

Survival analysis with R (Part I)

Juan R Gonzalez

8 marzo 2021

Contents

1	Introduction	2
2	Survival analysis	2
3	Survival time estimates using Kaplan-Meier estimator.	3
4	Comparing survival curves	5
5	Adjusting for other covariates: stratified analysis	7
6	Trend test	8
7	Exercise (to deliver)	9
8	References	10
9	Session information	10

1 Introduction

Objectives

- Understand the concept of survival analysis
- Learn how to perform survival analysis (Kaplan-Meier estimator and log-rank test) with R
- Perform data analyses where the scientific question is to determine factor associated with time until event

2 Survival analysis

To illustrate how to carry out different survival data analyses a real dataset is going to be used. The database belongs to the data presented in a paper that analysed 686 women enrolled in a clinical trial on breast cancer. The reference is *W. Sauerbrei and P. Royston. Building multivariable prognostic and diagnostic models: transformation of the predictors by using fractional polynomials. Journal of the Royal Statistical Society, Series A, 1999;162:71-94*. The information available is:

- Running-ID
- Hormonal Therapy (0- no treatment, 1-treatment)
- age (X1; in years)
- menopausal status (X2; 1- premenopausal, 2- postmenopausal)
- Tumour size (X3; in mm)
- Tumour grade (X4; 1,2,3)
- Number of positive nodes (X5)
- Progesterone receptor (X6; in fmol)
- Estrogen receptor (X7; in fmol)
- Survival time (in days)
- Censoring Indicator (0- censored, 1- event).

Data can be loaded by executing:

```
library(survival)
datos <- read.table("../data/sauerbre.txt", header=TRUE)
head(datos)
```

	id	therapy	age	meno.status	tumor.size	tumor.grade	nodes	progest	estrog	time	event
1	1	0	70	2	21	2	3	48	66	1814	1
2	2	1	56	2	12	2	7	61	77	2018	1
3	3	1	58	2	35	2	9	52	271	712	1
4	4	1	59	2	17	2	4	60	29	1807	1
5	5	0	73	2	35	2	1	26	65	772	1
6	6	0	32	1	57	3	24	0	13	448	1

Survival analysis requires an object of class `Surv` where "+" denotes the individual is right-censored.

```
Surv(datos$time, datos$event)
```

```
[1] 1814 2018 712 1807 772 448 2172+ 2161+ 471 2014+ 577 184 1840+ 1842+ 1821+ 1371
[17] 707 1743+ 1781+ 865 1684 1701+ 1701+ 1693+ 379 1105 548 1296 1483+ 1570+ 1469+ 1472+
[33] 1342+ 1349+ 1162 1342+ 797 1232+ 1230+ 1205+
```

3 Survival time estimates using Kaplan-Meier estimator

The survival function can be estimated by using `survfit` function

```
ans <- survfit(Surv(time, event)~1, datos)
ans
Call: survfit(formula = Surv(time, event) ~ 1, data = datos)
```

```
      n  events  median 0.95LCL 0.95UCL
686    299    1807    1587    2030
```

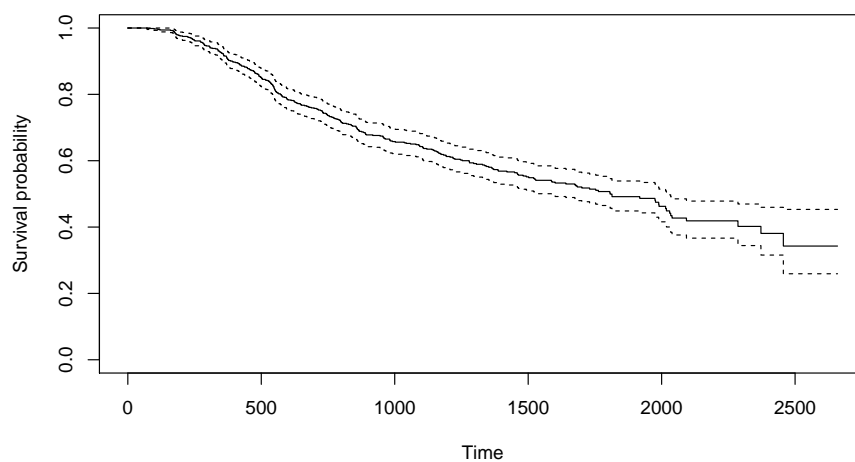
The `survfit` object contains more variables, including detailed time points with the number at risk `n.risk`, events `n.event`, and censors `n.censor` at each time point. Therefore, the Kaplan-Meier table can be obtained by:

```
library(tidyverse)
km <- tibble(
  time = ans$time,
  n.risk = ans$n.risk,
  n.event = ans$n.event,
  n.censor = ans$n.censor,
  surv = ans$surv,
  upper = ans$upper,
  lower = ans$lower
)
km
# A tibble: 574 x 7
   time n.risk n.event n.censor  surv upper lower
   <dbl> <dbl>   <dbl>   <dbl> <dbl> <dbl> <dbl>
1     8    686     0         1     1     1     1
2    15    685     0         1     1     1     1
3    16    684     0         1     1     1     1
4    17    683     0         2     1     1     1
5    18    681     0         1     1     1     1
6    29    680     0         1     1     1     1
7    42    679     0         1     1     1     1
8    46    678     0         1     1     1     1
9    57    677     0         1     1     1     1
10   63    676     0         1     1     1     1
# ... with 564 more rows
```

