Aim
R packages
Data
Survival data analysis
Estimating the Survival Function
Comparison of survival curves
Modeling survival data
Additional analyses
Competing risk
References

A not so short review on survival analysis in R

Alessio Crippa (https://alecri.github.io) *Institutet*

Biostatistics Team (http://stats4life.se/)
Department of Public Health Sciences, Karolinska (http://ki.se/en/phs/startpage)

March 23, 2017

(http://ki.se/en/phs/startpage)
(Last compiled on Mon Jan 29 14:18:04 2018)

(http://ki.se/en/phs/startpage)

Aim

(http://ki.se/en/phs/startpage)
The aim of this document is to give a short but yet comprehensive review on how to conduct survival analysis in R. The literature on the topic is very extensive and only a limited number of (common) problems/features will be covered.
The amount of R packages available reflects the extent of the research on the topic. A broad (but not complete) task view presenting useful R packages for different aspects of survival analysis can be found at (http://ki.se/en/phs/startpage)here (https://cran.r-project.org/web/views/Survival.html).

R packages

A variety of R packages can be used to tackle specific problems and there are alternative functions to address a common question. The following are the packages used in this review for reading, managing, analyzing, and displaying data.
Run the following lines for installing and loading the required packages.

```
pkg <- c("knitr", "kfigr", "tidyverse", "survival", "ggfortify", "survminer", "plotly",
         "gridExtra", "Epi", "KMsurv", "gnm", "cmprsk", "mstate", "flexsurv", "splines",
         "epitools", "eha", "shiny")
new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
if (length(new.pkg)) install.packages(new.pkg, dependencies = TRUE)
sapply(pkg, require, character.only = TRUE)
```

```
    knitr     kfigr tidyverse  survival ggfortify survminer    plotly gridExtra       Epi
     TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
   KMsurv       gnm    cmprsk    mstate  flexsurv   splines  epitools       eha     shiny
     TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
```

OBS Data manipulation and graphics will be based on a collection of packages that share common philosophies and are designed to work together (http://tidyverse.org/ (http://tidyverse.org/)). Additional useful resources can be found here 1 (http://r4ds.had.co.nz/), 2 (http://adv-r.had.co.nz/).

Data

The review will be based on the `orca` dataset which contains data from a population-based retrospective cohort design. The dataset can be found in a text file format at http://www.stats4life.se/data/oralca.txt (http://www.stats4life.se/data/oralca.txt). It includes a subset of 338 patients diagnosed with an oral squamous cell carcinoma (OSCC) between January 1, 1985 and December 31, 2005 from the 2 northernmost provinces of Finland. Follow-up of patients was started on the date of cancer diagnosis and ended on the date of death, migration, or the closing date of the follow-up, December 31, 2008. Cause of death was classified into the 2 categories: (1) deaths from OSCC; and (2) deaths of other causes.
The dataset contains the following variables:
`id` = a sequential number,
`sex` = sex, a factor with categories 1 = “Female”, 2 = “Male”,
`age` = age (years) at the date of diagnosing the cancer,
`stage` = TNM stage of the tumor (factor): 1 = “I”, …, 4 = “IV”, 5 = “unkn”
`time` = follow-up time (in years) since diagnosis until death or censoring,
`event` = event ending the follow-up (factor): 1 = censoring alive, 2 = death from oral cancer, 3 = death from other causes.

Thanks to Esa Läärä (http://stat.oulu.fi/laara/) for sharing the data.

Load data into R from the URL.

```
orca <- read.table("http://www.stats4life.se/data/oralca.txt", header = T)
```

Have a feeling of the data at hand.

```
head(orca)
```

```
  id    sex      age stage  time          event
1  1   Male 65.42274  unkn 5.081          Alive
2  2 Female 83.08783   III 0.419 Oral ca. death
3  3   Male 52.59008    II 7.915    Other death
4  4   Male 77.08630     I 2.480    Other death
5  5   Male 80.33622    IV 2.500 Oral ca. death
6  6 Female 82.58132    IV 0.167    Other death
```

```
str(orca)
```

```
'data.frame':   338 obs. of  6 variables:
 $ id   : int  1 2 3 4 5 6 7 8 9 10 ...
 $ sex  : Factor w/ 2 levels "Female","Male": 2 1 2 2 2 1 2 2 1 2 ...
 $ age  : num  65.4 83.1 52.6 77.1 80.3 ...
 $ stage: Factor w/ 5 levels "I","II","III",..: 5 3 2 1 4 4 2 5 4 2 ...
 $ time : num  5.081 0.419 7.915 2.48 2.5 ...
 $ event: Factor w/ 3 levels "Alive","Oral ca. death",..: 1 2 3 3 2 3 1 2 2 1 ...
```

```
summary(orca)
```

```
       id             sex           age         stage         time                   event
 Min.   :  1.00   Female:152   Min.   :15.15   I   :50   Min.   : 0.085   Alive         :109
 1st Qu.: 85.25   Male  :186   1st Qu.:53.24   II  :77   1st Qu.: 1.333   Oral ca. death:122
 Median :169.50                Median :64.86   III :72   Median : 3.869   Other death   :107
 Mean   :169.50                Mean   :63.51   IV  :68   Mean   : 5.662
 3rd Qu.:253.75                3rd Qu.:74.29   unkn:71   3rd Qu.: 8.417
 Max.   :338.00                Max.   :92.24             Max.   :23.258
```

Survival data analysis

Survival analysis focuses on time to event data, usually referred to as failure time T, $T \geq 0$. In our example, T is time to death after diagnosis.

In order to define a failure time random variable, we need:
1. a time origin (diagnosis of OSCC),
2. a time scale (years after diagnosis, age),
3. definition of the event. We will first consider total (or all-cause) mortality, pooling the two causes of death into a single outcome (Figure 1 (A)).

Show Source

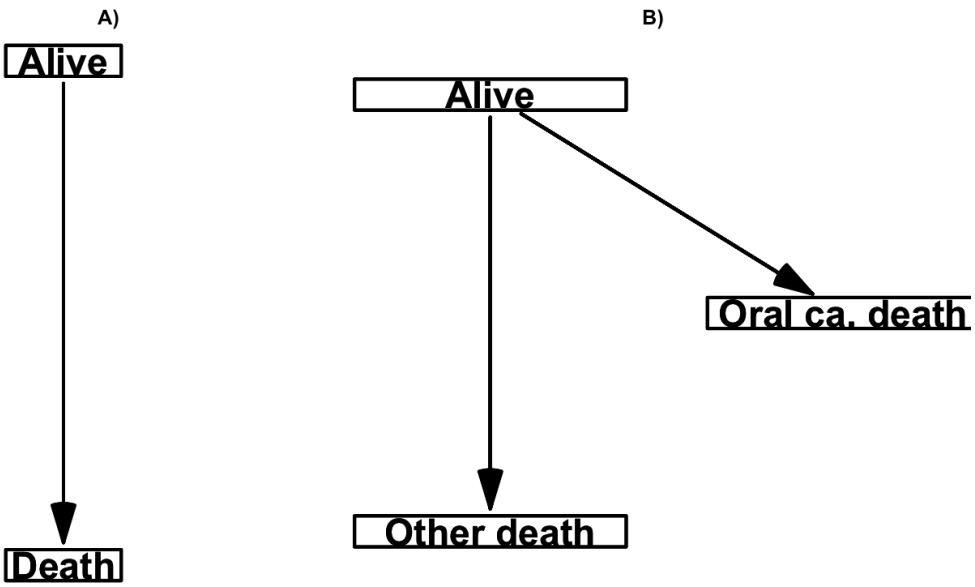


Figure 1: Box diagram for transitions.

```
table(orca$event)
```

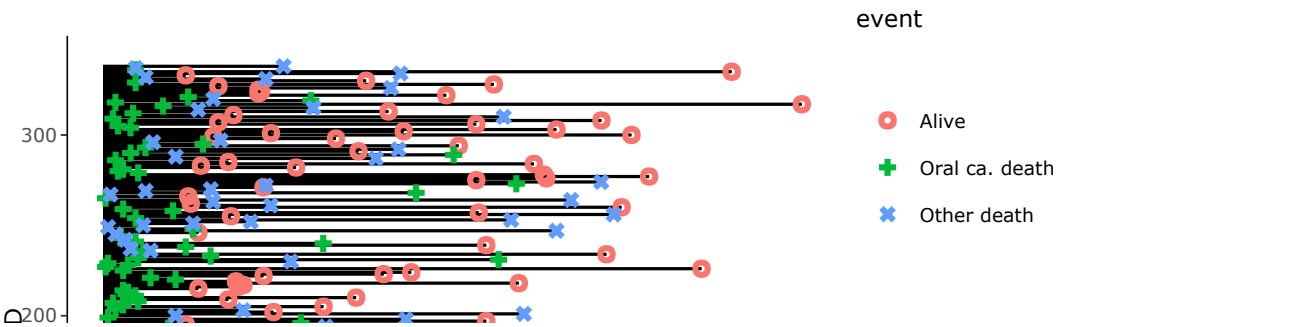
```
         Alive Oral ca. death    Other death
           109            122            107
```

```
orca$all <- 1*(orca$event != "Alive")
table(orca$all)
```

```
  0   1
109 229
```

A graphically presentation of the observed follow-up times can be of great aid in an analysis of survival data. The illustration of survival data in Figure 2 shows several features which are typically encountered in analysis of survival data.

Show Source



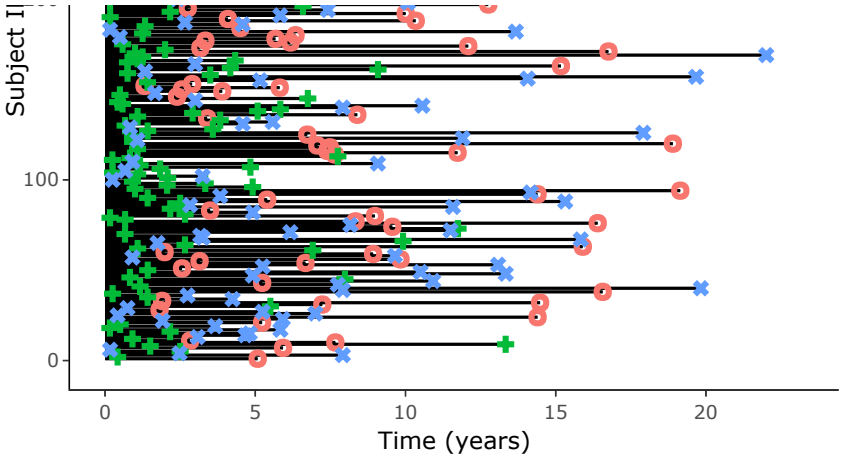


Figure 2: Possible representations of follow-up time.

Different time scales can give different perspectives.

Show Source

Show Plot

Death from OSCC is more likely to occur early after diagnosis, as opposed to death from other causes. What about the type of censoring?

A survival object is defined by the pair (y, δ), i.e. the *time* variable and the *failure* or *status* indicator. To create such an object in R, the `Surv()` function in the `survival` package can be used. See the help page by typing `?Surv` (https://stat.ethz.ch/R-manual/R-devel/library/survival/html/Surv.html) for a description of the different possibilities.

```
su_obj <- Surv(orca$time, orca$all)
str(su_obj)
```

```
 Surv [1:338, 1:2]  5.081+  0.419   7.915   2.480   2.500   0.167   5.925+  1.503  13.333   7.666+
...
 - attr(*, "dimnames")=List of 2
  ..$ : NULL
  ..$ : chr [1:2] "time" "status"
 - attr(*, "type")= chr "right"
```

The created survival object is then used as response variable in other specific functions for survival analysis.

There are several equivalent ways to characterize the probability distribution of a survival random variable:
1. the density function $f(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} Pr(t \leq T \leq t + \Delta t)$;
2. the cumulative distribution function $F(t) = P(T \leq t) = \int_0^t f(u)du$, i.e. the probability of dying within a certain time t;
3. the survivor function $S(t) = P(T > t) = \int_t^\infty f(u)du$, i.e. the probability of surviving longer than time t;
4. the hazard function $\lambda(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} Pr(t \leq T \leq t + \Delta t | T \geq T) = \frac{f(t)}{S(t)}$, usually referred to as instantaneous failure rate, the force of mortality;
5. the cumulative hazard function $\Lambda(t) = \int_0^t \lambda(u)du$.

