Survival Analysis

Contents

1 Introduction

1.1 Definition

The **Survival analysis** also known as time-to-event analysis corresponds to a set of statistical approaches used to **predict the time until an event of interest occurs**.

Most important concepts for this analysis:

- **Exposure** (clock starts), time zero of the analysis for this subject.
- **Event** (clock stops),
- **Survival time or time-to-event**, (difference between time of Event and time of Exposure).

A **Survival time** can be estimated in this examples:

- **Exposure** (Cancer diagnostic), Event (Death)
- **Exposure** (Marriage), Event (Divorce)

The survival analysis is a method of descriptive statistics to study pre-existing data.
A **sample of observations** is defined to represent a **population**, for each sample, we will have a value of Exposure, Event, and Time to event. For example a group of 100 patients (observations) with cancer, are studied.
The Clock starts at the cancer diagnosis for each patient, and stops at the Event, the moment of death in this case.

1.2 Censoring and truncation

Censoring: Censoring is the term used when we don't know the exact time-to-event for an included observation.

There are 2 main types of censoring:

- **Right censoring** (time to event greater than a value): this type of censoring could sometimes be informative when the censoring is actually correlated to the expected event (consequence). An example of right censoring is a time-to-event that continues after the experiment (patient that is still alive a the end of the study).
- **Left censoring** (time to event less than a value) : This corresponds to an unknown Exposure time, the clock has already started therefore the time to event has a variability because of this unknown.

Truncation : Truncation is the term used for the removal of observation based on outliers time-to event values (time-to-event too low or too high).

There are 2 different types of truncation :

- **left truncation** (short time-to-event values, small values that where not measured)
- **right truncation** (long time-to-event values, large values that where not measured)

The whole data set can be truncated, whereas data points can be censored.

2 Survival function, Hazard & Hazard ratio

2.1 The Survival function/Model

survival function $S(t) = P(T > t)$ = Probability of Survival **beyond time t**.

2.2 Hazard

$Hazard(Haz) = P(T < t + d \,|\, T > t)$ = probability of dying in the next few seconds **given alive now**.

For the Exponential survival model, the hazard function correspond to the rate of the exponential curve.

2.3 Hazard Ratio

$Hazard\,(HR) = \frac{Haz,x=1}{Haz,x=0}$= relative ratio,

At a given instant in time someone who is exposed is "relative ratio" times more important to someone who is not.

2.4 Different models

Two type of functions/models to illustrate the decrease in survival probability.

S(t) is the survival function

- **Kaplan-Meier survival model** (non-parametric)
  - Pros : Simple to interpret, can estimate S(t)
  - Cons : No functional form (no mathematical function, because of steps), can not estimate hazard ratio
- **Exponential survival model** (parametric)
  - Pros : Can estimate the S(t), and Hazard ratio
  - Cons : Not always realistic, because assumes constant hazard (death is not constant)
- **Cox proportional Hazard model** (semi-parametric), sort of a combination of KM model and Exponential model
  - Pros : Haz can fluctuate with time, Can estimate Hazard ratio
  - Cons : Can not estimate S(t)

2.5 (to change location), Equation

Y = time-to-event

Where the Y = outcome, depends on *Time* and on the *Event* (0 = NO, 1 = YES) :

- 0 : the event didn't occured
- 1 : the event occured

3 Survival analysis in R

3.1 Lung data set

```r
library(tidyverse); library(survival); library(survminer); library(knitr)

head(lung) %>% kable()
```

inst	time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss
3	306	2	74	1	1	90	100	1175	NA
3	455	2	68	1	0	90	90	1225	15
3	1010	1	56	1	0	90	90	NA	15
5	210	2	57	1	1	90	60	1150	11
1	883	2	60	1	0	100	90	NA	0
12	1022	1	74	1	1	50	80	513	0

Data set description:

- inst: Institution code
- time: Survival time in days
- status: censoring status 1=censored, 2=dead
- age: Age in years
- sex: Male=1 Female=2
- ph.ecog: ECOG performance score (0=good 5=dead)
- ph.karno: Karnofsky performance score (bad=0-good=100) rated by physician
- pat.karno: Karnofsky performance score as rated by patient
- meal.cal: Calories consumed at meals
- wt.loss: Weight loss in last six months

3.2 Compute survival curves with survfit()

The function survfit() [in survival package] can be used to compute kaplan-Meier survival estimate. Its main arguments include:

- a survival object created using the function Surv()
- and the data set containing the variables.

```
fit <- survfit(Surv(time, status) ~ sex, data = lung)
print(fit)
```

```
## Call: survfit(formula = Surv(time, status) ~ sex, data = lung)
##
##         n events median 0.95LCL 0.95UCL
## sex=1 138    112    270     212     310
## sex=2  90     53    426     348     550
```

3.2.1 Access to the value returned by survfit()

The function survfit() returns a list of variables, including the following components:

- n: total number of subjects in each curve.
- time: the time points on the curve.
- n.risk: the number of subjects at risk at time t
- n.event: the number of events that occurred at time t.
- n.censor: the number of censored subjects, who exit the risk set, without an event, at time t.
- lower,upper: lower and upper confidence limits for the curve, respectively.
- strata: indicates stratification of curve estimation. If strata is not NULL, there are multiple curves in the result. The levels of strata (a factor) are the labels for the curves.

```
fit <- survfit(Surv(time, status) ~ sex, data = lung)
print(fit)
```

```
## Call: survfit(formula = Surv(time, status) ~ sex, data = lung)
##
##         n events median 0.95LCL 0.95UCL
## sex=1 138    112    270     212     310
## sex=2  90     53    426     348     550
```

```
d <- data.frame(time = fit$time,
                  n.risk = fit$n.risk,
                  n.event = fit$n.event,
                  n.censor = fit$n.censor,
                  surv = fit$surv,
                  upper = fit$upper,
                  lower = fit$lower
                  )
head(d) %>% kable()
```

time	n.risk	n.event	n.censor	surv	upper	lower
11	138	3	0	0.9782609	1.0000000	0.9542301
12	135	1	0	0.9710145	0.9994124	0.9434235
13	134	2	0	0.9565217	0.9911586	0.9230952
15	132	1	0	0.9492754	0.9866017	0.9133612
26	131	1	0	0.9420290	0.9818365	0.9038355
30	130	1	0	0.9347826	0.9768989	0.8944820

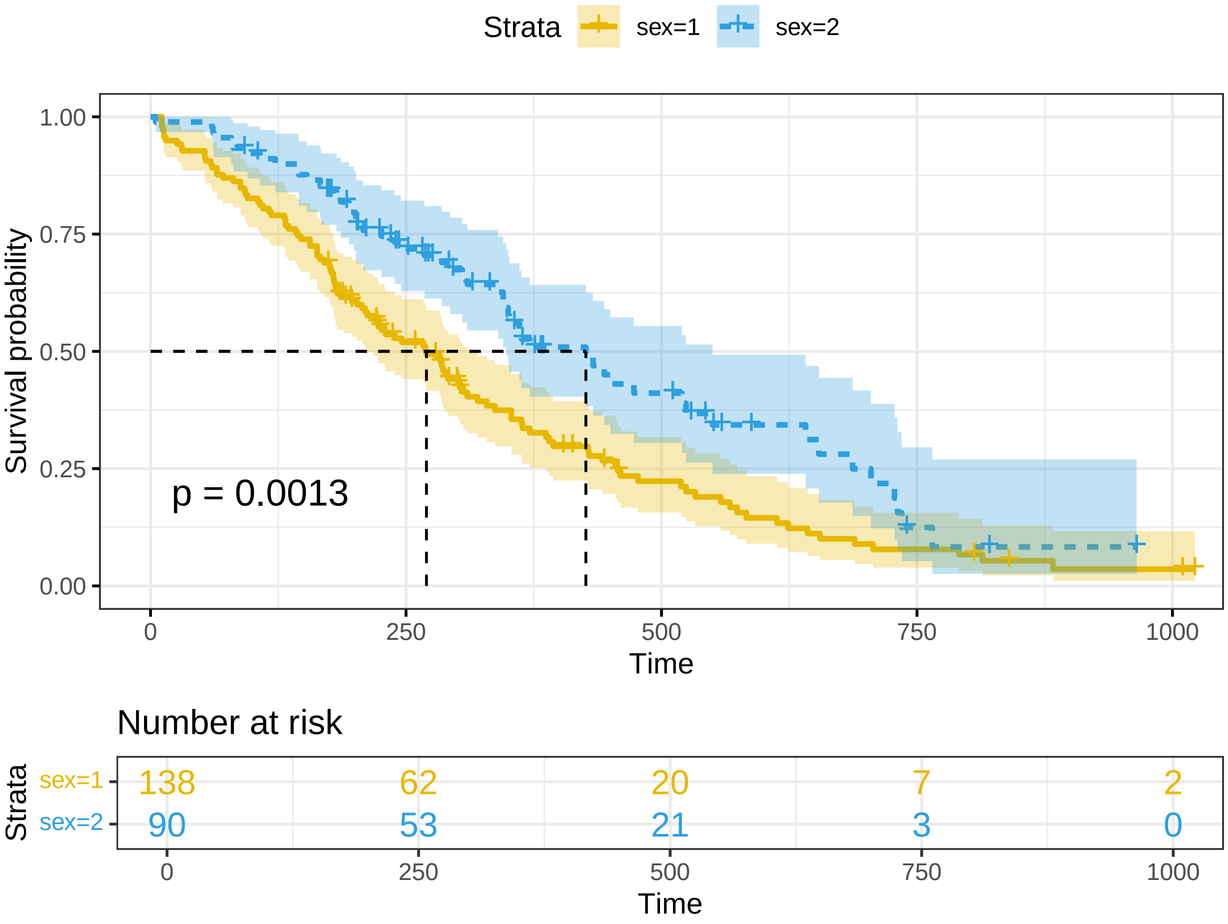
The events and the median for both groups are displayed.