The survival curve can be visualized by

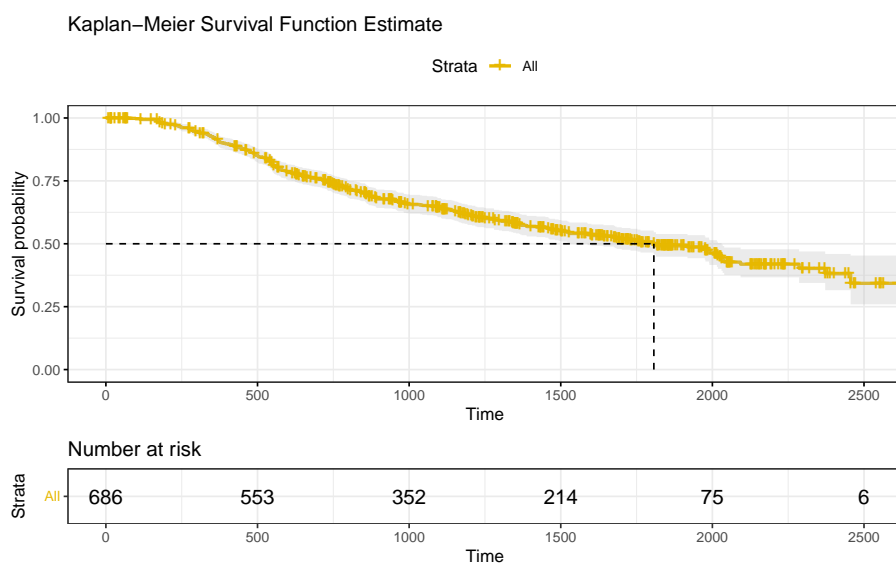
```
plot(ans, xlab="Time", ylab="Survival probability")
```

Survival analysis with R (Part I)



We can also use `ggsurvplot()` function from `survminer` package. A vertical drop in the curves indicates an event. The vertical tick mark on the curves means that a patient was censored.

```
library(survminer)
ggsurvplot(
  ans,
  pval = TRUE,
  conf.int = TRUE,
  risk.table = TRUE,
  surv.median.line = "hv", # median horizontal and vertical ref lines
  ggtheme = theme_bw(),
  palette = c("#E7B800", "#2E9FDF"),
  title = "Kaplan-Meier Survival Function Estimate"
)
```



4 Comparing survival curves

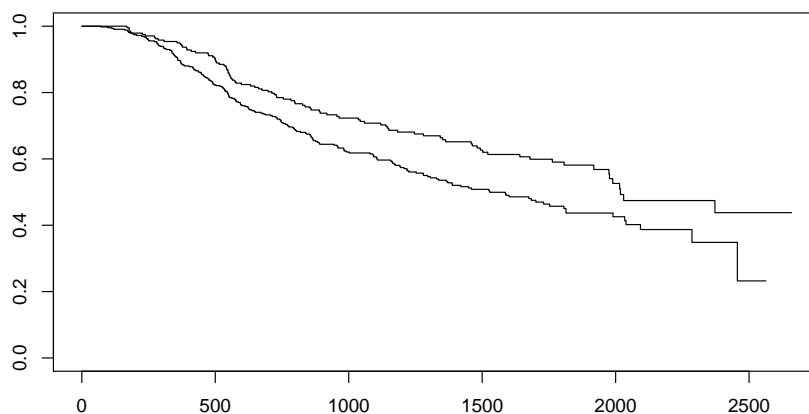
Let us illustrate how to compare survival curves for two groups. Researchers are interested in comparing the survival between patients who received or not a new therapy. The R code is:

```
ans.ther <- survfit(Surv(time, event)~as.factor(therapy), datos)
ans.ther
Call: survfit(formula = Surv(time, event) ~ as.factor(therapy), data = datos)
```

	n	events	median	0.95LCL	0.95UCL
as.factor(therapy)=0	440	205	1528	1296	1814
as.factor(therapy)=1	246	94	2018	1918	NA

