Stockholms universitet

Estimation of the risk of the occurrence of events of interest over time with applications to medical data

Emilia Forslin

Masteruppsats i matematisk statistik
Master Thesis in Mathematical Statistics

Masteruppsats 2023:10
Matematisk statistik
Juni 2023

www.math.su.se

Matematisk statistik
Matematiska institutionen
Stockholms universitet
106 91 Stockholm

Mathematical Statistics
Stockholm University
Master Thesis **2023:10**
http://www.math.su.se

Estimation of the risk of the occurrence of events of interest over time with applications to medical data

Emilia Forslin[*]

June 2023

Abstract

In medical research it is common to study the time to the occurrence of a single event that may only happen once per individual. One such typical event is the occurrence of a death and the studying of the survival time. Even though the analysis of this type of one-time-only event is very common, classical theory in event history analysis fail to give a logical measurement of the risk of occurrence of these events.

In the article by Bottai (2017) the theoretical ground for two new measurements, for these type of events, were presented; the incidence rate and the event-probability function. s The aim of this thesis is to make the usability of these two measurements visible and easily accessible by giving a thorough theoretical explanation, presenting new software, and providing a real data example.

[*] Postal address: Mathematical Statistics, Stockholm University, SE-106 91, Sweden. E-mail: emilia.forslin@icloud.com. Supervisor: Matteo Bottai.

Acknowledgements

First and foremost, I want to thank my supervisor Matteo Bottai for the idea of this thesis and for all help and support. You are a source of inspiration and I am really grateful for the opportunity I have been given having you as a supervisor.

I also want to thank Pär Villner for always showing such an interest in helping, not only me, but everyone around you. Never stop being that person! And to all my other colleagues at K.I. for an inspiring and fun time at the Division of Biostatistics.

Thank you Raid Saffar for informing me of our obligation to give back to the world in knowledge what we get in the opportunity to learn. Far from everyone get the chance to study.

And thank you Tommy, thank you Nina, thank you mom Ann-Kristin, and thank you grandma Mirjam, you make me thrive.

Thank you dad Bengt, you always where the unwavering support, but little did you know of what role you would play in my educational path.

Contents

1 Background

1.1 Introduction

In medical research, some of the most frequently used methods, in the search of answering the big questions of human health, lie within the field of *event history analysis*. Event history analysis, also know as *survival analysis*, is a family of statistical methods used for studying and analysing the time to, and the occurrence of, one or several events.

Many researchers study the occurrence of a one-time-only event of an individual, such as death or first cancer diagnosis. Even though the analysis of this type of one-time-only events is very common, classical theory in event history analysis fails to give appropriate measures of the risk of occurrence of these events.

In the article *A Regression Method for Modelling Geometric Rates* [Bottai, 2017], the theory of two new measurements in event history analysis, for the one-time-only case, were presented. These measurements were later defined as the *incidence risk* and the *event-probability function* [Bottai et al., 2021, Bottai, 2022].

The incidence risk is the geometric mean of the probability of occurrence of an event per unit time in a time interval, given that it has not happened yet. As shown by Bottai [2022], it is naturally connected to the cumulative hazard and can be estimated by the use of the Nelson-Aalen estimate of cumulative hazards, see Subsection 2.6.

Compared to the classical estimate of the incidence rate, which is an arithmetic mean of the same probability, the incidence risk is a measurement of risk, while the incidence rate is not.

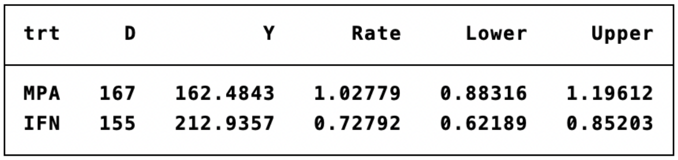
The event-probability function is the limit of the incidence risk, and it is the instantaneous risk of the occurrence of an event, given that it has not occurred yet. Compared to the hazard rate, this is a probability measure, which gives it an advantage when it comes to interpretability.

1.2 Motivating example

The following is a motivating example of a comparison of two measurements, the incidence rate and the incidence risk. As already mentioned, the incidence rate is a classical measure in event history analysis and is often reported in medical research, while the incidence risk is the new measurment presented by Bottai [2022].

These estimates are computed and compared in Stata by using one of Stata's built-in datasets, `kidney`. This data contains right censored survival times of patients with a diagnosis of metastatic renal carcinoma [Medical Research Council Renal Cancer Collaborators, 1999]. The theory and the logic of this example are found in the article by Bottai [2022], in which the incidence risk is presented and defined together with the new command `stprisk` for estimating it. Both the theory and a discussion about the use case of the incidence risk are revisited in later sections.

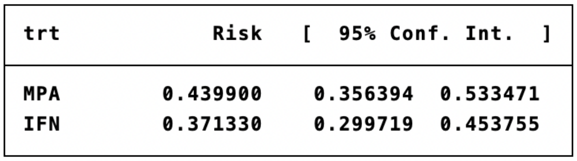
The incidence *rate* is estimated for the group variable `trt` containing two treatment groups, subcutaneous interferon-α (IFN) and oral medroxyprogesterone acetate (MPA). The incidence rate is an arithmetic mean of the occurrence of an event per individual and unit time. It is computed by counting all occurrences of the event and dividing it by the sum of all event times. The command `strate` on `trt` gives estimated mortality rates for the two treatment groups:



```
trt     D          Y      Rate     Lower     Upper

MPA   167   162.4843   1.02779   0.88316   1.19612
IFN   155   212.9357   0.72792   0.62189   0.85203
```

The question arises of how one interprets a mortality rate of 1.023 per unit time and person. Some might read the incidence rate as a measurement of risk but it should be clear that this is not a probability measurement.

The incidence *risks* is then estimated for the two treatment groups by running the command `stprisk` on `trt`:



```
trt            Risk   [  95% Conf. Int.  ]

MPA        0.439900    0.356394  0.533471
IFN        0.371330    0.299719  0.453755
```

The incidence risk is a geometric mean of the occurrence of event per unit time and person and, as shown later, this is a measurement of risk. We can interpret 0.44 as a 44% mortality risk per year, for an individual in treatment group MPA, and compare it to a 37% mortality risk per year, for an individual in treatment group IFN.

1.3 Goals

The goal of this thesis is to make the usability of the two measurements, the incidence rate, and the event-probability function, visible and easily accessible, both theoretically and practically. This is done in three steps:

(i) By giving a thorough theoretical explanation of the incidence risk, and the event-probability function,

(ii) by presenting two new software implementations of the incidence risk and the event-probability function, and

(iii) by giving a visual explanation of theory, through a real data example.

The first step (i) is done by first building a solid foundation of theory of event history analysis, and then new theory is introduced from it, all in Section 2. Hence, the method section can be read with or without the *Event history analysis* subsection.

The second step (ii) is done in Section 3, *Software implementation*, and the third step in Section 4, *A real data example.*

2 Methods

2.1 Introduction

Event history analysis, also referred to as *survival analysis*, is a family of statistical methods developed mainly within the field of medical research. As the second name suggests, the classical target of event history analysis has been to study *survival time*; the time to the occurrence of death, or any other event of interest.

Theory within event history analysis is used beyond the field of medical research and one can refer to *survival time* as *time-to-event.* The time of interest could for example be the time to giving birth to a first child, the lifetime of a light-bulb or the time to getting divorced after getting married. Events could be of a one-time-only nature, like the examples given above, but they could also be events with the possibility of reappearing; like the event of having a stroke or getting married.

Theory of this methods section is divided into four subsections. In Subsection 2.3 basic concepts of event history analysis is covered together with theory of the counting process and the martingale.

In Subsection 2.4 the Nelson-Aalen estimate of cumulative hazards is explained and proven, using theory of the counting process.

In Subsection 2.5 some regression models in event history analysis are explained and in Subsection 2.6, which is the last of the methods section, show theory of the incidence risk and the event-probability function, with some applications.

2.2 Notation and setup

These are the notations used throughout this thesis. Most of them are widely used notations and are easily recognized from books and papers.

t - Observed time-to-event

$t-$ - The time just before some time t

T - The random variable time-to-event

n - Number of observations

$f(t)$ - Probability density function

$F(t)$ - Cumulative probability function

$S(t)$ - Survival function

$h(t)$ - Hazard

$H(t)$ - Cumulative hazard

$\mathcal{F}(t)$ - Filtration

$M(t)$ - Martingale

$\langle M \rangle(t)$ - Predictable variation process

$[M](t)$ - Optional variation process

$J(t)$ - Stochastic integral

$N(t)$ - Counting process

$\lambda(t)$ - Intensity function of a counting process

$\Lambda(t)$ - Cumulative intensity function of a counting process

$G(t)$ - Predictable process

$B(t)$ - Brownian motion process

$I(t)$ - Indicator function

$Y(t)$ - Number at risk at time $t-$

$r_n(t)$ - Residual of the Nelson-Aalen estimator of cumulativ hazards

$\boldsymbol{x} = (x_1, .., x_q)'$ - Vector of q covariates

$\boldsymbol{\beta} = (\beta_1, .., \beta_q)'$ - Vector of q regression coefficients

$\psi(\boldsymbol{x}'\boldsymbol{\beta})$ - Hazard rate ratio function

$L(\theta)$ - Likelihood function

$l_n(\theta)$ - Log-likelihood function

$s(x)$ - Spline function

$G(t, t + dt)$ - Incidence risk

$g(t)$ - Event-probability function

$\Sigma(\theta)$ - Variance covariance matrix

$I(\theta)$ - Fisher information matrix

$\mathcal{H}(\theta)$ - Hessian matrix

2.3 Event history analysis

In this subsection basic concepts of event history analysis are covered and explained using theory of the counting process and the martingale. If not otherwize cited, the logical trail follow that of chapter 1 and 2 in the book by Aalen et al. [2008]. This is a theoretical foundation on which classical applications and new concepts are added in later subsection.

2.3.1 Survival function and hazard

As established earlier, the main theory within event history analysis focuses on analysing some *time-to-event* variable. At the base of the theory lie the *survival function*, the *hazard function* and the *cumulative hazard function.*

If one view the time-to-event as a random variable T defined on the positive real line, then the *survival function* $S(t)$ is the probability of *surviving*, or being *event free*, up to time t. Hence, the survival function can be written as

$$\begin{aligned} S(t) &= P(T > t) \\ &= 1 - F(t), \end{aligned}$$

where $F(t) = P(T \leq t)$ is the cumulative probability of seeing an event before time t. The cumulative distribution function $F(t)$ is also naturally called the *failure function.*

The *hazard function* or the *hazard rate* is the instantaneous rate of occurrence of an event per unit time at time t, given that it has not occurred yet. It is obtained by taking the limit of the conditional probability of T being in a small interval $[t, t + dt)$ divided by the length of the interval dt

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

In order to connect the hazard function and the survival function, one can use that $P(t \leq T < t + dt|T \geq t) = P(t \leq T < t + dt)/P(T \geq t)$. Then the hazard rate can be rewritten as the limit of the failure and the survival function

$$\begin{aligned} h(t) &= \lim_{dt \to 0} \left\{ \frac{F(t + dt) - F(t)}{dt} \right\} \frac{1}{S(t)} \\ &= \frac{f(t)}{S(t)}, \end{aligned} \tag{2}$$

where $f(t)$ is the derivative of the failure function $F(t)$ [Collett, 2003][page 12-13]. Taking a closer look at Equation 2, one could notice that this is in fact the derivative of $-\log\{S(t)\}$. Also, the *cumulative hazard function*

$$H(t) = \int_0^t h(x)dx,$$

is the integral of the hazard function up to time t. These two facts together give us the relation between the cumulative hazard function and the survival function

$$\begin{aligned} H(t) &= \int_0^t \frac{f(x)}{S(x)} dx \\ &= -\log\{S(t)\}, \end{aligned}$$

and this is equivalent to

$$S(t) = e^{-H(t)}. \tag{3}$$

The *survival function*, the *hazard function* and the *cumulative hazard function* all uniquely defines the time-to-event variable T.

2.3.2 The typical time-to-event data

When studying an event that may only occur once, a few times, or may not happen at all to an individual over a lifetime, one may not observe all events occurring in the sample population. It could be that the actual time to the occurrence of the event for an individual is longer than the time of the research period. This means that when the research period is ended some individuals are still at risk of having the event occurring later on in their life, we typically call these observations *censored*.

If for example one study the survival time of cancer patients, as in the example of the `kidney` data in Section 1.2, it is a fact that all individuals die, whether it is by cancer or a natural death, but not everyone will die during the time of the research period. If this is the case one have the limited information of the survival time being longer than the length of the study, but the survival time is censored beyond that.

There are different type of censoring where the type described above is called *right censored*; one can not see 'to the right' on the time line, if and when an event occurs. Then there is *left censored* event times; for example when checkups are only done on discrete time points, and if an individual is found to have an occurrence of the event of interest on one of those checkups, we do not know at what time between the last checkup and the current one it happened.

Also, there is the case where individuals entering the research have an unknown starting point of the time-to-event. As an example, if one want to study the time from a cancer diagnosis to the time of death, some of those that enter the study may have had cancer some time before they got their diagnosis, this is called *left truncation*.

The methods presented in this chapter are applicable to right censored event times and these are some times referred to as just censored event times.

The typical survival data with n observations contains observed *event times* $t_1, t_2, .., t_n$ defined on the positive real line. If censored event times are included, these are indicated by some *event indicator variable* $d_1, d_2, .., d_n$ that typically takes on the value 0 for censored and 1 for uncensored.

2.3.3 The past

When the subject of an analysis is a sequences of random variables presented over time, the past is often considered when looking at the presence. For a time t, we denote the time up to some time just before t as $t-$. For a sequence of random variables $\{X_n\}_{n\geq 0}$ generated by a process $X(t)$, where t represents the present time, the past can be presented by a *filtration* $\mathcal{F}_{t-}$. This filtration is then a collection of σ-algebras generated by all earlier sequences of random variables up to the time $t-$, it contains all possible outcomes of the past.

2.3.4 The martingale

The martingale is a sequence of random variables $\{M_n\}_{n\geq 0}$, typically generated over time, where the expected value of the process at the present time equals the most recent value of the process in the past, given that the past is known. This definition of the martingale is expressed by *the martingale property*, and for discrete time points labeled $n = 1, 2, ..$ this can be formally expressed as

$$E[M_n|\mathcal{F}_{n-1}] = M_{n-1},$$

where $\mathcal{F}_{n-1}$ is the past up to the discrete time point $n-1$. The rest of the theory of the *discrete time martingale* is left-out and the martingale generated in continuous time is considered. The continuous time case is an augmentation of theory of the discrete case and both are, for the interested reader, covered in the book by Aalen et al. [2008].