N.B. These distributions are closely related to each other. Hence one may estimate one and derive the remaining.

$$\lambda(t) = \frac{f(t)}{1 - F(t)} = -\frac{S'(t)}{S(t)} = -\frac{d \log[S(t)]}{dt}$$

$$\Lambda(t) = -\log[S(t)]$$

$$S(t) = \exp(-\Lambda(t)) = \exp\left(-\int_0^t \lambda(v)dv\right)$$

$$f(t) = \lambda(t)S(t)$$

$$F(t) = 1 - \exp(-\Lambda(t)) = \int_0^t \lambda(v)S(v)dv$$

In addition, measures of central tendency can be useful for summarizing the observed distributions:
1. mean survival $\mu = \int_0^\infty uf(u)du$;
2. median survival $\tau : \min_\tau S(\tau) \leq 0.5$;
3. any other quantiles.

Estimating the Survival Function

There are two alternatives for estimating the survival or the hazard:
a. an empirical estimate of the survival function (i.e., non-parametric estimation);
b. a parametric model for $\lambda(t)$ based on a particular density function $f(t)$.

Non-parametric estimators

We will first cover the class of non-parametric estimators (a.), which includes the Kaplan–Meier, Life-table, and Nelson-Aalen estimators.

Kaplan–Meier estimator

The Kaplan-Meier estimator is the most common estimator and can be explained using different strategies (product limit estimator, likelihood justification).

$$\hat{S}(t) = \prod_{j:\tau_j \leq t} \left(1 - \frac{d_j}{r_j}\right)$$

with τ_j = distinct death times observed in the sample, d_j = number of deaths at τ_j, and r_j = number of individuals at risk right before the j-th death time ($r_j = r_{j-1} - d_{j-1} - c_{j-1}$, c_j = number of censored observations between the j-th and $(j+1)$-st death times).

A survival curve is based on a tabulation of the number at risk and number of events at each unique death times. The `survfit()` function of the `survival` package creates (estimates) the survival curves using different methods (see `?survfit` (https://stat.ethz.ch/R-manual/R-devel/library/survival/html/survfit.html)).
Using the created survival object `su_obj` as response variable in the formula, the `survfit()` function will return the default calculations for a Kaplan–Meier analysis of the (overall) survival curve.

```
fit_km <- survfit(su_obj ~ 1, data = orca)
# str(fit_km)
print(fit_km, print.rmean = TRUE)
```

```
Call: survfit(formula = su_obj ~ 1, data = orca)

        n     events     *rmean *se(rmean)     median    0.95LCL    0.95UCL
  338.000    229.000      8.060      0.465      5.418      4.331      6.916
    * restricted mean with upper limit =  23.3
```

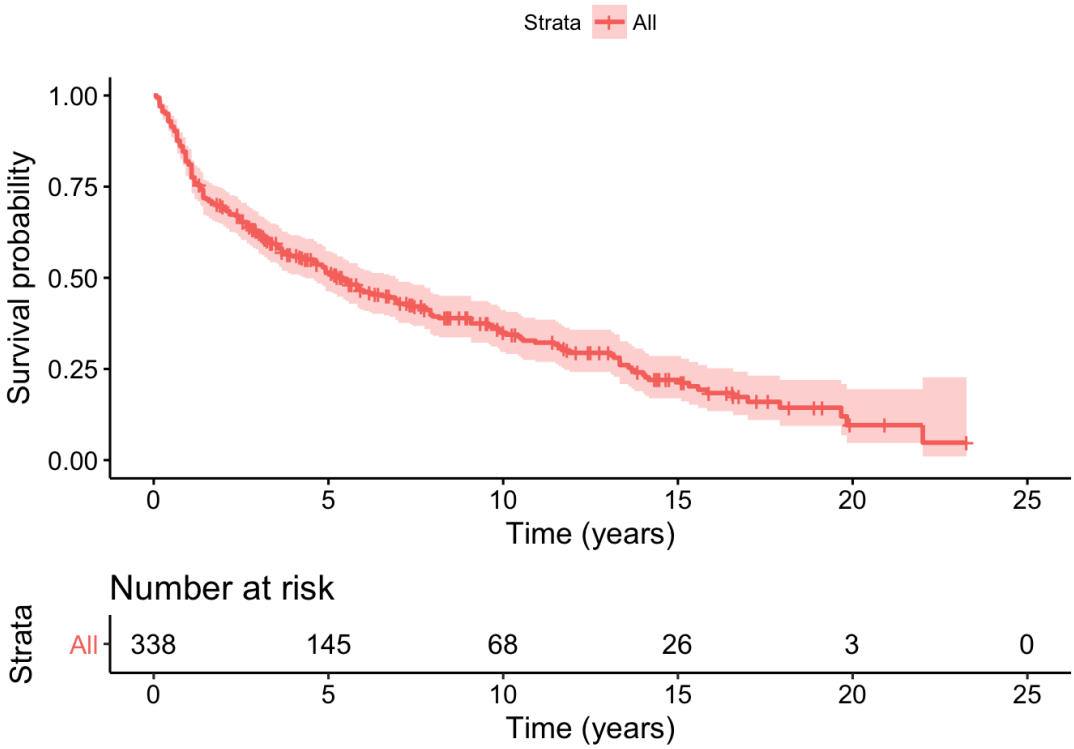
The `print()` function returns just a summary of the estimated survival curve. The `fortify()` function of the `ggfortify` package is useful to extract the whole survival table in a data.frame.

```
dat_km <- fortify(fit_km)
head(dat_km)
```

```
   time n.risk n.event n.censor      surv     std.err     upper     lower
1 0.085    338       2        0 0.9940828 0.004196498 1.0000000 0.9859401
2 0.162    336       2        0 0.9881657 0.005952486 0.9997618 0.9767041
3 0.167    334       4        0 0.9763314 0.008468952 0.9926726 0.9602592
4 0.170    330       2        0 0.9704142 0.009497400 0.9886472 0.9525175
5 0.246    328       1        0 0.9674556 0.009976176 0.9865584 0.9487228
6 0.249    327       1        0 0.9644970 0.010435745 0.9844277 0.9449699
```

The `ggsurvplot()` is a dedicated function in the `survminer` package to give an informative illustration of the estimated survival curve(s). See the help page `?ggsurvplot` for a description of the different possibilities (arguments).

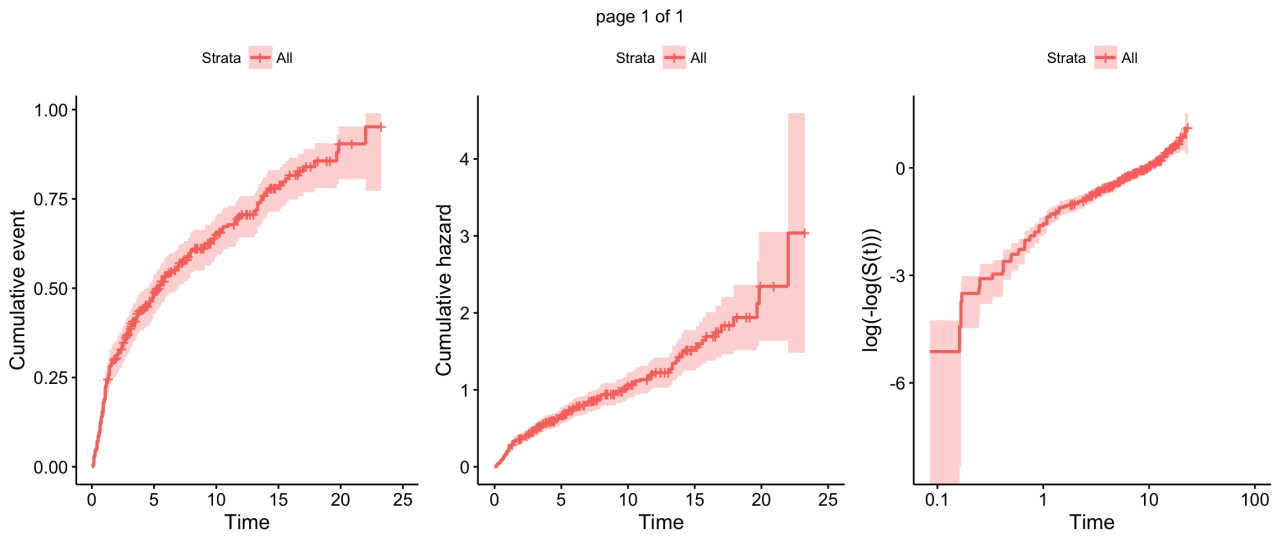
```
ggsurvplot(fit_km, risk.table = TRUE, xlab = "Time (years)", censor = T)
```



N.B.: see the help page `?survfit.formula` (https://stat.ethz.ch/R-manual/R-devel/library/survival/html/survfit.formula.html) for a description of the different methods for constructing confidence intervals (argument `conf.type`).

The default KM plot presents the survival function. Several alternatives/functions are available (see the help page `?plot.survfit` (https://stat.ethz.ch/R-manual/R-devel/library/survival/html/plot.survfit.html) or `?ggsurvplot` (http://www.sthda.com/english/rpkgs/survminer/reference/ggsurvplot.html)).

```
glist <- list(
  ggsurvplot(fit_km, fun = "event", main = "Cumulative proportion"),
  ggsurvplot(fit_km, fun = "cumhaz",  main = "Cumulative Hazard"),
  ggsurvplot(fit_km, fun = "cloglog", main = "Complementary log-log")
)
do.call(marrangeGrob, list(grobs = lapply(glist, function(x) x$plot), ncol = 3, nrow = 1))
```



Lifetable or actuarial estimator

The lifetable method is very common in actuary and demography. It is particularly suitable for grouped data.

$$\hat{S}(t_j) = \prod_{l \leq j} \hat{p}_l$$

with $\hat{p}_j = 1 - \hat{q}_j$ (conditional probability of surviving), $\hat{q}_j = d_j / r'_j$ (conditional probability of dying), and $r'_j = r_j - c_j/2$

In order to show this method on the actual example, we need first to create aggregated data, i.e. divide the follow-up in groups and calculate in each strata the number of people at risk, events, and censored.

```
cuts <- seq(0, 23, 1)
lifetab_dat <- orca %>%
  mutate(time_cat = cut(time, cuts)) %>%
  group_by(time_cat) %>%
  summarise(nlost = sum(all == 0),
            nevent = sum(all == 1))
```

Based on the grouped data, we will estimate the survival curve using the `lifetab()` in the `KMsurv` package. See the help page `?lifetab` (https://www.rdocumentation.org/packages/KMsurv/versions/0.1-5/topics/lifetab) for a description of the arguments and example.

```
dat_lt <- with(lifetab_dat, lifetab(tis = cuts, ninit = nrow(orca),
                                    nlost = nlost, nevent = nevent))
round(dat_lt, 4)
```

```
      nsubs nlost nrisk nevent   surv    pdf hazard se.surv se.pdf se.hazard
0-1     338     0 338.0     64 1.0000 0.1893 0.2092  0.0000 0.0213    0.0260
1-2     274     4 272.0     41 0.8107 0.1222 0.1630  0.0213 0.0179    0.0254
2-3     229     9 224.5     21 0.6885 0.0644 0.0981  0.0252 0.0136    0.0214
3-4     199    12 193.0     20 0.6241 0.0647 0.1093  0.0265 0.0140    0.0244
4-5     167     9 162.5     13 0.5594 0.0448 0.0833  0.0274 0.0121    0.0231
5-6     145    14 138.0     13 0.5146 0.0485 0.0989  0.0279 0.0131    0.0274
6-7     118     5 115.5      8 0.4662 0.0323 0.0717  0.0283 0.0112    0.0254
7-8     105     8 101.0      9 0.4339 0.0387 0.0933  0.0286 0.0126    0.0311
8-9      88     7  84.5      1 0.3952 0.0047 0.0119  0.0288 0.0047    0.0119
9-10     80     4  78.0      8 0.3905 0.0401 0.1081  0.0288 0.0137    0.0382
10-11    68     4  66.0      5 0.3505 0.0266 0.0787  0.0291 0.0116    0.0352
11-12    59     3  57.5      5 0.3239 0.0282 0.0909  0.0292 0.0123    0.0406
12-13    51     6  48.0      0 0.2958 0.0000 0.0000  0.0293    NaN       NaN
13-14    45     2  44.0      8 0.2958 0.0538 0.2000  0.0293 0.0180    0.0704
14-15    35     6  32.0      3 0.2420 0.0227 0.0984  0.0295 0.0128    0.0567
15-16    26     3  24.5      4 0.2193 0.0358 0.1778  0.0295 0.0171    0.0885
16-17    19     5  16.5      2 0.1835 0.0222 0.1290  0.0296 0.0152    0.0910
17-18    12     2  11.0      1 0.1613 0.0147 0.0952  0.0299 0.0142    0.0951
18-19     9     2   8.0      0 0.1466 0.0000 0.0000  0.0306    NaN       NaN
19-20     7     2   6.0      2 0.1466 0.0489 0.4000  0.0306 0.0300    0.2771
20-21     3     1   2.5      0 0.0977 0.0000 0.0000  0.0348    NaN       NaN
21-22     2     0   2.0      1 0.0977 0.0489 0.6667  0.0348 0.0387    0.6285
22-23     1     1   0.5      0 0.0489     NA     NA  0.0387     NA        NA
```

