Who needs the Cox model anyway?

SDCC
November 2023
http://bendixcarstensen.com/WntCma.pdf
http://bendixcarstensen.com/WntCma.R
Version 4

Compiled Sunday 19th November, 2023, 17:50
from: C:\Bendix\teach\AdvCoh\art\WntCma/WntCma.tex

Bendix Carstensen Steno Diabetes Center Copenhagen, Herlev, Denmark
& Department of Biostatistics, University of Copenhagen
bcar0029@regionh.dk b@bxc.dk
http://BendixCarstensen.com

Chapter 1

Theory

1.1 Introduction

First it should be noted that the content of this note by no means is original; John Whitehead devised a similar machinery already in 1980 [3], using GLIM. Here things are laid out using `R`.

In the last about 50 years, survival analysis has been virtually synonymous with application of the Cox-model. The common view of survival analysis (and teaching of it) from the Kaplan-Meier-estimator to the Cox-model is based on time as the response variable, incompletely observed (right-censored). This has automatically lent a certain aura of complexity to concepts such as time-dependent covariates, stratified analysis, delayed entry and time-varying coefficients.

More unfortunate, however, is that the use of this particular technique for survival analysis has become a dominant tool in epidemiology too, largely restricting models for occurrence rates to models with only one time scale.

If survival studies is viewed in the light of the demographic tradition, the basic observation is not one time to event (or censoring) for each individual, but rather many small pieces of follow up from each individual. This makes concepts clearer as modeling of rates rather than time to response becomes the focus; the basic response is now a $0/1$ outcome in each interval, albeit not independent anymore, but still with a likelihood which is a product across intervals.

In this set up, time(scale) is then correctly viewed as a covariate rather than a response. From a practical point of view time-dependent covariates will not have any special status relative to other covariates. Stratified analysis becomes a matter of interaction between time and a categorical covariate, and time-varying coefficients becomes interactions between time and a continuous covariate. Finally, the modeling tools needed reduces to Poisson regression (and ultimately logistic regression) — standard generalized linear models.

The Cox-model may actually be viewed as a special case of a Poisson model where the detail in modeling of the time covariate has been taken *ad absurdum*, namely with one parameter per failure time. The main advantage of the demographic view is therefore that researchers will be forced to explicitly consider which time-scale(s) to use and to what degree of detail it is relevant to model interactions between time scales and other covariates.

Contrary to this, Poisson modeling of disease rates and follow-up studies in epidemiology has traditionally (and until 1990 for good computational reasons) been restricted to analysis of tables where rates have been assumed constant over fairly broad time-spans, typically 5 years, as most methods have been developed in cancer epidemiology, where 5 years is considered a short age-span. This approach is essentially one where initial tabulation of data

unnecessarily limits the flexibility of modeling (and discards information).

If follow-up time both in survival and cohort studies are considered in small intervals, the smoothing of rates can be done with standard regression tools in Poisson modeling.

The practicalities of this type of modeling focus requires a splitting of the follow-up in many small intervals, and hence Poisson modeling of datasets with many records, each representing a small piece of the follow-up time.

The only remaining advantage of the Cox-model is the ability to easily produce estimates of survival probabilities in (clinical) studies with a well-defined common entry time for all individuals, and hence with a single timescale. This can however also be produced from a model using a smooth parametric form for the occurrence rates.

1.2 Time: Response or covariate?

Both, actually.

One common exposition of survival analysis is as analysis of data (X, Z), where we only observe $\min(X, Z)$ and $\delta = 1\{Z < X\}$. This is an approach which takes the survival time X, as response variable, albeit not fully observed, limited by the censoring time, Z.

The snag here is that apart from being a measure of *how long* the person has been at risk, X is also a measure of *when* the person was at risk; if we refer to time since entry to the study, namely at times 0 through X. The entry time to the study is implicitly taken to be 0, so easy to confuse risk time $(X - 0)$ and time scale $((0, X])$.

From a life-table (demographic) point of view, we will break the interval $(0, X]$ in, say, one-year intervals, each one with a different time associated (be that age or time since entry). In this sense time is a covariate, and only differences (*i.e. risk time*) on a timescale should be considered responses. In a life-table, differences on the timescale (interval length) are accumulated as risk time whereas the position on the age-scale for these are used as a covariate classifying the table.

Now consider a follow-up (survival) study where the follow-up time for each individual is divided into small intervals of equal length y, say, and each with an exit status recorded (this will be 0 for the vast majority of intervals and only 1 for the last interval for individuals experiencing an event).

Each small interval for an individual contributes an observation of what I will term an *empirical rate*, (d, y), where d is the number of events in the interval (0 or 1), and y is the length of the interval, i.e. the risk time. This is slightly different from the traditional definition of an empirical rate as d/y (or $\sum d / \sum y$); it is designed to keep the entire information content in the demographic observation, even if the number of events is 0. This is in order to make it usable as a response variable, showing that in a follow-up study the basic observation is a (set of) rate(s).

The *theoretical* rate of event occurrence is defined as a function, usually depending on some timescale, t:

$$\lambda(t) = \lim_{h \searrow 0} \frac{\mathrm{P}\{\text{event in } (t, t+h] | \text{ at risk at time } t\}}{h}$$

The rate may depend on any number of covariates; incidentally on none at all. Note that in this formulation time(scale) t has the status of a covariate and h the status of risk time, h is the difference between two points on the timescale (in this case $t + h$ and t).

1.2.1 Likelihood for a single rate

To derive the likelihood for a rate we must compute the probability of an observation from a single person as a function of the rate λ. Suppose a person is alive from time t_e (entry) to t_x (exit) and that the person's status at t_x is d, where $d = 0$ means alive and $d = 1$ means dead. If we choose, say, two time points, t_1, t_2 between t_e and t_x, standard use of conditional probability (formally, repeated use of Bayes' formula) gives

$$\begin{aligned} \mathrm{P}\{d \text{ at } t_x \mid \text{entry at } t_e\} = &\mathrm{P}\{\text{survive } (t_e, t_1] \mid \text{alive at } t_e\} \times \\ &\mathrm{P}\{\text{survive } (t_1, t_2] \mid \text{alive at } t_1\} \times \\ &\mathrm{P}\{\text{survive } (t_2, t_x] \mid \text{alive at } t_2\} \times \\ &\mathrm{P}\{d \text{ at } t_x \mid \text{alive just before } t_x\} \end{aligned} \tag{1.1}$$

By choosing more intermediate time points we can make the intervals arbitrarily small, so we just need to derive the probability of surviving a small piece of time, as a function of the mortality rate.

For a start assume that the mortality is constant over time $\lambda(t) = \lambda$. From the definition of a rate we have (conditional on being alive at t):

$$\begin{aligned} \mathrm{P}\{\text{death during } (t, t+h]\} &\approx \lambda h \\ \Rightarrow \mathrm{P}\{\text{survive } (t, t+h]\} &\approx 1 - \lambda h \end{aligned} \tag{1.2}$$

where the approximation gets better the smaller h is. Suppose we have survival for a time span $y = t_x - t_e$ and that this is subdivided in N intervals, each of length $h = y/N$, then the survival probability for the entire span from t_e to t_x is the product of probabilities of surviving each of the small intervals, conditional on being alive at the beginning of each interval:

$$\mathrm{P}\{\text{survive } t_e \text{ to } t_x\} \approx (1 - \lambda h)^N = \left(1 - \frac{\lambda y}{N}\right)^N$$

From mathematics it is known that $(1 + x/n)^n \to \exp(x)$ as $n \to \infty$ (some define $\exp(x)$ this way). So if we divide the time span y in successively smaller pieces we will have that $N \to \infty$, and hence that:

$$\mathrm{P}\{\text{survive } t_e \text{ to } t_x\} \approx \left(1 - \frac{\lambda y}{N}\right)^N \to \exp(-\lambda y), \quad N \to \infty \tag{1.3}$$

Therefore the contribution to the likelihood from a person observed for a time span of length y is $\exp(-\lambda y)$, and the contribution to the log-likelihood is therefore $-\lambda y$.

If we observe a person dying at the end of the last interval, the contribution to the likelihood from the last interval will be the probability surviving till just before the end of the interval, multiplied by the probability of dying in the last tiny instant (of length ϵ, say) of the interval (the last term in (1.1)). This probability is by (1.2) $\lambda\epsilon$, and hence the log-likelihood contribution from this last instant is $\log(\lambda\epsilon) = \log(\lambda) + \log(\epsilon)$.

The total likelihood for one person is the product of all these terms from the follow-up intervals (i) for the person; and the log-likelihood (ℓ) is therefore:

$$\begin{aligned} \ell(\lambda) &= -\lambda \sum_i y_i + \sum_i d_i \log(\lambda) + \sum_i d_i \log(\epsilon) \\ &= \sum_i \left(d_i \log(\lambda) - \lambda y_i\right) + \sum_i d_i \log(\epsilon) \end{aligned}$$

where y_i is observation time (risk time or person-years) in the i^{th} interval and d_i the death indicator (0/1) for the i^{th} interval (which is 0 for all intervals, except possibly for the last). We want to estimate λ, so terms that does not involve λ can be ignored—such as the last term. Very convenient, we then do not have to decide precisely how small ϵ is.

Thus we have established that the contribution to the log-likelihood from a single person's follow-up is the sum of a number of terms of the form $d_i \log(\lambda) - \lambda y_i$ where d_i is the event indicator (0/1) and y_i the length of the i^{th} interval.

If we subdivide the follow-up of a person across several records, each representing y_i of follow-up, and keep track of the deaths in each interval, d_i (which is 0 for all intervals, except possibly for the last if the person dies), then each record (d_i, y_i) will represent a contribution to the log-likelihood of $d_i \log(\lambda) - \lambda y_i$. This will be exploited in the practical modeling of rates.

By the assumption that the rate λ is constant over time, the log-likelihood contribution from a person with d deaths (0 or 1) at the end of a follow-up period of y is $d \log(\lambda) - \lambda y$. But once we have subdivided follow-up we do not need the assumption of a constant rate across the intervals, we can allow separate rates (λ_is) in different intervals, relaxing the assumption to only constant rates within each of the very small intervals. These rates can of course not be estimated based on the observation from a single person. A model for the λ_is must be specified in terms of covariates associated with each interval. Among these will normally be the value of one or more time scales at the beginning of each interval.

1.2.2 Poisson likelihood

The Poisson distribution with mean μ is a distribution on the non-negative integers ($x = 0, 1, 2, 3, \ldots$), where:

$$\mathrm{P}\{X = x\} = \frac{\mu^x \exp(-\mu)}{x!}$$

Thus, the log-likelihood from observation of a Poisson variate x with mean μ is $x \log(\mu) - \mu - \log(x!)$. The last term does not depend on the parameter μ so it can be omitted when maximizing the log-likelihood over values of μ.

Changing the notation, the log-likelihood contribution from a Poisson-variate d with mean λy is $d \log(\lambda y) - \lambda y = d \log(\lambda) - \lambda y + d \log(y)$, which is the same as the likelihood for d events during y follow-up time with rate λ, except for the term $d \log(y)$. But this term does not depend on the parameter λ and therefore can be ignored when maximizing the log-likelihood.

This means that maximum-likelihood estimation for observations from follow up, (d, y), can be done using a function that can maximize the likelihood for independent Poisson variables. We just need to pretend that the ds are independent Poisson variates with means λy. But recall that the ds are neither independent nor Poisson distributed.

In R we have the functions `glm` and `gam` to do this, depending on how we will model the covariate effects.

1.3 The Cox-likelihood as a profile likelihood

The Cox model [2] specifies the intensity (rate) as a function of time (t) and the covariates through the linear predictor $\eta_i = \beta_1 x_{1i} + \cdots + \beta_p x_{pi}$ as:

$$\lambda(t, x_i) = \lambda_0(t) \exp(\eta_i)$$

leaving the baseline hazard λ_0 unspecified.

Cox devised the *partial* (log-)likelihood for the parameters $\beta = (\beta_1, \ldots, \beta_p)$ in the linear predictor

$$\ell(\beta) = \sum_{\text{death times}} \log\left(\frac{\mathrm{e}^{\eta_{\text{death}}}}{\sum_{i\in\mathcal{R}_t}\mathrm{e}^{\eta_i}}\right)$$

where $\mathcal{R}_t$ is the risk set at time t, i.e. the set of individuals at risk at time t.