Let the random variables $\{M_n\}_{n\geq 0}$ be generated by a process $M(t)$ over continuous time t, defined on $[0, \tau]$. This process is a *continuous time martingale* if it fulfills *the martingale property* given by

$$E[M(t)|\mathcal{F}_{t-}] = M(t-), \tag{4}$$

where $M(t-)$ is the value of the process at some time just before t. By properties of expectations, Equation 4 can also be expressed as

$$E[M(t)|\mathcal{F}_s] = M(s), \tag{5}$$

for all $s < t$. Also, if the starting point of the martingale at time 0 is $M(0) = 0$, one can use the *tower property of expectations* together with Equation 5, to show that

$$\begin{aligned} E[M(t)] &= E[E[M(t)|\mathcal{F}_0]] \\ &= E[M(0)] \\ &= 0 \end{aligned}$$

for all $t \geq 0$. This is the *zero mean martingale* for which a change over a small time interval $[t, t+dt)$, by linearity of expectations and Equation 5, is given by

$$\begin{aligned} E[dM(t)|\mathcal{F}_{t-}] &= E[M(t+dt) - M(t)|\mathcal{F}_{t-}] \\ &= E[M(t+dt)|\mathcal{F}_{t-}] - E[M(t)|\mathcal{F}_{t-}] \\ &= 0. \end{aligned} \tag{6}$$

The zero mean martingale is a powerful tool to show properties of residuals in *stochastic process theory*, and all martingales mentioned below are assumed to be of zero mean. Equation 4, 5 and 6 are three important expressions of the martingale property and these are utilized when showing applications of martingale theory in event history analysis.

2.3.5 The variation processes of the martingale

For a martingale $M(t)$ in continuous time, the *predictable variation process* $\langle M\rangle(t)$ and *the optional variation process* are defined as

$$\langle M\rangle(t) = \lim_{n\to\infty} \sum_{k=1}^{n} Var(\Delta M_k | \mathcal{F}_{(k-1)t/n}), \tag{7}$$

and

$$[M](t) = \lim_{n\to\infty} \sum_{k=1}^{n} (\Delta M_k)^2, \tag{8}$$

where M_k is a discrete time martingale and $\Delta M_k = M(kt/n) - M((k-1)t/n)$.

The predictable variation process is related to the conditional variance of the increment $dM(t)$ by

$$d\langle M\rangle(t) = Var(dM(t)|\mathcal{F}_{t-}),$$

where the increment is over a small time interval $[t, t+dt)$. One can show that $M^2(t) - \langle M\rangle(t)$ and $M^2(t) - [M](t)$ are both zero mean martingales, and that

$$\begin{aligned} Var(M(t)) &= E[M(t)^2] \\ &= E[\langle M\rangle(t)] \\ &= E[[M](t)] \end{aligned}$$

is the variance of the martingale $M(t)$ at time t. Also the predictable variation process of the sum of two mean zero martingales can be written as

$$\langle M_1 + M_2\rangle = \langle M_1\rangle + \langle M_2\rangle + 2\langle M_1, M_2\rangle \tag{9}$$

where $\langle M_1, M_2\rangle$ is the predictable covariance process of the two martingales M_1 and M_2.

2.3.6 The stochastic integral

The martingale property can be preserved under many transformations, making it useful in many applications. One of these transformations is the *stochastic integral*, which has an important role in the derivation of the *Nelson-Aalen estimate* of the cumulative hazard in Subsection 2.4.3.

The stochastic integral $J(t)$ is a stochastic process generated on a filtration $\mathcal{F}_t$, including a predictable part $G(t)$, known at time $t-$, and a stochastic part $dM(t)$, fulfilling the martingale property in Equation 6. Then this stochastic integral is defined as

$$J(t) = \int_0^t G(s)dM(s), \tag{10}$$

and it is a zero mean martingale with respect to the filtration $\mathcal{F}_{t-}$.

This can be shown by looking at a process $\{Z_n\}_{n\geq 0}$, defined in *discrete* time but otherwise in the same way, with one predictable part $\{G_n\}_{n\geq 0}$ and one stochastic martingale $\{M_n\}_{n\geq 0}$. Then Z_n equals the sum

$$Z_n = \sum_{i=0}^{n} G_i \Delta M_i,$$

where $\Delta M_0 = M_0$ and $\Delta M_i = M_i - M_{i-1}$. Taking the expected value of the difference $Z_n - Z_{n-1}$ and using *linearity of expectations*, as follows

$$\begin{aligned} E[Z_n - Z_{n-1}|\mathcal{F}_{n-1}] &= E[G_n(M_n - M_{n-1})|\mathcal{F}_{n-1}] \\ &= G_n E[M_n - M_{n-1}|\mathcal{F}_{n-1}] \\ &= 0, \end{aligned}$$

take us to the conclusion that $\{Z_n\}_{n\geq 0}$ is a martingale in discrete time, fulfilling the martingale property in Equation 6. Then $J(t)$ can be expressed as a limit of Z_n by dividing the interval $[0, t]$ in to n intervals of length t/n and letting n go to infinity. Hence, this limit is given by

$$J(t) = \lim_{n\to\infty} \sum_{k=0}^{n} G_k \Delta M_k,$$

where $G_k = G((k-1)t/n)$ and $\Delta M_k = M(kt/n) - M((k-1)t/n)$, and it is the zero mean martingale in Equation 10. This shows that the martingale property is preserved under stochastic integration.

The *predictable variation process* and the *optional variation process*, Equation 7 and 8, of the stochastic integral are given by

$$\Big\langle \int GdM \Big\rangle = \int G^2 d\langle M \rangle, \tag{11}$$

$$\Big[\int GdM \Big] = \int G^2 d[M]. \tag{12}$$

2.3.7 The Doob-Meyer decomposition

The Doob-Mayer (D-M) decomposition gives another transformation of a stochastic process for which useful characteristics can be obtained. First the definition of a special type of stochastic process called a *sub martingale* is given, then it is to this process the D-M decomposition is applied.

A *sub martingale* $X(t)$, for $t \in [0, \tau]$, is a non decreasing process that for all $s < t$ fulfills the *sub martingale property* given by

$$E[X(t)|\mathcal{F}_{t-}] \geq X(s). \tag{13}$$

To this sub martingale $X(t)$ one can apply the D-M decomposition, which is unique and given by

$$X(t) = X^*(t) + M(t), \tag{14}$$

where $X^*(t)$ is a predictable process and $M(t)$ is a martingale. Another way of expressing this is by saying that the increment of X^* is given by

$$dX^*(t) = E[dX(t)|\mathcal{F}_{t-}],$$

and the increment of $M(t)$, the residual, is given by

$$dM(t) = dX(t) - E[dX(t)|\mathcal{F}_{t-}].$$

2.3.8 Counting process

A counting process $N(t)$ is a stochastic process, describing the number of occurrences of an event up to some time t, where $N(t)$ is right continuous from zero and integer valued. The process increases one point whenever there is an event time, and for $s < t$ the difference $N(t) - N(s)$ is the number of events occurring in the time interval $(s, t]$.

The counting process is a non decreasing process fulfilling the *sub martingale property* of Equation 13, and hence, it is a sub martingale.

The *intensity function* $\lambda(t) > 0$ of a counting process $N(t)$ is the conditional probability of a new event occurring in a small time interval, given the past, divided by the length of the interval. Let $\mathcal{F}_{t-}$ be the past, then the change in intensity of the counting process, in a very small time interval $[t, t + dt)$, can be expressed as

$$\lambda(t)dt = P(dN(t) = 1|\mathcal{F}_{t-}). \tag{15}$$

In this very small time frame dt one can expect the process to increase with one event or none, and in this context the increment $dN(t)$ can be view as a binary variable. Hence, the probability of $dN(t) = 1$ equals the expected value, and

$$\lambda(t)dt = E[dN(t)|\mathcal{F}_{t-}]. \tag{16}$$

Now, using Equation 16 together with the D-M decomposition (14) one can define a process $M(t)$ such that

$$M(t) = N(t) - \int_0^t \lambda(s)ds, \tag{17}$$

is a martingale. It is the uniquely defined decomposition of the sub martingale $N(t)$ into a predictable part $\int_0^t \lambda(s)ds$ and a zero mean martingale $M(t)$. This implies that

$$dM(t) = dN(t) - \lambda(t)dt, \tag{18}$$

and taking expectations on both sides of the equation give

$$\begin{aligned} E[dM(t)|\mathcal{F}_{t-}] &= E[dN(t) - \lambda(t)dt|\mathcal{F}_{t-}] \\ &= 0. \end{aligned}$$

The process $M(t)$ fulfills the martingale property of Equation 4, and one can conclude that $M(t)$ is a martingale. The integral of the intensity function, called the *cumulative intensity process*,

$$\Lambda(t) = \int_0^t \lambda(s)ds,$$

can be seen as the signal of the process. When subtracting the signal from the process $N(t)$, one get a zero mean martingale $M(t)$, the randomness of the process. To this martingale $M(t)$ the cascade of martingale theory that follows with it can be applied, which gives a logical way of presenting methods used for computing measurements of the counting process.

The derivation continues with the predictable and the optional variation process, starting with the latter, which is somewhat natural. Looking at the definition of the optional variation process in Equation 8 one can notice that, as $n \to \infty$, time intervals gets smaller and smaller, the increment $(\Delta M_k)^2$ will either equal one or none and

$$[M](t) = N(t). \tag{19}$$

In the case of the predictable variation process (7), one can consider the increment $d\langle M\rangle(t)$, which, by Equation 18 and the fact that $\lambda(t)$ is predictable, can be written as

$$\begin{aligned} d\langle M\rangle(t) &= Var(dN(t) - \lambda(t)dt|\mathcal{F}_{t-}) \\ &= Var(dN(t)|\mathcal{F}_{t-}). \end{aligned}$$

This variance can, due to the binomial behavior of $dN(t)$ in a small time frame, be computed by taking the expected value of $dN(t)$. It follows from Equation 16 that

$$d\langle M\rangle(t) \approx \lambda(t)dt. \tag{20}$$

Now the predictable variation process of the counting process martingale $M(t) = N(t) - \Lambda(t)$ can be expressed as

$$\langle M\rangle(t) = \int_0^t \lambda(s)ds,$$

giving the name a logical explanation, because it is the integral of the predictable part of the process.

One should also consider a case where two counting processes N_1 and N_2 have jump times in continuous time, and hence, they have no jump times at the same time. This implies that the two corresponding martingale processes M_1 and M_2 are independent and

$$\langle M_1, M_2\rangle = 0,$$

for all t.

Last but not least, theory of the counting process martingale (17) can be connected with the stochastic integral (10) by re-expressing the stochastic integral as

$$J(t) = \int_0^t G(s)dN(s) - \int_0^t G(s)\lambda(s)ds.$$

Then the predictable variation process of $J(t)$, in Equation 11, together with Equation 20, can be rewritten as

$$\langle J\rangle(t) = \int_0^t G^2(s)\lambda(s)ds. \tag{21}$$

2.3.9 The martingale central limit theorem

The martingale central limit theorem is given here without an extensive explanation. For a more thorough proof, see Rebolledo [1980].

One can begin by defining a *Brownian motion process* $B(t)$ as a continuous time process with continuous sample path where $B(t) - B(s)$ is normally distributed with expected value

$$E[B(t) - B(s)] = 0,$$

and variance

$$Var[B(t) - B(s)] = t - s$$

for $s < t$. Also, the process $U = B(V(t))$ where $V(t)$ is a strictly increasing process and $V(0) = 0$, is called the Gaussian martingale and just as the Brownian motion its increment, $B(V(t)) - B(V(s))$ on $(s, t]$, is a zero mean martingale with predictable variation process $\langle U \rangle = V(t)$. The Gaussian martingale is uniquely determined by its variation process.

An increasing sequence of continuous time martingales $M^{(n)} = \{M_n(t)\}_{n \geq 0}$ produced by a sequence of counting processes $\{N_n(t)\}_{n \geq 0}$, as in Equation 17, and normalized by some proper factor, converges toward this Gaussian martingale if

(i) the predictable variation process $\langle M^{(n)}(t) \rangle$ converges to a deterministic function, and

(ii) jumps between values in the sequence converges toward zero.

For an increasing sequence of mean zero martingales $M^{(n)}(t)$ defined on some interval $[0, \tau]$, one can define $M_\varepsilon^{(n)}(t)$ as all jumps greater than some $\varepsilon > 0$. Expressed more formally, the martingale central limit theorem then says that for some strictly increasing function $V(t)$, where $V(0) = 0$, if

(i) $\langle M^{(n)} \rangle(t) \xrightarrow{p} V(t)$, and

(ii) $M_\varepsilon^{(n)}(t) \to 0$,

for all $t \in [0, \tau]$, as $n \to \infty$, then $M^{(n)}(t)$ converges to a Gaussian martingale.

Without immediate explanation but for further use, these two conditions can be rewritten as expressions of the sum of a stochastic integral

$$\sum_{i=1}^{k} J_i^{(n)}(t) = \sum_{i=1}^{k} \int_0^t G_i^{(n)}(s) dM_i^{(n)}(s),$$

where $G_i^{(n)}(t)$ is the predictable part of a counting process for each n, and $dM_i^{(n)}(t)$ the residual. Assuming independence between these integrals for all $i = 1, .., k$, which implies that $\langle J_i^{(n)}, J_j^{(n)} \rangle = 0$ for $i \neq j$, the two conditions *(i)* and *(ii)* can be written with this sum of stochastic integrals. By Equation 9 the two conditions are given by

$$\sum_{i=1}^{k} \int_0^t (G_i^{(n)}(s))^2 \lambda_i^{(n)}(s) ds \xrightarrow{p} V(t), \text{ and} \tag{22}$$

$$\sum_{i=1}^{k} \int_0^t (G_i^{(n)}(s))^2 I\{|G_i^{(n)}(s)| > \varepsilon\} \lambda_i^{(n)}(s) ds \xrightarrow{p} 0, \tag{23}$$

which in both cases should hold for all t in $[0, \tau]$.

2.3.10 Counting events

The theory of *the counting process* is well suited for applications on time-to-event data and is a good way to show concepts in event history analysis. Assuming that event times $\{T_i\}_{i=1}^n$ are independent random variables and that no event time is exactly the same, one can order them in a way that $T_1 < T_2 < ... < T_n$. Then the number of events that have occurred at time t can be described by a counting process $N(t)$, where the process at time T_i equals its label number i.

For one uncensored event time T_i, with hazard rate $h(t)$, one can define a variable $N_i(t) = I\{T_i \leq t\}$, indicating whether the event has happened at time t or not. The intensity function $\lambda_i(t)$ can by the definition of the hazard rate (1) be derived as follows

$$P(dN_i(t) = 1|\mathcal{F}_{t-}) = P(t \leq T < t + dt|\mathcal{F}_{t-}) = \begin{cases} h(t)dt \text{ for } T \geq t \\ 0 \text{ for } T < t \end{cases} \tag{24}$$

for $i = 1, 2, .., n$. In plain words; given that the past up to some time before t is known, one know if $T_i \geq t$, and so, the intensity process $\lambda_i(t) = h(t)I\{T_i \geq t\}$.