Nelson-Aalen estimator

The focus of the Nelson-Aalen estimator is on the cumulative hazard at time t $\left(\hat{\Lambda}_{NA}(t)\right)$.

$$\hat{\Lambda}_{\mathrm{NA}}(t) = \sum_{j:\tau_j \leq t} \frac{d_j}{r_j}$$

Once we have $\hat{\Lambda}(t)_{NA}$, we can derive the Fleming-Harrington estimator of $S(t)$

$$\hat{S}_{\mathrm{FH}} = \exp(-\hat{\Lambda}(t)_{\mathrm{NA}})$$

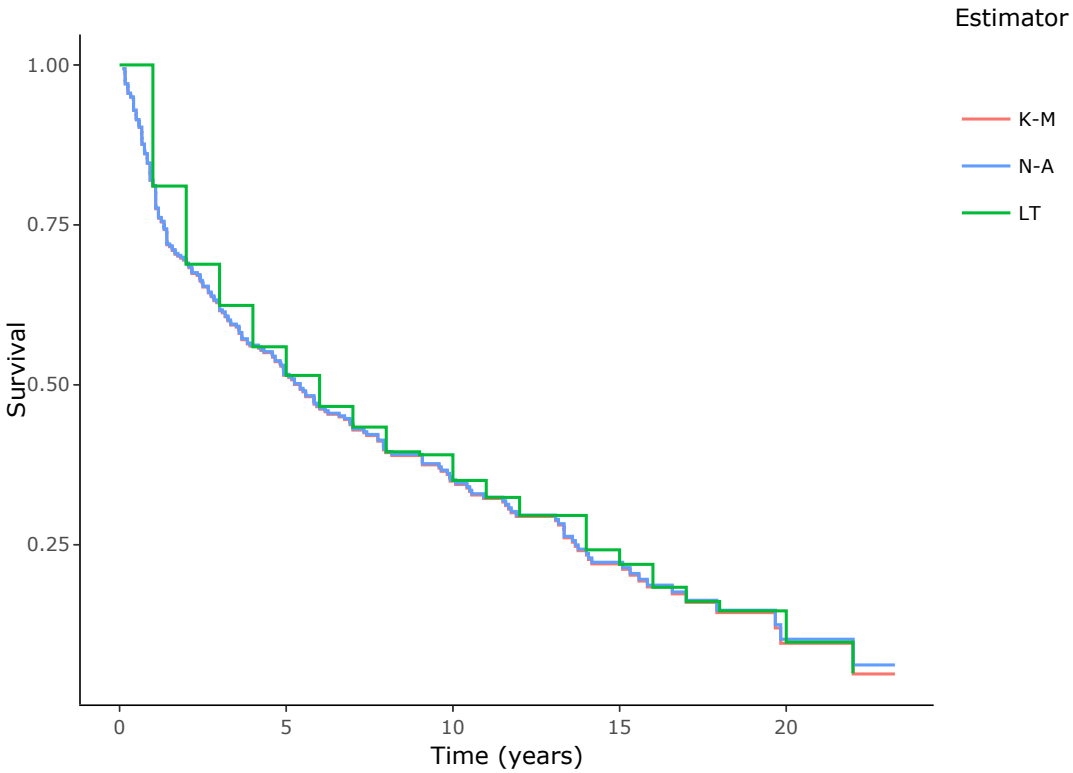
```
fit_fh <- survfit(su_obj ~ 1, data = orca, type = "fleming-harrington", conf.type = "log-log")
dat_fh <- fortify(fit_fh)
## for the Nelson-Aalen estimator of the cumulative hazard
#dat_fh <- fortify(fit_fh, fun = "cumhaz")
head(dat_fh)
```

```
   time n.risk n.event n.censor      surv     std.err     upper     lower
1 0.085    338       2        0 0.9941003 0.004196498 0.9985273 0.9765229
2 0.162    336       2        0 0.9882006 0.005952486 0.9955680 0.9687798
3 0.167    334       4        0 0.9764365 0.008468952 0.9881827 0.9532939
4 0.170    330       2        0 0.9705366 0.009497400 0.9840793 0.9457956
5 0.246    328       1        0 0.9675821 0.009976176 0.9819575 0.9420958
6 0.249    327       1        0 0.9646277 0.010435745 0.9797988 0.9384268
```

Graphical comparison

It is possible to plot different estimates of the survival functions to evaluate potential differences.

```
ggplotly(
ggplot() +
  geom_step(data = dat_km, aes(x = time, y = surv, colour = "K-M")) +
  geom_step(data = dat_fh, aes(x = time, y = surv, colour = "N-A")) +
  geom_step(data = dat_lt, aes(x = cuts[-length(cuts)], y = surv, colour = "LT")) +
  labs(x = "Time (years)", y = "Survival", colour = "Estimator") +
  theme_classic()
)
```



Measures of central tendency

Measures of central tendency such as quantiles can be derived from the estimated survival curves.

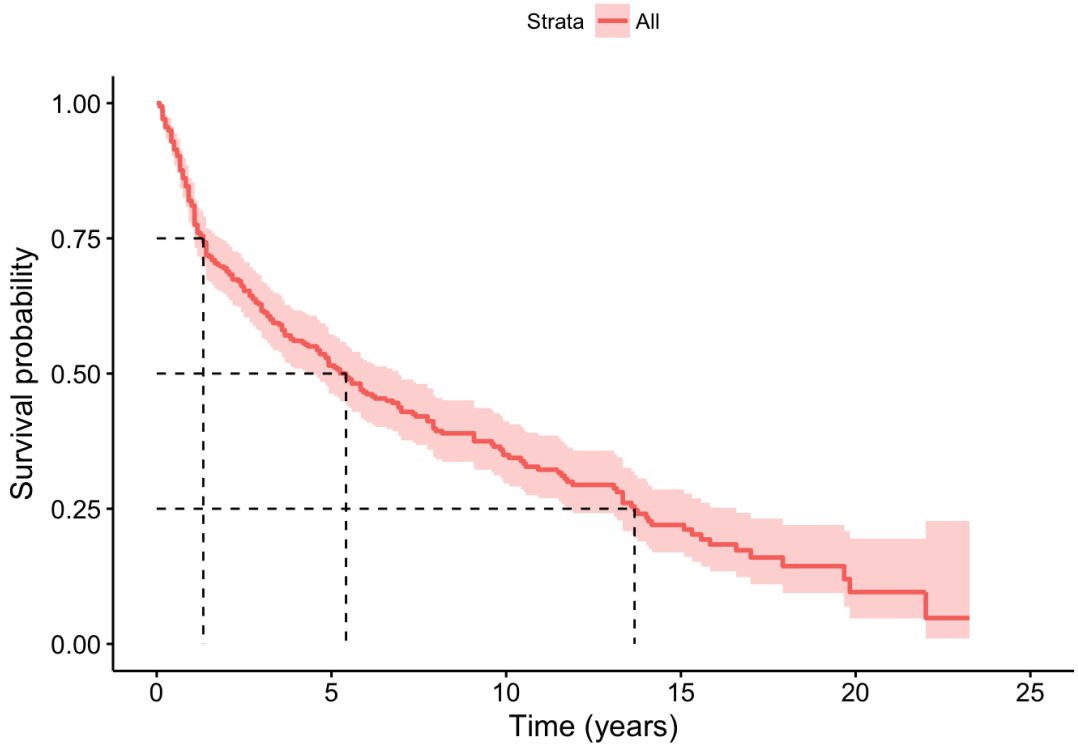
```
(mc <- data.frame(q = c(.25, .5, .75),
                  km = quantile(fit_km),
                  fh = quantile(fit_fh)))
```

```
      q km.quantile km.lower km.upper fh.quantile fh.lower fh.upper
25 0.25       1.333    1.084    1.834       1.333    1.084    1.747
50 0.50       5.418    4.331    6.916       5.418    4.244    6.913
75 0.75      13.673   11.748   16.580      13.673   11.748   15.833
```

Half of the individuals is estimated to live longer than 5.4 years.
The first one-fourth of the individuals died within 1.3 year while the top three-fourths of the individuals lived longer than 1.3 days.
The first three-fourths of the individuals died within 13.7 year while the top one-fourth of the individuals lived longer than 13.7 days.

A graphical presentation of the estimated quantities (based on the survival curve using K-M).

```
ggsurvplot(fit_km, xlab = "Time (years)", censor = F)$plot +
  geom_segment(data = mc, aes(x = km.quantile, y = 1-q, xend = km.quantile, yend = 0), lty = 2) +
  geom_segment(data = mc, aes(x = 0, y = 1-q, xend = km.quantile, yend = 1-q), lty = 2)
```



Parametric estimators

As opposed to a non-parametric approach, a parametric one assumes a distribution for the survival distribution. The family of survival distributions can be written by introducing location and scale changes of the form

$$\log(T) = \mu + \sigma W$$

where the parametric assumption is made on W.
The `flexsurvreg()` function in the `flexsurv` package estimates parametric accelerated failure time (AFT) models. See the help page `?flexsurvreg` (https://www.rdocumentation.org/packages/flexsurv/versions/1.0.0/topics/flexsurvreg) for a description of different parametric distributions for W.
We are going to consider three common choices: the exponential, the Weibull, and the log-logistic models. In addition, the flexible parametric modelling of time-to-event data using the spline model of Royston and Parmar (2002) is also considered.

Model	Hazard	Survival
Exponential	$\lambda(t) = \lambda$	$S(t) = \exp(-\lambda t)$
Weibull	$\lambda(t) = \lambda^p p t^{p-1}$	$S(t) = \exp((-\lambda t)^p)$
Log logistic	$\lambda(t) = \frac{\lambda p (\lambda t)^{p-1}}{1+(\lambda t)^p}$	$S(t) = \frac{1}{1+(\lambda t)^p}$

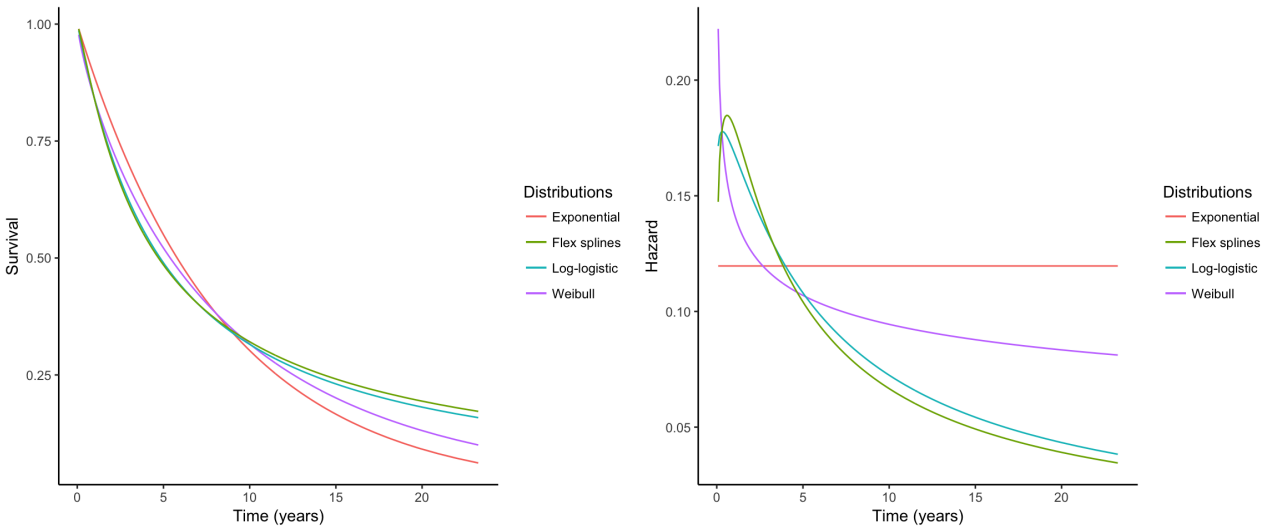
```
fit_exp <- flexsurvreg(su_obj ~ 1, data = orca, dist = "exponential")
fit_exp
```

```
Call:
flexsurvreg(formula = su_obj ~ 1, data = orca, dist = "exponential")

Estimates:
      est      L95%     U95%     se
rate  0.11967  0.10513  0.13621  0.00791

N = 338,  Events: 229,  Censored: 109
Total time at risk: 1913.673
Log-likelihood = -715.1802, df = 1
AIC = 1432.36
```

```
fit_w <- flexsurvreg(su_obj ~ 1, data = orca, dist = "weibull")
fit_ll <- flexsurvreg(su_obj ~ 1, data = orca, dist = "llogis")
fit_sp <- flexsurvspline(su_obj ~ 1, data = orca, k = 1, scale = "odds")
```

Again, different approaches can be graphically compared (estimated survival curves or hazard functions). *OBS* `ggsurvplot()` has not yet implemented graphical functions for object of class `flexsurvreg`. We can create the plot of our self with some extra lines of code.