Suppose the time-scale has been divided into small time intervals with at most one death in each, and that we in addition to the regression parameters describing the effect of covariates use one parameter per time interval to describe the effect of time (i.e. the chosen timescale). Thus the model with constant rates in each small interval is:

$$\log\big(\lambda(t, x_i)\big) = \log\big(\lambda_0(t)\big) + \beta_1 x_{1i} + \cdots + \beta_p x_{pi} = \alpha_t + \eta_i$$

using $\alpha_t = \log\big(\lambda_0(t)\big)$ Assume w.l.o.g. the y for these empirical rates are 1. The log-likelihood contributions that contain information on a specific time-scale parameter α_t, relating to a particular time t, will be contributions from the empirical rate $(d, y) = (1, 1)$ with the death at time t, and the empirical rates $(d, y) = (0, 1)$ from all other individuals at risk at time t.

Note that there is exactly one contribution from each individual at risk to this part of the log-likelihood:

$$\ell_t(\alpha_t, \beta) = \sum_{i\in\mathcal{R}_t}\left\{d_i(\alpha_t + \eta_i) - \mathrm{e}^{\alpha_t+\eta_i}\right\} = \alpha_t + \eta_{\text{death}} - \mathrm{e}^{\alpha_t}\sum_{i\in\mathcal{R}_t}\mathrm{e}^{\eta_i}$$

where η_{death} is the linear predictor for the individual that died at t. For those intervals on the time-scale where no deaths occur the estimate of the α_t will be $-\infty$[1], and so these intervals will not contribute to the log-likelihood.

The derivative w.r.t. α_t is:

$$\mathrm{D}_{\alpha_t}\ell(\alpha_t, \beta) = 1 - \mathrm{e}^{\alpha_t}\sum_{i\in\mathcal{R}_t}\mathrm{e}^{\eta_i} = 0 \quad \Rightarrow \quad \widehat{\mathrm{e}^{\alpha_t}} = \frac{1}{\sum_{i\in\mathcal{R}_t}\mathrm{e}^{\eta_i}}$$

If this estimate of e^{α_t} is fed back into the log-likelihood for α_t, we get the *profile likelihood* (with α_t "profiled out"):

$$\log\left(\frac{1}{\sum_{i\in\mathcal{R}_t}\mathrm{e}^{\eta_i}}\right) + \eta_{\text{death}} - 1 = \log\left(\frac{\mathrm{e}^{\eta_{\text{death}}}}{\sum_{i\in\mathcal{R}_t}\mathrm{e}^{\eta_i}}\right) - 1$$

which is the same as the contribution from time t to Cox's partial likelihood.

Thus we may estimate the regression parameters from the Cox model by standard Poisson-regression software by splitting the data finely and specifying the model as having one rate parameter per time interval.

The Cox model could therefore have been formulated as model with a baseline rate modeled by a timescale parameter for each time recorded. This is an exchangeable model for the baseline rate parameters, thus using neither the ordering nor the absolute scaling of the times.

The results will be the same, also for the s.e.s, since everything is derived . This is illustrated in section 2.1, where fully parametric alternatives to the Cox model is described too.

[1] This is because the term $\alpha_t + \eta_{\text{death}}$ vanishes if all $d_i = 0$, and the last term is maximal if $\mathrm{e}^{\alpha_t} = 0 \Leftrightarrow \alpha_t = -\infty$

1.4 Practical data processing

Implementation of the Poisson-approach in practice requires that follow-up for each individual is split in small pieces of follow-up along one or more time scales. The relevant time-varying covariates should be computed for each interval and fixed covariates should be carried over to all intervals for a given individual.

Presently there are (at least the following) tools for this in:

Stata: The function `stsplit` is part of standard Stata, it is a descendant of `stlexis` written by Michael Hills & David Clayton.

SAS: A macro `%Lexis`, available at http://BendixCarstensen/Lexis, written by Bendix Carstensen. Another macro is by Klaus Rostgaard [5] https://sourceforge.net/p/pyrsstep/wiki/Home/.

R: Function `survSplit` from the `survival` package does the job. The `Epi` package has a function `splitLexis` that does this for `Lexis` objects [4, 1], and in the `popEpi` package there is a faster `data.table` based version, `splitMulti`, which also has a more friendly syntax.

These tools expand a traditional survival dataset with one record per individual to one with several records per individual, one record per follow-up interval. In the following we shall restrict attention to the `Lexis` tools in R.

The split data makes a clear distinction between *risk time* which is the length of each interval and *time scale* which is the value of the timescale at (the beginning of) each interval, be that time since entry, current age, calendar time, etc.

In traditional Poisson modeling the log-risk time is used as offset and the time is used as covariate. Thus Poisson modeling of follow-up data makes a clear distinction between risk time as the response variable and time scale(s) as covariate(s). A recent addition to the `Epi` package is the family `poisreg`[2], which uses the more intuitive specification of the response as a two-column vector of events and person-years.

1.4.1 Estimation of baseline hazard

Once data has been split in little pieces of follow-up time, the effect of any timescale can be estimated using parametric regression tools such as splines. This will directly produce estimated baseline rates by using standard prediction machinery for generalized linear models with a given set of covariates.

Suppose $h(t)$ is a parametric function which is parametrized linearly by the parameters in τ, $h(t) = w'\tau$ (w and τ are column vectors). The model can be formulated as:

$$\log\bigl(\lambda(t,x)\bigr) = h(t) + x'\gamma = w'\tau + x'\gamma = (w\,x)'\begin{pmatrix}\tau\\ \gamma\end{pmatrix}$$

Standard prediction machinery can be used to produce estimates of log-rates with standard errors for a set of values of t (and hence w), and some chosen values of the variables in x. This is a standard tool in any statistical package able to fit generalized linear models. Rate

[2] This means that in a `glm` or `gam` model you can specify `family=poisreg`

estimates with confidence intervals are then derived by taking the exponential function of the estimates for the log-rates with confidence intervals.

In the `Epi` package this is handled by the `ci.pred` function that produces predicted rates for a specified set of prediction points.

1.4.2 Estimation of survival function

In studies where entry time 0 is meaningful the survival function is a simple, albeit non-linear function of the rates:

$$S(t) = \exp\left(-\int_0^t \lambda(s)\ \mathrm{d}s\right)$$

so in order to estimate this from a parametric model for the log-rates we need to derive the integral, i.e. a cumulative sum of predictions. If we want standard errors for this we must have not only standard errors for the λs, but the entire the variance-covariance matrix of estimated values of λ.

From a generalized linear model we can easily extract estimates for $\log\bigl(\lambda(t)\bigr)$ at any set of points. This is just a linear function of the parameters, and so the variance-covariance matrix of these can be computed from the variance-covariance matrix of the parameters.

A Taylor approximation of the variance-covariance matrix for $\lambda(t)$ can be obtained from this by using the derivative of the function that maps $\log\bigl(\lambda(t)\bigr)$ to $\lambda(t)$. This is the coordinate-wise exponential function, so the matrix required is the diagonal matrix with entries $\log\bigl(\lambda(t)\bigr)$.

Finally the cumulative sum is obtained by multiplying with a matrix with 1s on and below the diagonal, so this matrix just needs to be pre and post-multiplied in order to produce the variance-covariance of the cumulative hazard at the prespecified points.

In technical terms we let $\hat{f}(t_i)$ be estimates for the log-rates for a certain set of covariate values (x) at points $t_i, i = 1, \ldots, I$, derived by:

$$\hat{f}(t_i) = \mathbf{B}\,\hat{\beta}$$

where $\beta = (\tau, \gamma)$ is the parameter vector in the model (of length p, say), including the parameters that describe the baseline hazard. Here, $\hat{f}(t_i)$ is $I \times 1$, $\mathbf{B}$ is $I \times p$ and $\hat{\beta}$ is $p \times 1$.

Now, let the estimated variance-covariance matrix of β be Σ, a $p \times p$ matrix. Then the variance-covariance of $\hat{f}(t_i)$ is $\mathbf{B}\Sigma\mathbf{B}'$—which is $I \times I$. The transformation to the rates is the coordinate-wise exponential function so the derivative of this (evaluated at the m.l.e.) is the diagonal matrix with entries $\exp\bigl(\hat{f}(t_i)\bigr)$, so the variance-covariance matrix of the rates at the points t_i is

$$\mathrm{diag}(\mathrm{e}^{\hat{f}(t_i)})\,\mathbf{B}\,\Sigma\,\mathbf{B}'\,\mathrm{diag}(\mathrm{e}^{\hat{f}(t_i)})'$$

Finally, the transformation to the cumulative hazard (assuming that all intervals have length ℓ) is by an $I \times I$ matrix of the form:

$$\mathbf{L} = \ell \times \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

so the (approximate) variance-covariance matrix for the cumulative hazard is:

$$\mathbf{L}\ \mathrm{diag}(\mathrm{e}^{\hat{f}(t_i)})\,\mathbf{B}\,\Sigma\,\mathbf{B}'\ \mathrm{diag}(\mathrm{e}^{\hat{f}(t_i)})'\,\mathbf{L}'$$

This formula for the variance of the cumulative hazard does not guarantee that the derived confidence intervals' lower endpoints are larger than 0. This can be fixed by computing confidence intervals for the log-cumulative hazard using the delta-rule, and back-transforming to the rate scale.

These calculations are implemented in the `Epi` package function `ci.cum`, which requires (at least) 3 objects as arguments: 1) a model object representing a multiplicative model for occurrence rates, 2) a prediction data frame which will produce (log) rate-estimates from the model at a set of equidistant times since some origin, and 3) a scalar representing the distance between the prediction times (in the units in which the person-years was supplied to the model). The function also has a facility for computing the confidence limits on the log-cumulative hazard scale and back transforming to ensure positive lower confidence bounds for the integrated hazard.

Once we have estimated the cumulative hazard function as a function of time we can transform it to the survival function by the exponential—this is implemented in the function `ci.surv`.

Chapter 2

Examples

2.1 Equality of Cox and Poisson modeling: The lung cancer example

In this section we use the lung cancer example data from the `survival` package to illustrate that the results from a Cox model are identical to results from a particular Poisson model—albeit quite an absurd one. We also illustrate two ways to use a parametrically smoothed version of the linear predictor in the Poisson model to obtain a sane estimate for the baseline hazard.

```
> library(Epi)
> library(popEpi)
> library(survival)
> library(mgcv)
> data(lung)
> str(lung)
'data.frame':        228 obs. of  10 variables:
 $ inst     : num  3 3 3 5 1 12 7 11 1 7 ...
 $ time     : num  306 455 1010 210 883 ...
 $ status   : num  2 2 1 2 2 1 2 2 2 2 ...
 $ age      : num  74 68 56 57 60 74 68 71 53 61 ...
 $ sex      : num  1 1 1 1 1 1 2 2 1 1 ...
 $ ph.ecog  : num  1 0 0 1 0 1 2 2 1 2 ...
 $ ph.karno : num  90 90 90 90 100 50 70 60 70 70 ...
 $ pat.karno: num  100 90 90 60 90 80 60 80 80 70 ...
 $ meal.cal : num  1175 1225 NA 1150 NA ...
 $ wt.loss  : num  NA 15 15 11 0 0 10 1 16 34 ...
```

```
> lung[1:10, ]
```

	inst	time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss
1	3	306	2	74	1	1	90	100	1175	NA
2	3	455	2	68	1	0	90	90	1225	15
3	3	1010	1	56	1	0	90	90	NA	15
4	5	210	2	57	1	1	90	60	1150	11
5	1	883	2	60	1	0	100	90	NA	0
6	12	1022	1	74	1	1	50	80	513	0
7	7	310	2	68	2	2	70	60	384	10
8	11	361	2	71	2	2	60	80	538	1
9	1	218	2	53	1	1	70	80	825	16
10	7	166	2	61	1	2	70	70	271	34

How many deaths are there?

```
> table(lung$status)
  1   2
 63 165
```

Convert sex to a factor:

```
> lung$sex <- factor(lung$sex, labels = c("M", "F"))
```

How many distinct event times?

```
> addmargins(table(table(lung$time)))
  1   2   3 Sum
146  38   2 186
```

To avoid tied event times we add a small random quantity to each time:

```
> set.seed(1952)
> lung$time <- lung$time + runif(lung$time, -2, 2)
> addmargins(table(table(lung$time)))
  1 Sum
228 228
```