Extending this to a *sample* of n uncensored event times $\{T_i\}_{i=1}^n$, one can sum the individual counting processes $\{N_i(t)\}_{i=1}^n$ to an aggregate

$$N(t) = \sum_{i=1}^{n} N_i(t).$$

This aggregated process is the number of events that has happened at time t and it is in itself a counting process with intensity process $\lambda(t)$. Assuming event times have the same underlying hazard rate, $h(t)$, this intensity process is found using Equation 16 and *linearity of expectations*. The aggregated intensity process is then given by

$$\begin{aligned}\lambda(t) &= \frac{E[dN(t)|\mathcal{F}_{t-}]}{dt} \\ &= \sum_{i=1}^{n} \frac{E[dN_i(t)|\mathcal{F}_{t-}]}{dt} \\ &= \sum_{i=1}^{n} \lambda_i(t) \\ &= h(t)Y(t),\end{aligned}$$

where $Y(t) = \sum_{i=1}^{n} I\{T_i \geq t\}$ is the number of individuals at risk just before time t. This is called the *multiplicative form* of the intensity process, and it is intuitively understood by seeing that the hazard rate is the individual probability of experiencing the event per unit time. Multiplying the hazard rate by the number at risk just before time t give the probability of one more occurrence of the event per unit time.

Assuming $\lambda(t)$ is right continuous Equation 17 holds, and one can rewrite Equation 18 as

$$dN(t) = \lambda(t) + dM(t),$$

showing that a change in number of occurrences of the event has a predictable part $\lambda(t)$, in the form of the intensity process, and a residual $dM(t)$, in the shape of a martingale.

The *optional variation process* of the martingale $M(t)$ equals that of Equation 19.

2.3.11 Independent censoring

When working with survival data one can expect the presence of right censored event times, which were briefly explained in Subsection 2.3.2. Some assumptions have to be made about the censored data in order for this theory to be applicable. The weakest assumption needed for the theory discussed in this thesis is that of *independent censoring*.

To explain the concept of independent censoring one can start by imagining a set of n uncensored and independent event times $T_1, T_2, .., T_n$. If all these *true* event times cannot be observed, but rather some of them are censored, one can denote these n *censored* and *uncensored* event times $\tilde{T}_1, \tilde{T}_2, .., \tilde{T}_n$. If an event time is censored; $\tilde{T}_i < T_i$, and if the event time is uncensored; $\tilde{T}_i = T_i$.

The *event indicator variable*, D_i, is set to 0 for all censored event times and 1 for uncensored, for all $i = 1, 2, .., n$. Then the definition of *independent censoring* is expressed as

$$P(t \leq \tilde{T}_i < t + dt, D_i = 1 | \tilde{T}_i \geq t, \mathcal{F}_t) = P(t \leq T_i < t + dt | T_i \geq t).$$

In plain words, independent censoring assumes that the probability of an event occurring within a time interval $[t, t+dt)$, given that the event has not occurred before time t, does not depend on the presence of censored event times.

The theory of the counting process can be extend, assuming independent censoring. For the n censored and uncensored event times, $\{\tilde{T}_i\}_{i=1}^{n}$, the individual counting processes for $i = 1, 2, .., n$ is the indicator variable

$$N_i(t) = I\{\tilde{T}_i \leq t, D_i = 1\},$$

and the aggregated counting is the sum of the individual processes

$$N(t) = \sum_{i=1}^{n} I\{\tilde{T}_i \leq t, D_i = 1\}.$$

Now, one can use the same logic as for the sample of uncensored event times and calculate the intensity process of the aggregated counting process. Equation 24 is rewritten as $\lambda_i(t)dt = P(t \leq \tilde{T}_i \leq t + dt, D_i = 1 | \mathcal{F}_{t-})$, and the aggregated intensity process

$$\begin{aligned} \lambda(t) &= \sum_{i=1}^{n} \lambda_i \\ &= h(t)Y(t), \end{aligned} \tag{25}$$

where $Y(t) = \sum_{i=1}^{n} I\{\tilde{T}_i \geq t\}$ is the number of individuals at risk at some time just before time t. One can conclude that the form of the intensity process is preserved under independent censoring.

2.4 Estimation

In this subsection, the *Nelson-Aalen estimatior of cumulative hazards* is explained, and to some extent proven, using theory of the counting process. Again, the logical trail of the theory follows the book by Aalen et al. [2008], Chapter 3. This estimator is used for calculating the *incidence risk*, explained in Subsection 2.6.

2.4.1 Nelson-Aalen

To estimate the cumulative hazard rate, Nelson (1969, 1972) and Aalen (1970) developed this non-parametric method; the *empirical cumulative hazard estimator*, also called the *Nelson-Aalen* (N-A) *estimator*.

For a sample of n event times $T_1, T_2, ..., T_n$, both censored and observed, the counted number of observed events $N(t)$ is described by a counting process with intensity function given by the multiplicative form in Equation 25. The N-A estimatimator of the cumulative hazard function $H(t) = \int_0^t h(s)ds$ is then given by

$$\hat{H}(t) = \sum_{T_i \leq t} \frac{1}{Y(T_i)}, \tag{26}$$

where $Y(T_i)$ is the number at risk just before the event time T_i.

The intuitive way of understanding this estimate is to think of the time interval $[0, t]$ as a sum of many really small intervals. Then the change in cumulative hazard rate in one small time interval, $[s, s + ds)$, equals $h(s)ds$, the conditional probability of observing the event given that it has not occurred yet.

In this small time frame, either one event is observed or none. Hence, the conditional probability $h(s)ds$ is estimated by one divided by the number at risk just before the time s, in case one event is observed during that time, or zero otherwise. A good estimator of the cumulative hazard is then given by the sum of these estimations of the increments, $h(s)ds$, which is exactly the N-A estimate in Equation 26.

Also, the N-A estimator $\hat{H}(t)$ is approximately normally distributed with estimated variance

$$\hat{\sigma}^2(t) = \sum_{T_i \leq t} \frac{1}{Y(T_i)^2}.$$

The $100(1-\alpha)\%$ confidence interval is given by

$$\hat{H}(t) \pm z_{1-\alpha/2}\hat{\sigma}(t),$$

and these intervals can be improved by a log transformation

$$\hat{H}(t)exp\bigg\{ \pm z_{1-\alpha/2}\hat{\sigma}(t)/\hat{A}(t)\bigg\}. \tag{27}$$

The derivation of the N-A estimator is given in later subsections.

2.4.2 Tied event-times

One can take two different approaches about how to view event-times; as absolute continuous and no event-time can be exactly the same, or event-times being discrete, making it possible for several event-times of the same size.

In the first approach ties arises from the need of rounding of event-times and these are handled by using a special formula for the increment of the N-A estimate $\Delta\hat{H}(T_i)$. If k_i is the number of observed event times at a specific discrete time point T_i, then

$$\Delta\hat{H}(T_i) = \sum_{l=0}^{k_i-1} \frac{1}{Y(T_i) - l}, \tag{28}$$

gives the increment of the N-A estimator and

$$\Delta\hat{\sigma}^2(T_i) = \sum_{l=1}^{k_i-1} \frac{1}{(Y(T_i) - l)^2}$$

the increment of the estimated variance for this N-A estimator.

For the second approach the increment for the N-A estimator is computed as

$$\Delta\hat{H}(T_i) = \frac{k_i}{Y(T_i)}, \tag{29}$$

and the increment of its the estimated variance as

$$\Delta\hat{\sigma}^2(T_i) = \frac{(Y(T_i) - k_i)k_i}{Y(T_i)^3}. \tag{30}$$

2.4.3 The Nelson-Aalen estimate with martingale theory

If one let the number of occurrences of an event at time $t > 0$ be described by the counting process $N(t)$, with intensity process $\lambda(t)$, one can use the multiplicative form in Equation 25 and rewrite Equation 18 as

$$dN(t) = h(t)Y(t)dt + dM(t). \tag{31}$$

Then, in a very small time frame $[t, t + dt)$, the increment $dN(t)$ will at most equals 1 and more often equals 0. One can dividing both sides in Equation 31 by $Y(t)$, the number at risk just before time t, but because $Y(t)$ can equal 0, we also need to include an indicator function $I(t) = I(\{Y(t) > 0\})$. This gives the expression

$$\frac{I(t)}{Y(t)} dN(t) = I(t)h(t) + \frac{I(t)}{Y(t)} dM(t), \tag{32}$$

which equals 0, whenever $Y(t) = 0$. To show that the N-A estimator (26) is a true estimator of $\int_0^t h(t)$, one can define $H^*(t) = \int_0^t I(s)h(s)ds$, which, by integration of Equation 32, is equivalent to

$$H^*(t) = \int_0^t \frac{I(s)}{Y(s)} dN(s) - \int_0^t \frac{I(s)}{Y(s)} dM(s).$$

The integral $\int_0^t \frac{I(t)}{Y(t)} dN(t)$ is then the N-A estimator in Equation 26, and this is seen by the fact that integration of a counting process is indeed given by a step function. The integral takes one step for every event-time, and so

$$\begin{aligned}\hat{H}(t) &= \int_0^t \frac{I(s)}{Y(s)} dN(s) \\ &= \sum_{T_i \leq t} \frac{1}{Y(T_i)},\end{aligned}$$

which is exactly what was requested, the N-A estimator (26). Now, subtracting $H^*(t)$ from the N-A estimator gives an expression

$$\hat{H}(t) - H^*(t) = \int_0^t \frac{I(s)}{Y(s)} dM(s),$$

the residual. This integral contains two parts; a predictable part $1/Y(t)$ and a martingale $dM(t)$. This meets the definition of a stochastic integral (10) and it is in it self a zero mean martingale. One can conclude that the residual mean $E[H^*(t) - \hat{H}(t)] = 0$, and hence, that the N-A estimator (26) is a true estimator of $H(t)$, for all $Y(t) > 0$.

Using the *optional variance process of a stochastic integral* (12), the optional variance process of the N-A estimator is given by

$$[\hat{H}(t) - H^*(t)] = \int_0^t \frac{I(s)}{Y(s)^2} dN(s). \tag{33}$$

The expected value of this process equals the variance of the N-A estimator, and one can conclude, with the same argument as for the estimate $\hat{H}(t)$, that

$$\begin{aligned}\hat{\sigma}^2(t) &= \int_0^t \frac{I(s)}{Y(s)^2} dN(s) \\ &= \sum_{T_i \leq t} \frac{1}{Y(T_i)^2},\end{aligned} \tag{34}$$

is an unbiased estimator of the N-A estimator variance.

2.4.4 Asymptotic behavior of the N-A estimator

The large sample properties of the N-A estimator is derived, using the fact that the residual (33) is a zero mean martingale. The residual is transformed to show that it fulfills the two conditions of the *martingale central limit theorem* (MCLT), from Subsection 2.3.9, and then the theorem is applied.

The N-A estimator is an aggregate of n independent counting processes $N_1, N_2, ..., N_n$ for the event times $T_1, T_2, .., T_n$, and its residual $[\hat{H}(t) - H^*(t)]$ is an aggregate of the n underlying counting process martingales $M_1, M_2, .., M_n$. The idea is to show that the residual times the square root of the sample size

$$\sqrt{n}(\hat{H}(t) - H^*(t)) = \int_0^t \sqrt{n}\frac{I(s)}{Y(s)} dM(s),$$

fulfills the two conditions of the MCLT, and hence, is approximately normally distributed with mean 0 and variance $\sigma^2(t)$.

This residual, $r_n(t) = \sqrt{n}(\hat{H}(t) - H^*(t))$, is a stochastic integral with a predictable part, $G(t) = \sqrt{n}\frac{I(s)}{Y(s)}$, and a martingale, $dM(s)$.

The predictable variation process of a stochastic integral (21) is used, and hence, the predictable variation process of the residual, given the sample size n, can be expressed as

$$\begin{aligned}\langle r_n(t) \rangle &= \left\langle \int_0^t G(s) dM(s) ds \right\rangle \\ &= \int_0^t G^2(s)\lambda(s) ds.\end{aligned}$$

Also, one can use the multiplicative form of $\lambda(t) = Y(t)h(t)$ to rewrite this as

$$\begin{aligned}\langle r_n(t) \rangle &= \int_0^t \left(\sqrt{n}\frac{I(s)}{Y(s)}\right)^2 Y(s)h(s) ds \\ &= \int_0^t \frac{I(s)h(s)}{Y(s)/n} ds.\end{aligned}$$

To ensure that the two conditions in Equation 22 and 23 are fulfilled it suffices to show that:

(i) $G^2(t)\lambda(t)$ converges in probability toward some function $v(t)$, and

(ii) $I\{|G(t)| > \varepsilon\}$ converges to 0, as $n \to \infty$.

The first requirement is fulfilled by looking at the denominator $Y(t)/n$, which can be assumed to stabilize for greater values of n. If it converges in probability to some function $y(t)$, $Y(t)/n \xrightarrow{p} y(t)$, one can conclude that $G^2(t)\lambda(t) \xrightarrow{p} v(t)$, where the function $v(t) = \frac{h(t)}{y(t)}$.

Also, the predictable process converges in probability to zero, $G(t) = \frac{1}{\sqrt{n}} \frac{I(t)}{Y(t)/n} \xrightarrow{p} 0$, which is to say that the probability of $|G(t)| > \varepsilon$ converges to zero, and so the second condition is fulfilled.

One can conclude that the residual, $r_n(t) = \sqrt{n}(\hat{H}(t) - H^*(t))$, converges in distribution to a Gaussian martingale with mean zero and variance function given by

$$\sigma^2(t) = \int_0^t \frac{h(t)}{y(t)}.$$

2.5 Regression in event history analysis

In this section, the theory of two regression models are covered; The *Cox proportional hazards model* and the *proportional odds model.* The first of these two models is one of the most popular regression models within event history analysis, and its theory has inspired much subsequent research. The second model is also important for this thesis and an adjusted version of it is presented for event-probability regression, in Subsection 2.6.4.

2.5.1 Cox proportional hazards model

The proportional hazards model, proposed by Cox [1972], is a semi-parametric regression model, using partial likelihood estimation for the coefficients.

For a q dimensional vector of covariates $\boldsymbol{x}$, the hazard rate $h(t|\boldsymbol{x})$ is assumed to be proportional to the *base line hazard rate*, $h_0(t)$, by some scale parameter, θ. This relation is formally shown by the equation

$$\frac{h(t|\boldsymbol{x})}{h_0(t)} = \theta \tag{35}$$

where $\boldsymbol{\beta} \in \mathbb{R}^q$ is a q dimensional vector of regression coefficients. The base line hazard rate $h_0(t)$ is a theoretical case, for all covariates $\boldsymbol{x}_0 = \mathbf{0}$, and it is proportional to the hazard rate by some predefined *hazard rate ratio function* $\theta = \psi(\boldsymbol{x}'\boldsymbol{\beta})$

The hazard ratio only takes on positive values, and a natural choice of ratio function is the exponential function

$$\psi(\boldsymbol{x}'\boldsymbol{\beta}) = \exp\{\boldsymbol{x}'\boldsymbol{\beta}\},$$

for which $\psi(\boldsymbol{x}_0'\boldsymbol{\beta}) = 1$ [Collett, 2003, p. 57-58].