Show Source



Comparison of survival curves

A common research question is to compare the survival functions between 2 or more groups. Several alternatives (as well as packages) are available. Have a look at the *testing* section in the survival analysis task view (https://cran.r-project.org/web/views/Survival.html).

Tumor stage, for example, is an important prognostic factor in cancer survival studies. We can estimate and plot separate survival curves for the different groups (stages) with different colors.

```
#ci.exp(glm(all ~ 0 + stage, data = orca, family = "poisson", offset = log(time)))
with(orca %>% group_by(stage) %>%
  summarise(D = sum(all),
            Y = sum(time)),
  cbind(stage, pois.approx(x = D, pt = Y)))
```

```
  stage  x      pt       rate      lower     upper conf.level
1     I 25 336.776 0.07423332 0.04513439 0.1033322       0.95
2    II 51 556.700 0.09161128 0.06646858 0.1167540       0.95
3   III 51 464.836 0.10971611 0.07960454 0.1398277       0.95
4    IV 57 262.552 0.21709985 0.16073995 0.2734597       0.95
5  unkn 45 292.809 0.15368380 0.10878136 0.1985862       0.95
```

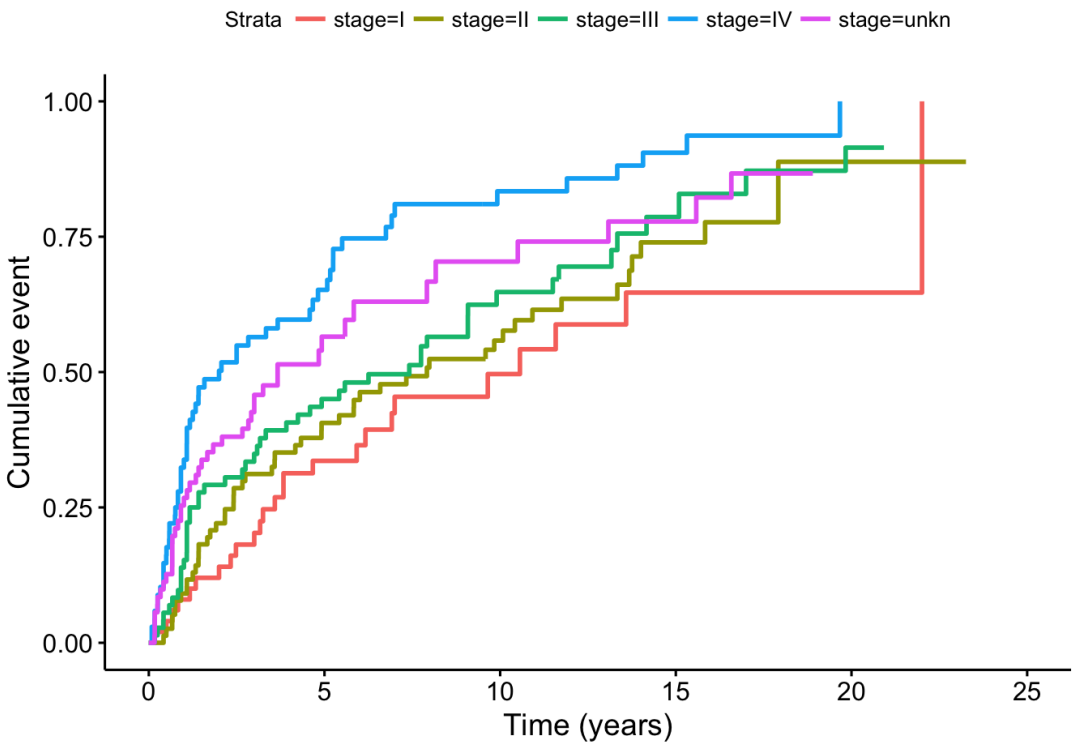
In general, patients diagnostic with a lower stage tumor has a lower (mortality) rate as compared to patients with high stage tumor. An overall comparison of the survival functions can be performed using the `survfit()` function.

```
su_stg  <- survfit(su_obj ~ stage, data = orca)
su_stg
```

```
Call: survfit(formula = su_obj ~ stage, data = orca)

            n events median 0.95LCL 0.95UCL
stage=I    50     25  10.56    6.17      NA
stage=II   77     51   7.92    4.92   13.34
stage=III  72     51   7.41    3.92    9.90
stage=IV   68     57   2.00    1.08    4.82
stage=unkn 71     45   3.67    2.83    8.17
```

As the incidence rates are lower for low tumoral stages, the median survival times also decrease for increasing levels of tumoral stage. The same behavior can be observed plotting the K-M survival curves separately for the different tumoral stages.

```
ggsurvplot(su_stg, fun = "event", censor = F, xlab = "Time (years)")
```



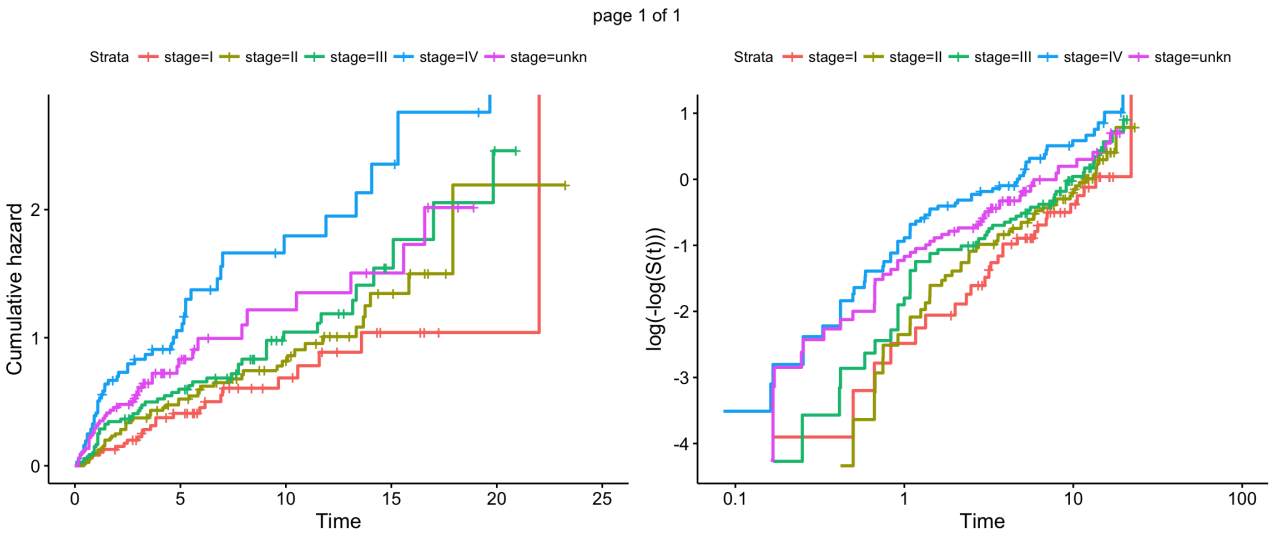
It is also possible to construct the whole survival table for each stage level. Here the first 3 lines of the survival table in each tumoral stage.

```
lifetab_stg <- fortify(su_stg)
lifetab_stg %>%
  group_by(strata) %>%
  do(head(., n = 3))
```

```
# A tibble: 15 x 9
# Groups:   strata [5]
    time n.risk n.event n.censor      surv    std.err     upper     lower strata
   <dbl>  <dbl>   <dbl>    <dbl>     <dbl>      <dbl>     <dbl>     <dbl> <fctr>
 1 0.170     50       1        0 0.9800000 0.02020305 1.0000000 0.9419529      I
 2 0.498     49       1        0 0.9600000 0.02886751 1.0000000 0.9071919      I
 3 0.665     48       1        0 0.9400000 0.03572948 1.0000000 0.8764252      I
 4 0.419     77       1        0 0.9870130 0.01307217 1.0000000 0.9620459     II
 5 0.498     76       1        0 0.9740260 0.01860968 1.0000000 0.9391392     II
 6 0.665     75       1        0 0.9610390 0.02294560 1.0000000 0.9187760     II
 7 0.167     72       1        0 0.9861111 0.01398636 1.0000000 0.9594462    III
 8 0.249     71       1        0 0.9722222 0.01992048 1.0000000 0.9349948    III
 9 0.413     70       1        0 0.9583333 0.02457366 1.0000000 0.9132706    III
10 0.085     68       2        0 0.9705882 0.02111002 1.0000000 0.9312497     IV
11 0.162     66       1        0 0.9558824 0.02605251 1.0000000 0.9082983     IV
12 0.167     65       1        0 0.9411765 0.03031695 0.9987962 0.8868807     IV
13 0.162     71       1        0 0.9859155 0.01418475 1.0000000 0.9588830   unkn
14 0.167     70       2        0 0.9577465 0.02492740 1.0000000 0.9120787   unkn
15 0.170     68       1        0 0.9436620 0.02899769 0.9988478 0.8915251   unkn
```

Alternatively, the cumulative hazards and the log-cumulative hazards for the different stages can be presented.

```
glist <- list(
  ggsurvplot(su_stg, fun = "cumhaz"),
  ggsurvplot(su_stg, fun = "cloglog")
)
# plot(su_stg, fun = "cloglog")
do.call(marrangeGrob, list(grobs = lapply(glist, function(x) x$plot), ncol = 2, nrow = 1))
```



Several methods have been developed for formally testing the overall equivalence of survival curves. The `survdiff()` in the `survival` package implements the *G-rho* family of tests for evaluating differences in survival curves. See the help page (`?survdiff`) for the different options .

Mantel-Haenszel logrank test

The default argument `rho = 0` implements the log-rank or Mantel-Haenszel test.

```
survdiff(su_obj ~ stage, data = orca)
```

```
Call:
survdiff(formula = su_obj ~ stage, data = orca)

            N Observed Expected (O-E)^2/E (O-E)^2/V
stage=I    50       25     39.9     5.573     6.813
stage=II   77       51     63.9     2.606     3.662
stage=III  72       51     54.1     0.174     0.231
stage=IV   68       57     33.2    16.966    20.103
stage=unkn 71       45     37.9     1.346     1.642

 Chisq= 27.2  on 4 degrees of freedom, p= 1.78e-05
```

Peto & Peto modification of the Gehan-Wilcoxon test

With `rho = 1` it is equivalent to the Peto & Peto modification of the Gehan-Wilcoxon test.

```
survdiff(su_obj ~ stage, data = orca, rho = 1)
```

```
Call:
survdiff(formula = su_obj ~ stage, data = orca, rho = 1)

            N Observed Expected (O-E)^2/E (O-E)^2/V
stage=I    50     14.5     25.2     4.500     7.653
stage=II   77     29.3     39.3     2.549     4.954
stage=III  72     30.7     33.8     0.284     0.521
stage=IV   68     40.3     22.7    13.738    21.887
stage=unkn 71     32.0     25.9     1.438     2.359

 Chisq= 30.9  on 4 degrees of freedom, p= 3.22e-06
```

Different tests use different weights in comparing the survival functions depending on the failure times. In the actual example, they give comparable results, suggesting that the survival functions for different tumoral stage are different.

Modeling survival data

Non-parametric tests are particularly feasible when comparing survival functions across the levels of a factor. They are fairly robust, efficient, and usually simple/intuitive.
As the number of factors of interest increases, however, non-parametric tests become difficult to conduct and interpret. Regression models, instead, are more flexible for exploring the relationship between survival and predictors.

We will cover two different broad class of models: semi-parametric (i.e. proportional hazard) and parametric (accelerated failure time) models.

Cox PH model

A cox proportional hazards model assumes a baseline hazard function $\lambda_0(t)$, i.e. the hazard for the reference group ($Z_1, \dots, Z_p = 0$). Each predictor Z_i has a multiplicative effect on the hazard.

$$\lambda(t, Z) = \lambda_0(t) e^{Z\beta}$$

The semi-parametric nature of the Cox model is that the baseline rate may vary over time and it is not required to be estimated. The major assumption of the Cox model is that the hazard ratio for a predictor Z_i is constant (e^{β_i}) and does not depend on the time, i.e. the hazards in the two groups are proportional over time.