First we fit a traditional Cox-model for the Mayo clinic data

```
> m0.cox <- coxph(Surv(time, status == 2) ~ age + sex, data = lung)
> summary(m0.cox)
Call:
coxph(formula = Surv(time, status == 2) ~ age + sex, data = lung)

  n= 228, number of events= 165

          coef exp(coef)  se(coef)      z Pr(>|z|)
age   0.016985  1.017130  0.009222  1.842  0.06550 .
sexF -0.518435  0.595452  0.167496 -3.095  0.00197 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     exp(coef) exp(-coef) lower .95 upper .95
age     1.0171     0.9832    0.9989    1.0357
sexF    0.5955     1.6794    0.4288    0.8268

Concordance= 0.603  (se = 0.025 )
Likelihood ratio test= 14.3  on 2 df,   p=8e-04
Wald test            = 13.64  on 2 df,   p=0.001
Score (logrank) test = 13.9  on 2 df,   p=0.001
```

Now create a `Lexis` object from the dataset `lung`, representing the follow-up time and events:

```
> Lung <- Lexis(exit = list(tfe = time),
+        exit.status = factor(status,
+                             labels = c("Alive", "Dead")),
+               data = lung)
NOTE: entry.status has been set to "Alive" for all.
NOTE: entry is assumed to be 0 on the tfe timescale.
```

```
> summary(Lung)
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive    63  165       228      165   69618.73       228
```

Split data in small intervals, defined by all recorded events and censoring times

```
> Lung.s <- splitMulti(Lung, tfe = c(0, sort(unique(Lung$time))))
> summary(Lung.s)
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive 25941  165     26106      165   69618.73       228
> Lung.s[1:10, 1:10]
 lex.id   tfe lex.dur lex.Cst lex.Xst inst    time status age sex
      1  0.00    6.78   Alive   Alive    3 307.804      2  74   M
      1  6.78    3.25   Alive   Alive    3 307.804      2  74   M
      1 10.04    0.15   Alive   Alive    3 307.804      2  74   M
      1 10.19    0.38   Alive   Alive    3 307.804      2  74   M
      1 10.57    1.83   Alive   Alive    3 307.804      2  74   M
      1 12.40    0.63   Alive   Alive    3 307.804      2  74   M
      1 13.03    1.62   Alive   Alive    3 307.804      2  74   M
      1 14.65    1.01   Alive   Alive    3 307.804      2  74   M
      1 15.66    8.94   Alive   Alive    3 307.804      2  74   M
      1 24.59    5.02   Alive   Alive    3 307.804      2  74   M
```

Now fit the Cox model to the `Lexis` data set as well as the time-split `Lexis` data set; note the code is exactly the same, only the `data=` argument differs:

```
>  mL.cox <- coxph(Surv(tfe, tfe+lex.dur, lex.Xst == "Dead") ~ age + sex,
+                  eps = 10^-11, iter.max = 25, data = Lung)
> mLs.cox <- coxph(Surv(tfe, tfe+lex.dur, lex.Xst == "Dead") ~ age + sex,
+                  eps = 10^-11, iter.max = 25, data = Lung.s)
> round(rbind(ci.exp( m0.cox),
+             ci.exp( mL.cox),
+             ci.exp(mLs.cox))[c(1,3,5,2,4,6),], 6)
     exp(Est.)     2.5%    97.5%
age   1.017130 0.998911 1.035681
age   1.017130 0.998911 1.035681
age   1.017130 0.998911 1.035681
sexF  0.595452 0.428818 0.826837
sexF  0.595452 0.428818 0.826837
sexF  0.595452 0.428818 0.826837
```

We see we get the same results from the three different sets of data — they contain exactly the same amount of information about the rates.

Now we fit the corresponding Poisson model with factor modeling of the time scale — note that we use the `poisreg` family where we enter events and person-years as a 2 column matrix:

```
> nlevels(factor(Lung.s$tfe))
[1] 228
```

```
> system.time(
+ mLs.pois.fc <- glm(cbind(lex.Xst == "Dead", lex.dur) ~ 0 + factor(tfe) + age + sex,
+                     family = poisreg, data = Lung.s))
   user  system elapsed
  15.39    0.47   15.86
> length(coef(mLs.pois.fc))
[1] 230
> rbind(ci.exp(mLs.cox),
+       ci.exp(mLs.pois.fc, subset = c("age", "sex")))[c(1,3,2,4),]
     exp(Est.)      2.5%     97.5%
age  1.0171302 0.9989114 1.0356813
age  1.0171302 0.9989114 1.0356813
sexF 0.5954519 0.4288185 0.8268369
sexF 0.5954519 0.4288185 0.8268369
```

In accordance with the mathematical derivations above, we see that the estimates of the regression coefficients are exactly the same from the Cox model and the Poisson model. The latter has an extra 228 parameters estimated, which is what causes the very long estimation time.

2.1.1 Parametric baseline

To get a more realistic model for the baseline rate we now define knots for a spline basis and fit the model with natural splines for the baseline effect of `tfe`. These knots are basically taken out of thin air:

```
> t.kn <- c(0, 25, 100, 500, 1000)
> system.time(
+ mLs.pois.sp <- glm(cbind(lex.Xst == "Dead", lex.dur)
+                     ~ Ns(tfe, knots = t.kn) + age + sex,
+                   family = poisreg,
+                     data = Lung.s))
   user  system elapsed
   0.09    0.04    0.12
```

Finally we fit the model with a penalized spline model for the effect of `tfe` using `gam` from the `mgcv` package:

```
> system.time(
+ mLs.pois.ps <- gam(cbind(lex.Xst == "Dead", lex.dur)
+                     ~ s(tfe) + age + sex,
+                     family = poisreg,
+                       data = Lung.s))
   user  system elapsed
   0.66    0.39    1.05
> summary(mLs.pois.ps)

Family: poisson
Link function: log

Formula:
cbind(lex.Xst == "Dead", lex.dur) ~ s(tfe) + age + sex
```

```
Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -6.94486    0.59468 -11.678  < 2e-16 ***
age          0.01629    0.00920   1.771  0.07663 .
sexF        -0.50706    0.16731  -3.031  0.00244 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
         edf Ref.df Chi.sq  p-value
s(tfe) 2.131  2.687  17.74 0.000649 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  9.74e-06   Deviance explained =  1.7%
UBRE = -0.93042  Scale est. = 1         n = 26106
```

We see that the effective d.f. for the time scale effect (`tfe`) is about 2, so some indication that the arbitrary spline may be over-modeling data.

Finally we make an overall comparison of estimates of age and sex effects from the different approaches:

```
> ests <-
+ rbind(ci.exp(m0.cox),
+       ci.exp(mLs.cox),
+       ci.exp(mLs.pois.fc, subset = c("age", "sex")),
+       ci.exp(mLs.pois.sp, subset = c("age", "sex")),
+       ci.exp(mLs.pois.ps, subset = c("age", "sex")))
> cmp <- cbind(ests[c(1, 3, 5, 7, 9)  , ],
+              ests[c(1, 3, 5, 7, 9)+1, ])
> rownames(cmp) <-
+  c("Cox", "Cox-split", "Poisson-factor", "Poisson-spline", "Poisson-Pspline")
> colnames(cmp)[c(1, 4)] <- c("age", "sex")
> round(cmp, 7)
                     age      2.5%    97.5%       sex      2.5%     97.5%
Cox             1.017130 0.9989114 1.035681 0.5954519 0.4288185 0.8268369
Cox-split       1.017130 0.9989114 1.035681 0.5954519 0.4288185 0.8268369
Poisson-factor  1.017130 0.9989114 1.035681 0.5954519 0.4288185 0.8268369
Poisson-spline  1.016214 0.9980575 1.034701 0.5994793 0.4317352 0.8323977
Poisson-Pspline 1.016423 0.9982591 1.034917 0.6022668 0.4338818 0.8360001
```

So even if the factor model, and by that token also the Cox-model, seem pretty far fetched in their (lack of) assumptions, there is minimal difference to the regression parameter estimates from the models with more realistic assumptions for the baseline rates.

2.1.2 Rates, cumulative rates and survival

Parametric models

Now we compute the estimated rates and cumulative rates over 10-day periods for 60 year old men, and then the survival function at these points.

In order to get the predictions from the spline model we specify a prediction data frame, where we predict rates at equidistant points, using `ci.pred` for the rates. Note that since we used the `poisreg` family, the predicted rates are by definition per one unit of `lex.dur`, which in our case is days, so we multiply by 365.25 to get rates per 1 PY.

Further, when we compute the survival function (and by that token also the cumulative rates), we must supply the interval length (distance between values of `tfe` in the prediction data frame) by `intl=` in the units of `lex.dur`. If we do not supply it, it will be taken as the difference between the two first elements of the first column in the prediction data frame:

```
> # midpoints of 10-day intervals and other covariates
> nd <- data.frame(tfe = seq(5, 995, 10), age = 60, sex = "M")
> #
> # the rates from the spline model (events/year)
> lambda <- ci.pred(mLs.pois.sp, nd) * 365.25
> # the survival function
> survP  <- ci.surv(mLs.pois.sp, nd)
NOTE: interval length chosen from  as tfe[2] - tfe[1]
> # same same for the penalized spline model:
> lambdap <- ci.pred(mLs.pois.ps, nd) * 365.25
> survPp  <- ci.surv(mLs.pois.ps, nd)
NOTE: interval length chosen from  as tfe[2] - tfe[1]
```

So now we have the incidence rates per 1 PY as well as cumulative incidence rates and the corresponding survival function(s) based both on natural splines and a penalized likelihood via `gam`.

Cox models

The Breslow-estimator of the survival curve from the corresponding Cox-model for a male aged 60 is obtained from the `m0.cox` object:

```
> sf <- survfit(m0.cox, newdata = data.frame(sex = "M", age = 60))
```

We can extract the baseline rates from the corresponding Poisson model as well. Since `lex.dur` is supplied in units of days to `mLs.pois.fc`, the predicted rates from using `ci.exp` on the model will be in events per day, hence we rescale to events per year. We extract the times from the names of the parameters:

```
> (nc <- length(coef(mLs.pois.fc)))
[1] 230
> br <- ci.exp(mLs.pois.fc,
+              ctr.mat = cbind(diag(nc - 2), 60, 0)) * 365.25
> bt <- as.numeric(gsub("factor\\(tfe)", "", names(coef(mLs.pois.fc))[1:(nc - 2)]))
> head(cbind(bt, br))
           bt  exp(Est.)       2.5%     97.5%
[1,]  0.00000  0.2653605 0.03724572  1.890585
[2,]  6.78194  0.5547910 0.07787287  3.952507
[3,] 10.03619 11.8535350 1.66384425 84.446782
[4,] 10.18946  4.8591849 0.68210629 34.615834
[5,] 10.56603  1.0050413 0.14108059  7.159794
[6,] 12.39700  2.9190141 0.40975739 20.794362
```

Now we have the predicted rates in intervals between the times observed; the Poisson model fitted implicitly assumes that event rates are constant within intervals between times. Since the deaths occur at the *end* of the intervals, and intervals are named by their *left* endpoint, plotting of the rates must use `type = "s"`, which creates steps between successive points where the curve *first* moves horizontally, then vertically.