The idea of the model is that the underlying distribution of the hazard function $h_0(t)$ is not of interest, but rater, the impact the covariates $\boldsymbol{x}$ has on the hazard, e.g. then regression coefficients $\boldsymbol{\beta}$. Coefficients are estimated by maximizing the partial likelihood function

$$\begin{aligned} L(\theta) &= \prod_{j=1}^{n} \frac{h_0(t)\exp\{\boldsymbol{x}_j'\boldsymbol{\beta}\}}{\sum_{l\in R(t_{(j)})} h_0(t)\exp\{\boldsymbol{x}_l'\boldsymbol{\beta}\}} \\ &= \prod_{j=1}^{n} \frac{\exp\{\boldsymbol{x}_j'\boldsymbol{\beta}\}}{\sum_{l\in R(t_{(j)})} \exp\{\boldsymbol{x}_l'\boldsymbol{\beta}\}}, \end{aligned}$$

where $R(t_{(j)})$ is the set of individuals at risk at time $t_{(j)}$, for the n ordered event-times $t_{(1)} < t_{(2)} < .. < t_{(n)}$ [Collett, 2003, p.66].

Described in words, the likelihood is the product of individual hazard rates, divided by the sum of the hazard rates for those that are at risk at time $t_{(j)}$, for all individuals $j = 1, 2, .., n$.

One problem with the Cox model is that the assumption of proportional hazards is not always a suitable approximation of real data, when hazard rates of different groups may converge, or diverge more than proportionally, over time. One way of relaxing the assumption of proportionality is to include a time varying covariate or using proportional odds models [Kirmani and Gupta, 2001].

For more information on Cox proportional hazards model see Liu [2012, ch.5] and Collett [2003, ch.3].

2.5.2 Proportional odds model

Benet [1982] proposed *the proportional odds model* as an alternative to the Cox proportional hazards model, for some cases where the assumption of proportionality of hazards ratios does not hold. The non-proportionality is a common case in medical studies where the effect of a disease on hazard rates often wear off, and hazard rates converges with time. To instead assume proportionality of survival odds is a relaxation of the assumption of proportional hazards [Kirmani and Gupta, 2001].

In the proportional odds model, the odds ratio of event-free time, or survival, are assumed to be proportional in a scale parameter θ as follows

$$\frac{S(t)}{1 - S(t)} = \frac{S_0(t)}{1 - S_0(t)}\theta,$$

where $\theta = \psi(\boldsymbol{x}'\boldsymbol{\beta})$ is a function of the q dimensional vector of covariates $\boldsymbol{x}$.

Compared to the proportional hazards model, the coefficients in the proportional odds model can *not* be estimated by partial likelihood, where one factor out the baseline function. The *baseline odds function* $\frac{S_0(t)}{1-S_0(t)}$ is estimated together with the coefficients by maximization of the likelihood function

$$L(\theta) = \prod_{i=1}^{n} f(t_i|\theta)^{d_i} S(t_i|\theta)^{1-d_i}, \tag{36}$$

where $d_1, d_2, .., d_n$ is the indicator variable for event times [Benet, 1982]. Uncensored event times are indicated by $d_i = 1$ and contribute to the likelihood through their probability function $f(t_i|\theta)$, while censored event times $d_i = 0$ contributes to the likelihood through the survival time $S(t_i|\theta)$.

2.5.3 Restricted cubic splines

When fitting a regression model to variables dependent on time, one often find it difficult to fit just one model to the whole time-line.

One way of handling structural variations over time is to divide the time line into several segments and fit different models within these segments. A *spline* is a function that does just that, splits the data into segments on predetermined points called *knots*.

A *cubic spline* is a spline function with existing and continuous first and second derivatives, where polynomials are fitted between knots, and the function gives smooth transition over the knots. Polynomials are advanced enough to approximate most of the structural variations of reality without over fitting, but two problems arise from using cubic splines:

(i) The model can give unreliable and volatile results beyond the end knots, due to data points being scars, and

(ii) the splines are costly in degree of freedom.

One solution to ease these problems is the use of *restricted cubic splines* [Harrell Jr., 2001, p.19]. The restriction is to fit linear functions beyond end knots, releasing 2 degree of freedom and reducing problems with volatility.

The *restricted cubic spline function*, for a covariate x with K knots ε_i, for $i = 1, 2, .., K$, contains $K - 1$ *basis functions* $B_i(x)$, for $i = 1, 2, .., K - 1$. The spline function

$$s(x) = \alpha_0 + \sum_{i=1}^{K-1} \alpha_i B_i(x) \tag{37}$$

generates different covariates, in the form of the basis functions B_i, for different levels of x, with regression coefficients α_i. The basis functions have different structures between knots ε_i and are computed as

$$B_i(x) = (x - \varepsilon_i)^3_+ - \rho_i(x - \varepsilon_1)^3_+ - (1 - \rho_i)(x - \varepsilon_K)^3_+,$$

where $(\cdot)^3_+ = (\cdot)^3$ for positive values of $(\cdot)^3$ and 0 otherwise, and

$$\rho_i = \frac{\varepsilon_K - \varepsilon_i}{\varepsilon_K - \varepsilon_1}.$$

In event history regressions, the covariate x is usually time, $x = t$, or log time, $x = log(t)$. For more reading on restricted cubic splines in survival analysis see Rutherford et al. [2015].

2.6 Risk

The foundation of basic theory of event history analysis, with some applications, is given in previous subsections. Now, this thesis has arrived at the introduction of two new measurements in event history analysis; the incidence risk, and the event-probability function, that was presented theoreticaly for the first time by Bottai [2017].

This subsection begins with showing two ways of computing averages, the arithmetic mean and the geometric mean. This is done by looking at the theory of the incidence rate and the incidence risk, and by highlighting the differences between them, in Subsections 2.6.1, 2.6.2, and 2.6.3.

The aim is to show concept of the *incidence risk*; a geometric average of the conditional probability of the occurrence of an event per unit time. The outline of theory of the incidence risk follows the logical trail of the article by Bottai [2022].

The following subsections extends this theory of the incidence risk to the *event-probability function.* The event-probability function is the instantaneous and conditional probability of the occurrence of an event at time t given that the event has not occurred yet.

Theory of the event-probability function, with some applications, follows the article by Bottai et al. [2021], and are presented in Subsections 2.6.4, and 2.6.5.

2.6.1 Incidence rate

In event history analysis *the incidence rate* or *the failure rate* is an often reported measure, and it is the rate of occurrences of an event per unit time and person. It is a *mean* rate per unit time and it is calculated by

$$\frac{N(t)}{\sum_{T_i \leq t} T_i}, \tag{38}$$

the number of incidences at time t divided by the sum of all event-times up to some time, t, see StataCorp [2021, p.386].

2.6.2 Geometric mean

The incidence rate in 38 is calculated as an *arithmetic mean* of occurrences of an event per unit time and individual, but there is another method for calculating averages; the *geometric mean.* The use of this geometric mean is a logical method for calculating averages of probability, which is here demonstrated by some theoretical and algebraic workout.

The probability of incidence of an event in any time interval $(t, t+x]$, given that it has not yet happened at time t, can be calculated using the ratio of the survival functions

$$\begin{aligned} P(T \leq t+x \mid T > t) &= \frac{P(t < T \leq t+x)}{P(T > t)} \\ &= 1 - \frac{P(T > t+x)}{P(T > t)} \\ &= 1 - \frac{S(t+x)}{S(t)}. \end{aligned}$$

Expressing this probability as a mean value per unit time can then be done by the use of the geometric mean.

To demonstrate this one can assume a special case, where x is integer valued, and divide this time interval, $(t, t+x]$, in to several unit time intervals. In this special case, the survival function ratio can be expressed as

$$\frac{S(t+x)}{S(t)} = \frac{S(t+x)}{S(t+(x-1))} \frac{S(t+(x-1))}{S(t+(x-2))} \cdots \frac{S(t+2)}{S(t+1)} \frac{S(t+1)}{S(t)},$$

which is the product of the survival ratios per disjoint unit time interval. The geometric mean of this probability ratio $\frac{S(t+x)}{S(t)}$, per unit time, is then given by

$$\left[\frac{S(t+x)}{S(t)}\right]^{1/x}. \tag{39}$$

Hence, the average probability of incidence per unit time, given that the event has not occurred yet, is

$$1 - \left[\frac{S(t+x)}{S(t)}\right]^{1/x}.$$

2.6.3 Incidence risk

When the event of interest can only happen once, for example when the event is a death, the incidence *rate* can be difficult to interpret. This measurement can take on any positive number and is not a probability measure of incidence but the average rate of incidence.

The geometric mean, of Equation 39, on the other hand, is a way to calculate an average probability of occurrence of an event; it is a measurement of risk.

Bottai [2022] defines the *incidence risk* in a time interval $(t, t+dt]$ as the geometric rate

$$G(t, t+dt) = 1 - [S(t+dt)/S(t)]^{1/dt},$$

which is the average probability of the occurrence of an event per unit time, in the interval, given that it has not yet occurred. Using the relationship between the survival function and the cumulative hazard, in Equation 3, one can rewrite this geometric rate as

$$G(t, t + dt) = 1 - \exp\left\{\frac{H(t + dt) - H(t)}{dt}\right\}. \tag{40}$$

Also, in the case of a time interval starting from zero, $(0, t)$, it can be simplified to

$$G(0, t) = 1 - \exp\left\{ - \frac{H(t)}{dt}\right\}, \tag{41}$$

for $H(0) = 0$.

The incidence risk is a measurement which, in the case of a one-time-only event analysis, gives a clear interpretation: It could for example be, the risk of a first heart failure per year in a population, or, the mortality risk per month after getting a cancer diagnosis.

2.6.4 Event probability

Where the theory of the incidence risk ends, the theory of the *event-probability function* takes off. As the name suggests, this is a probability measure and it is defined as the limit of the incidence risk $G(t, t+dt)$, in Equation 40, for $dt \to 0$.

In this and the next subsection, the outline of theory follows that of the article by Bottai et al. [2021].

The event-probability function $g(t) = \lim_{dt\to 0} G(t, t+dt)$ is the instantaneous risk of the occurrence of an event at time t, given that it has not occurred yet. The definition is given by

$$g(t) = \lim_{dt\to 0}\left[1 - \left(\frac{S(t + dt)}{S(t)}\right)^{\frac{1}{dt}}\right], \tag{42}$$

the limit of the geometric mean of the risk of incidence.

In order to express the event-probability function as a function of the hazard rate $h(t)$, one can rewrite Equation 42, using the relationship $log(S(t)) = -H(t)$. By the definition of a derivative of a function, which is the limit of an increment of the function divided by the small change in the function variable, the event-probability function can be written as

$$\begin{aligned} g(t) &= \lim_{dt\to 0}\left[1 - \exp\left\{\frac{\log(S(t + dt) - \log S(t)}{dt}\right\}\right] \\ &= 1 - \exp\{-h(t)\}, \end{aligned}$$

a function of the hazard rate.

Both the event-probability function and the hazard function, are defined as limits of a mean, where

$$g(t) = 1 - \lim_{dt \to 0} P(T > t + dt | T > t)^{1/dt},$$

and

$$h(t) = \lim_{dt \to 0} P(T \leq t + dt | T > t)/dt.$$

In words, the event-probability $g(t)$ is defined as the limit of a geometric mean, while the hazard rate $h(t)$ is the limit of an arithmetic mean. The event-probability is strictly smaller than the hazard rate $g(t) < h(t)$, for all positive values of t. They are related to each other in the same way as the cumulative distribution function and the cumulative hazard function are related, where

$$g(t) = 1 - \exp\{-h(t)\}, \text{ and}$$

$$F(t) = 1 - \exp\{-H(t)\}.$$

These relationships, put side by side, are illustrative of the role of the event-probability function, and show that the existence of the event-probability is some what natural in event history analysis.

Also, defining $\bar{g}(t) = 1 - g(t)$ as the *no-event probability function*, one can extend theory to include the following relations;

$$h(t) = -\log[\bar{g}(t)], \tag{43}$$

and

$$S(t) = \exp\left\{ \int_o^t \log[\bar{g}(u)] du \right\}. \tag{44}$$

Compared to the probability function $f(t)$, the event-probability function is a conditional probability for the case when the event has not occurred yet, whereas $f(t) = F'(t)$ is the *unconditional* probability of the occurrence of an event at an instance in time, t. The relationship between the probability function $f(t)$ and the event-probability function $g(t)$ is therefore shown by taking the derivative of $F(t) = 1 - S(t)$, where

$$\begin{aligned} f(t) &= \frac{d}{dt}(1 - S(t)) \\ &= \frac{d}{dt} \exp\left\{ \int_o^t \log[\bar{g}(u)] du \right\} \\ &= -\log[\bar{g}(t)] \exp\left\{ \int_o^t \log[\bar{g}(u)] du \right\}. \end{aligned}$$

More properties of the event-probability function is found in the article by Bottai et al. [2021].

2.6.5 Regression methods for event-probability

The event-probability function, which is a true measure of probability, can, in the same way as the survival function, be modeled using a *proportional odds regression model* (45). The event-probability proportional odds model is defines as

$$\frac{g(t|\boldsymbol{\beta}, \boldsymbol{x})}{1 - g(t|\boldsymbol{\beta}, \boldsymbol{x})} = \frac{g_0(t)}{1 - g_0(t)}\theta, \tag{45}$$

with *rate ratio function* $\theta = \psi(\boldsymbol{x}'\boldsymbol{\beta})$ for a q dimensional vector of covariates $\boldsymbol{x}$, and regression coefficients $\boldsymbol{\beta} \in \mathbb{R}^q$.

As in the case of the proportional odds model in Subsection 2.5.2, the *baseline odds ratio* $\frac{g_0(t)}{1-g_0(t)}$ represents the event probability in a theoretical case, where all covariates $\boldsymbol{x}$ equals zero. Also, the rate ratio function is defined for positive values of θ and a logical choice of function is to use the exponential function, $\psi(\boldsymbol{x}'\boldsymbol{\beta}) = \exp\{\boldsymbol{x}'\boldsymbol{\beta}\}$.

Expressing the logarithm of odds as $logit(p) = \log\{\frac{p}{1-p}\}$, and taking the logarithm of Equation 45, with exponential ration function $\theta = \exp\{\boldsymbol{x}'\boldsymbol{\beta}\}$, the proportional odds model is given by

$$logit[g(t|\boldsymbol{\beta}, \boldsymbol{x})] = logit[g_0(t)] + \boldsymbol{x}'\boldsymbol{\beta},$$

where the odds $\frac{g(t|\boldsymbol{\beta},\boldsymbol{x}_i)}{1-g(t|\boldsymbol{\beta},\boldsymbol{x}_i)}$ is defined on the positive real line and $logit[g(t|\boldsymbol{\beta}, \boldsymbol{x}_i)]$ on the entire real line. The baseline odds function $logit[g_0(t)]$ is estimated together with the regression coefficients $\boldsymbol{\beta}$ by maximization of the likelihood function in Equation 36.

For more information on event-probability regression see Bottai et al. [2021].

3 Software implementation

The methods for computing *incidence risks*, and preforming *event-probability regressions*, are still at an early stage of introduction to the world of research. For most users of statistical methods, the availability of existing software are crucial for the usability of such new methods.

In this section, existing computer software for estimating incidence risks and preforming event-probability regressions is covered, and two new implementations is introduced, available for users anywhere.