In this case we observe that there are statistically significant differences between the median survival of the two groups since their confidence intervals do not overlap. However we are normally interested in determining differences at any time. This can be visualized by comparing survival curves between groups

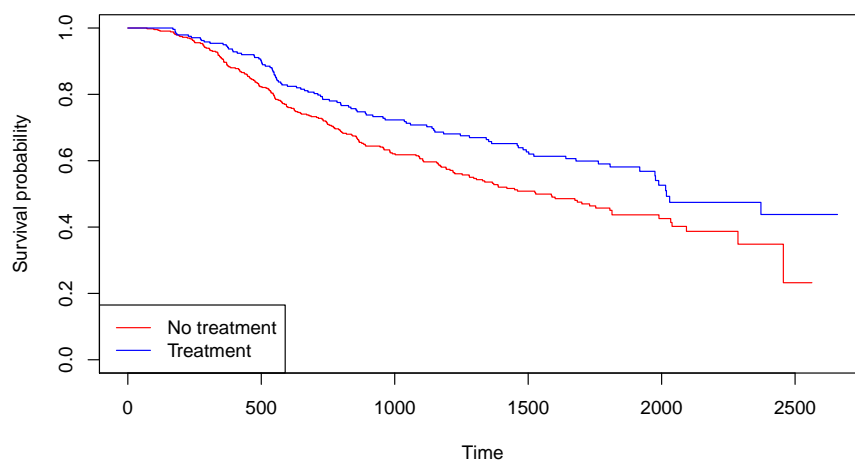
```
plot(ans.ther)
```



The plot can be improved by

```
plot(ans.ther, xlab="Time", ylab="Survival probability",
      col=c("red", "blue"))
legend("bottomleft", c("No treatment", "Treatment"),
      col=c("red", "blue"), lty=c(1,1))
```

Survival analysis with R (Part I)



The comparison across time is computed by using log-rank test

```
ans.logrank<-survdif(Surv(time, event)~as.factor(therapy), datos,
                    rho=0)

ans.logrank
Call:
survdif(formula = Surv(time, event) ~ as.factor(therapy), data = datos,
        rho = 0)

      N Observed Expected (O-E)^2/E (O-E)^2/V
as.factor(therapy)=0 440      205      180      3.37      8.56
as.factor(therapy)=1 246       94      119      5.12      8.56

Chisq= 8.6 on 1 degrees of freedom, p= 0.003
```

We observe that the differences in the curves, are statistically significant at 5% level. Notice that the argument `rho` is not necessary since it is the default value. It corresponds to the log-rank test. Wilcoxon test is computed by setting `rho=1`

```
ans.wilcox<-survdif(Surv(time, event)~as.factor(therapy), datos,
                    rho=1)

ans.wilcox
Call:
survdif(formula = Surv(time, event) ~ as.factor(therapy), data = datos,
        rho = 1)

      N Observed Expected (O-E)^2/E (O-E)^2/V
as.factor(therapy)=0 440      157.8      138.6      2.66      8.71
as.factor(therapy)=1 246       69.3      88.5      4.16      8.71

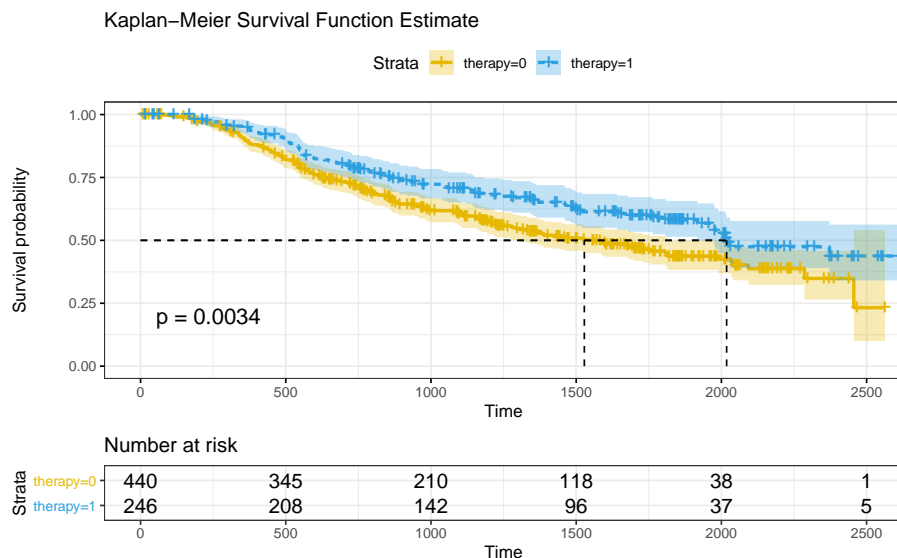
Chisq= 8.7 on 1 degrees of freedom, p= 0.003
```