In our example we will consider modeling time to death as a function sex, age, and tumoral stage.
A cox proportional hazards model can be fitted using the `coxph()` function in the `survival` package.

```
m1 <- coxph(su_obj ~ sex + I((age-65)/10) + stage, data = orca)
summary(m1)
```

```
Call:
coxph(formula = su_obj ~ sex + I((age - 65)/10) + stage, data = orca)

  n= 338, number of events= 229

                    coef exp(coef) se(coef)     z Pr(>|z|)
sexMale          0.35139   1.42104  0.14139 2.485 0.012947
I((age - 65)/10) 0.41603   1.51593  0.05641 7.375 1.65e-13
stageII          0.03492   1.03554  0.24667 0.142 0.887421
stageIII         0.34545   1.41262  0.24568 1.406 0.159708
stageIV          0.88542   2.42399  0.24273 3.648 0.000265
stageunkn        0.58441   1.79393  0.25125 2.326 0.020016

                 exp(coef) exp(-coef) lower .95 upper .95
sexMale              1.421     0.7037    1.0771     1.875
I((age - 65)/10)     1.516     0.6597    1.3573     1.693
stageII              1.036     0.9657    0.6386     1.679
stageIII             1.413     0.7079    0.8728     2.286
stageIV              2.424     0.4125    1.5063     3.901
stageunkn            1.794     0.5574    1.0963     2.935

Concordance= 0.674  (se = 0.021 )
Rsquare= 0.226   (max possible= 0.999 )
Likelihood ratio test= 86.76  on 6 df,   p=1.11e-16
Wald test            = 80.5  on 6 df,   p=2.776e-15
Score (logrank) test = 82.86  on 6 df,   p=8.882e-16
```

We can check whether the data are sufficiently consistent with the assumption of proportional hazards with respect to each of the variables separately as well as globally, using the `cox.zph()` function.

```
cox.zph.m1 <- cox.zph(m1)
cox.zph.m1
```

```
                     rho    chisq     p
sexMale          -0.00137 0.000439 0.983
I((age - 65)/10)  0.07539 1.393597 0.238
stageII          -0.04208 0.411652 0.521
stageIII         -0.06915 1.083755 0.298
stageIV          -0.10044 2.301780 0.129
stageunkn        -0.09663 2.082042 0.149
GLOBAL                NA 4.895492 0.557
```

No evidence against proportionality assumption could apparently be found.

Additional functions for exploring departure from the (hazards) proportionality assumption (`ggcoxzph`) and for diagnostic (`ggcoxdiagnostics`) are implemented in the `survminer` package.

```
ggcoxzph(cox.zph.m1)
```

Show Plot

Results from the Cox model suggested a significant effect of sex, age, and stage. In particular, every 10 year increase in age the mortality rate increased by 50%. The HR for all-cause mortality comparing men to women was 1.42. Moreover, no differences could be observed between stages I and II in the estimates. On the other hand, the group with stage unknown is a complex mixture of patients from various true stages. Therefore, it may be prudent to exclude these subjects from the data and to pool the first two stage groups into one.

```
orca2 <- orca %>%
  filter(stage != "unkn") %>%
  mutate(st3 = Relevel(droplevels(stage), list(1:2, 3, 4)))
m2 <- coxph(Surv(time, all) ~ sex + I((age-65)/10) + st3, data = orca2, ties = "breslow")
round(ci.exp(m2), 4)
```

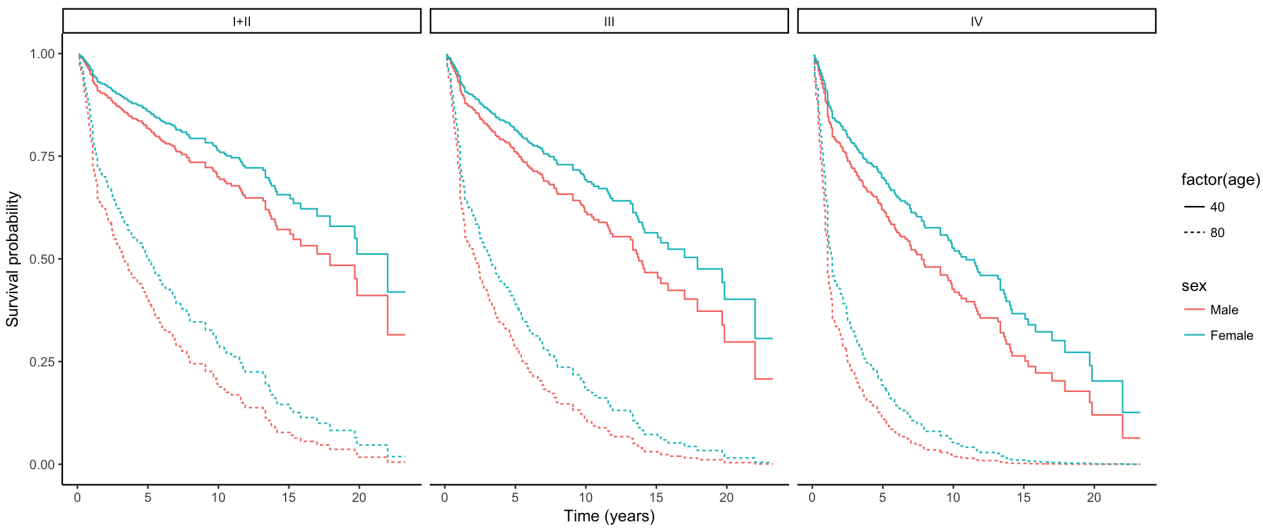
```
                exp(Est.)   2.5%  97.5%
sexMale             1.3284 0.9763 1.8074
I((age - 65)/10)    1.4624 1.2947 1.6519
st3III              1.3620 0.9521 1.9482
st3IV               2.3828 1.6789 3.3818
```

Let's plot the predicted survival curves by stage, fixing the values for sex and age (focusing only on 40 and 80 year old patients), based on the fitted model `m2` .
In order to do that, we first create a new artificial data frame containing the desired values for the covariates.

```
newd <- expand.grid(sex = c("Male", "Female"), age = c(40, 80), st3 = levels(orca2$st3))
newd$id <- 1:12
newd
```

```
      sex age  st3 id
1    Male  40 I+II  1
2  Female  40 I+II  2
3    Male  80 I+II  3
4  Female  80 I+II  4
5    Male  40  III  5
6  Female  40  III  6
7    Male  80  III  7
8  Female  80  III  8
9    Male  40   IV  9
10 Female  40   IV 10
11   Male  80   IV 11
12 Female  80   IV 12
```

```
fortify(survfit(m2, newdata = newd)) %>%
  gather(strata, surv, surv.1:surv.12) %>%
  mutate(id = gsub("surv.","", strata)) %>%
  merge(newd, by = "id") %>%
  ggplot(aes(x = time, y = surv, col = sex, linetype = factor(age))) +
  geom_step() + facet_grid(. ~ st3) +
  labs(x = "Time (years)", y = "Survival probability") + theme_classic()
```



AFT model

A parametric model assumes a distribution for the survival time. The model can be written as

$$\log(T) = \mu + Z\beta + \sigma W$$

We will consider a popular strategy, i.e. $W \sim \text{Weibull}(\lambda, \gamma)$

```
m2w <- flexsurvreg(Surv(time, all) ~ sex + I((age-65)/10) + st3, data = orca2, dist = "weibull")
m2w
```

```
Call:
flexsurvreg(formula = Surv(time, all) ~ sex + I((age - 65)/10) +
    st3, data = orca2, dist = "weibull")

Estimates:
                  data mean  est       L95%      U95%      se        exp(est)  L95%
shape                   NA    0.93268   0.82957   1.04861   0.05575        NA        NA
scale                   NA   13.53151   9.97582  18.35456   2.10472        NA        NA
sexMale            0.53184   -0.33905  -0.66858  -0.00951   0.16813   0.71245   0.51243
I((age - 65)/10)  -0.15979   -0.41836  -0.54898  -0.28773   0.06665   0.65813   0.57754
st3III             0.26966   -0.32567  -0.70973   0.05839   0.19595   0.72204   0.49178
st3IV              0.25468   -0.95656  -1.33281  -0.58030   0.19197   0.38421   0.26374
                  U95%
shape                   NA
scale                   NA
sexMale            0.99053
I((age - 65)/10)   0.74996
st3III             1.06012
st3IV              0.55973

N = 267,  Events: 184,  Censored: 83
Total time at risk: 1620.864
Log-likelihood = -545.858, df = 6
AIC = 1103.716
```

The interpretation of the coefficients is in terms of t. For example, men have a 30% $(100 \cdot (1 - 0.71))$ reduction in the survival time compared to women, adjusting for age and tumoral stage. Every ten years increase in age are associated with a 35% reduction in survival times, adjusting for sex and tumoral stage.

It can be shown that an AFT models assuming an exponential or a Weibull distribution can be reparametrized as proportional hazards models (with baseline hazard from the exponential/Weibull family of distributions).
This can be shown using the `weibreg()` function in the `eha` package.

```
m2wph <- weibreg(Surv(time, all) ~ sex + I((age-65)/10) + st3, data = orca2)
summary(m2wph)
```

```
Call:
weibreg(formula = Surv(time, all) ~ sex + I((age - 65)/10) +
    st3, data = orca2)

Covariate           Mean       Coef Exp(Coef)  se(Coef)    Wald p
sex
          Female    0.490     0         1           (reference)
            Male    0.510     0.316     1.372     0.156     0.043
I((age - 65)/10)   -0.522     0.390     1.477     0.062     0.000
st3
             I+II   0.551     0         1           (reference)
              III   0.287     0.304     1.355     0.182     0.095
               IV   0.162     0.892     2.440     0.178     0.000

log(scale)                    2.605    13.532     0.156     0.000
log(shape)                   -0.070     0.933     0.060     0.244

Events                    184
Total time at risk        1620.9
Max. log. likelihood      -545.86
LR test statistic         68.7
Degrees of freedom        4
Overall p-value           4.30767e-14
```

The (exponential of the) coefficients have an equivalent interpretation to the coefficients of a Cox proportional model (the estimates are similar as well).

Any function of the parameters of a fitted model can be summarized or plotted by supplying the argument `fn` to the `summary` or `plot` methods. For example, median survival under the Weibull model can be summarized as

```
median.weibull <- function(shape, scale) qweibull(0.5, shape = shape, scale = scale)
set.seed(2153)
newd <- data.frame(sex = c("Male", "Female"), age = 65, st3 = "I+II")
summary(m2w, newdata = newd, fn = median.weibull, t = 1, B = 10000)
```

```
sex=Male, I((age - 65)/10)=0, st3=I+II
  time      est      lcl      ucl
1    1 6.507834 4.898889 8.631952

sex=Female, I((age - 65)/10)=0, st3=I+II
  time      est      lcl      ucl
1    1 9.134466 6.801322 12.33771
```

Compare the results with those from the Cox model.

```
survfit(m2, newdata = newd)
```

```
Call: survfit(formula = m2, newdata = newd)

    n events median 0.95LCL 0.95UCL
1 267    184   7.00    5.25    10.6
2 267    184   9.92    7.33    13.8
```

Poisson regression

It can be shown that the Cox model is mathematically equivalent to a Poisson regression model on a particular transformation of the data.
The idea is to split the follow-up time every time an event is observed in such a way every time interval contains only one event. In this augmented dataset subjects may be represented several times (multiple rows).