To make it possible to understand this section, even for those not familiar with JavaScript or programming at large, most of the code is excluded from this section. For those that are more curious of the implementations in the

JavaScript language, some of the functions, from the back end programs introduced in this section, are displayed in Appendix A.

3.1 Available software

The method for computing *incidence risks*, described in detail in Subsection 2.6.3, and the *event-probability regression*, described in Subsection 2.6.4, are at the time of writing only available for users of Stata. The Stata commands `stprisk` and `stpreg` can be downloaded to Stata, using the command:

```
. net from http://www.imm.ki.se/biostatistics/stata
```

These commands will be demonstrated here through some short examples.

Again, the data set `kidney` from Subsection 1.2 is used. This data set contains of right censored survival times for patients with diagnosis of metastatic renal carcinoma, see Medical Research Council Renal Cancer Collaborators [1999]. Two treatment groups are compared: *subcutaneous interferon-α* (IFN) and *oral medroxyprogesterone acetate* (MPA).

After importing the data set, with the command `use`, one need to specify the time-to-event variable `survtime`, by running the command:

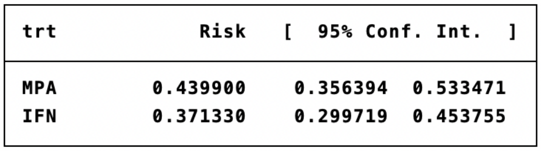
```
    . stset survtime, failure(cens=1) scale(365.5)
```

The sub-command `failure()` sets the event-indicator variable, in this case `cens`, and what value indicates a failure, opposed to being a censored event. The sub-command `scale()` is used to scale the event-times, in this case from days to years.

The first command, `stprisk`, is used on time-to-event data to estimate the incidence risk. The mortality risks is estimated for the two treatment groups in `trt`, by running:

```
    . stprisk trt
```

This generates the following output, recognized from the earlier example:



```
trt            Risk   [  95% Conf. Int.  ]

MPA        0.439900    0.356394  0.533471
IFN        0.371330    0.299719  0.453755
```

The estimated mortality risk in the MPA group is 44% and 37% in the IFN group.

The second command, `stpreg`, is used on time-to-event data to preform an *event-probability regression*. The command have multiple options, for choice of regression model, but the default performs a regression using the proportional odds explained in Subsection 2.6.5.

The other, non-default, regression models are risk ratios model `rr`, risk difference model `rd`, and power probability model `power`. More information about the different, optional, regression models are found in Bottai et al. [2021]. Coefficients are estimated by maximization of the likelihood and more options can be found, running the help command:

```
. help stpreg
```

The command `stpreg` is demonstrated by using the same `kidney` data. The following two regression models are estimated

$$logit[g(t|\theta)] = \beta_0 + \beta_1 \log(t) + \beta_2 trt, \tag{46}$$

and

$$logit[g(t|\theta)] = \beta_0 + \beta_1 \log(t) + \beta_2 s(\log(t)) + \beta_3 trt, \tag{47}$$

where trt is the treatment-group-variable, and $s(.)$ a RCS function defined in Equation 37, with 3 knots. The RCS function is specified using the sub-command `df(.)`, which specifies how many degree of freedom to spend.

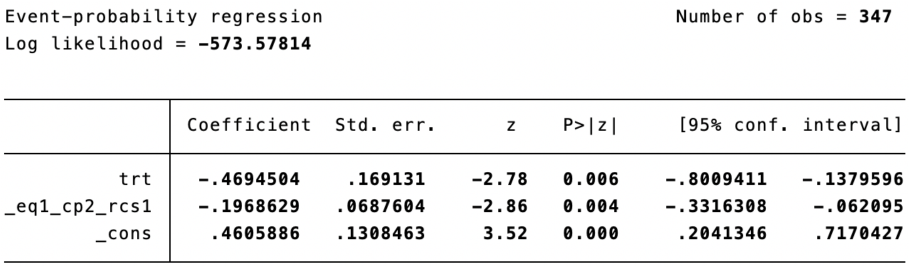
The first regression, in Equation 46, is estimated by running the following command:

```
. stpreg trt, coef df(1)
```

The second regression, in Equation 47, is estimated by:

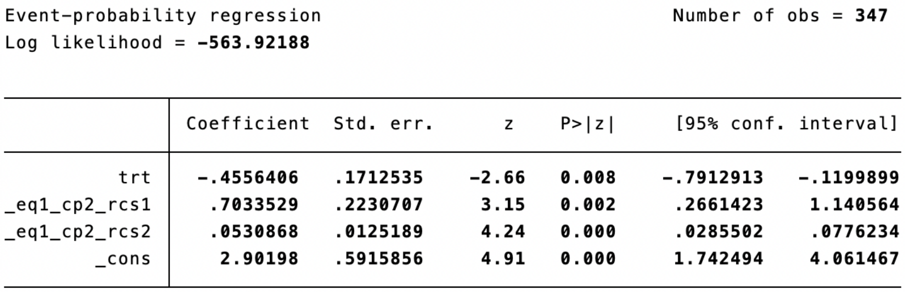
```
. stpreg trt, coef df(2) noorthog
```

The sub-command `coef` give an output showing estimated coefficients, and the sub-command `df(2)` is used for adding a RCS with three knots, while `df(1)` only give the $log(t)$ regressor. The sub-command `noorthog` is for specifying no orthogonalization of the splines. The results from the first model (46) is then given by:



```
Event-probability regression                              Number of obs = 347
Log likelihood = -573.57814

---------------------------------------------------------------------------------
             |  Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+-------------------------------------------------------------------
         trt |   -.4694504    .169131    -2.78   0.006    -.8009411   -.1379596
_eq1_cp2_rcs1|   -.1968629   .0687604    -2.86   0.004    -.3316308    -.062095
       _cons |    .4605886   .1308463     3.52   0.000     .2041346    .7170427
---------------------------------------------------------------------------------
```

The variable `_eq1_cp2_rcs1` show the coefficient of $\log(t)$ and `_cons` the intercept. All three coefficients are statistically significant on the 1%-level.
The results from the second model (47) is given by:



```
Event-probability regression                              Number of obs = 347
Log likelihood = -563.92188

---------------------------------------------------------------------------------
             |  Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+-------------------------------------------------------------------
         trt |   -.4556406   .1712535    -2.66   0.008    -.7912913   -.1199899
_eq1_cp2_rcs1|    .7033529   .2230707     3.15   0.002     .2661423    1.140564
_eq1_cp2_rcs2|    .0530868   .0125189     4.24   0.000     .0285502    .0776234
       _cons |     2.90198   .5915856     4.91   0.000     1.742494    4.061467
---------------------------------------------------------------------------------
```

Another regressor is added, `_eq1_cp2_rcs2`, which gives the coefficient of the second spline function $s(\cdot)$. All four coefficients are statistically significant at the 1%-level.

3.2 Current limitations

The users in Stata are just a part of all those active in the world of research. Many bio-statisticians and researchers within the field of time-to-event analysis are users of the programming languages R and Python, or other statistical analysis software like SAS and SPSS.

For anyone with the right skills in programming and knowledge in statistical methodology, implementing new methods is possible. Python and R libraries give users good tools, for optimization and analysis, making the implementation easier and less time consuming.

But not everyone is comfortable with developing their own software. Researchers may not always have sufficient knowledge in statistical methods and even though statisticians may be able to understand the methods, building software implementations may demand to much programming skills fore some.

An important task for those with knowledge in statistical methodology, data science and software development is to make both new methods and implementations of them understandable and easy to use. In other words, making it available to those who may lack time or the demanded skills to do it themselves.

As one of those who wants to make it more available, the thought of implementing an online application was appealing. The question is what options do one have when working with HTML and back-end programming languages for data science.

One option is to use Python, which is a high-level programming language with many existing libraries for data analysis. There are some development frameworks available for putting applications built in Python on the world-wide web.

Two of these frameworks are Flask and Django which enable HTML protocols interacting with Python applications [Breuss, MDN, 2023]. One disadvantage, if you are a statistician and not a software engineer, is that one need to learn and set up these frameworks to make the application work on the web and on a server that supports running Python code. Using Python together with Flask or Django is not as easy as just letting a JavaScript script interact with your HTML protocol, that is if one only look at the technical setup needed for running a web application.

One solution for making Python more accessible for development of web applications appeared when Anaconda launched the new PyScript, in 2022. PyScript is the new programming language for web scripting with Python. It was still very new and experimental in autumn 2022 [Yegulalp, 2022]. With only a released beta version, and only for browsers supporting WebAssembly, it could become complicated to use PyScript to produce more advanced statistical web applications, and reach out to users. Something to still look forward to in a near future.

Another option is to use plain JavaScript. Building software in JavaScript requires the minimal setup in order to make it work in any browser, but it raises questions about how well suited JavaScript is for programming in data science. The idea awoken a curiosity of exploring this possibility. How difficult would it be to build something fairly complex using the most available tools for building applications: HTML, CSS, JavaScript.

3.3 JavaScript as programming language for data science

Classic programming languages for data science, like R and Python, have libraries for numerical methods and statistical modeling. JavaScript is not among these languages, even though more and more tools are available for applications within the data science field. And so, there are data scientists and software engineers discussing the pros and cons of JavaScript for data analysis.

Lin and Gebaly [2016] consider the performance and speed in analyzing big data. This was the year 2016, and the authors of the article discussed the development of the JavaScript language. There were many new available supporting software programs, making JavaScript run faster and smoother even when handling a lot of information. One type of such supporting software they pointed at was more effective compilers available for JavaScript programmers in 2016. They concluded that there was not much standing in the way of advanced data analysis in JavaScript in 2016.

The article by Bostock [2021] is another who is addressing the subject of using JavaScript in data science, declaring the advantage of availability; one can implement data science tools and make them available for anyone, anywhere. Also, because JavaScript is the most used programming language for the web; its development is fast, trying to meet requirements of so many different types of applications. Last but not least, the article points out that web applications in JavaScript serves users on many different levels. JavaScript programs on the web are available for everyone, not only when it comes to using the application, but also when it comes to the availability of the underlying source code of the application. In that sense it also serves as a learning tool for other developers to use.

The article by Schmidt [2021] establishes the speed benefits of JavaScript, which in many cases outruns Python and R. One of the reasons for JavaScript high performance in case of speed has the web to thank for. Speed is essential for developing advanced software on the web and web developers are generally demanding when it comes to computing speed. With more packages appearing, as for example Tidyverse, with a range of statistical libraries for machine learning, data science with JavaScript becomes easier.

And last, a personal note. As a statistician, with focus on finding good tools for implementing statistical methods, working in JavaScript one can notice that even though there is a variety of useful libraries for data science, these are many times developed for software engineers and computer scientists interested in machine learning. The result is a difficulty to find written information about these libraries, addressing those interested in extensive use of methods for statistical inference programming. There is little information on how functionalities of these libraries work, and how to use functions outside the machine learning environment.

In the next two Subsections, 3.4, and 3.5, two web applications will be presented. These applications run JavaScript implementations of the new methods presented in Subsection 2.6. As reference literature for the programming in JavaScript, Html and CSS, I have used the book by [Collins, 2017]. Also I worked with the development tools Visual Studio Code and Github.

3.4 A JavaScript implementation of incidence risk estimation

The first web application to be presented is a JavaScript implementation of the incidence risk (2.6.3). This application is designed for easy use on `.txt` or `.csv` files containing right censored event history data. The idea is to present a software for users without extensive knowledge in statistical methods or statistical software programming. The application is in the format of a web form. User fill in the requested information about the uploaded data, and the back-end JavaScript program estimates incidence risks, using the Nelson-Aalen estimation of cumulative hazards.

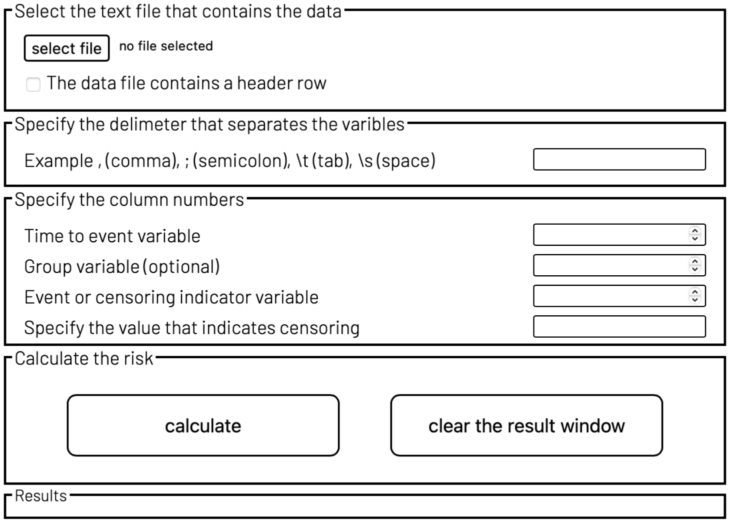
3.4.1 The web form

The design of the form is basic, with simple instructions on how to upload a data file and what information to fill in about the uploaded data, see Figure 1.

Allowed file formats are `.txt` or `.csv` files and the user can decide to specify an optional group variable. This group variable can be used for estimation of incidence risks within different groups, making comparison between them possible. The user submit the information given in the form by clicking the *calculate* button at the end of the form, and delete results displayed in the result window by clicking the *clear the result window* button.

3.4.2 Estimating cumulative hazards and the incidence risk

From the definition in Equation 41, the incidence risk is a function of the cumulative hazard. Hence, an appropriate estimate of this measurement is to use the Nelson-Aalen (N-A) estimator of the cumulative hazard in Equation 26, see [Bottai, 2022].



Select the text file that contains the data

no file selected

□ The data file contains a header row

Specify the delimeter that separates the varibles

Example , (comma), ; (semicolon), \t (tab), \s (space)

Specify the column numbers

Time to event variable

Group variable (optional)

Event or censoring indicator variable

Specify the value that indicates censoring

Calculate the risk

Results

Figure 1: Web form for the incidence risk program.

For a sample of n censored event-times, $\{t_i\}_{i=1}^n$, and event-indicator variables, $\{d_i\}_{i=1}^n$, this estimator of the incidence risk $\hat{G}(t)$ is given by

$$\hat{G}(t) = \exp\left\{ -\frac{\hat{H}(t)}{T^{obs}(t)} \right\},$$

where $\hat{H}(t)$ is the N-A estimator of the cumulative hazard, and $T^{obs}(t) = \{T_i : \max\ T_i < t, D_i = 1\}$ is the last observed event-time, both at time t. Now, assuming the presence of tied event-times, this estimator can be rewritten as

$$\hat{G}(t) = \exp\left\{ -\frac{\sum_{t_i<t} \Delta\hat{H}(t_i)}{T_n^{obs}(t)} \right\},$$

where $\Delta\hat{H}(t_i)$ is the increment given by Equation 28, and k_i is the number of duplicates of the event-time t_i. Also, according to conventions, event-times are counted before censored times [Aalen et al., 2008, page 84].