Natural spline vs. penalized splines

First we just compare the two smooth curves:

```
> par(mfrow = c(1, 2), mar = c(3, 3, 1, 1), mgp = c(3, 1, 0)/1.6,
+     bty = "n", las = 1, lend = "butt")
> matshade(nd$tfe, cbind(lambda, lambdap), plot = TRUE,
+          col = c("blue", "red"), lwd = 3, lty = c("solid", "21"),
+          xlim = c(0, 900), xaxs = "i", ylim = c(1/5, 20), log = "y",
+          xlab = "Days since diagnosis",
+          ylab = "Mortality rate per 1 year")
> matshade(nd$tfe-5, cbind(survP[, -4], survPp[, -4]), plot = TRUE,
+          col = c("blue", "red"), lwd = 3, , lty = c("solid", "21"),
+          xlim = c(0, 900), xaxs = "i", yaxs = "i", ylim = 0:1,
+          xlab = "Days since diagnosis",
+          ylab = "Survival probability")
```

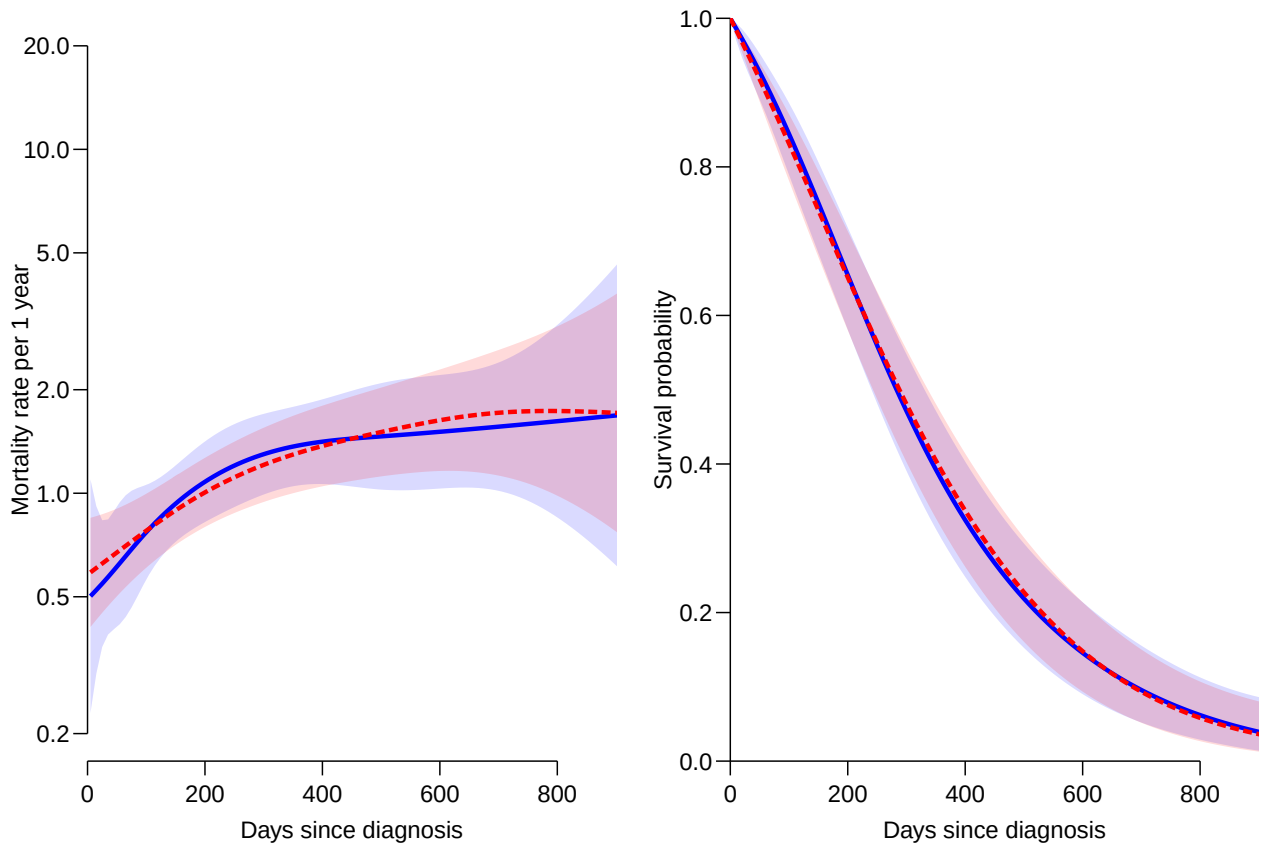

Figure 2.1: *Left panel: Estimated mortality rates by Poisson models; blue is the natural spline with pre-chosen knots, the red is the* `gam` *model with penalization. Right panel: The resulting survival curves. Shaded areas indicate 95% confidence intervals.* `./lung-rtsurv-sm`

Comparison with Cox

The we make the same plots overlaid with the results from the Cox model:

```
> par(mfrow = c(1, 2), mar = c(3, 3, 1, 1), mgp = c(3, 1, 0)/1.6,
+     bty = "n", las = 1, lend = "butt")
> plot(NA, xlim = c(0, 900), xaxs = "i", ylim = c(1/5, 20), log = "y",
+          xlab = "Days since diagnosis",
+          ylab = "Mortality rate per 1 year")
> lines(bt, br[, 1], type = "s", col = gray(0.6))
> matshade(nd$tfe, cbind(lambda, lambdap), # plot = TRUE,
+          col = c("blue", "red"), lwd = 3, lty = c("solid", "21"))
> matshade(nd$tfe, cbind(survP[, -4], survPp[, -4]), plot = TRUE,
+          col = c("blue", "red"), lwd = 3, , lty = c("solid", "21"),
+          xlim = c(0, 900), xaxs = "i", yaxs = "i", ylim = 0:1,
+          xlab = "Days since diagnosis",
+          ylab = "Survival probability")
> lines(sf, lwd = 1, lty = c(1, 1))
> lines(sf, lwd = 2, conf.int = FALSE)
```

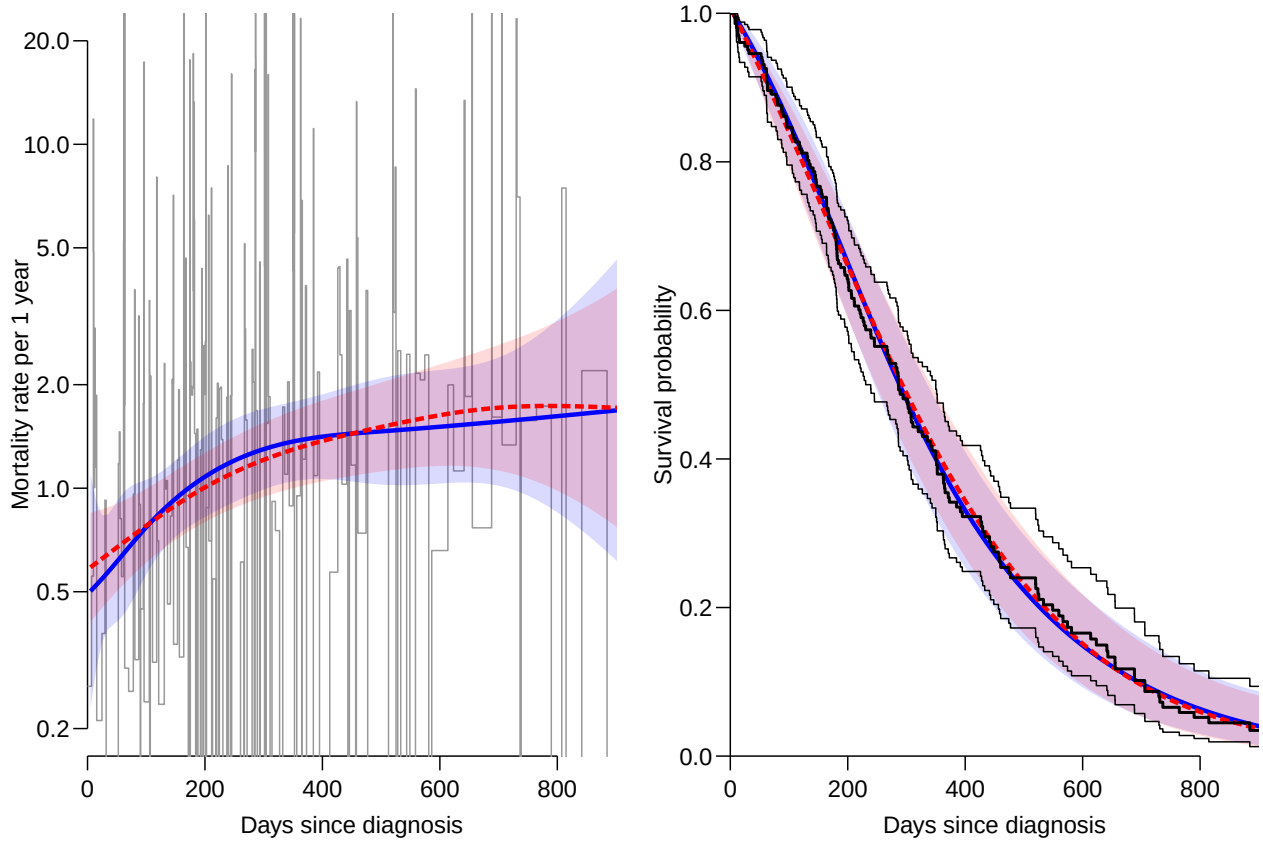

Figure 2.2: *Left panel: Estimated mortality rates by Poisson models; blue is the natural spline with pre-chosen knots, the red is the* `gam` *model with penalization, and the thin gray line indicate the estimated baseline hazard from the Poisson (Cox) model with one parameter per event/censoring time. Right panel: The resulting survival curves, over-laid in black with the Breslow-estimator of the survival curve. Shaded areas and thin lines indicate 95% confidence intervals.* `./lung-rtsurv-cp`

From figure 2.1 we see that there is only slight difference between the two parametric approaches; the penalized splines (red broken curve) smooths a bit more hat the natural splines with arbitrarily chosen knots. When transformed to the survival scale, the two approaches are practically indistinguishable.

Figure 2.1 has the Cox-model estimates overlaid; strictly speaking the baseline hazard is not really part of the Cox-model, the underlying hazard comes from the corresponding Poisson model, the survival curve is the Breslow estimator. There is a faint indication that the parametric curves produces slightly narrower confidence bands for the survival probabilities than the Breslow-estimator.

2.1.3 Practical splitting

In practical applications the splitting of time need not be at the times of events and censorings; this was only done above to demonstrate the connection between the Cox model and the Poisson model.

The assumption behind the Poisson approach is essentially only the assumption that a model with constant rates in each small interval gives an adequate description of data. So in practice we would split data in small equidistant intervals. In the lung cancer dataset there are 165 deaths and the total observation period is some 1000 days, some 2.8 years, so we split the follow-up in intervals of 20 days:

```
> sL <- splitMulti(Lung, tfe = seq(0, 1100, 20))
> summary(Lung)
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive    63  165       228      165   69618.73       228
> summary(sL)
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  3443  165      3608      165   69618.73       228
```

so we have much fewer records but the same number of events and person-time.

We can then compare to the estimates from the parametric model `mLs.pois.sp`, if we instead use the equidistantly cut dataset:

```
> mLs.pois.se <- update(mLs.pois.sp, data = sL)
> ee <- cbind(ci.exp(mLs.pois.sp),
+             ci.exp(mLs.pois.se),
+             ci.exp(mLs.pois.sp)/
+             ci.exp(mLs.pois.se))
> colnames(ee)[c(1,4,7)] <- "expEst"
> round(ee, 3)
                       expEst  2.5%  97.5% expEst  2.5%  97.5% expEst  2.5% 97.5%
(Intercept)             0.001 0.000  0.002  0.001 0.000  0.002  0.869 0.774 0.976
Ns(tfe, knots = t.kn)1  2.773 1.037  7.415  2.639 1.103  6.312  1.051 0.940 1.175
Ns(tfe, knots = t.kn)2  2.817 0.902  8.797  2.454 0.880  6.841  1.148 1.025 1.286
Ns(tfe, knots = t.kn)3  3.836 0.536 27.479  2.894 0.586 14.296  1.325 0.914 1.922
Ns(tfe, knots = t.kn)4  3.253 0.705 15.010  3.329 0.792 13.999  0.977 0.891 1.072
age                     1.016 0.998  1.035  1.016 0.998  1.035  1.000 1.000 1.000
sexF                    0.599 0.432  0.832  0.599 0.432  0.832  1.000 1.000 1.000
```

We see only minor differences in the estimated values of the regression parameters, while it appears that the spline parameters are somewhat different. This does however not translate any relevant differences in the estimated curves:

```
> plot(NA, xlim = c(0, 900), xaxs = "i", ylim = c(1/5, 20), log = "y",
+          xlab = "Days since diagnosis",
+          ylab = "Mortality rate per 1 year")
> matshade(nd$tfe, cbind(ci.pred(mLs.pois.sp, nd),
+                        ci.pred(mLs.pois.se, nd)) * 365.25,
+          lwd = 2, col = c('blue', 'black'), log = "y" )
```

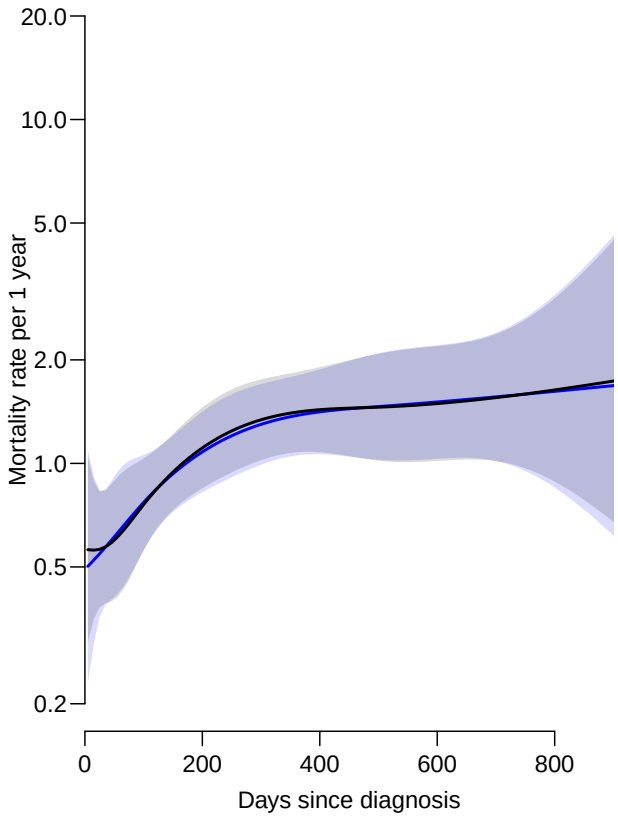

Figure 2.3: *Comparing the same model fitted to data split at all 228 recorded event and censoring times (26,106 records) (blue), and fitted to a data set only cut every 20 days (3,591 records) (black).* `./lung-spcmp`

Thus from figure 2.3 it appears that the splitting of the follow-up time in 20-day intervals is sufficient to render the estimation of the baseline hazard reliable.