3.2.2 Visualize survival curves

We'll use the function ggsurvplot() [in Survminer R package] to produce the survival curves for the two groups of subjects.

```r
ggsurvplot(
  fit,
  pval = TRUE, # p-value of the Log-Rank test comparing the groups using pval = TRUE
  conf.int = TRUE, # the 95% confidence limits
  risk.table = TRUE, # Add risk table,
                     # the number and/or the percentage of individuals at risk by time
  risk.table.col = "strata", # Change risk table color by groups
  linetype = "strata", # Change line type by groups
  surv.median.line = "hv", # Specify median survival, horizontal/vertical line
  ggtheme = theme_bw(), # Change ggplot2 theme
  palette = c("#E7B800", "#2E9FDF")
)
```

```
## Warning in geom_segment(aes(x = 0, y = max(y2), xend = max(x1), yend = max(y2)), : All aesthetics hav
## i Did you mean to use ‘annotate()‘?
## All aesthetics have length 1, but the data has 2 rows.
## i Did you mean to use ‘annotate()‘?
## All aesthetics have length 1, but the data has 2 rows.
## i Did you mean to use ‘annotate()‘?
## All aesthetics have length 1, but the data has 2 rows.
## i Did you mean to use ‘annotate()‘?
```



Strata
sex=1
sex=2
Survival probability
1.00
0.75
0.50
0.25
0.00
p = 0.0013
0
250
500
750
1000
Time
Number at risk
Strata
sex=1
sex=2
138
62
20
7
2
90
53
21
3
0
0
250
500
750
1000
Time

3.3 Log-Rank test comparing survival curves: survdiff()

The log-rank test is the most widely used method of comparing two or more survival curves and assesses whether the KM survival curves from two subpopulation are significantly different.
The null hypothesis is that there is no difference in survival between the two groups :

- H_0: survival in two groups is **the same**.
- H_1: survival in the two groups is **not the same**.

The log-rank test is a non-parametric test, which makes no assumptions about the survival distributions. Essentially, the log rank test compares the observed number of events in each group to what would be expected if the null hypothesis were true (i.e., if the survival curves were identical). The log rank statistic is approximately distributed as a chi-square test statistic.

The function survdiff() [in survival package] can be used to compute log-rank test comparing two or more survival curves.
survdiff() can be used as follow:

```
surv_diff <- survdiff(Surv(time, status) ~ sex, data = lung)
surv_diff
```

```
## Call:
## survdiff(formula = Surv(time, status) ~ sex, data = lung)
##
##         N Observed Expected (O-E)^2/E (O-E)^2/V
## sex=1 138      112     91.6      4.55      10.3
## sex=2  90       53     73.4      5.68      10.3
##
##  Chisq= 10.3  on 1 degrees of freedom, p= 0.001
```

The function returns a list of components, including:

- n: the number of subjects in each group.
- obs: the weighted observed number of events in each group.
- exp: the weighted expected number of events in each group.
- chisq: the chisquare statistic for a test of equality.
- strata: optionally, the number of subjects contained in each stratum.

The log rank test for difference in survival gives a p-value of p = 0.0013, indicating that the sex groups differ significantly in survival.

4 Old part

```
time   <- c(2,4,6,8,11,15,16,18,18,20,22,22,25,27,28,32,32,34,34)
death  <- c(1,1,0,1,1,1,1,0,0,1,0,1,1,1,1,0,1,0,0) # Censoring or not, 1 = died, 0 = censored
over40 <- c(1,1,0,1,1,1,1,0,0,1,0,1,0,1,1,0,1,1,0)   # Is over 40 or not, 1 = YES, 0 = NO

df <- tibble(time,death,over40)

# We use ~1 when there is no X variable ( additional categorical variable )
km.model <- survfit(Surv(time = time, event = death) ~ 1,
                    type = "kaplan-meier")   # Kaplan-Meier is the default value

km.model
```

```
## Call: survfit(formula = Surv(time = time, event = death) ~ 1, type = "kaplan-meier")
##
##       n events median 0.95LCL 0.95UCL
## [1,] 19     12     25      16      NA
```