Continuing using theory of the N-A estimator, the 95% confidence intervals of the estimated incidence risk $I(\hat{G}(t))$ are derived using Equation 27. These intervals are then given by

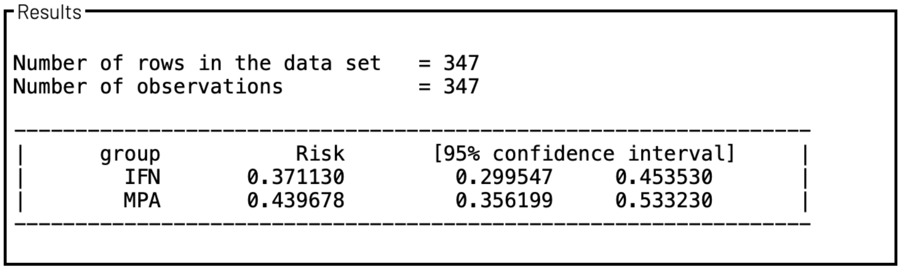
$$I(\hat{G}(t)) = \hat{H}(t) \exp\left\{ \pm 1.96 \frac{\hat{\sigma}_H}{\hat{H}(t)} \right\},$$

where $\hat{\sigma}_H$ is the estimated N-A variance of the estimated cumulative hazard, in Equation 34.

3.4.3 Result window and output

The back-end JavaScript program calculates estimations of the incidence risk for the uploaded time-to-event data, given a group variable or not, and the corresponding confidence intervals. Results are presented in the result window together with the number of rows in the data set and number of observations used for the estimates.

To demonstrate the output given by the program, the `kidney` data from Subsection 1.2 is revisited. The `.csv` file is uploaded, information is filled into the form, and the variable `trt` is specified as a group variable. Our program estimates the incidence risks for the two treatment groups, IFN and MPA, and display the results in the results-window, which expands with output:



```
Number of rows in the data set   = 347
Number of observations           = 347

-----------------------------------------------------------------
|      group           Risk       [95% confidence interval]     |
|        IFN      0.371130         0.299547     0.453530       |
|        MPA      0.439678         0.356199     0.533230       |
-----------------------------------------------------------------
```

Comparing these results with those given by Stata, when running the `strisk` command on the same data, Section 3.1, one can conclude that these are the same.

With results still displayed in the results-window users can upload new data, change some information in the form, and submit again by clicking the *calculate* button. New results from another submit will then be displayed bellow the first output. Clicking the *clear the result window* button deletes all output and empties the result window.

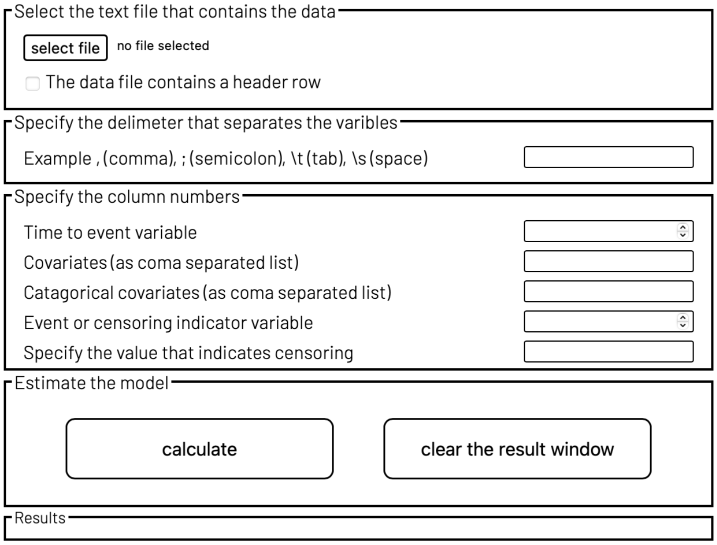
3.5 A JavaScript implementation of event-probability regression

The second web application runs a JavaScript implementation of the event probability regression from Subsection 2.6.5. Just as the one for estimation of incidence risks, it is designed to be an easily understandable application, in the format of a web form. Users upload a `.txt` or `.csv` file containing censored event history data and fill information about the variables in the web form. The back-end JavaScript program read the time-to-event data and preform a proportional odds regression, using the specified covariates from the uploaded data. Regression coefficients are estimated by maximizing the log-likelihood through a simple optimization algorithm. The coefficients are presented in the result window together with standard errors, p-values, and 95% confidence intervals.

3.5.1 The web form

Figure 2 shows the web form for the event-probability regression application. The form is divided into sections with explaining headers that guide the user through the process of uploading data and specifying the regression variables. Users upload a `.csv` or `.txt` file and fill in the information in the form. Covariates are specified with two comma-separated lists, where numerical and categorical covariates are specified in the two lists.

After filling out the form users submit the information by clicking on the *calculate* button and then delete output with the *clear the result window* button. After submitting, the back-end JavaScript program estimates the regression coefficients, calculates standard errors, and writes the results in the results window.



Select the text file that contains the data

no file selected

□ The data file contains a header row

Specify the delimeter that separates the varibles

Example , (comma), ; (semicolon), \t (tab), \s (space)

Specify the column numbers

Time to event variable

Covariates (as coma separated list)

Catagorical covariates (as coma separated list)

Event or censoring indicator variable

Specify the value that indicates censoring

Estimate the model

Results

Figure 2: The web form for the event-probability regression.

3.5.2 Model set up and log-likelihood

This Subsection begins with the derivation of the log-likelihood of the proportional odds model, for which regression coefficients are estimated through maximization. The definitions of the proportional odds model and the likelihood function are given in the article by Bottai et al. [2021].

For a set of event history data, containing n observations of right censored event-times, $\{t_i\}_{i=1}^n$, event-indicators, $\{d_i\}_{i=1}^n$, and a q-dimensional covariate vectors $\{\boldsymbol{x}_i\}_{i=1}^n$, the log-likelihood is specified and optimized with respect to the regression coefficient vector $\boldsymbol{\beta}$, defined on $\mathbb{R}^{q+1}$. The covariate vectors can contain both numerical and categorical variables where the later are included by means of $r-1$ indicator variables, for a variable with r categories.
With some algebra the likelihood function in Equation 36 can be written as

$$\begin{aligned} L(\theta) &= \prod_{i=1}^n f(t_i|\theta)^{d_i} S(t_i|\theta)^{1-d_i} \\ &= \prod_{i=1}^n \left[\frac{f(t_i|\theta)}{S(t_i|\theta)}\right]^{d_i} S(t_i|\theta) \\ &= \prod_{i=1}^n h(t_i|\theta)^{d_i} S(t_i|\theta), \end{aligned}$$

for a scale parameter θ.

This likelihood can then be expressed as a function of the no-event-probability function $\bar{g}(t|\theta)$, using the formulations of the hazard and the survival function in Equation 43, and 44, where

$$L(\theta) = \prod_{i=1}^n -\log[\bar{g}(t_i|\theta)]^{d_i} \exp\left\{\int_o^{t_i} \log[\bar{g}(u|\theta)]du\right\}. \tag{48}$$

The rate ratio function $\theta = \exp\{\boldsymbol{x}'\boldsymbol{\beta}\}$, is a function of the covariate vector $\boldsymbol{x}' = (1, x_1, .., x_q)'$, for the q covariates, and a $(q+1)$ dimensional coefficient vector $\boldsymbol{\beta} = (\beta_0, \beta_1, .., \beta_q)$. Then, given the proportional odds model,

$$\frac{1-\bar{g}(t|\theta)}{\bar{g}(t|\theta)} = \exp\left\{\boldsymbol{x}'\boldsymbol{\beta}\right\},$$

one can formulate the no-event-probability as an expression of the rate ratio function

$$\bar{g}(t|\theta) = \frac{1}{\exp\left\{\boldsymbol{x}'\boldsymbol{\beta}\right\} + 1}. \tag{49}$$

Note that the baseline log-odds function $logit[g_0(t)]$ is integrated into the coefficient vector $\boldsymbol{\beta}$.

Using Equation 48 together with the new expression of the no-event-probability (49), one can then derive the log-likelihood function $l_n(\theta)$ for the regression model as

$$l_n(\theta) = \log\left[\prod_{i=0}^{n} -\log[\bar{g}(t_i|\theta)]^{d_i} \exp\left\{\int_{o}^{t_i} \log[\bar{g}(u|\theta)]du\right\}\right]$$

$$= \sum_{i=1}^{n}\left(d_i \log\left[\log\left[\exp\left\{\boldsymbol{x}_i'\boldsymbol{\beta}\right\}+1\right]\right] + \int_{0}^{t_i} -\log\left[\exp\left\{\boldsymbol{x}_i'\boldsymbol{\beta}\right\}+1\right]du\right). \quad (50)$$

Hence, the maximum likelihood estimator of the coefficient vector $\boldsymbol{\beta}$ is found by maximizing $l_n(\theta)$ for the coefficients over the parameter space $\mathbb{R}^{q+1}$

$$\hat{\boldsymbol{\beta}}_{ML} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{q+1}}{\operatorname{argmax}}\; l_n(\theta).$$

The maximization is done using the optimization algorithm explained in the next subsection.

The integral of the log-likelihood (50) does not have a closed-form solution and must be estimated by numerical approximation. The integral is approximated by the Simpson's rule, where

$$\int_a^b f(u)du \approx \frac{b-a}{2}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right],$$

for some continuous function $f(x)$ [Süli and Mayers, 2003, p. 203].

3.5.3 Compass optimization

This algorithm is a simpler version of the gradient search suggested by Bottai et al. [2015]. The algorithm is slow but ensures convergence.

The algorithm
For a vector of coefficients $\boldsymbol{\beta}$, defined on $\mathbb{R}^{q+1}$, this optimization algorithm takes a small step δ in any direction k. It computes the log-likelihood for the new coefficient vector, lets call it $\boldsymbol{\beta}_{k\delta}$, and then it makes one of the following choices:

- If $l_n(\boldsymbol{\beta}) < l_n(\boldsymbol{\beta}_{k\delta})$ the length of the step is increased by some positive factor $\gamma > 1$, and the algorithm restart from the top, with the new coefficient vector $\boldsymbol{\beta}_{k\delta}$, and the longer step $\gamma\delta$.
- If $l_n(\boldsymbol{\beta}) \geq l_n(\boldsymbol{\beta}_{k\delta})$ the algorithm does not carry through with the step, but change to the next direction.

- If all directions been searched, with the given step-length, δ, and no one was found with a higher log-likelihood, the length of the step is decreased by some positive factor, $\rho < 1$. The algorithm restarts with the old coefficient vector $\boldsymbol{\beta}$ and the new step length $\rho\delta$.
- otherwise it restarts with the new direction and the old coefficient vector $\boldsymbol{\beta}$.

The algorithm finish when $l_n(\boldsymbol{\beta}) < l_n(\boldsymbol{\beta}_{k\delta})$ and the difference $l_n(\boldsymbol{\beta}_{k\delta}) - l_n(\boldsymbol{\beta})$ is smaller than some tolerance factor. It returns $\boldsymbol{\beta}_{k\delta}$ which is the estimated coefficient vector $\hat{\boldsymbol{\beta}}_{ML}$.

3.5.4 Estimating the maximum likelihood variances

The maximum likelihood variances of the estimated coefficients $\hat{\boldsymbol{\beta}}_{ML}$ are found by computing the inverse of the Fisher information matrix, $I(\boldsymbol{\beta})$, at the maximal point, $\hat{\boldsymbol{\beta}}_{ML}$. The estimated variances-covariance matrix is then given by

$$\begin{aligned}\hat{\Sigma}_{ML}(\boldsymbol{\beta}) &= I(\hat{\boldsymbol{\beta}}_{ML})^{-1} \\ &= -\mathcal{H}(\hat{\boldsymbol{\beta}}_{ML})^{-1},\end{aligned}$$

where $\mathcal{H}(\hat{\boldsymbol{\beta}}_{ML})$ is the hessian matrix of the log-likelihood at the maxima $l_n(\hat{\beta}_{ML})$ [Held and Bové, 2014, p.28].

The hessian matrix $\mathcal{H}(\boldsymbol{x})$ of a function $f(\boldsymbol{x})$, and variable vector $\boldsymbol{x}$, defined on $\mathbb{R}^n$, is the matrix containing the n^2 second-order partial derivatives

$$\mathcal{H}(\boldsymbol{\beta}) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}. \tag{51}$$

The second-order partial derivatives of the log-likelihood (50), for regression coefficients β_i and β_j, for all $i, j = 0, 1, .., q$, are approximated using numerical methods. For small values of $h, k > 0$, these approximations are computed by

$$\frac{\partial^2 f}{\partial x \partial y} \approx \frac{f(x+h, y+k) - f(x+h, y-k) - f(x-h, y+k) + f(x-h, y-k)}{4hk},$$

see Ames [1977, page 17].

To find the inverse of the hessian matrix $\mathcal{H}(\hat{\boldsymbol{\beta}}_{ML})$, the LUP-decomposition algorithm for solving linear equations is used. An explanation of the LUP-decomposition methods is given here, and it follows the same outline as in the book by Strang [2019, p.21-23].

To find the inverse $A^{-1} = B$ of an n-dimensional matrix, A, is to solve n linear equations in $AB = I$, where I is the identity matrix.

The LUP-decomposition algorithm decomposes an $n \times n$ matrix A, into a lower triangular matrix L, an upper triangular matrix U, and a permutation matrix P. The permutation matrix P rank the rows of A with the highest values, starting from column 1 top-to-bottom, then PA is decomposed into L and U.
The LU decomposition is given by

$$PA = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \ell_{2,1} & 1 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ \ell_{n,1} & \ell_{n,2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n} \\ 0 & u_{2,2} & \cdots & u_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & u_{n,n} \end{bmatrix}.$$

The first row of U is the first row of PA, and the first column of L is given by $\ell_{1,1} = 1$ and $\ell_{i,1} = \frac{a_{i,1}}{a_{1,1}}$ for $i = 1, .., n$.

Then, The second row of U is the first row of A_2^* which contains an $(n-1)$-dimensional matrix A_2 as given by the equation

$$PA = \ell_1 u_1^* + A_2^* \quad = \begin{bmatrix} 1 \\ \ell_{2,1} \\ \vdots \\ \ell_{n,1} \end{bmatrix} \begin{bmatrix} u_{1,1} & u_{2,1} & \cdots & u_{1,n} \end{bmatrix} + \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & A_2 & \\ 0 & & & \end{bmatrix},$$

where u_1^* is the first row vector of U. One can find this A_2^* matrix by subtracting $\ell_1 u_1^*$ from PA.

The decomposition continues in the same way until it is finished. Row vector u_2^* is the second row of the upper triangular matrix U, and of A_2^*, then $\ell_2 = (0, 1, \ell_{3,2}, .., \ell_{n,2})^T$ is obtained by $\ell_{i,2} = \frac{u_{2,i}}{u_{2,2}}$ for $2 < i \leq n$. Subtracting $\ell_2 u_2^*$ from A_2^* gives the matrix A_3^*, which third row vector is the third row vector u_3^*, and so on.

After the decomposition is found the inverse $B = A^{-1}$ is computed by, using *forward* and *back substitution*, solving the linear equations $Ab_i = e_i$, for

$i = 1, .., q$, where b_i is the i:th column vector of B, and e_i the i:th column vector of the identity matrix.