2.2 Stratified models

A stratified Cox-model is a model where the underlying hazard is allowed to differ between strata, i.e. between levels of a categorical variable.

Thus this is merely an interaction between time and a categorical variable. If a spline basis has been chosen as model for the time variable a model with separate baseline hazards for each level of a factor F this is easily modelled by saying:

For illustration we use the lung cancer example again, the `Lexis` object where time is split every 20 days:

```
NOTE: entry.status has been set to "Alive" for all.
NOTE: entry is assumed to be 0 on the tfe timescale.


> summary(sL)
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  3432  165      3597      165      69593       228
```

For the modeling of the baseline rate (timescale `tfe`) we define the knots and fit a natural spline, one with main effect of sex, the other with an interaction. Note that there is no requirement that the time-part of the interaction is parametrized in the same way as the main effect. The model `m3` below uses a simpler time-effect in the interaction:

```
> kn <- c(0, 50, 150, 450)
> m1 <- glm(cbind(lex.Xst=="Dead", lex.dur)
+           ~ Ns(tfe, knots = kn) + sex + age,
+           family = poisreg,
+             data = sL)
> m2 <- glm(cbind(lex.Xst=="Dead", lex.dur)
+           ~ Ns(tfe, knots=kn) * sex + age,
+           family = poisreg,
+             data = sL )
> m3 <- update(m1, . ~ . + Ns(tfe, knots = c(0, 50, 200)):sex )
> anova(m3, m1, m2, test = "Chisq")
Analysis of Deviance Table

Model 1: cbind(lex.Xst == "Dead", lex.dur) ~ Ns(tfe, knots = kn) + sex +
    age + sex:Ns(tfe, knots = c(0, 50, 200))
Model 2: cbind(lex.Xst == "Dead", lex.dur) ~ Ns(tfe, knots = kn) + sex +
    age
Model 3: cbind(lex.Xst == "Dead", lex.dur) ~ Ns(tfe, knots = kn) * sex +
    age
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1      3588     1293.2
2      3591     1298.6 -3  -5.3226   0.1496
3      3588     1294.2  3   4.3292   0.2280
```

There is no significant interaction here, but the test statistic would actually have been significant if the interaction were only on two degrees of freedom. Hence we may want to inspect the two fitted baseline rates, as well as their ratio:

```
> par(mfrow = c(1, 2), mar = c(3, 3, 1, 3), mgp = c(3, 1, 0), las = 1)
> nm <- data.frame(tfe = seq(0, 1000, 10),
+                  age = 65,
+                  sex = factor("M", levels = c("M", "F")))
> nf <- data.frame(tfe = seq(0, 1000, 10),
+                  age = 65,
+                  sex = factor("F", levels = c("M", "F")))
> plot(NA, xlim = c(0,900), xaxs = "i", ylim = c(1/100, 5), log = "y",
+          xlab = "Days since diagnosis",
+          ylab = "Mortality rate per 1 year" )
> matshade(nm$tfe, cbind(ci.pred(m2, nm) * 365.25,
+                        ci.pred(m2, nf) * 365.25,
+                        ci.exp (m2, list(nm,nf)) / 20),
```

```
+          lwd = 2, col = c('blue', 'red', 'black'))
> abline(h = 1/20, lty = 3)
> axis(side = 4, at = c(2, 5, 10, 15, 20) / 200, labels = c(2, 5, 10, 15, 20) / 10 )
> axis(side = 4, at = c(2:9)/200, labels = NA, tcl = -0.3 )
> plot(NA, xlim = c(0,900), xaxs = "i", ylim = c(1/100,5), log = "y",
+          xlab = "Days since diagnosis",
+          ylab = "Mortality rate per 1 year" )
> matshade(nm$tfe, cbind( ci.pred(m3,nm)*365.25,
+                         ci.pred(m3,nf)*365.25,
+                         ci.exp (m3,list(nm,nf))/20 ),
+          lwd = 2, col = c('blue','red','black') )
> abline(h = 1 / 20, lty = 3)
> axis(side = 4, at = c(2, 5, 10, 15, 20) / 200,
+            labels = c(2, 5, 10, 15, 20) / 10)
> axis(side = 4, at = c(2:9) / 200, labels = NA, tcl = -0.3)
```

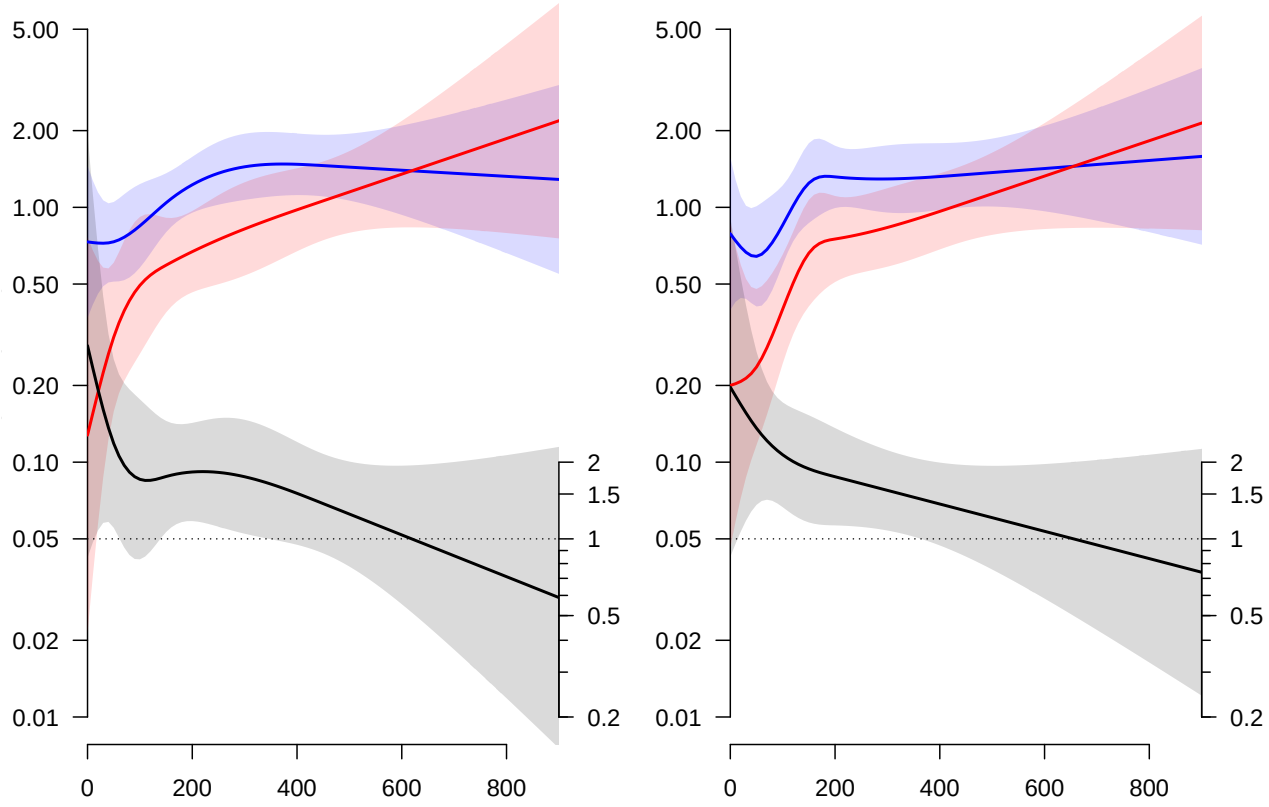

Figure 2.4: *Baseline rates for 65 year old men (blue) resp. women (red), and the rate-ratio between these (black). The leftmost panel uses the same set of knots for the main effect and the interaction, the rightmost a more parsimonious interaction specification.* `./strat-prcmp`

From figure 2.4, it is clear that although there is no formal interaction (p-values are 15-20%), there is a clear tendency that the mortality among men is higher during the first year or so after diagnosis.

For illustration we repeat the same exercise with the `gam` machinery. Note that the interaction specification `s(tfe,by = sex)` does not contain the main effect of sex, so this must be specified:

```
> p1 <- gam(cbind(lex.Xst == "Dead", lex.dur) ~ s(tfe)           + sex + age,
+           family = poisreg,
+             data = sL)
> p2 <- gam(cbind(lex.Xst == "Dead", lex.dur) ~ s(tfe, by = sex) + sex + age,
+           family = poisreg,
+             data = sL)
> anova(p1, p2, test = "Chisq")
Analysis of Deviance Table

Model 1: cbind(lex.Xst == "Dead", lex.dur) ~ s(tfe) + sex + age
Model 2: cbind(lex.Xst == "Dead", lex.dur) ~ s(tfe, by = sex) + sex +
    age
  Resid. Df Resid. Dev      Df Deviance Pr(>Chi)
1    3591.4     1299.5
2    3590.9     1298.0 0.52822   1.4688   0.1092
> par(mar = c(3,3,1,3), mgp = c(3,1,0), las = 1 )
> plot(NA, xlim = c(0,900), xaxs = "i", ylim = c(1/100,5), log = "y",
+           xlab = "Days since diagnosis",
+           ylab = "Mortality rate per 1 year" )
> matshade(nm$tfe, cbind(ci.pred(p2, nm) * 365.25,
+                        ci.pred(p2, nf) * 365.25,
+                        ci.exp (p2, ctr.mat = list(nm, nf)) / 20 ),
+          lwd = 2, col = c('blue', 'red', 'black') )
> abline(h = 1 / 20, lty = 3)
> axis(side = 4, at = c(2, 5, 10, 15, 20) / 200,
+            labels = c(2, 5, 10, 15, 20) / 10)
> axis(side = 4, at = c(2:9) / 200, labels = NA, tcl = -0.3)
```

From figure 2.5 we see the same overall tendency, but substantially more smoothed. But with this type of analysis we have a more firm evidence that male mortality actually is higher in the first year or so.

This is one of the main advantages of a fully parametric approach to modeling of rates: It is possible to show how the different baseline rates from a stratified model looks. The baseline rates are not readily available from a Cox model.

2.3 Time-varying coefficients

When it is suspected that effects of a given variable is not constant, one may allow the coefficient of a variable to vary by time:

$$\lambda_i(t) = \lambda_0(t) \exp(\beta(t) x_i + \cdots)$$

When we think of time as a covariate, this corresponds to an interaction between the covariate and time, which is restricted by letting the x-effect be linear for any fixed value of time.

The substantial reason for this particular choice of this particular form of interaction is slightly opaque. Given that one variable (time) is meticulously modelled it seems strange to insist on a conditionally linear effect of x. It seems to be more productive to explore more parsimonious parametrizations of interactions that were more directly addressing biologically meaningful deviations from the log-linear additivity of the effects.