We first define the unique time where we observed an event (`all == 1`) and use the `survSplit()` function in the `survival` package to create the augmented or splitted data.

```
cuts <- sort(unique(orca2$time[orca2$all == 1]))
orca_splitted <- survSplit(Surv(time, all) ~ ., data = orca2, cut = cuts, episode = "tgroup")
head(orca_splitted, 15)
```

Show Output

The `gnm()` function in the `gnm` package fits a conditional Poisson on splitted data, where the effects of the time (as a factor variable) can be marginalized (not estimated to improve computational efficiency).

```
mod_poi <- gnm(all ~ sex + I((age-65)/10) + st3, data = orca_splitted,
               family = poisson, eliminate = factor(time))
summary(mod_poi)
```

Show Output

Compare the estimates obtained from the conditional Poisson with the cox proportional hazard model.

```
round(data.frame(cox = ci.exp(m2), poisson = ci.exp(mod_poi)), 4)
```

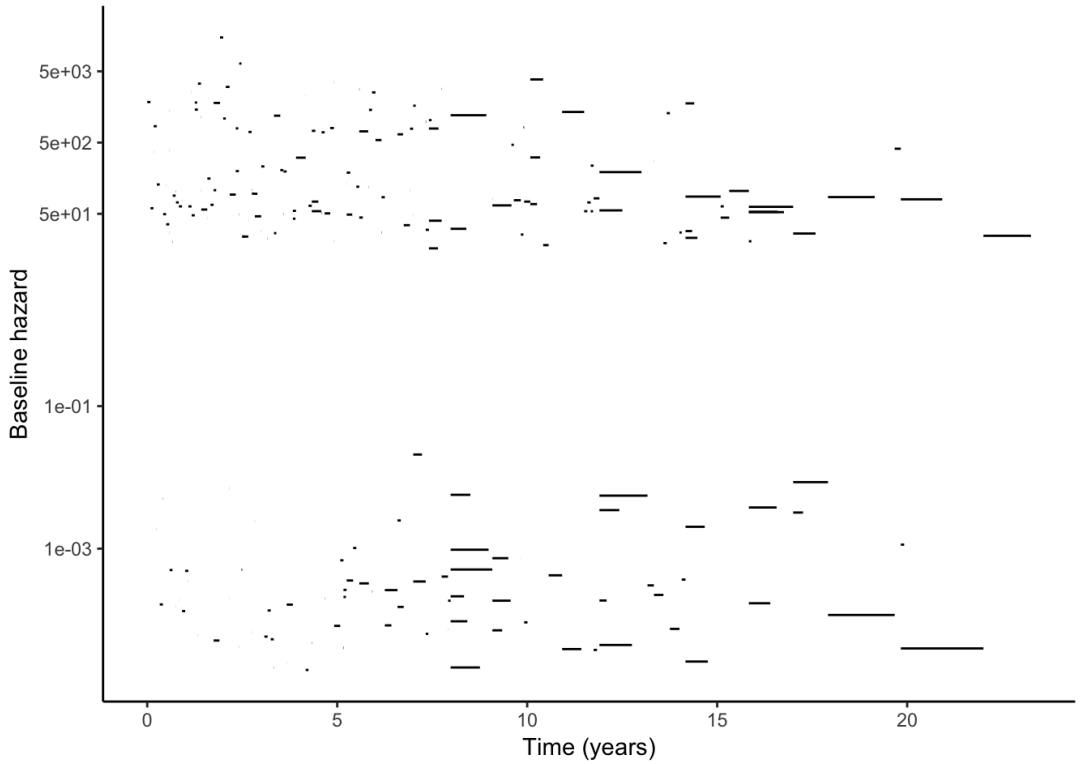
```
                cox.exp.Est.. cox.2.5. cox.97.5. poisson.exp.Est.. poisson.2.5. poisson.97.5.
sexMale                1.3284   0.9763    1.8074            1.3284       0.9763        1.8074
I((age - 65)/10)       1.4624   1.2947    1.6519            1.4624       1.2947        1.6519
st3III                 1.3620   0.9521    1.9482            1.3620       0.9521        1.9482
st3IV                  2.3828   1.6789    3.3818            2.3828       1.6789        3.3818
```

If we want to estimate the baseline hazard, we need also to estimate the effects of time in the Poisson model (OBS we also need to include the (log) length of the time intervals as offset).

```
orca_splitted$dur <- with(orca_splitted, time - tstart)
mod_poi2 <- glm(all ~ -1 + factor(time) + sex + I((age-65)/10) + st3,
                data = orca_splitted, family = poisson, offset = log(dur))
```

The baseline hazard consists of a step function, where the rate is constant in each time interval.

```
newd <- data.frame(time = unique(orca2$time), dur = 1,
                   sex = "Female", age = 65, st3 = "I+II")
blhaz <- 1000*data.frame(ci.pred(mod_poi2, newdata = newd))
xint <- unique(cbind(orca_splitted$tstart, orca_splitted$time))
ggplot(blhaz, aes(x = xint[, 1], y = Estimate, xend = xint[, 2], yend = Estimate)) + geom_segment(
) +
  scale_y_continuous(trans = "log", breaks = c(.001, .1, 50, 500, 5000)) +
  theme_classic() + labs(x = "Time (years)", y = "Baseline hazard")
```

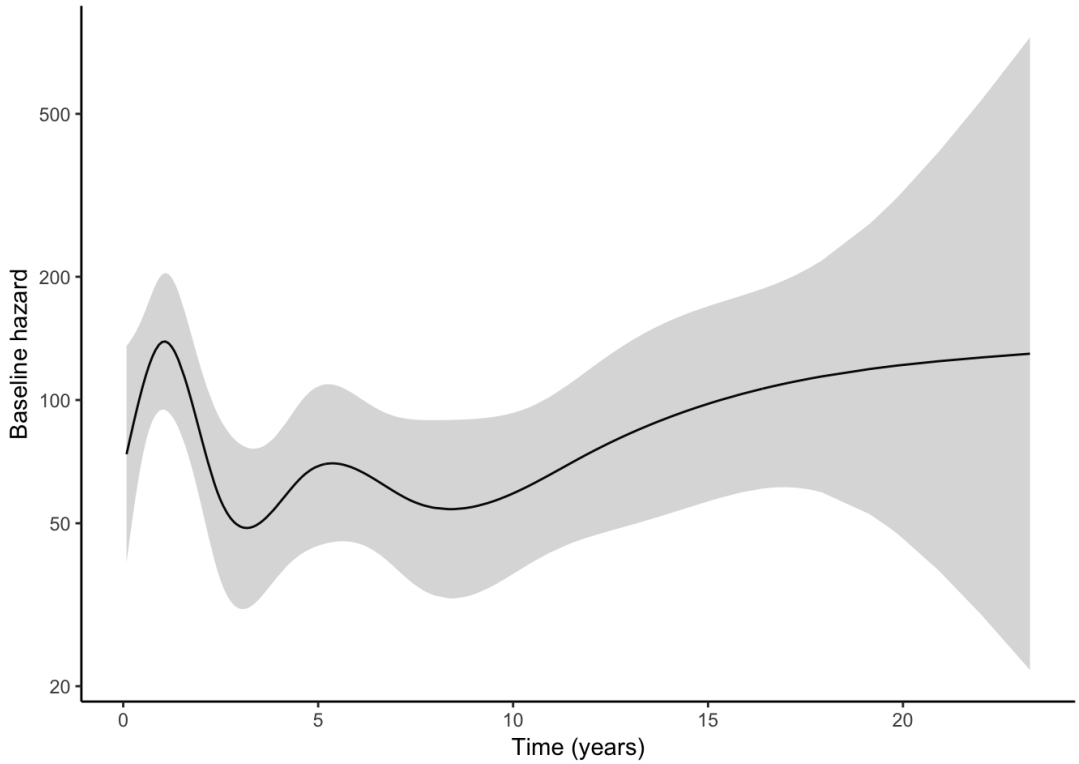


A better approach would be to flexibly model the baseline hazard by using, for instance, splines with knots k.

```
k <- quantile(orca2$time, 1:5/6)
mod_poi2s <- glm(all ~ ns(time, knots = k) + sex + I((age-65)/10) + st3,
                 data = orca_splitted, family = poisson, offset = log(dur))
round(ci.exp(mod_poi2s), 3)
```

```
                    exp(Est.)  2.5%  97.5%
(Intercept)             0.074 0.040  0.135
ns(time, knots = k)1    0.402 0.177  0.912
ns(time, knots = k)2    1.280 0.477  3.432
ns(time, knots = k)3    0.576 0.220  1.509
ns(time, knots = k)4    1.038 0.321  3.358
ns(time, knots = k)5    4.076 0.854 19.452
ns(time, knots = k)6    1.040 0.171  6.314
sexMale                 1.325 0.975  1.801
I((age - 65)/10)        1.469 1.300  1.659
st3III                  1.360 0.952  1.942
st3IV                   2.361 1.665  3.347
```

```
blhazs <- 1000*data.frame(ci.pred(mod_poi2s, newdata = newd))
ggplot(blhazs, aes(x = newd$time, y = Estimate)) + geom_line() +
  geom_ribbon(aes(ymin = X2.5., ymax = X97.5.), alpha = .2) +
  scale_y_continuous(trans = "log", breaks = c(20, 50, 100, 200, 500, 1000)) +
   theme_classic() + labs(x = "Time (years)", y = "Baseline hazard")
```



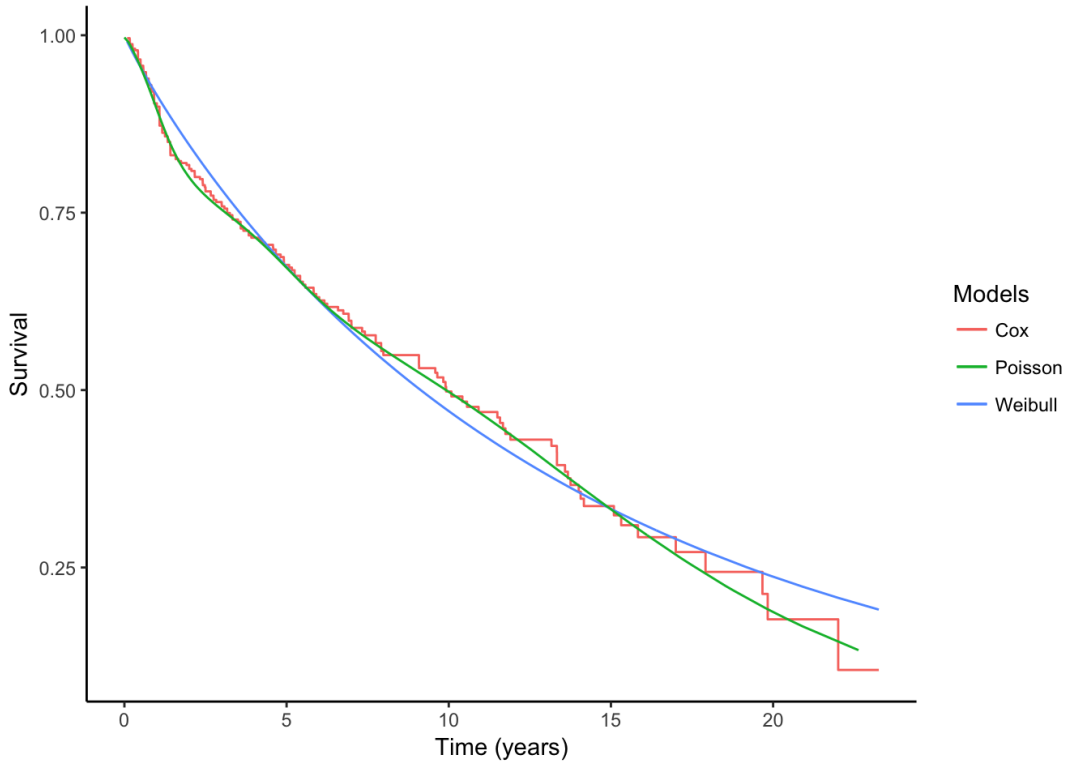
Comparison of different strategies

We can compare the previous strategies based on the predicted survival curves for a specific covariate patters, saying 65 years-old women with tumoral stage I or II.

```
newd <- data.frame(sex = "Female", age = 65, st3 = "I+II")
surv_cox <- fortify(survfit(m2, newdata = newd))
surv_weibull <- summary(m2w, newdata = newd, tidy = TRUE)
## For the poisson model we need some extra steps
tbmid <- sort(unique(.5*(orca_splitted$tstart + orca_splitted$time)))
mat <- cbind(1, ns(tbmid, knots = k), 0, 0, 0, 0)
Lambda <- ci.cum(mod_poi2s, ctr.mat = mat, intl = diff(c(0, tbmid)))
surv_poisson <- data.frame(exp(-Lambda))
```

A graphical representation of the survival function facilitates the comparison.

```
ggplot(surv_cox, aes(time, surv)) + geom_step(aes(col = "Cox")) +
  geom_line(data = surv_weibull, aes(y = est, col = "Weibull")) +
  geom_line(data = surv_poisson, aes(x = c(0, tbmid[-1]), y = Estimate, col = "Poisson")) +
  labs(x = "Time (years)", y = "Survival", col = "Models") + theme_classic()
```



Get interactive! Shiny tutorial (https://shiny.rstudio.com/tutorial/).

Show Source

Additional analyses

Non-linearity

We have assumed that the effect of age on the (log) mortality rate is linear. A possible strategy to relax this assumption is fit a Cox model where age is modeled with a quadratic effect.