Equation $LUb_1 = e_1$ can be solved by defining a vector c_1 as $Ub_1 = c_1$ and then start by solving the equation $Lc_1 = e_1$. Expressing these linear equations in matrix form

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ \ell_{2,1} & 1 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ \ell_{n,1} & \ell_{n,2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} c_{1,1} \\ c_{2,1} \\ \vdots \\ c_{n,1} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

show that $c_{1,1} = 1$, $c_{2,1} = \ell_{2,1}$ and so on. By forward substituting the $c_{i,1}$'s, from $i = 1$ to n, these equations are solved.

Then, using the back substituting give the solution to the equations in

$$\begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n} \\ 0 & u_{2,2} & \cdots & u_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & u_{n,n} \end{bmatrix} \begin{bmatrix} b_{1,1} \\ b_{2,1} \\ \vdots \\ b_{n,1} \end{bmatrix} = \begin{bmatrix} c_{1,1} \\ c_{2,1} \\ \vdots \\ c_{n,1} \end{bmatrix}.$$

When all $b_1, .., b_n$ are solved, the inverse of the matrix A can be computed by $A^{-1} = (P^{-1}PA)^{-1} = PB = P[b_1, ..., b_n]$. The estimated maximum likelihood variances of the estimated coefficients is then the negated diagonal values of the inverted hessian matrix.

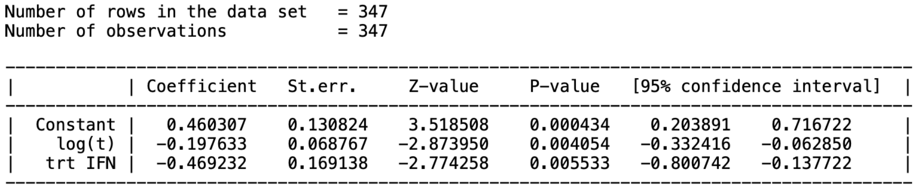
3.5.5 Result window and output

The application is demonstrated, again using the `kidney` data from Section 1.2. A proportional odds regression is estimated, with two regressors `trt` and `log(t)`. The back-end program maximizes the log-likelihood for the regression line

$$logit[g(t)] = \beta_0 + \beta_1 trt + \beta_2 \log(t),$$

using the compass optimization algorithm. The variances of the coefficients are estimated using the LUP-decomposition algorithm to compute the inverse of the hessian matrix, and z, and p-values are calculated, using JavaScript standard library `jStat`.

Estimated coefficients are displayed, together with standard errors, z-values, p-values and the 95% confidence intervals, in the result window:



```
Number of rows in the data set   = 347
Number of observations           = 347

----------------------------------------------------------------------------------------
|           | Coefficient   St.err.     Z-value     P-value   [95% confidence interval]  |
----------------------------------------------------------------------------------------
|  Constant |   0.460307    0.130824    3.518508    0.000434    0.203891    0.716722     |
|    log(t) |  -0.197633    0.068767   -2.873950    0.004054   -0.332416   -0.062850     |
|   trt IFN |  -0.469232    0.169138   -2.774258    0.005533   -0.800742   -0.137722     |
----------------------------------------------------------------------------------------
```

The output can be compared with the result in Subsection 3.1 and one can conclude that they are equal up to some negligible numerical approximation.

4 A real data example

In this section applications and functionalities of the *incidence risk* and the *event-probability function* are demonstrated with a data example, using 1000 data points from the *Whitehall* data.

The British Whitehall study is a cross-sectional cohort study on more than 17.5 thousand civil servants in London. The data contain information on health factors from men in the ages 20 to 64 collected over ten years between 1967 and 1977. The main idea of the study was to investigate the impact of social and economical status on health and mortality [Marmot et al., 1978, 1987]. Over the years, the data from the Whitehall study has been used in medical and statistical research.

Table 1 shows a list of the variables included in the sample, those of interest for this example are marked with bold text.

The variable `pyall10` contains the follow-up times of the study, and the time is set to 9.99 years for those still alive at the end of the follow-up, and the time of death for those that died during the research period. The mortality indicator `all10` denotes observed survival times and censored observations. Data is heavily right censored, naturally, and contain 99 deaths observations.

For a demonstration of the theory of the incidence risk and the event-probability function, two group variables `smoke` and `jobgrade` are used to test for differences in mortality risk between smokers and non smokers and between groups with different social and economical status.

The analysis is done using the two Stata commands `stprisk` and `stpreg`, presented in Subsection 3.1.

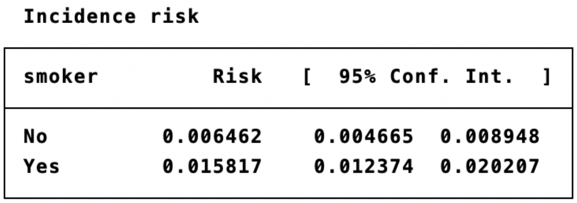
variable name	variable label
serno	Serial Number
all10	**All cause mortality**
pyall10	**Years of follow-up (all10)**
chd	CHD mortality
pyar	Years of follow-up (chd)
jobgrade	**Job grade**
age	Age (years)
sysbp	Systolic blood pressure (mm Hg)
map	Mean arterial pressure (mm Hg)
ht	Height (cm)
chol	Cholesterol (mmol/l)
agecat	Age categories (years)
bmi	Body mass index (kg/m^2)
cigs	Daily cigarette consumption
diasbp	Diastolic blood pressure (mm Hg)
wt	Weight (kg)
smoke	**Smoking status**
sample	Indicator for the sample

Table 1: Variables from the Whitehall study with short descriptions.

This example begins by estimating the mortality risk for the two groups defined by the variable `smoke`, smokers and non smoker. The command

```
stprisk smoke
```

gives the following results.



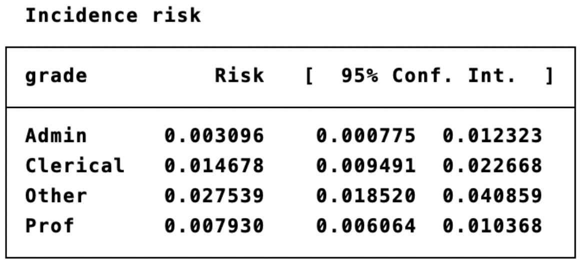
```
Incidence risk

smoker         Risk   [  95% Conf. Int.  ]

No         0.006462    0.004665  0.008948
Yes        0.015817    0.012374  0.020207
```

Smokers have an estimated mortality risk of 1.6% per year, compared to the non-smokers whose mortality risk is 0.6% per year.

Then, the mortality risks are estimated for the variable `jobgrade`, containing four levels of professional grade. These different professional groups constitutes different social and economical classes in the British society. In descending order, these four grades are: `Admin` for administrator, `Prof` for professional and executive, `Clerical` is self explanatory, and `Other` for messengers and doorkeepers etc [Marmot et al., 1987]. The incidence risks for the four levels of professional grade are estimated by running the command:

```
stprisk jobgrade
```

This gives the following results:



```
Incidence risk

grade          Risk    [  95% Conf. Int.  ]

Admin      0.003096    0.000775  0.012323
Clerical   0.014678    0.009491  0.022668
Other      0.027539    0.018520  0.040859
Prof       0.007930    0.006064  0.010368
```

Estimated mortality is lower for the higher professional grades and ascending with lower grades. Administrative personnel have an estimated mortality risk of 0.3%, professional and executive personnel 0.8%, clerical personnel 1.4% and those with the lowest grade 2.7% per year. One can statistically establish a difference in mortality risk between the highest level of professional grades `Admin` and the lowest grade `Other` and also between `Prof` and `Other`, on a 95% level of confidence.

This data example continues with the event-probability regression from Subsection 2.6.5. Again, the difference in mortality between smokers and non smokers are explored and then the differences in mortality between different social and economical classes.
Three proportional odds regressions models are estimated with an increasing level of complexity.

$$logit[g(t)] = \beta_0 + \beta_1 smoke, \tag{52}$$

$$logit[g(t)] = \beta_0 + \beta_1 smoke + \beta_2 log(t), \tag{53}$$

$$logit[g(t)] = \beta_0 + \beta_1 smoke + \beta_2 log(t) + \beta_3 s_2(log(t)). \tag{54}$$

The second model in Equation 53 includes the covariate $s_1 = log(t)$, and the third model in Equation 54 includes a spline function $s_2(log(t))$. This spline function is the restricted cubic spline (RCS) given by Equation 37.

Table 2 show estimated coefficients for the three regression models, together with standard errors, in parenthesis, and p-values. The low p-value of the variable `smoke` statistically assert the impact of smoking on the mortality, also, the coefficients for $log(t)$ are significant but not for the spline function $s_2(log(t))$.

	Model 1		**Model 2**		**Model 3**	
	coef.	p-val	coef.	p-val	coef.	p-val
cons.	-5.044(0.167)	0.000	-4.499(0.191)	0.000	-4.439(0.221)	0.000
smoke	0.892(0.210)	0.000	0.905(0.210)	0.000	0.906(0.210)	0.000
$\mathbf{s_1}$			0.198(0.045)	0.000	0.212(0.051)	0.000
$\mathbf{s_2}$					-0.019(0.037)	0.597
loglik.	-372.079		-357.942		-357.808	

Table 2: Estimates of the three regression models in Equation 52, 53, and 54, with standard errors in brackets and p-values. The coefficient for *cons* gives the constant β_0, s_1 the coefficient for $log(t)$ and s_2 the coefficient for the RCS. The last row, *loglik*, show the log-likelihoods.

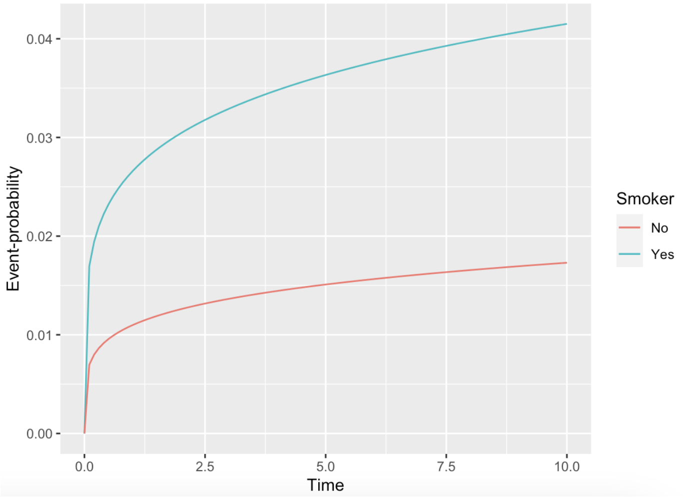


Figure 3: Estimated mortality risks for smokers and non smokers over a period of ten years.

Figure 3 shows the estimated event-probabilities of smokers and non smokers from the model in Equation 53. Over the ten year period, the estimated probability of dying for a smoker is both higher and steeper than that of a non smoker, ranging from around 2% in the first year and over 4% after ten years. The probability of dying for a non smoker is estimated to range between 1% an 1.7% during this period.

Three more proportional odds regression models are estimated, exploring the mortality of the different groups of social and economical classes including three indicator variables, representing the four levels of the variable `jobgrade`. The lowest job grade level `other` is left as baseline and `cler`, `prof`, and `admin` are added as regressors. Again, Three proportional odds regressions models are estimated with an increasing level of complexity.

$$logit[g(t)] = \beta_0 + \beta_1 cler + \beta_2 prof + \beta_3 admin, \tag{55}$$

$$logit[g(t)] = \beta_0 + \beta_1 cler + \beta_2 prof + \beta_3 admin + \beta_4 log(t), \tag{56}$$

$$logit[g(t)] = \beta_0 + \beta_1 cler + \beta_2 prof + \beta_3 admin + \beta_4 log(t) + \beta_5 s_2(log(t)). \tag{57}$$

Table 3 shows the estimated coefficients of the three models. Statistical evidence is found for that mortality vary with log-time, while the impact of the spline function s_2 gives insignificant results.

	Model 4		**Model 5**		**Model 6**	
	coef.	p-val	coef.	p-val	coef.	p-val
cons.	-3.599(0.207)	0.000	-3.013(0.230)	0.000	-2.949(0.257)	0.000
cler.	-0.612(0.306)	0.045	-0.632(0.307)	0.039	-0.634(0.307)	0.039
prof	-1.243(0.249)	0.000	-1.276(0.249)	0.000	-1.278(0.249)	0.000
admin	-2.180(0.738)	0.003	-2.223(0.738)	0.003	-2.226(0.738)	0.003
$\mathbf{s_1(t)}$			0.201(0.045)	0.000	0.215(0.051)	0.000
$\mathbf{s_2(t)}$					-0.020(0.037)	0.576
loglik.	-367.762		-353.320		-353.171	

Table 3: Estimated coefficients for the three regression models in Equation 55, 56, and 57, together with standard errors in brackets and p-values. The coefficient for *cons.* give the constant β_0, s_1 the coefficient for $\log(t)$ and s_2 the coefficient for the spline function. The last row, *loglik*, give the log-likelihoods of the models.

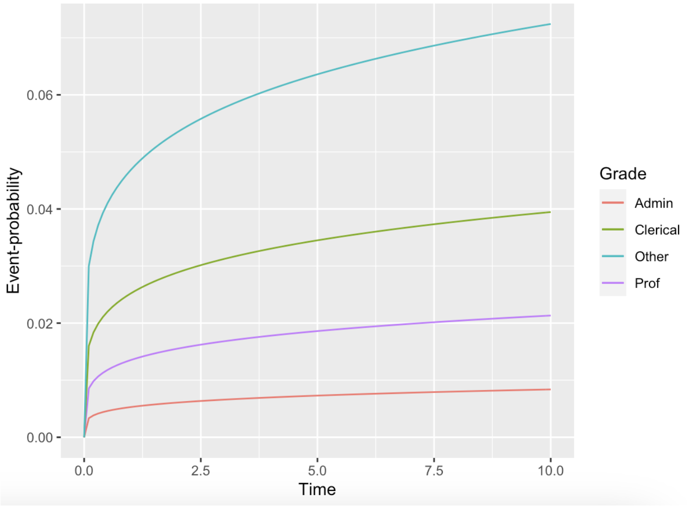


Figure 4: Estimated mortality risks for different professional grades over a period of ten years.

The event-probabilities for the different levels of `jobgrade` of the fifth model, in Equation 56, are shown in Figure 4. The difference in mortality is evident between individuals of the lowest professional grade `other`, which in the first year has a mortality over 4.5%, compared to an individual of the highest professional grade `administrator`, which has a mortality just reaching 0.5%, during the same time. Also, an individual with professional grade `other`, which has the highest probability of dying.

5 Future possibilities

This thesis describes and elucidates the features of the theory of the incidence risk, the event-probability function and event history analysis in general. The idea is to present it with mathematical rigor and at the same time make it understandable and usable. The theory of the incidence risk and the event-probability offers interpretable measurements of risk in the case of the one-time-only event, where classical theory does not.

After two applications for computing incidence risks and preforming event-probability regression where presented and demonstrated with some data examples, the question was where to go next.

I can begin with some thoughts on the two new web applications, which where presented in Section 3, and some changes spring to mind.