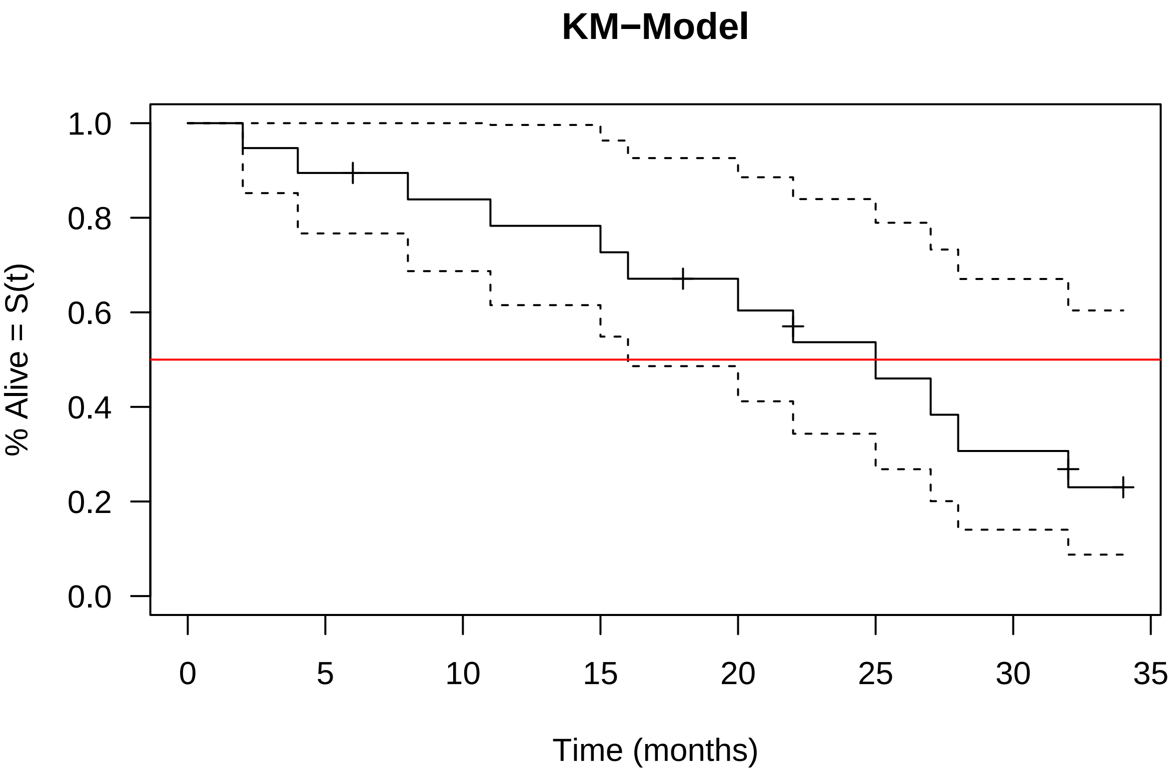
To visualize the survival at specific time, and confidence intervals.

```
summary(km.model)
```

```
## Call: survfit(formula = Surv(time = time, event = death) ~ 1, type = "kaplan-meier")
##
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     2     19       1    0.947  0.0512       0.8521        1.000
##     4     18       1    0.895  0.0704       0.7669        1.000
##     8     16       1    0.839  0.0854       0.6871        1.000
##    11     15       1    0.783  0.0963       0.6152        0.996
##    15     14       1    0.727  0.1044       0.5487        0.963
##    16     13       1    0.671  0.1103       0.4862        0.926
##    20     10       1    0.604  0.1179       0.4119        0.886
##    22      9       1    0.537  0.1224       0.3433        0.839
##    25      7       1    0.460  0.1267       0.2682        0.789
##    27      6       1    0.383  0.1267       0.2007        0.733
##    28      5       1    0.307  0.1224       0.1404        0.671
##    32      4       1    0.230  0.1133       0.0876        0.604
```

4.1 Kaplan-Meier curve

```
plot(
  km.model,    # used model
  conf.int = T,   # include confidence intervals
  xlab = "Time (months)",
  ylab = "% Alive = S(t)",
  main = "KM-Model",
  las = 1,       # rotatates the values on the y axis for better readability
  mark.time = T  # adds the censored values to the graph as a tick
  )

abline(h = 0.5, col = "red")
```



4.2 Kaplan-Meier survival model

Also known as Product-Limit Method, or the life table method.

This is a non-parametric curve, explains the selected data. The ticks are censored data