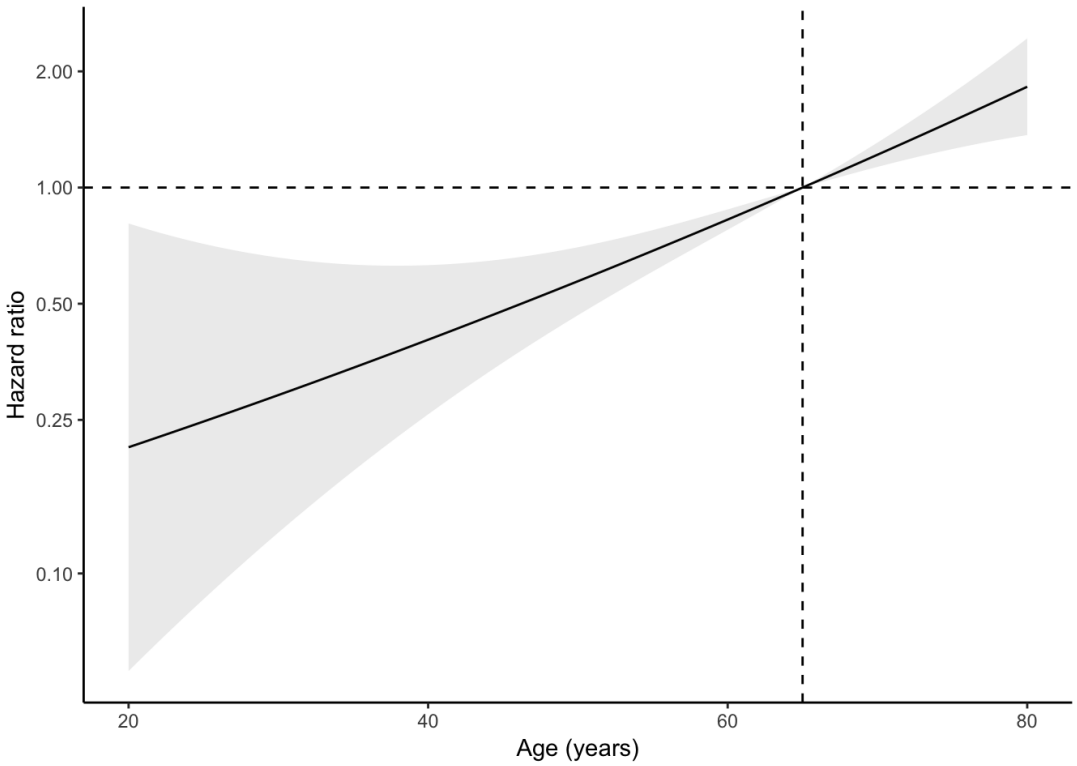
```
m3 <- coxph(Surv(time, all) ~ sex + I(age-65) + I((age-65)^2) + st3, data = orca2)
summary(m3)
```

```
Call:
coxph(formula = Surv(time, all) ~ sex + I(age - 65) + I((age -
    65)^2) + st3, data = orca2)

  n= 267, number of events= 184

                    coef exp(coef)  se(coef)     z Pr(>|z|)
sexMale         2.903e-01 1.337e+00 1.591e-01 1.825   0.0681
I(age - 65)     3.868e-02 1.039e+00 6.554e-03 5.902 3.59e-09
I((age - 65)^2) 9.443e-05 1.000e+00 3.576e-04 0.264   0.7917
st3III          3.168e-01 1.373e+00 1.838e-01 1.724   0.0847
st3IV           8.691e-01 2.385e+00 1.787e-01 4.863 1.16e-06

                exp(coef) exp(-coef) lower .95 upper .95
sexMale             1.337     0.7481    0.9787     1.826
I(age - 65)         1.039     0.9621    1.0262     1.053
I((age - 65)^2)     1.000     0.9999    0.9994     1.001
st3III              1.373     0.7284    0.9576     1.968
st3IV               2.385     0.4193    1.6801     3.385

Concordance= 0.674  (se = 0.024 )
Rsquare= 0.216   (max possible= 0.999 )
Likelihood ratio test= 64.89  on 5 df,   p=1.183e-12
Wald test            = 63.11  on 5 df,   p=2.756e-12
Score (logrank) test = 67.64  on 5 df,   p=3.176e-13
```

The p-value for non-linearity (i.e. quadratic term) is high and thus there is no evidence to reject the null hypothesis (i.e. the linearity assumption is appropriate).

If the relation would be non-linear, the coefficients for age are no longer directly interpretable. We can instead present the HR graphically as a function of age. We need to specify a referent value; we chose the median age value of 65 years old.

```
age <- seq(20, 80, 1) - 65
hrtab <- ci.exp(m3, ctr.mat = cbind(0, age, age^2, 0, 0))
ggplot(data.frame(hrtab), aes(x = age+65, y = exp.Est.., ymin = X2.5., ymax = X97.5.)) +
  geom_line() + geom_ribbon(alpha = .1) +
  scale_y_continuous(trans = "log", breaks = c(.1, .25, .5, 1, 2)) +
  labs(x = "Age (years)", y = "Hazard ratio") + theme_classic() +
  geom_vline(xintercept = 65, lty = 2) + geom_hline(yintercept = 1, lty = 2)
```



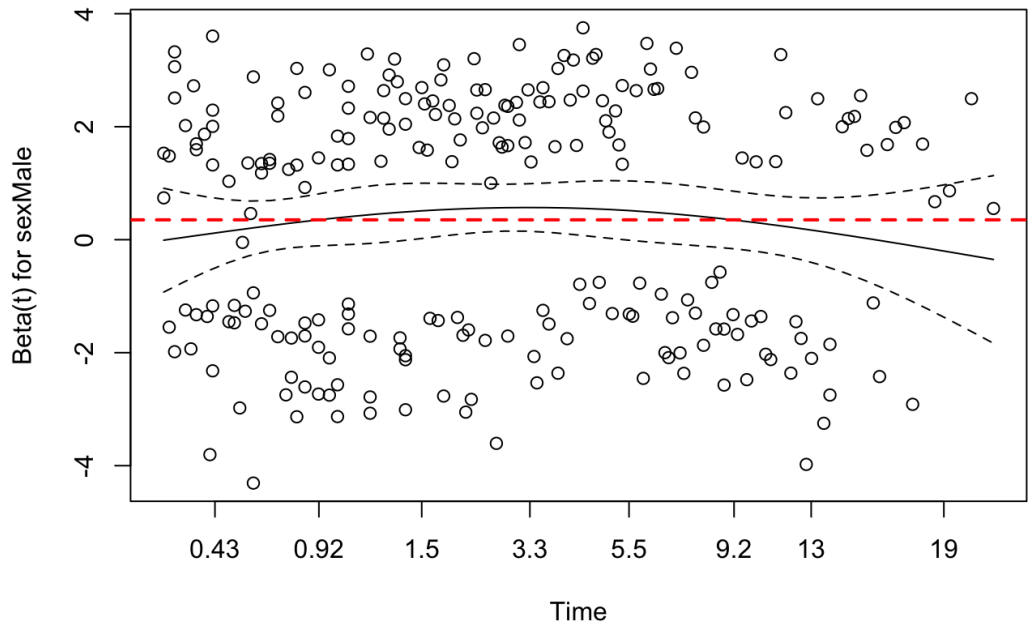
Time dependent coefficients

Let's consider the Cox proportional hazard model fitted and saved in the object `m1`. As already mentioned, the main assumption is that the effect of a predictor, e.g. sex, is constant over time.

$$\lambda(t, Z) = \lambda_0(t)e^{Z\beta}$$

The `cox.zph()` function is useful to plot the effect of individual predictors over time, and is thus used to diagnose and understand non-proportional hazards.

```
plot(cox.zph.m1[1])
abline(h= m1$coef[1], col = 2, lty = 2, lwd = 2)
```



We can relax the proportional hazard assumption by fitting a step function for $\beta(t)$, which implies different βs over different time intervals.

$$\lambda(t, Z) = \lambda_0(t)e^{Z\beta(t)}$$

The `survSplit()` function in the `survival` package breaks the data set into time dependent parts. Let's consider as time breaks 5 and 15.

```
orca3 <- survSplit(Surv(time, all) ~ ., data = orca2, cut = c(5, 15), episode = "tgroup")
head(orca3)
```

```
  id    sex      age stage          event  st3 tstart  time all tgroup
1  2 Female 83.08783   III Oral ca. death  III      0 0.419   1      1
2  3   Male 52.59008    II    Other death I+II      0 5.000   0      1
3  3   Male 52.59008    II    Other death I+II      5 7.915   1      2
4  4   Male 77.08630     I    Other death I+II      0 2.480   1      1
5  5   Male 80.33622    IV Oral ca. death   IV      0 2.500   1      1
6  6 Female 82.58132    IV    Other death   IV      0 0.167   1      1
```

```
m3 <- coxph(Surv(tstart, time, all) ~ relevel(sex, 2):strata(tgroup) + I((age-65)/10) + st3, data
= orca3)
m3
```

```
Call:
coxph(formula = Surv(tstart, time, all) ~ relevel(sex, 2):strata(tgroup) +
    I((age - 65)/10) + st3, data = orca3)

                                          coef exp(coef) se(coef)     z       p
I((age - 65)/10)                        0.3818    1.4650   0.0626  6.10 1.0e-09
st3III                                  0.2886    1.3345   0.1839  1.57   0.117
st3IV                                   0.8758    2.4008   0.1796  4.88 1.1e-06
relevel(sex, 2)Male:strata(tgroup)tgroup=1    0.4208    1.5231   0.1905  2.21   0.027
relevel(sex, 2)Female:strata(tgroup)tgroup=1      NA        NA   0.0000    NA      NA
relevel(sex, 2)Male:strata(tgroup)tgroup=2   -0.1027    0.9024   0.2812 -0.37   0.715
relevel(sex, 2)Female:strata(tgroup)tgroup=2      NA        NA   0.0000    NA      NA
relevel(sex, 2)Male:strata(tgroup)tgroup=3    1.1319    3.1014   1.0944  1.03   0.301
relevel(sex, 2)Female:strata(tgroup)tgroup=3      NA        NA   0.0000    NA      NA

Likelihood ratio test=68.1  on 6 df, p=1.02e-12
n= 416, number of events= 184
```

Although not significant, the hazard ratio comparing male to female is lower than 1 for the second time period (between 5 and 15 years) while it's higher than one for the other two time periods. The `cox.zph()` function can be used to check if there are still departure from the proportionality assumption on the splitted analysis.

Competing risk

Oftentimes the main interest if on the risk or hazard of dying from one specific cause. The cause-specific event may not be observed because of a competing cause which prevents the subject to develop the event. Competing events occur not only for cause-specific mortality but more in general every time an event prevents a concurrent event to happen.
In our example we are interested in modeling the risk of mortality from oral cancer, and dying from other causes will be considered as competing event.

In a competing risk scenario, event-specific survival obtained censoring the other event (aka naive Kaplan–Meier estimates of cause-specific survival) are generally not appropriate.
We will consider instead the cumulative incidence function (CIF) for the event c

$$F_c(t) = P(T \leq t \text{ and } C = c)$$

From these, it is possible to recover the CDF of event-free survival time T, i.e. cumulative risk of any event by t:
$F(t) = \sum_c F_c(t)$
and the event-free survival function, i.e. probability of avoiding all events by t: $S(t) = 1 - F(t)$

The CIF (risk of event c over risk period $[0, t]$ in the presence of competing risks) can also be obtained as