First, people might have other types of file formats they want to use, other than text- and csv-files. Today, virtually all statistical software programs can upload excel-files, and researchers are used to these file formats.

Secondly, the compass optimizer is an easy implemented algorithm that will always find its optima, but it is slow. For two regressors the application will only take a few seconds to finish but when running a proportional odds regression with RCS, seconds turn in to minutes while the Stata command `stpreg` finish in fractions of seconds. There is nothing standing in the way of implementing an algorithm that uses the gradient in the search of an optima, except lack of time on my part.

A third improvement would be to add more options. The option of using RCS or time varying covariates is one example. The option of running a proportional ratio model in stead of a proportional odds model is another possibility. The article by Bottai et al. [2021] present a lot of other options for implementations of event-probability regression models, and there are more to be found.

And last but not least, the fourth improvement is to consider a more accurate approximation of the log-likelihood integral. Integration of $\log(t)$ can be tricky because small values of $t < 1$ will produce highly negative values and more precision is needed for those values, than for values of $t > 1$. Some adaptive approximation algorithm could be preferable.

I do not think that one should ignore the benefits of less-is-more, when reaching out to users. Sometimes, all one needs is something that is easy to use. But, a natural step would be to consider a more advanced implementations of theory presented, in one of the most used programming languages in data science; R.

R is a programming language that can reach out to many users that may be interested in more advanced statistical methodology. Implementing R-libraries for the methods presented here would make it available to them, and they are prone to continue where this theory ends and develop it further.

What is possible for a user of the simple web applications is just a fraction of what a statistician would be able to do when using an R-library. With multiple options of model set-up and output, theory becomes a building-block.

Finally, I may speculate on where this theory could lead in the future.

The options of how one could utilize the event-probability are many. The case where one can go from one state to several others states represents a possible avenue for future research. An example is the case where on can move from a cancer diagnosis to death, cancer free, or a new diagnosis.

References

O. O. Aalen, Ø. Borgan, and H. K. Gjessing. *Survival and Event History Analysis*. Statistics for Biology and Health. Springer, 2008. ISBN 9780387202877.

W. F. Ames. *Numerical Methods for Partial Differential Equations, second edition*. Academic Press, inc., 1977. ISBN 0120567601.

S. Benet. Analysis of survival data by the proportional odds model. *Statistics in Medicine*, 2(2):273–277, 1982.

M. Bostock. Javascript for data analysis. *Towards Data Science*, (May 25, 2021), 2021. URL `https://towardsdatascience.com/javascript-for-data-analysis-2e8e7dbf63a7`.

M. Bottai. A regression method for modelling geometric rates. *Statistical Methods in Medical Research*, 26(6):2700–2707, 2017. doi: https://doi-org.ezp.sub.su.se/10.1177/0962280215606474.

M. Bottai. Estimating the risk of events with stprisk. *The Stat Journal*, 22(4): 969–974, 2022. doi: https://doi.org/10.1177/1536867X221141057.

M. Bottai, N. Orsini, and M. Geraci. A gradient search maximization algorithm for the asymmetric laplace likelihood. *Journal of Statistical Computation and Simulation*, 85(10):1919–1925, 2015. doi: https://www.tandfonline.com/doi/epdf/10.1080/00949655.2014.908879?needAccess=true&role=button.

M. Bottai, A. Discacciati, and G. Santoni. Modeling the probability of occurrence of events. *Statistical Methods in Medical Research*, 30(9):1976–1987, 2021. doi: https://doi.org/10.1177/09622802211022403.

M. Breuss. Python web applications: Deploy your script as a flask app. URL `https://realpython.com/python-web-applications/`.

C and ASM. *Systems of Linear Equations*. Mathematics Source Library, C and ASM. URL `http://www.mymathlib.com/matrices/linearsystems/doolittle.html`.

D. Collett. *Modelling Survival Data in Medical Research, third edition*. Texts in Statistical Science Series. Chapman and Hall, 2003. ISBN 9781439856789.

M. J. Collins. *Pro HTML5 with CSS, JavaScript, and Multimedia: Complete Website Development and Best Practices*. Apress, 2017. ISBN 9781484224625.

D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society, Series B*, 34(2):187–220, 1972.

F. E. Harrell Jr. *Regression Modelining Strategies - With aplications to linear models, logistic regression and survival analysis*. Springer series in statistics. Springer, 2001. ISBN 9781475734621.

L. Held and D. S. Bové. *Applied Statistical Inference*. Springer, 2014. ISBN 9783642378874.

S. Kirmani and R. C. Gupta. On the proportional odds model in survival analysis. *Annals of the Institute of Statistical Mathematics*, 53(2):203–216, 2001.

J. Lin and K. E. Gebaly. The future of big data is ... javascript? *IEEE Internet Computing*, 20(Sept.-Oct. 2016):82–88, 2016. doi: 10.1109/MIC.2016.109. URL `https://www.computer.org/csdl/magazine/ic/2016/05/mic2016050082/13rRUxYrbQM`.

X. Liu. *Survival Analysis - Models and Applications*. John Wiley Sons, Ltd., 2012. ISBN 9780470977156.

M. G. Marmot, G. Rose, M. Shipley, and P. J. S. Hamilton. Employment grade and coronary heart disease in british civil servants. *J Epidemiol Community Health*, 32(4):244–249, 1978. doi: 10.1136/jech.32.4.244.

M. G. Marmot, M. Kogevinas, and M. A. Elston. Social/economic status and disease. *Ann. Rev. Public Health*, (8):111–135, 1987. doi: 10.1146/annurev.pu.08.050187.000551.

MDN. *Django Introduction*. MDN web docs, 2023. URL `https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction`.

M. R. C. Medical Research Council Renal Cancer Collaborators. Interferon-alpha and survival in metastatic renal carcinoma: early results of a randomised controlled trial. *The Lancet*, 353(9146):14–17, 1999.

R. Rebolledo. Central limit theorems for local martingales. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 51:269–286, 1980. doi: https://doi.org/10.1007/BF00587353.

M. J. Rutherford, M. J. Crowther, and P. C. Lambert. The use of restricted cubic splines to approximate complex hazard functions in the analysis of time-to-event data: a simulation study. *Journal of Statistical Computation and Simulation*, 85(4):777–793, 2015. doi: 10.1080/00949655.2013.845890.

B. Schmidt. Javascript and the next decade of data programming, 2021. URL `https://benschmidt.org/post/2020-01-15/2020-01-15-webgpu/`.

StataCorp. *Stata: Release 17*. StataCorp LLC, 2021. URL `https://www.stata.com/manuals/st.pdf`.

G. Strang. *Linear Algebra and Learning from Data*. Wellesley- Cambridge Press, 2019. ISBN 9780692196380.

E. Süli and D. F. Mayers. *An Introduction to Numerical Analysis*. Cambridge University Press, 2003. ISBN 9780511801181. doi: 10.1017/CBO9780511801181.

S. Yegulalp. Intro to pyscript: Run python in your web browser. *Info World*, (June 15, 2022), 2022. URL `https://www.infoworld.com/article/3661628/get-started-with-pyscript-the-in-browser-python-by-anaconda.html`.

Appendix A Functions in JavaScript

Here follows some JavaScript implementations of the algorithms explained in Subsection 3.5.

A.1 Integral approximation

The integral approximation function, used for computing the log-likelihood function in Equation 50, takes three variables

1. `index` - the index of the time-to-event observation which is the end point of the integral,
2. `theta` - the coefficient vector of the likelihood function, for which the integral will be computed,
3. `data` - the multi-dimensional list containing all the data points used for the regression.

```javascript
integralApprox = function(index, theta, data) {

    // The end point of the time-to-event variable (y):
    const yi = data.y[index];

    // # of intervals for used for the approximation:
    var n=200;

    // The increment:
    var dy = yi/n;

    var integral = 0;
    const k = theta.length - data.x.length;

    const fx_i = data.x.reduce((tot,value,i)=>{
        return tot + theta[i+k]*value[index];
    },0)

    const fy_i = ((y) => {
        return theta[1]*Math.log(y);
    });

    var e_prev = Math.exp(theta[0]+ fy_i(0.00000001) + fx_i);

    for (let i=1; i <= n; i++) {
        if(i==n){
            const dy_last = yi - (n-1)*dy;
            const e_curr = Math.exp(theta[0]+ fy_i(yi) + fx_i);
            integral -= ((Math.log(e_prev + 1) +
                        Math.log(e_curr + 1))/2)*dy_last;
        }
        else if (i==1){
            const e_curr = Math.exp(theta[0]+ fy_i(i*dy) + fx_i);
            integral -= Math.log(e_curr + 1)*dy;
            e_prev = e_curr;
        }
        else{
            const e_curr = Math.exp(theta[0]+ fy_i(i*dy) + fx_i);
            integral -= ((Math.log(e_prev + 1)+Math.log(e_curr + 1))/2)*dy;
            e_prev = e_curr;
        }
    }
    return integral;
}
```

A.2 Optimization algorithm

The optimization algorithm function takes five variables

1. lh - a log-likelihood function for the data which takes one variable, a coefficient vector theta, and return the value of the log-likelihood computed for the data and the given coefficients,

2. theta0 - a list, the initial coefficient vector and the starting point of the algorithm,

3. tol - tolerance level,

4. maxIter - an integer, a stopping time for the algorithm in case it converges to slow,

5. INITIALIZE - a boolean, set to True if the algorithm is used with a lower amount of data points to initialize the starting point theta0.

```
compassOptimizer = function(lh, theta0, tol, maxIter, INITIALIZE) {
    const nCoef = theta0.length;
    var step = tol*100*2;
    var optimal = false;
    var direction = 0;
    var count = 0;
    var curr = lh(theta0); //current log-likelihood value
    var noStep = true;

    while(!optimal) {
        if (direction < nCoef){
            if (this.takeAStepp(curr, lh, theta0, direction, step)){
                curr = lh(theta0);
                step *= 1.75;
                noStep = false;
            }
            else {
                direction++;
            }
        }
        else {
            if (this.takeAStepp(curr, lh, theta0, (direction-nCoef), -step)){
                curr = lh(theta0);
                step *= 1.75;
                noStep = false;
            }
            else {
                direction++;
            }
        }
        if(step < tol) {
            optimal = true;
        }
        if(direction == 2*nCoef){
            direction = 0;
            step *= 0.65;
        }
        count++;

        if (count >= maxIter) {
            if(!INITIALIZE)
                this.writeError("Likelihihood function was not able to converge.")
            return theta0;
        }
        else if(count % 1000 == 0){
            console.log("==========> " + count);
        }
    }
    return theta0;
}
```

A.3 Computing covariance matrix

The covariance matrix is computed in three parts

(i) approximation of hessian matrix,

(ii) finding the LUP-decomposition of the hessian,

(iii) invert the hessian using the LUP decomposition.

The hessian matrix is approximated using Equation 3.5.4 in Section 3.5 and the algorithm is given by the function `hessianMatrixApprox()` which takes two variables

1. `lh` - a log-likelihood function for the data which takes one variable, a coefficient vector `theta`, and return the value of the log-likelihood computed for the data and the given coefficients,

2. `thetaHat` - a list of maximum likelihood estimated coefficients.

The function returns a matrix of approximated second partial derivatives.

```
hessianMatrixApprox = function(lh,thetaHat) {
    const underOver = thetaHat.map((x) => {
        const dTheta = x/100;
        const under = x-dTheta;
        const over = x+dTheta;
        return [under,over,dTheta];
    });
    const derivatives = underOver.map((val_i,i) => {
        const row = underOver.map((val_j,j)=>{
            if(i==j){
                var thetaUnder = [...thetaHat];
                thetaUnder.splice(i,1,val_i[0]);

                var thetaOver = [...thetaHat];
                thetaOver.splice(i,1,val_i[1]);
                return
                (
                    (
                        lh(thetaUnder) + lh(thetaOver) - 2*lh(
                            thetaHat)
                    )/Math.pow(val_i[2],2)
                );
            }
            else {
                var thetaPP = [...thetaHat];
                thetaPP.splice(j,1,val_j[1]);
                thetaPP.splice(i,1,val_i[1]);
                var thetaPN = [...thetaHat];
                thetaPN.splice(j,1,val_j[0]);
                thetaPN.splice(i,1,val_i[1]);
                var thetaNP = [...thetaHat];
                thetaNP.splice(j,1,val_j[1]);
                thetaNP.splice(i,1,val_i[0]);
                var thetaNN = [...thetaHat];
                thetaNN.splice(j,1,val_j[0]);
                thetaNN.splice(i,1,val_i[0]);

                return ((lh(thetaPP)-lh(thetaPN)-lh(thetaNP)+lh(thetaNN))/(4*val_j[2]*val_i[2]));
            }
        });
        return row;
    });
    return derivatives;
}
```

The LUP-decomposition and the inversion algorithms used are found at C and ASM. The function `LUPDecompose()` takes three variables

1. `A` - a matrix,

2. `N` - an integer, the dimension of `A`, and

3. `tol` - a number, the tolerance level.

The function returns a decomposed matrix containing both the upper triangular matrix and the lower triangular matrix, and a permutation matrix.

```
LUPDecompose = function(A, N, tol) {
    var imax = 0;
    var maxA;
    var ptr = 0.0;
    var absA = 0.0;
    var P = Array(N+1).fill(0);
    for (var i = 0; i <= N; i++) {
        P[i] = i; //Unit permutation matrix, P[N] initialized with N
    }
    for (var i = 0; i < N; i++) {
        maxA = 0.0;
        imax = i;
        for (var k = i; k < N; k++) {
            if ((absA = Math.abs(A[k][i])) > maxA) {
                maxA = absA;
                imax = k;
            }
        }
        if (maxA < tol) {return 0;} //failure, matrix is degenerate
        if (imax != i) {
            //pivoting P
            j = P[i];
            P[i] = P[imax];
            P[imax] = j;

            //pivoting rows of A
            ptr = A[i];
            A[i] = A[imax];
            A[imax] = ptr;

            //counting pivots starting from N (for determinant)
            P[N]++;
        }
        for (var j = i + 1; j < N; j++) {
            A[j][i] /= A[i][i];
            for (var k = i + 1; k < N; k++)
                A[j][k] -= A[j][i] * A[i][k];
        }
    }
    return [A,P];
}
```

The inversion algorithm is a function `LUPInvert()` that takes

- `A` - a matrix containing the LUP-decomposition given by the function `LUPDecompose()`,
- `P` - a permutation matrix, and
- `N` - an integer, the dimension of `A`.

```
LUPInvert = function(A, P, N) {
    var IA = new Array(N).fill(0).map(()=>{
        return new Array(N).fill(0);
    })
    for (var j = 0; j < N; j++) {
        for (var i = 0; i < N; i++) {
            if(P[i] == j) {
                IA[i][j] = 1.0;
            }
            else {
                IA[i][j] = 0.0;
            }
            for (var k = 0; k < i; k++)
```

```
                IA[i][j] -= A[i][k] * IA[k][j];
        }
        for (var i = N - 1; i >= 0; i--) {
            for (var k = i + 1; k < N; k++)
                IA[i][j] -= A[i][k] * IA[k][j];
            IA[i][j] /= A[i][i];
        }
    }
    return IA;
}
```