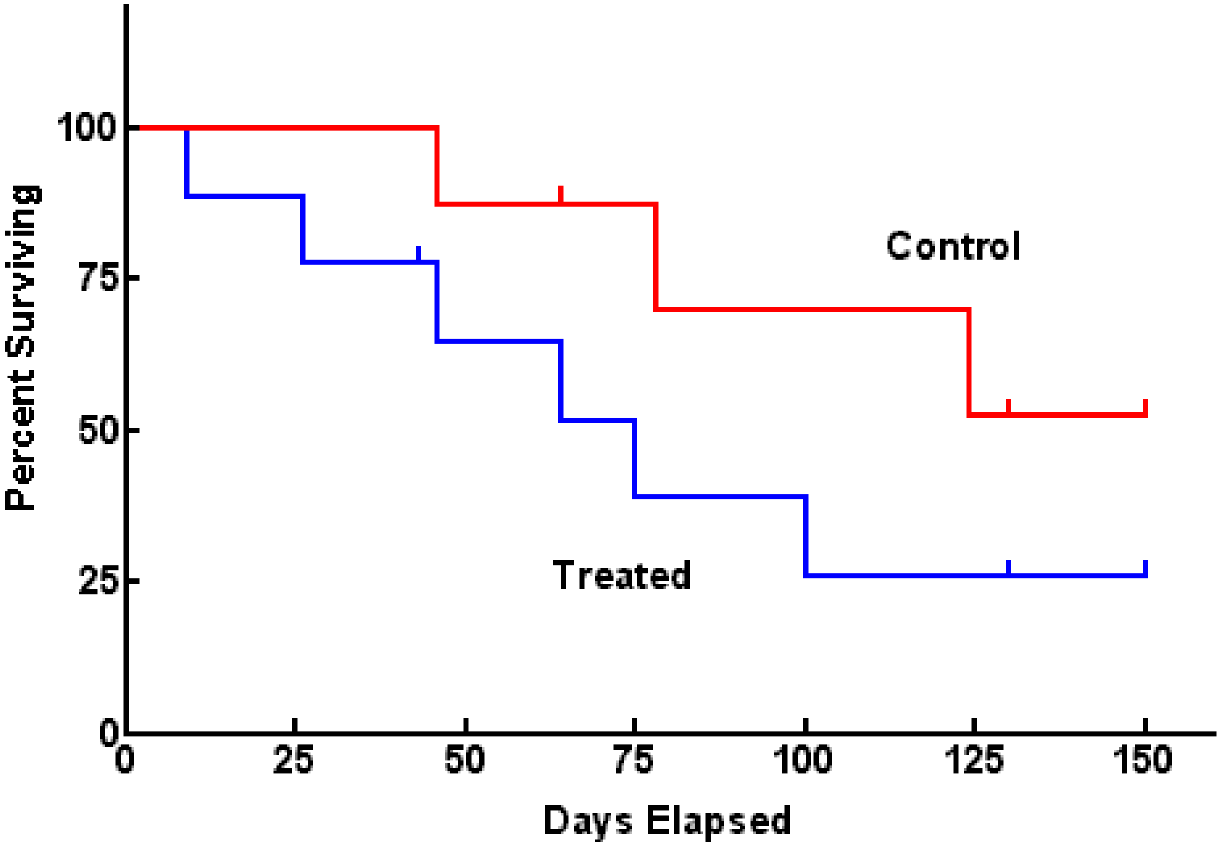


Figure 1: Kaplan Meier Curve

5 Kaplan Meier Model in R

5.1 Kaplan-Meier model with an X variable

5.2 Summary

```
# We use ~1 when there is no X variable ( additional categorical variable )
km.model2 <- survfit(Surv(time = time, event = death) ~ over40,
                     type = "kaplan-meier")  # Kaplan-Meier is the default value

km.model2
```

```
## Call: survfit(formula = Surv(time = time, event = death) ~ over40,
##     type = "kaplan-meier")
##
##           n events median 0.95LCL 0.95UCL
## over40=0  7      1     NA      25      NA
## over40=1 12     11     18      11      NA
```

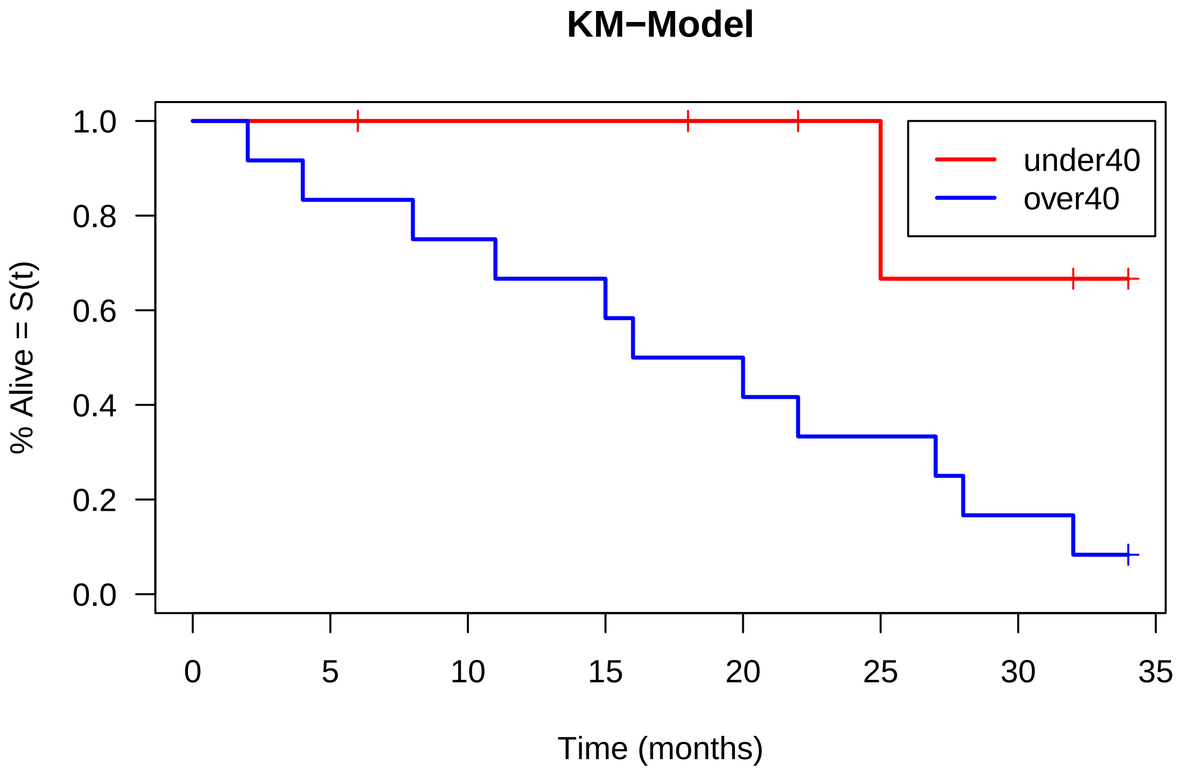
To visualize the survival at specific time, and confidence intervals.

```
summary(km.model2)
```

```
## Call: survfit(formula = Surv(time = time, event = death) ~ over40,
##     type = "kaplan-meier")
##
##                   over40=0
##         time       n.risk      n.event     survival      std.err lower 95% CI
##       25.000        3.000        1.000        0.667        0.272        0.300
## upper 95% CI
##        1.000
##
##                   over40=1
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     2     12       1   0.9167  0.0798       0.7729        1.000
##     4     11       1   0.8333  0.1076       0.6470        1.000
##     8     10       1   0.7500  0.1250       0.5410        1.000
##    11      9       1   0.6667  0.1361       0.4468        0.995
##    15      8       1   0.5833  0.1423       0.3616        0.941
##    16      7       1   0.5000  0.1443       0.2840        0.880
##    20      6       1   0.4167  0.1423       0.2133        0.814
##    22      5       1   0.3333  0.1361       0.1498        0.742
##    27      4       1   0.2500  0.1250       0.0938        0.666
##    28      3       1   0.1667  0.1076       0.0470        0.591
##    32      2       1   0.0833  0.0798       0.0128        0.544
```