$$F_c(t) = \int_0^t \lambda_c(v) S(v) dv$$

Depends on the hazard of the competing event as well

$$S(t) = \exp\left(- \int_0^t [\lambda_1(v) + \lambda_2(v)] dv \right)$$

The hazard of the subdistribution is defined as

$$\gamma_c(t) = f_c(t)/[1 - F_c(t)]$$

and is not the same as $\lambda_c(t) = f_c(t)/[1 - F(t)]$

CIF

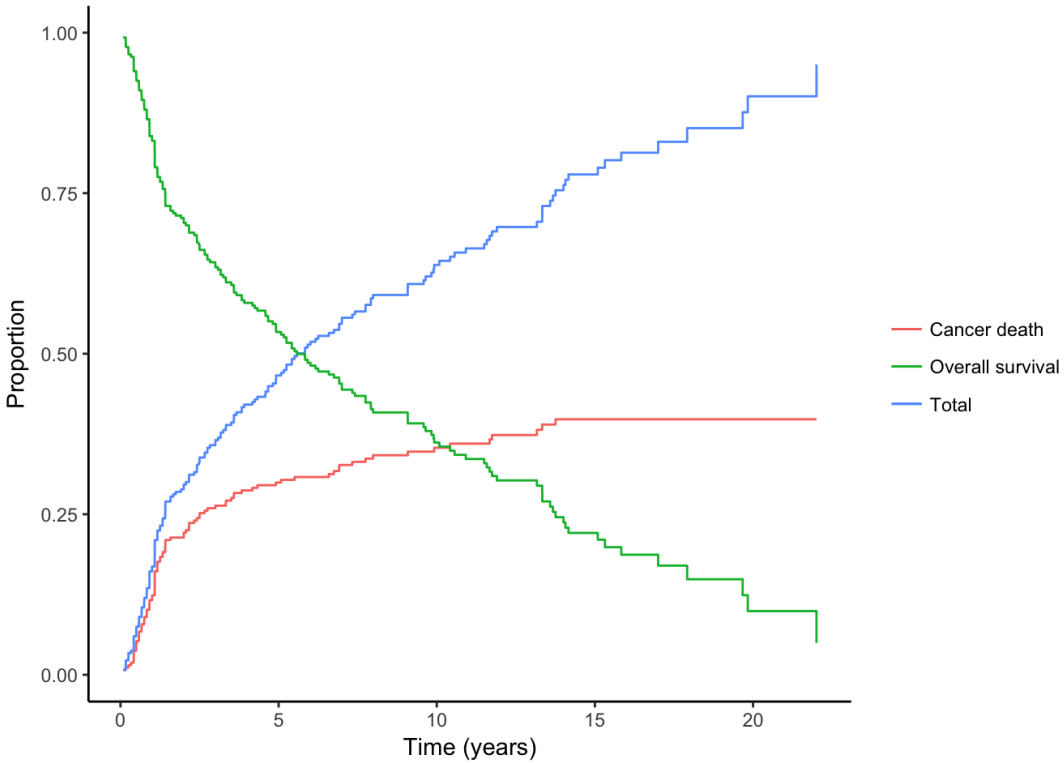
The `Cuminc()` function in the `mstate` package calculates non-parametric CIF (aka Aalen–Johansen estimates) and associated standard errors for the competing events.

```
cif <- Cuminc(time = "time", status = "event", data = orca2)
head(cif)
```

```
   time      Surv CI.Oral ca. death CI.Other death      seSurv seCI.Oral ca. death
1 0.085 0.9925094       0.007490637    0.000000000 0.005276805         0.005276805
2 0.162 0.9887640       0.011235955    0.000000000 0.006450534         0.006450534
3 0.167 0.9812734       0.011235955    0.007490637 0.008296000         0.006450534
4 0.170 0.9775281       0.011235955    0.011235955 0.009070453         0.006450534
5 0.249 0.9737828       0.011235955    0.014981273 0.009778423         0.006450534
6 0.252 0.9662921       0.014981273    0.018726592 0.011044962         0.007434315
  seCI.Other death
1      0.000000000
2      0.000000000
3      0.005276805
4      0.006450534
5      0.007434315
6      0.008296000
```

We can plot the CIF (one stacked of the other) together with the derived event-free survival function.

```
ggplot(cif, aes(time)) +
  geom_step(aes(y = `CI.Oral ca. death`, colour = "Cancer death")) +
  geom_step(aes(y = `CI.Oral ca. death`+ `CI.Other death`, colour = "Total")) +
  geom_step(aes(y = Surv, colour = "Overall survival")) +
  labs(x = "Time (years)", y = "Proportion", colour = "") +
  theme_classic()
```

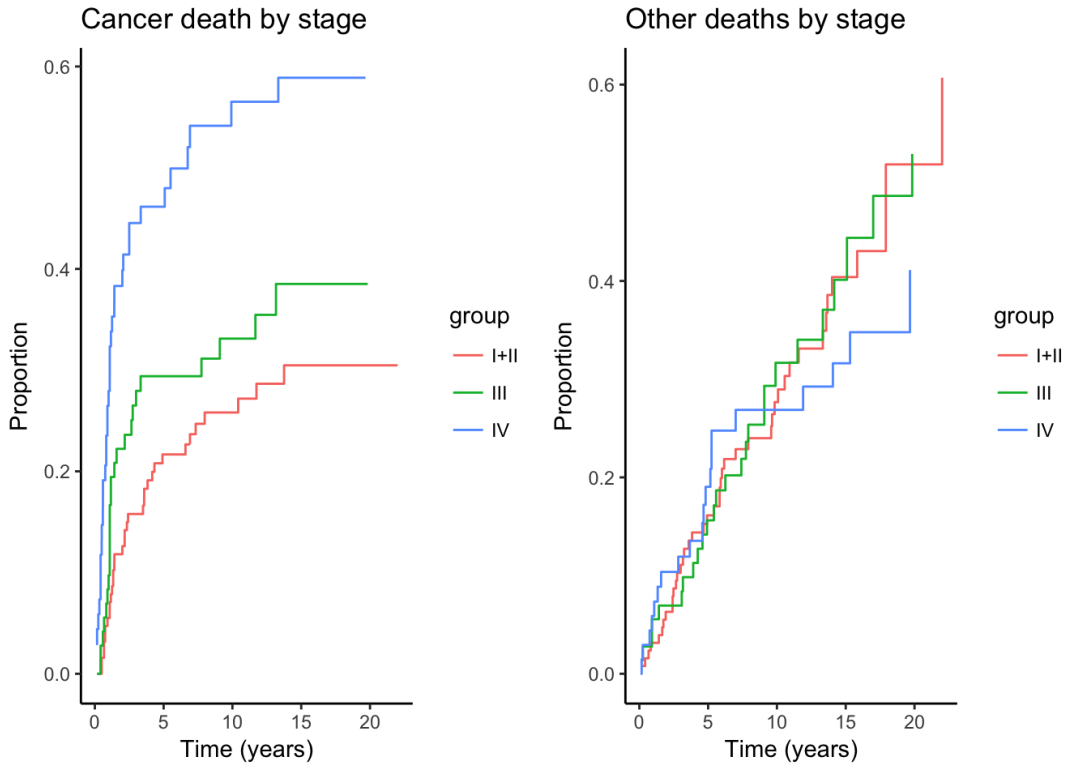


Extensions have been implemented to estimate the cumulative incidences functions by the levels of a factor variable, e.g by stage in 3 levels (`st3`) for both causes of death.

```
cif_stage <- Cuminc(time = "time", status = "event", group = "st3", data = orca2)
cif_stage %>%
  group_by(group) %>%
  do(head(., n = 3))
```

```
# A tibble: 9 x 8
# Groups:   group [3]
   group  time      Surv `CI.Oral ca. death` `CI.Other death`      seSurv
  <fctr> <dbl>     <dbl>               <dbl>            <dbl>       <dbl>
1   I+II 0.170 0.9921260          0.00000000      0.007874016 0.007842954
2   I+II 0.419 0.9842520          0.00000000      0.015748031 0.011047510
3   I+II 0.498 0.9685039          0.01574803      0.015748031 0.015498047
4    III 0.167 0.9861111          0.00000000      0.013888889 0.013792101
5    III 0.249 0.9722222          0.00000000      0.027777778 0.019367130
6    III 0.413 0.9583333          0.01388889      0.027777778 0.023549757
7     IV 0.085 0.9705882          0.02941176      0.000000000 0.020489134
8     IV 0.162 0.9558824          0.04411765      0.000000000 0.024903130
9     IV 0.167 0.9411765          0.04411765      0.014705882 0.028533603
# ... with 2 more variables: `seCI.Oral ca. death` <dbl>, `seCI.Other death` <dbl>
```

```
grid.arrange(
  ggplot(cif_stage, aes(time)) +
    geom_step(aes(y = `CI.Oral ca. death`, colour = group)) +
    labs(x = "Time (years)", y = "Proportion", title = "Cancer death by stage") +
    theme_classic(),
  ggplot(cif_stage, aes(time)) +
    geom_step(aes(y = `CI.Other death`, colour = group)) +
    labs(x = "Time (years)", y = "Proportion", title = "Other deaths by stage") +
    theme_classic(),
  ncol = 2
)
```



We can see that the CIF foo oral cancer death for stage IV are higher as compared to III, and even more to I+II. For other cause mortality, instead, the curves do not seem to vary according to tumoral stage.

When we want to model survival data in a competing risk setting, there are two common strategies that address different questions:
- Cox model for event-specific hazards, when e.g. the interest is in the biological effect of the prognostic factors on the fatality of the very disease that often leads to the relevant outcome.
- Fine–Gray model for the hazard of the subdistribution when we want to assess the impact of the factors on the overall cumulative incidence of event c.

Cox model for competing risk

```
m2haz1 <- coxph(Surv(time, event == "Oral ca. death") ~ sex + I((age-65)/10) + st3, data = orca2)
round(ci.exp(m2haz1), 4)
```

```
                exp(Est.)   2.5%  97.5%
sexMale            1.0171 0.6644 1.5569
I((age - 65)/10)   1.4261 1.2038 1.6893
st3III             1.5140 0.9012 2.5434
st3IV              3.1813 1.9853 5.0978
```

```
m2haz2 <- coxph(Surv(time, event == "Other death") ~ sex + I((age-65)/10) + st3, data = orca2)
round(ci.exp(m2haz2), 4)
```

```
                exp(Est.)   2.5%  97.5%
sexMale            1.8103 1.1528 2.8431
I((age - 65)/10)   1.4876 1.2491 1.7715
st3III             1.2300 0.7488 2.0206
st3IV              1.6407 0.9522 2.8270
```

The results of the cause-specific Cox models agree with the graphical presentation of the cause-specific CIFs, i.e. tumoral stage IV to be a significant risk factor only for oral cancer mortality. Increasing levels of age are associated with higher mortality rates for both causes (HR = 1.42 for oral cancer mortality, HR = 1.48 for mortality from other causes). Differences according to gender are observed only for other cause mortality (HR = 1.8).

Fine–Gray model

The `crr()` function in the `cmprsk` package can be used for regression modeling of subdistribution functions in case of competing risks. We present the results for the Fine–Gray model for the hazard of the subdistribution for both oral cancer deaths and other cause deaths with the same covariates as above.

```
m2fg1 <- with(orca2, crr(time, event, cov1 = model.matrix(m2), failcode = "Oral ca. death"))
summary(m2fg1, Exp = T)
```

```
Competing Risks Regression

Call:
crr(ftime = time, fstatus = event, cov1 = model.matrix(m2), failcode = "Oral ca. death")

                   coef exp(coef) se(coef)      z p-value
sexMale          -0.0953     0.909    0.213 -0.447 6.5e-01
I((age - 65)/10)  0.2814     1.325    0.093  3.024 2.5e-03
st3III            0.3924     1.481    0.258  1.519 1.3e-01
st3IV             1.0208     2.775    0.233  4.374 1.2e-05

                 exp(coef) exp(-coef)  2.5% 97.5%
sexMale              0.909      1.100 0.599  1.38
I((age - 65)/10)     1.325      0.755 1.104  1.59
st3III               1.481      0.675 0.892  2.46
st3IV                2.775      0.360 1.757  4.39

Num. cases = 267
Pseudo Log-likelihood = -501
Pseudo likelihood ratio test = 31.4  on 4 df,
```

```
m2fg2 <- with(orca2, crr(time, event, cov1 = model.matrix(m2), failcode = "Other death"))
summary(m2fg2, Exp = T)
```

```
Competing Risks Regression

Call:
crr(ftime = time, fstatus = event, cov1 = model.matrix(m2), failcode = "Other death")

                   coef exp(coef) se(coef)      z p-value
sexMale           0.544     1.723   0.2342  2.324   0.020
I((age - 65)/10)  0.197     1.218   0.0807  2.444   0.015
st3III            0.130     1.139   0.2502  0.521   0.600
st3IV            -0.212     0.809   0.2839 -0.748   0.450

                 exp(coef) exp(-coef)  2.5% 97.5%
sexMale              1.723      0.580 1.089  2.73
I((age - 65)/10)     1.218      0.821 1.040  1.43
st3III               1.139      0.878 0.698  1.86
st3IV                0.809      1.237 0.464  1.41

Num. cases = 267
Pseudo Log-likelihood = -471
Pseudo likelihood ratio test = 9.43  on 4 df,
```

References

- Läärä E, Korpi JT, Pitkänen H, Alho OP, Kantola S. Competing risks analysis of cause-specific mortality in patients with oral squamous cell carcinoma. Head & neck. 2017 Jan 1;39(1):56-62.
- Survival data analysis course by Michal Pešta http://www.karlin.mff.cuni.cz/~pesta/NMFM404/survival.html (http://www.karlin.mff.cuni.cz/~pesta/NMFM404/survival.html)
- Kleinbaum DG, Klein M. Survival analysis: a self-learning text. Springer Science & Business Media; 2006 Jan 2.
- Jackson CH. flexsurv: a platform for parametric survival modelling in R. Journal of Statistical Software. 2016 May 1;70(8):1-33.
- Therneau, T.M. and Grambsch, P.M., 2013. Modeling survival data: extending the Cox model. Springer Science & Business Media.
- Carstensen B. Who needs the Cox model anyway. Life. 2004;3:46.

- Xu S, Gargiullo P, Mullooly J, McClure D, Hambidge SJ, Glanz J. Fitting parametric and semi-parametric conditional Poisson regression models with Cox's partial likelihood in self-controlled case series and matched cohort studies. J Data Sci. 2010;8:349-60.
- Therneau T, Crowson C, Atkinson E. Using time dependent covariates and time dependent coefficients in the cox model. Survival Vignettes. 2016 Oct 29.