There is however a tradition in epidemiological analysis of trends in rates to summarize calendar time trends separately in each age-group by computing the average trend within

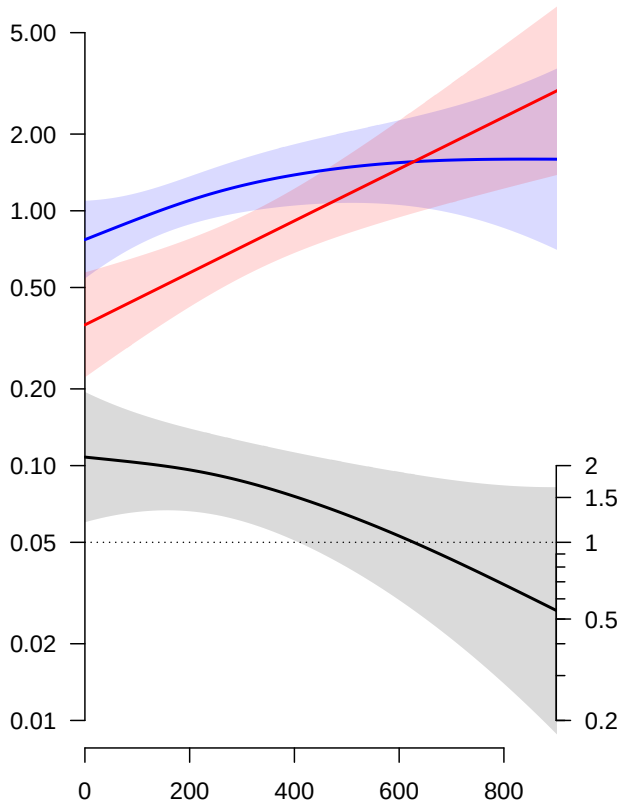

Figure 2.5: *Estimated rates and rate-ratio by the* `gam` *fitting machinery.* `./strat-pnsh`

each age class. The continuous time version of this is precisely a varying coefficients model where the effect of calendar time is taken as linear at each age.

The simplest parametric form of such an interaction is to have separate effects of x, say, at different times. This would correspond to adding an interaction between x and some grouping of time. Again this approach can be taken *ad absurdum* with increasingly fine groupings of time until we end up with the Cox-model formulation of the problem.

But when the main effect of time is modelled by a spline or any other smooth function, implemented as columns of the model matrix in the Poisson regression model, we can estimate time-varying coefficients by adding the same columns multiplied by x to the model matrix. The coefficients of these will then be the ones that determine the (time-varying) effect of the covariate x.

We use the same dataset as before, but now we have the interaction with the quantitative variable `age`.

```
> summary(sL)
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  3432  165      3597      165      69593       228
> kn <- c(0, 50, 150, 450)
> m1 <- glm(cbind(lex.Xst == "Dead", lex.dur)
+           ~ Ns(tfe, knots=kn) + age + sex,
+           family = poisreg,
+             data = sL)
```

```
> mv <- glm(cbind(lex.Xst == "Dead", lex.dur)
+           ~ Ns(tfe, knots=kn) + sex + Ns(tfe, knots=kn, i=T):age,
+           family = poisreg,
+             data = sL)
> anova(m1, mv, test = "Chisq")
Analysis of Deviance Table

Model 1: cbind(lex.Xst == "Dead", lex.dur) ~ Ns(tfe, knots = kn) + age +
    sex
Model 2: cbind(lex.Xst == "Dead", lex.dur) ~ Ns(tfe, knots = kn) + sex +
    Ns(tfe, knots = kn, i = T):age
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      3591     1298.6
2      3588     1280.3  3   18.245 0.0003914 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here we see that there actually is a massive interaction — the age-effects *does* vary considerably by time.

The parameters of interest are those from the second `Ns` term in the model, but of course taken out as a curve. But it is also possible to extract the age-effect as a difference between two predictions, namely the rate-ratio between two persons, say 5 years apart in age, computed at range of times (`tfe`):

```
> nx <- data.frame(tfe = seq(0, 1000, 10), age = 70, sex = "M")
> nr <- data.frame(tfe = seq(0, 1000, 10), age = 65, sex = "M")
> aRR <- ci.exp(mv, list(nx, nr)) # computes the ratio of predictions between nx and nr
> matshade(nx$tfe / 365.25 * 12, aRR, plot = TRUE,
+          lwd = 3,
+          log = "y", xlab = "Time since diagnosis (months)",
+          ylab = "RR per 5 years of age at diagnosis")
> abline( h=1 )
```

From the figure 2.6 we see that the age at diagnosis matters a lot for the mortality the first few months after diagnosis, but after about 3 months there is no effect.

Parametrizations

Note that when we use prediction data frames to tease out the effects, the particular parametrization does not matter, so we could have used a simple expression for the r.h.s. of the model formula:

```
> ~ Ns( tfe, knots=kn ) * age + sex
```

and we would have obtained the same results.

Simplifying code

Since we are using a `Lexis` object as data base for the analysis, we already have specified the time-structure in data, so we can shorten the code even further using the `glm.Lexis` function for fitting data. This function is really is designed to simplify analysis of rates in multistate models; basically it is just a wrapper using the `poisreg` family. It will by default analyze all transition to any absorbing state, which in this case is "Dead", so the analysis could be done by:

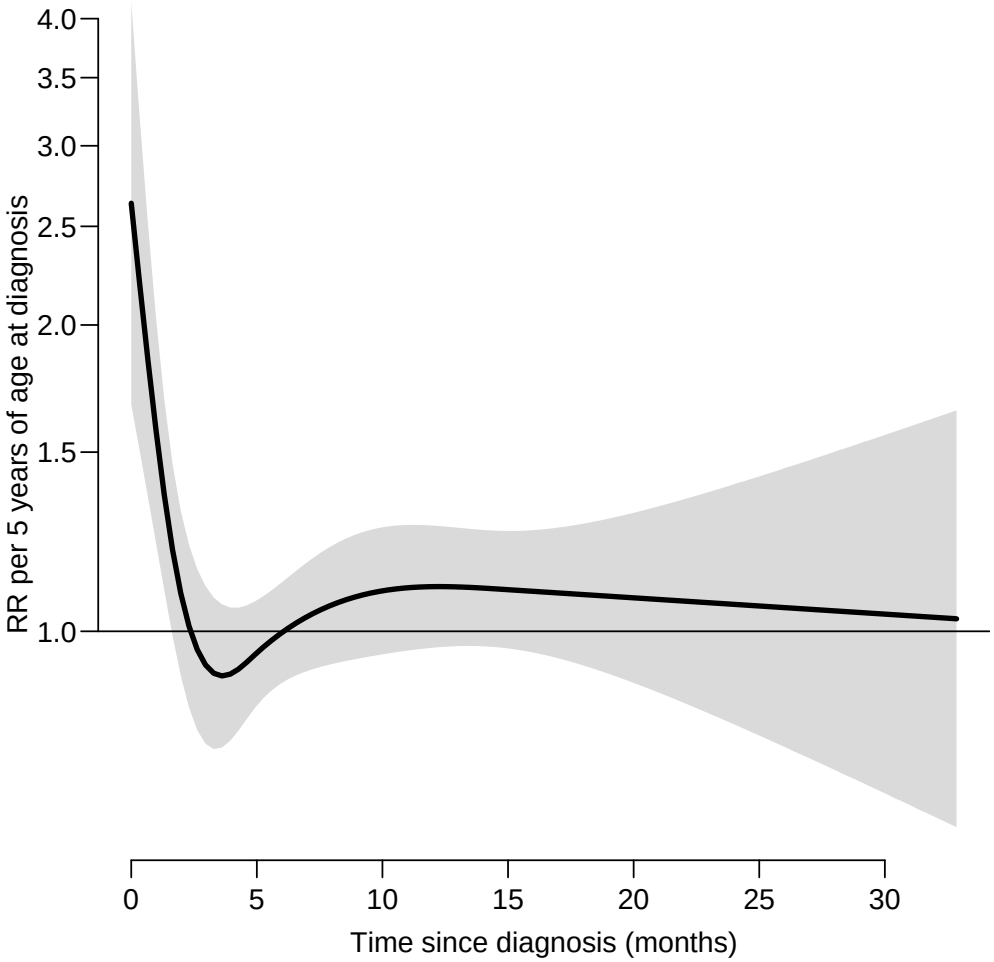

Figure 2.6: *RR of death for the lung cancer patients, per 5 years of age at diagnosis. Results from a "varying-coefficients" model — interaction between two continuous variables, where the effect of age is constrained to be linear at any time since diagnosis.* `./time-var-Aeff`

```
> mL <- glm.Lexis(sL, formula = ~ Ns(tfe, knots = kn) * age + sex)
stats::glm Poisson analysis of Lexis object sL with log link:
Rates for the transition:
Alive->Dead
> c(deviance(mv), deviance(mL))
[1] 1280.321 1280.321
```

You want to look up the help page:

```
> ?glm.Lexis
```

Chapter 3

So who do need the Cox-model?

Since everything which is possible using the Cox-model can be done using the Poisson modeling of split data, there is no loss only substantial gain of capability by switching to Poisson modeling.

The Cox-model is computationally vastly more efficient, and it is easier to produce a survival curve by standard software, which is relevant in most clinical studies. A drawback is the overly detailed modeling of survival curves that may lead to over-interpretation of little humps and notches on an estimated curve.

When stratification or time-dependent variables are involved, the facilities in the standard Cox-analysis programs limits the ways in which the desired interactions can be modeled, and moreover distracts the user from realizing that other interactions between covariates may be of interest.

Thus it seems that the Cox model is useful in the following cases:

- Clinical follow-up studies with only one relevant timescale and the focus on the effect of other covariates than time.
- Studies analyzed on computing equipment pre-1985.

In other settings it seems preferable to split time and use the Poisson approach, because:

- it clarifies the distinction between events and (risk) time as response variables and time(scales) as covariates — reflected in the `poisreg` family in the `Epi` package.
- it enables smoothing of the effect of timescales using standard regression tools. In particular it allows more credible estimates of survival functions in the simple case with only time since entry as timescale.
- it enables sensible modeling of interactions between timescales and other variables (and between timescales), using standard regression tools.

Moreover, as the necessary computing power and software is available, the computational problems encountered previously are now non-existent. Extraction of the relevant functions of model parameters has been facilitated by the introduction of the possibility of supplying pairs of prediction data frames to extract rate-ratios (see the help page for `ci.lin` in the `Epi` package).

That said, the user-interface to the Poisson modeling is slightly more complex than that offered by standard packages for the Cox-model. This is because the Poisson approach requires an explicit specification of a model for the effect of the timescale.

Chapter 4

Simple code for parametric model

The code for this chapter is here: http://bendixcarstensen.com/simple.R

4.1 Cookbook for a parametric survival curve

We use the `lung` dataset for illustration; categorical variables should always be declared as factors:

```
> lung$sex <- factor(lung$sex, labels = c("M", "W"))
```

First declare the follow-up as a `Lexis` data frame:

```
> Lx <- Lexis(exit = list(tfe = time),
+      exit.status = factor(status,
+                           labels = c("Alive", "Dead")),
+             data = lung)
NOTE: entry.status has been set to "Alive" for all.
NOTE: entry is assumed to be 0 on the tfe timescale.
```

```
> summary(Lx)
```

```
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive    63  165       228      165      69593       228
```

Note that the (*current*) time variable is called `tfe`—this will be the time variable to use in a parametric model.

Then split data in small intervals (so small that an assumption of constant rates in each is reasonable)—in this case we use 20 days:

```
> sL <- splitLexis(Lx, seq(0, 1100, 20))
```

Then specify a model for the effect of time using the `gam` function from the `mgcv` package through the `gam.Lexis` function:

```
> pmod <- gam.Lexis(sL, ~ s(tfe))
mgcv::gam Poisson analysis of Lexis object sL with log link:
Rates for the transition:
Alive->Dead
```

This estimates how the mortality rate depends on `tfe`.

The Kaplan-Meier estimator provides estimates of the survival function at the observed event times; a parametric model at any time. Thus, when predicting from a parametric model we must specify the times at which we want the survival function calculated. This is in the form of a data frame of equidistant times (intervals of 10 days, say, this is unrelated to the interval lengths used to split the follow-up for modeling):

```
> nd <- data.frame(tfe = seq(0, 1000, 10))
```

The survival function can then be computed (with confidence intervals) at each of the times supplied in `nd` (new data):

```
> Sf <- ci.surv(pmod, nd)
NOTE: interval length chosen from  as tfe[2] - tfe[1]
> head(Sf)
      Estimate      2.5%     97.5%
[1,] 1.0000000 1.0000000 1.0000000
[2,] 0.9852730 0.9893000 0.9797459
[3,] 0.9704037 0.9782776 0.9597349
[4,] 0.9553965 0.9669351 0.9399581
[5,] 0.9402563 0.9552769 0.9204052
[6,] 0.9249886 0.9433092 0.9010654
```

We can then plot the estimated survival curve:

```
> matshade(nd$tfe, Sf, plot = TRUE,
+          lwd = 2, ylim = c(0, 1), yaxs = "i",
+          xlab = "Time since lung cancer (days)",
+          ylab = "Survival probability")
```

4.1.1 In summary

what you need to do in order to get a parametric survival curve is:

```
> # Lexis object:
> Lx <- Lexis(exit = list(tfe = time),
+      exit.status = factor(status,
+                           labels = c("Alive", "Dead")),
+             data = lung)
NOTE: entry.status has been set to "Alive" for all.
NOTE: entry is assumed to be 0 on the tfe timescale.
> # Split follow up in small intervals
> sL <- splitLexis(Lx, seq(0, 1100, 20))
> # Model rates in these intervals
> pM <- gam.Lexis(sL, ~ s(tfe))
mgcv::gam Poisson analysis of Lexis object sL with log link:
Rates for the transition:
Alive->Dead
> # Survival function times
> nd <- data.frame(tfe = seq(0, 1000, 10))
> # Survival function estimate at these times
> Sf <- ci.surv(pM, nd)
NOTE: interval length chosen from  as tfe[2] - tfe[1]
```

```
> # Survival function plot
> matshade(nd$tfe, Sf, plot = TRUE,
+          lwd = 2, ylim = c(0, 1), yaxs = "i",
+          xlab = "Time since lung cancer (days)",
+          ylab = "Survival probability")
```

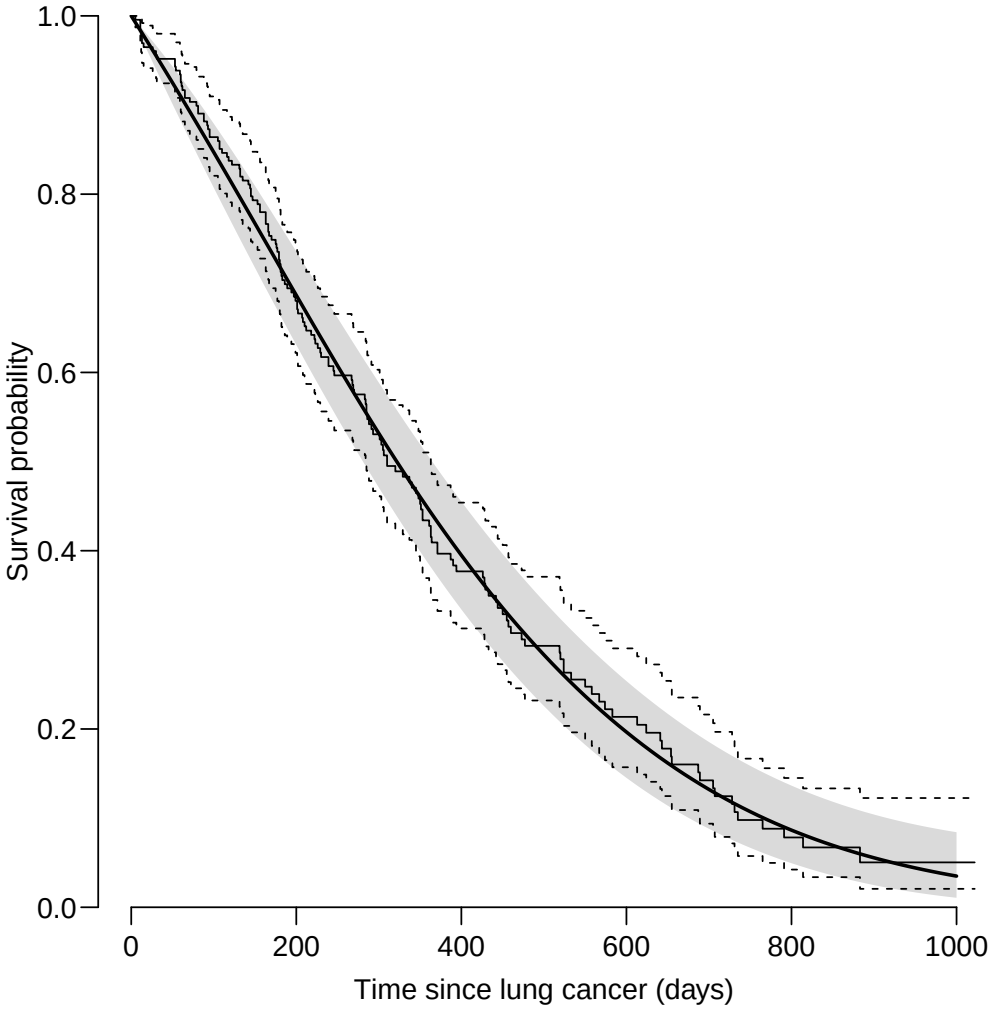

Figure 4.1: *Parametric survival curve based on a* `gam` *model for the rates, compared with the K-M estimator (step function).* `./simple-psurv`

4.1.2 Comparison with K-M:

For illustration we can plot the Kaplan-Meier curve for comparison with the parametric curve:

```
> matshade(nd$tfe, Sf, plot = TRUE,
+          lwd = 2, ylim = c(0, 1), yaxs = "i",
+          xlab = "Time since lung cancer (days)",
+          ylab = "Survival probability")
> km <- survfit(Surv(time, status) ~ 1, data = lung)
> lines(km, lty = 1)
```

4.1.3 The rates:

The parametric model provides estimates of rates (in this case in units of events / day), so we multiply by 365.25 to get rates per 1 year:

```
> Rf <- ci.pred(pM, nd) * 365.25
> head(Rf)
   Estimate      2.5%     97.5%
1 0.5419047 0.3929219 0.7473767
2 0.5554210 0.4091894 0.7539112
3 0.5692682 0.4257944 0.7610862
4 0.5834469 0.4426640 0.7690038
5 0.5979540 0.4597171 0.7777586
6 0.6127833 0.4768741 0.7874267
```

...and we can plot the mortality rates as a function of time:

```
> matshade(nd$tfe, Rf, lwd = 2, log = "y", plot = TRUE,
+          xlab = "Time since lung cancer (days)",
+          ylab = "Mortality rate (per 1 PY)")
```

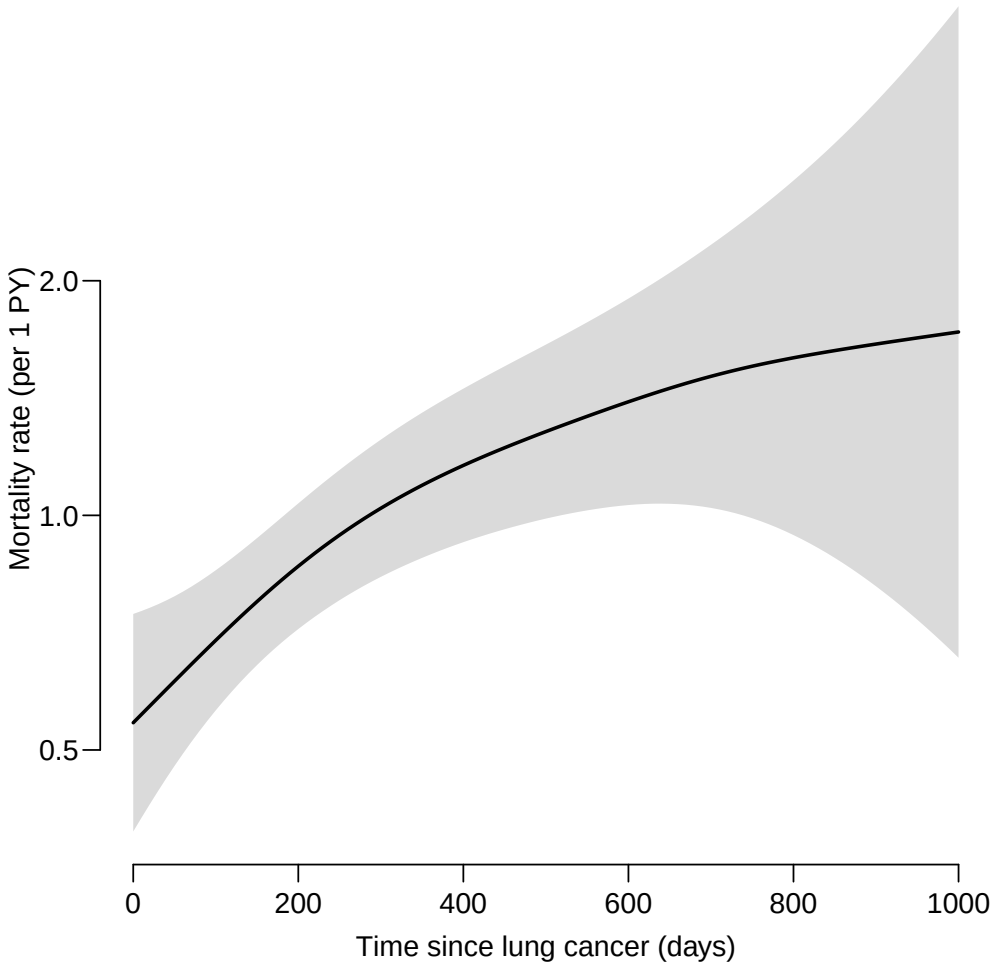

Figure 4.2: *Mortality rates from a parametric model based on a* `gam` *model.* `./simple-rates`

Note that when we show the rates it is necessary to consider the *units* in which they are measured—they are *scaled* quantities as opposed to survival probabilities. Note here that we have rate estimates at more than 0.5 per 1 PY, something like 50% per year—compatible with the 1-year survival of approximately 50%.

4.2 Parametric proportional hazards model

The parametric model can easily be expanded with other (fixed) covariates:

```
> fM <- gam.Lexis(sL, ~ s(tfe) + age + sex)
mgcv::gam Poisson analysis of Lexis object sL with log link:
Rates for the transition:
Alive->Dead
> round(ci.exp(fM), 3)
            exp(Est.)  2.5% 97.5%
(Intercept)     0.001 0.000 0.003
age             1.016 0.998 1.035
sexW            0.603 0.435 0.837
s(tfe).1        0.747 0.370 1.507
s(tfe).2        1.360 0.537 3.446
s(tfe).3        1.208 0.774 1.884
s(tfe).4        0.862 0.523 1.422
s(tfe).5        0.850 0.575 1.255
s(tfe).6        0.853 0.564 1.291
s(tfe).7        1.171 0.809 1.695
s(tfe).8        1.787 0.417 7.663
s(tfe).9        1.293 0.849 1.969
```

This is a proportional hazards model; the hazards (rates, occurrence rates, mortality rates) as function of `tfe` are proportional between, say, women and men—the W/M hazard ratio is the same at all times, namely 0.60.

The model `fM` corresponds to a Cox-model; except that a Cox model does not provide estimates of rates, but only of the hazard ratios:

```
> cM <- coxph(Surv(time, status) ~ age + sex, data = lung)
> round(ci.exp(cM), 3)
     exp(Est.)  2.5% 97.5%
age      1.017 0.999 1.036
sexW     0.599 0.431 0.831
```

—we see that the estimates of covariate effects are identical for all practical purposes.

There is a machinery to produce estimated survival curves from a Cox-model using the so-called Breslow-estimator, say for 60 year old men and women:

```
> Sc <- survfit(cM, data.frame(age = 60, sex = c("M", "W")))
> plot (Sc, col = c("blue", "red"), conf.int = TRUE)
> lines(Sc, col = c("blue", "red"), lwd = 2)
```

From figure 4.3 you observe that the location of the jumps is the same for the two curves, namely at each of the death times in the data set `lung`.

We can estimate the same curves from the parametric proportional hazards model, and overlay these with the step-curves from the Cox model:

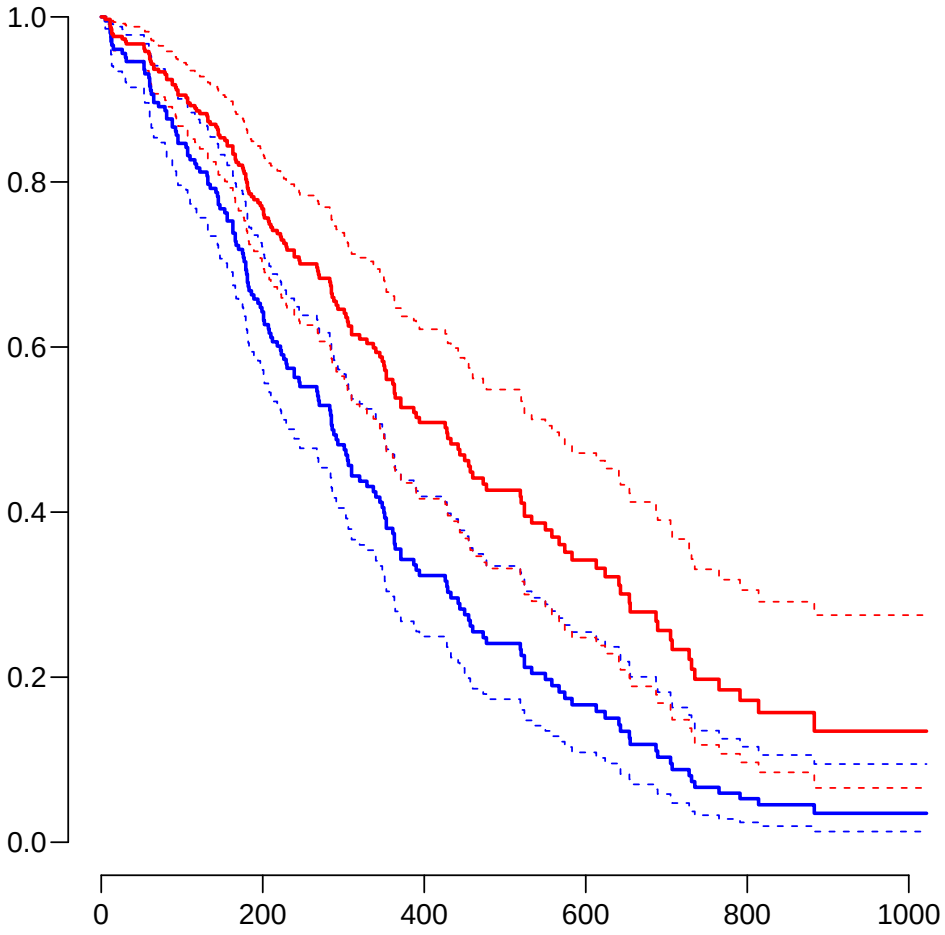

Figure 4.3: *Estimated survival curves for 60 year old men (blue) and women (red). Breslow-estimator from a Cox model.* `./simple-plCox`