5.3 Kaplan-Meier curve with an X variable

```
plot(
  km.model2,     # used model
  conf.int = F,    # include confidence intervals
  xlab = "Time (months)",
  ylab = "% Alive = S(t)",
  main = "KM-Model",
  col = c("red","blue"),    # add colors to the plots
  lwd = 2,        # makes the line a bit more thick
  las = 1,        # rotatates the values on the y axis for better readability
  mark.time = T  # adds the censored values to the graph as a tick
  )

legend(26,     # x coordinate for the box
       1,     # y coordinate for the box
       legend = c("under40","over40"),  # Names for the legend
       lty = 1, # linetype
       lwd = 2, # linewidth
       col = c("red","blue"),
       bty = "",  # boxtype shape
       cex = 1    # boxfont size
       )
```



5.4 The LOG-RANK-TEST

The log-rank test assesses whether the KM survival curves from two subpopulation are significantly different. Comparing the survival curves to see if they are different.

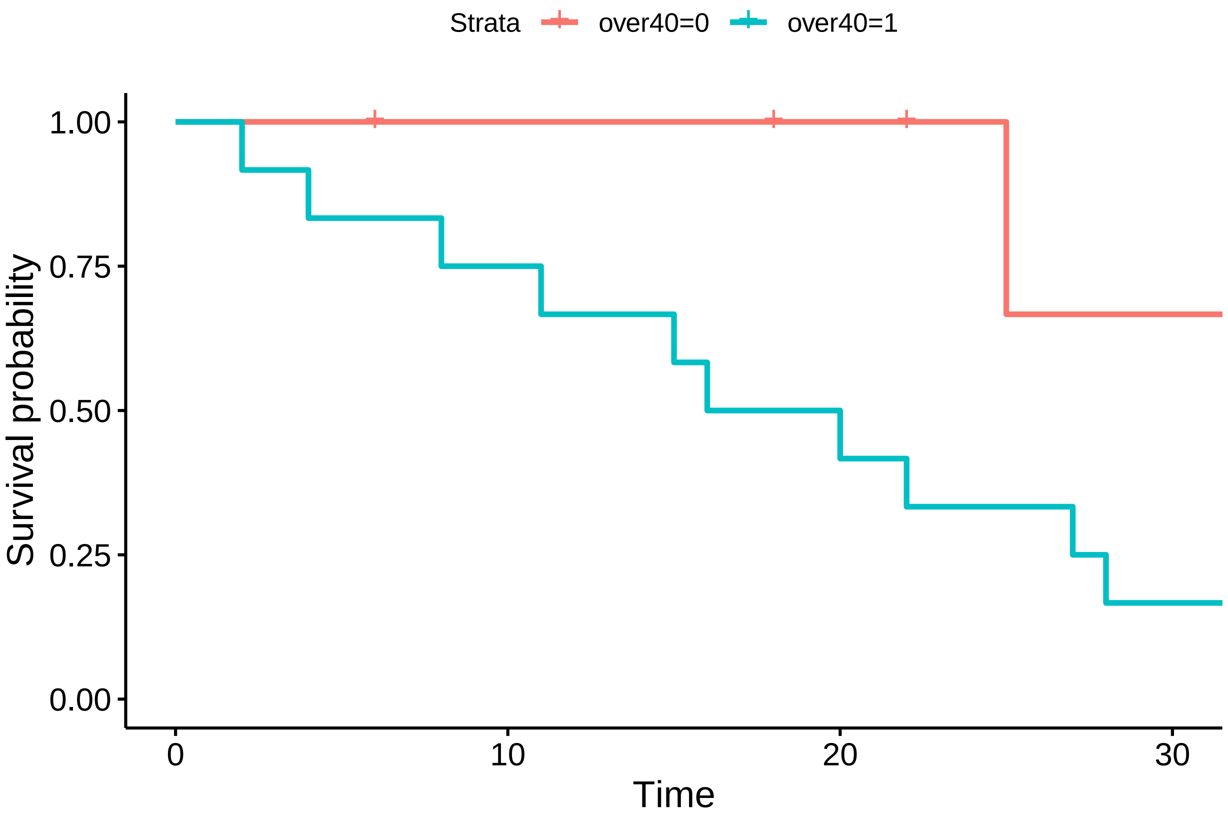
H_0: surivival in two groups is **the same**.
H_1: survival in the two groups is **not the same**.

```
survdiff(Surv(time,death)~ over40) # This can work with also more than 2 levels ( )
```

```
## Call:
## survdiff(formula = Surv(time, death) ~ over40)
##
##           N Observed Expected (O-E)^2/E (O-E)^2/V
## over40=0  7        1     4.93      3.13      5.36
## over40=1 12       11     7.07      2.18      5.36
##
##  Chisq= 5.4  on 1 degrees of freedom, p= 0.02
```

