. Improved horvitz-thompson estimation of model parameters for two-phase stratified samples: Applications in epidemiology. *Stat. Biosci*, 1(1):32–49, May 2009.

[5] N. E. Breslow, T. Lumley, C. M. Ballantyne, L. E. Chambless, and M. Kulich. Using the whole cohort in the analysis of case-cohort data. *American Journal of Epidemiology*, 169(11):1398–1405, Feb 2009.

[6] K. N. Chen. Generalized case-cohort sampling. *J. Roy. Staist. Soc. Ser. B*, 63(4):791–809, 2001.

[7] D. R. Cox. Regression models and life tables. *J. R. Statist. Soc B*, 34(2):187–220, 1972.

[8] D. R. Cox. Partial likelihood. *Biometrika*, 62(2):269–276, 1975.

[9] D.R. Cox and D. Oakes. *Analysis of Survival Data.* Monographs on Statistics and Applied Probability. Chapman and Hall, first edition, 1984.

[10] J. C. Deville, C. E. Sarndal, and O. Sautory. Generalized raking procedures in survey sampling. *Journal of the American Statistical Association*, 88(423):1013–1020, 1993.

[11] J. C Deville and C. E. Särndal. Calibration estimators in survey sampling. *Journal of the american statistical association*, 87(418):376–382, Jun 1992.

[12] J. D. Kalbfleisch and J.F. Lawless. Likelihood analysis of multi-state models for disease incidence and mortality. *Statistics in Medicine*, 7(1-2):149–160, Jan-Feb 1988.

[13] S. Kulathinal and E. Arjas. Bayesian inference from case-cohort data with multiple end-points. *Scandinavian Journal of Statistics*, 33(1):25–33, Jan 2006.

[14] B. Langholz and Ø. Borgan. Counter-matching: A stratified nested case-control sampling method. *Biometrika*, 82(1):69–79, Mar 1995.

[15] D. Y. Lin M. Kulich. Improving the efficiency of relative-risk estimation in case-cohort studies. *Journal of the american statistical association*, 99(467):832–844, Sep 2004.

[16] R. L. Prentice. A case-cohort design for epidemiologic cohort studies and disease prevention trials. *Biometrika*, 42(1):1–11, Apr 1986.

[17] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2007. ISBN 3-900051-07-0.

[18] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer, second edition, 2004.

[19] O. Saarela, A. Kulathinal, E. Arjas, and E. Läärä. Nested case-control data utilized for multiple outcomes: A likelihood approach and alternatives. *Statist. Med.*, 27:5991–6008, Sep 2008.

[20] S. O. Samuelsen. A pseudolikelihood approach to analysis of nested case-control studies. *Biometrika*, 84(2):379–394, Jun 1997.

[21] S. O. Samuelsen, P. Magnus, and L. S. Bakketeig. Birth weight and mortality in childhood in norway. *American Journal of Epidemiology*, 148(10):983–991, Apr 1998.

[22] S. O. Samuelsen, H. Ånestad, and A. Skrondal. Stratified case-cohort analysis of general cohort sampling designs. *Scandinavian Journal of Statistics*, 34(1):103–119, Mar 2007.

[23] T. H. Scheike and A. Juul. Maximum likelihood estimation for cox's regression model under nested case-control sampling. *Biostatistics*, 5(2):193–206, Apr 2004.

[24] S. G. Self and R. L. Prentice. Asumptotic distributions theory and efficiency results for case-cohort studies. *The Annals of Statistics*, 16(1):16–81, 1988.

[25] C. E. Särndal, B. Swensson, and J. H. Wretman. The weighted residual technique for estimating the variance of the general regression estimator of the finite population total. *Biometrika*, 76(3):527–537, Sep 1989.

[26] T. M. Therneau and P. M. Grambsch. *Modeling Survival Data: Extending the Cox Model*. Springer-Verlag, 2000.

[27] D. C. Thomas. Addendum to "Methods of cohort analysis: appraisal by application to asbestos mining" by F. D. K Liddell, J. C. McDonald and D. C. Thomas. *J. Roy. Stat. Soc.*, A 140:469–491, 1977.