```
> Mp <- ci.surv(fM, transform(nd, age = 60, sex = "M"))
NOTE: interval length chosen from  as tfe[2] - tfe[1]
> Wp <- ci.surv(fM, transform(nd, age = 60, sex = "W"))
NOTE: interval length chosen from  as tfe[2] - tfe[1]
> matshade(nd$tfe, cbind(Mp, Wp), col = c("blue","red"),
+          xlab = "Time since lung cancer (days)",
+          ylab = "Survival probability",
+          lwd = 2, ylim = c(0, 1), yaxs = "i", plot = TRUE)
> lines(Sc, conf.int = TRUE)
```

4.2.1 Proportional hazards

can be tested using the `cox.zph` function that tests the absence of interactions between covariates and time ("non-proportionality"):

```
> cox.zph(cM)
```

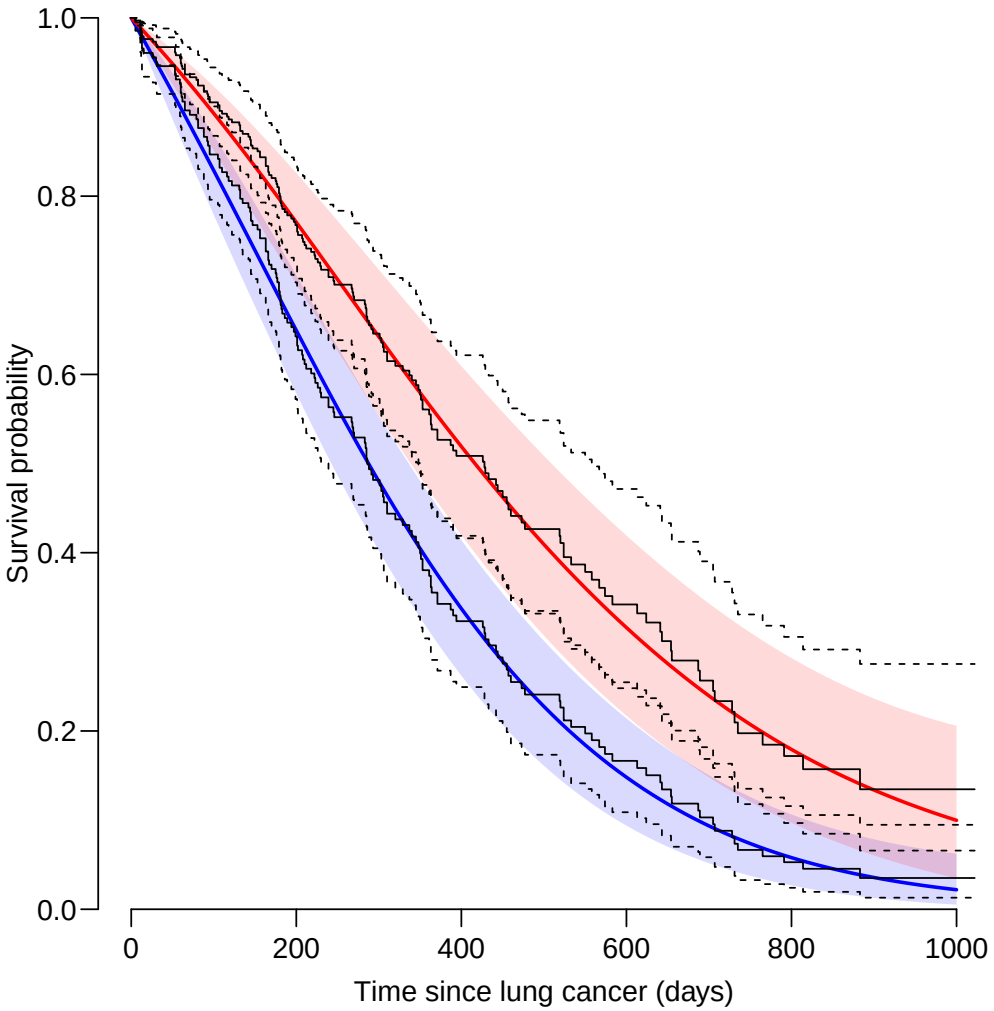

Figure 4.4: *Estimated survival curves for 60 year old men (blue) and women (red) from a model with smooth main effects of time since lung cancer and age. Breslow-estimators from a Cox model overlaid in black.* `./simple-plPH`

```
       chisq df    p
age    0.209  1 0.65
sex    2.608  1 0.11
GLOBAL 2.771  2 0.25
```

The parametric counterpart to this is to compare the smooth parametric models with and without the interaction; for sex we see that the p-value is pretty much the same:

```
> mM <- gam.Lexis(sL, ~ s(tfe, bs = "cr"          ) + sex + age)
mgcv::gam Poisson analysis of Lexis object sL with log link:
Rates for the transition:
Alive->Dead
> iM <- gam.Lexis(sL, ~ s(tfe, bs = "cr", by = sex) + sex + age)
mgcv::gam Poisson analysis of Lexis object sL with log link:
Rates for the transition:
Alive->Dead
> anova(mM, iM, test = "Chisq")
```

```
Analysis of Deviance Table

Model 1: cbind(trt(Lx$lex.Cst, Lx$lex.Xst) %in% trnam, Lx$lex.dur) ~ s(tfe,
    bs = "cr") + sex + age
Model 2: cbind(trt(Lx$lex.Cst, Lx$lex.Xst) %in% trnam, Lx$lex.dur) ~ s(tfe,
    bs = "cr", by = sex) + sex + age
  Resid. Df Resid. Dev      Df Deviance Pr(>Chi)
1    3591.5     1299.5
2    3590.9     1298.0 0.54129   1.4919   0.1102
```

But testing for an interaction without showing the shape of it is a bit odd; so we first plot the estimated rates for 60 year old men and women together, and beside that the estimated M/W rate-ratio as functions of time since lung cancer:

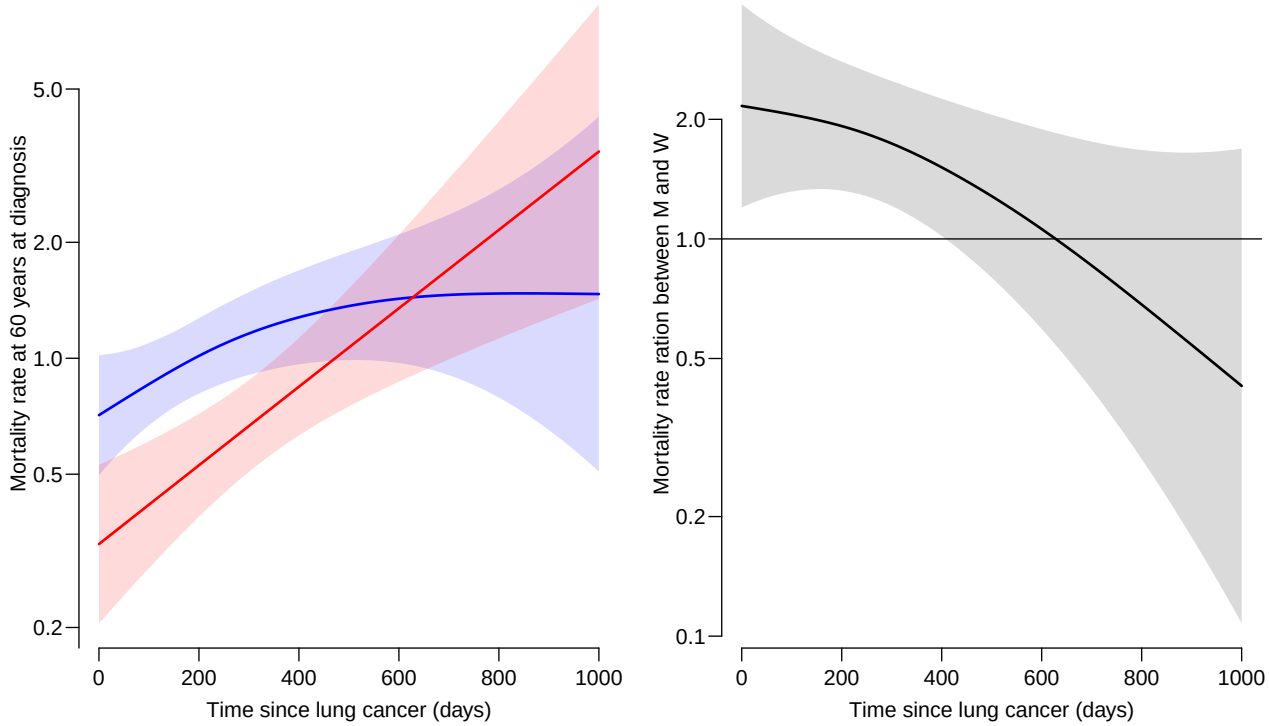

Figure 4.5: *Left: Mortality rates for men (blue) and women (red); right: M/W rate ratio.*
`./simple-2rates`

```
> par(mfrow = c(1,2))
> # rates from M and W
> nd <- data.frame(tfe = seq(0, 1000, 10), age = 60)
> Mp <- ci.pred(iM, mutate(nd, sex = "M"))
> Wp <- ci.pred(iM, mutate(nd, sex = "W"))
> matshade(nd$tfe, cbind(Mp, Wp) * 365, plot = TRUE,
+          col = c("blue","red"),
+          xlab = "Time since lung cancer (days)",
+          ylab = "Mortality rate at 60 years at diagnosis",
+          lwd = 2, log = "y")
> # rate-ratio between M and W
> RR <- ci.exp(iM, list(mutate(nd, age = 60, sex = "M"),
+                       mutate(nd, age = 60, sex = "W")))
```

```
> matshade(nd$tfe, RR, plot = TRUE,
+          xlab = "Time since lung cancer (days)",
+          ylab = "Mortality rate ration between M and W",
+          lwd = 2, log = "y")
> abline(h = 1)
```

It is pretty clear that there is some kind of interaction even if the test produces a p-value of some 10% (both in the parametric and the Cox model). The mortality rates in both sexes increase by time, but more so among women, so the M/W rate ratio is clearly decreasing. Neither of these two features of the mortality rates are available from the Cox-model, even if they are essential for judging the interaction ("non-proportionality").

References

[1] Bendix Carstensen and Martyn Plummer. Using Lexis objects for multi-state models in R. *Journal of Statistical Software*, 38(6):1–18, 1 2011.

[2] DR Cox. Regression and life-tables (with discussion). *J. Roy. Statist. Soc B*, 34:187–220, 1972.

[3] Whitehead J. Fitting Cox's regression model to survival data using GLIM. *Applied Statistics*, 29(3):268–275, 1980.

[4] Martyn Plummer and Bendix Carstensen. Lexis: An R class for epidemiological studies with long-term follow-up. *Journal of Statistical Software*, 38(5):1–12, 1 2011.

[5] K. Rostgaard. Methods for stratification of person-time and events - a prerequisite for Poisson regression and SIR estimation. *Epidemiol Perspect Innov*, 5:7, Nov 2008.