5.5 Exemple of plot with ggplot2

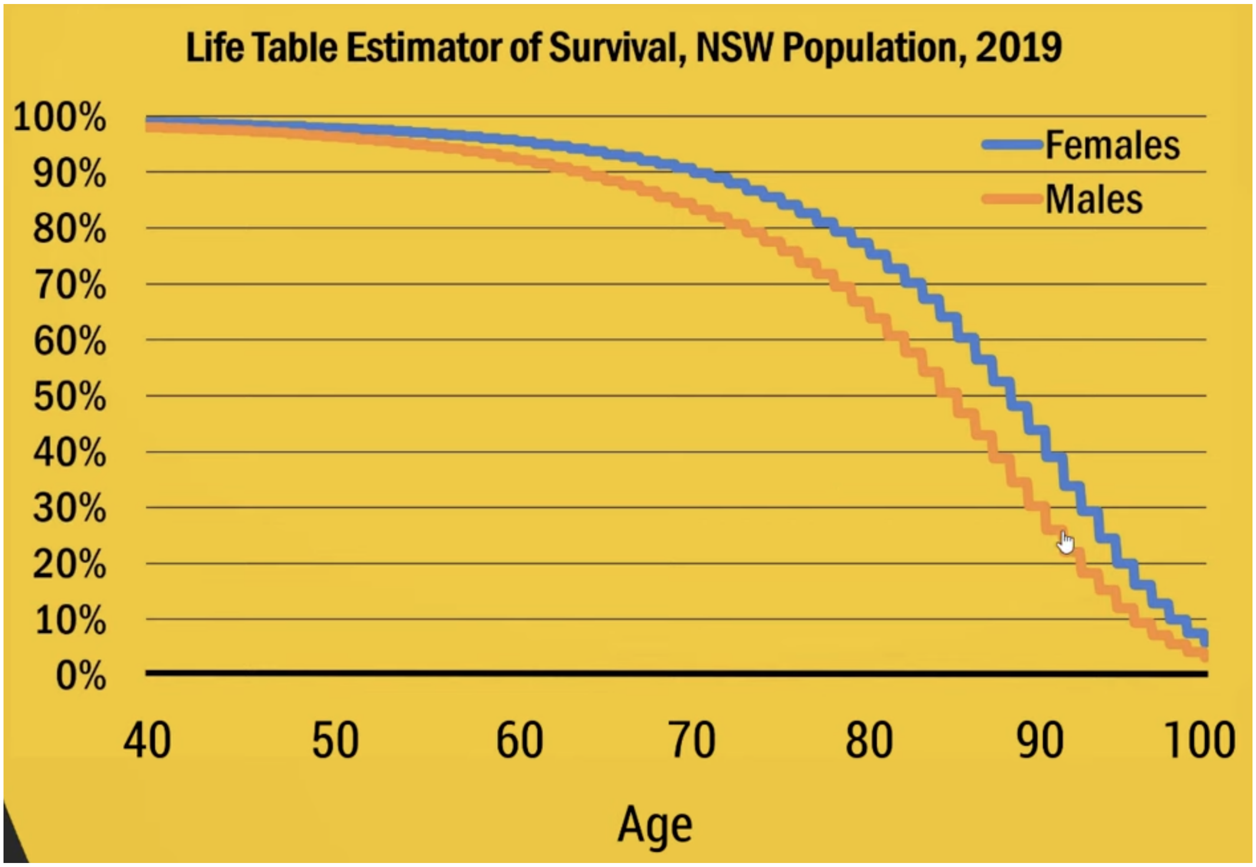
```
km.model2 <- survfit(Surv(time = time, event = death) ~ over40,
                    type = "kaplan-meier")  # Kaplan-Meier is the default value

p <- survminer::ggsurvplot(fit = km.model2,data = df)

p
```



6 Life tables

6.1 Definitions

Life tables illustrate the pattern of survivorship of a population by considering the probability of death at each consecutive age.



Input :

- population by age group
- deaths in each age group

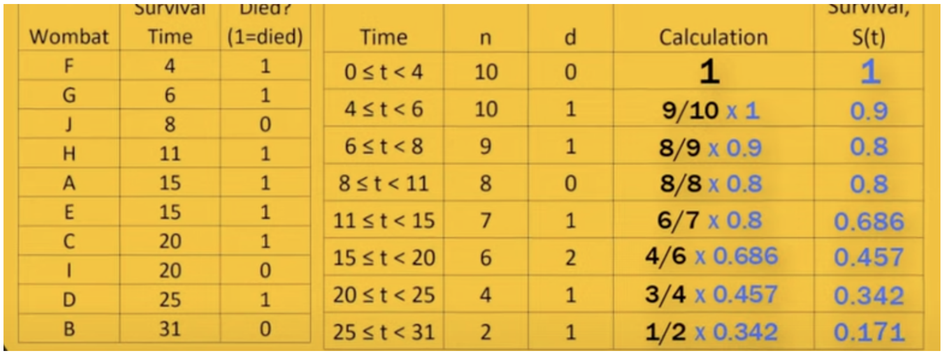
Output :

- the survival functions at each age
- conditional life expectancy at each age (probability of being alive)
- median, mean quantile survival

7 Kaplan-Meier Curves and Log-rank Test

7.1 Calculating the steps

It's a **non-parametric** estimator of survival. Doesn't have an estimator (like the mean or the standard deviation for an normal distribution). A Kaplan-Meier curve looks more like steps than a curve, and doesn't use parameters, it represents the data.



Wombat	Survival Time	Died? (1=died)
F	4	1
G	6	1
J	8	0
H	11	1
A	15	1
E	15	1
C	20	1
I	20	0
D	25	1
B	31	0

Time	n	d	Calculation	Survival, S(t)
$0 \le t < 4$	10	0	1	1
$4 \le t < 6$	10	1	9/10 x 1	0.9
$6 \le t < 8$	9	1	8/9 x 0.9	0.8
$8 \le t < 11$	8	0	8/8 x 0.8	0.8
$11 \le t < 15$	7	1	6/7 x 0.8	0.686
$15 \le t < 20$	6	2	4/6 x 0.686	0.457
$20 \le t < 25$	4	1	3/4 x 0.457	0.342
$25 \le t < 31$	2	1	1/2 x 0.342	0.171

Figure 2: Survival Table

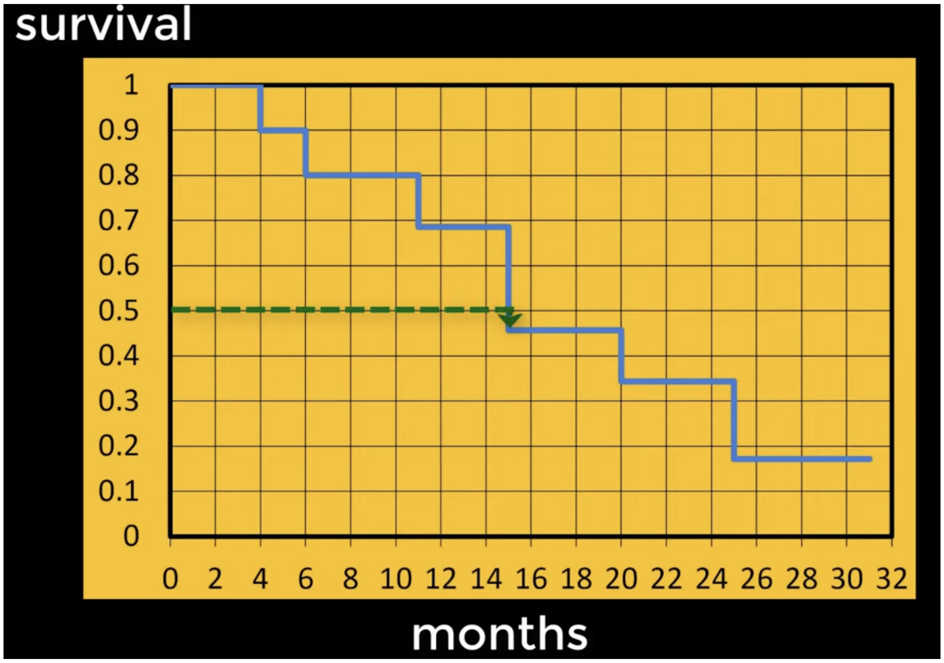


Figure 3: Survival Curve