Survival analysis with R (Part I)

We also observe that both tests are providing the same conclusion. We can produce a figure containing all this information by using the `survminer` package. To this end, `survfit` object should be created instead of a `survdif`. In that case, log-rank test is used.

```
ans.km <- survfit(Surv(time, event) ~ therapy, datos)

ggsurvplot(
  ans.km,
  linetype = "strata", # Change line type by groups
  pval = TRUE,
  conf.int = TRUE,
  risk.table = TRUE,
  surv.median.line = "hv", # median horizontal and vertical ref lines
  ggtheme = theme_bw(),
  palette = c("#E7B800", "#2E9FDF"),
  title = "Kaplan-Meier Survival Function Estimate"
)
```



5 Adjusting for other covariates: stratified analysis

In some occasions researchers are interested in comparing survival curves between two groups of patients but they know that there are differences in the survival due to a third variable. For instance, in this data, menopausal status influence survival. Therefore, observed differences between women received therapy or not must be adjusted for this third variable. This can be performed by using an stratified test

```
ans.strat <- survdiff(Surv(time, event) ~ as.factor(therapy)
  + strata(meno.status), datos)

ans.strat
Call:
survdiff(formula = Surv(time, event) ~ as.factor(therapy) + strata(meno.status),
  data = datos)
```

Survival analysis with R (Part I)

```

              N Observed Expected (O-E)^2/E (O-E)^2/V
as.factor(therapy)=0 440      205      180      3.52      9.51
as.factor(therapy)=1 246       94      119      5.31      9.51

Chisq= 9.5 on 1 degrees of freedom, p= 0.002

ans.logrank
Call:
survdif(formula = Surv(time, event) ~ as.factor(therapy), data = datos,
        rho = 0)

              N Observed Expected (O-E)^2/E (O-E)^2/V
as.factor(therapy)=0 440      205      180      3.37      8.56
as.factor(therapy)=1 246       94      119      5.12      8.56

Chisq= 8.6 on 1 degrees of freedom, p= 0.003
```

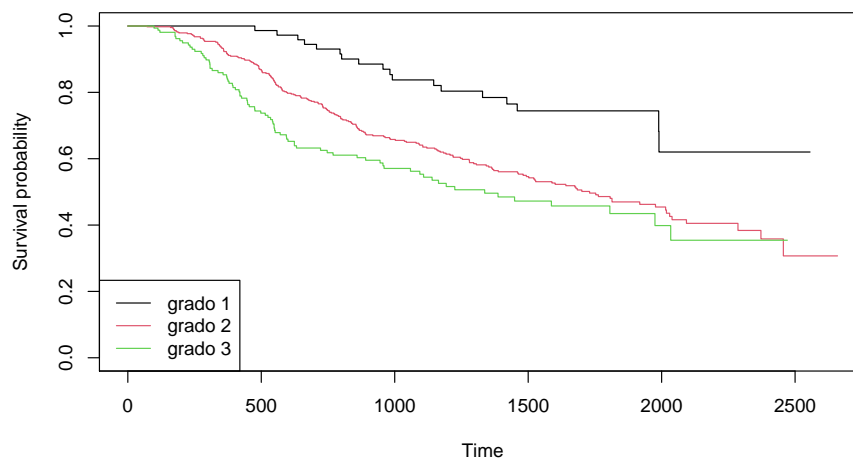
We can observe as both tests are providing similar answer. In both cases the differences in survival times after receiving or not the therapy are statistically significant. This implies that the menopausal status is not confounding the results.

6 Trend test

In other occasions researchers are interested in addressing the next scientific question: Is there any linear relationship between the survival and the tumor grade? In other words, the worse tumor grade implies worse survival? This question makes sense in the situations where the independent variable has any order. In this figure we also check whether this question is well addressed.

```
ans.grade<-survfit(Surv(time, event)~as.factor(tumor.grade), datos)
plot(ans.grade, xlab="Time", ylab="Survival probability", col=1:3)
legend("bottomleft", c("grado 1", "grado 2", "grado 3"),
      col=1:3, lty=1)
```


Survival analysis with R (Part I)



The statistical test is performed by considering `tumor.grade` variable as numeric. This cannot be performed by using `survfit` function since it assumes that the covariates are categorical. These type of question can be answered by fitting Cox proportional hazards models (this will be seen in Part II). Here can also addressed he question: Are there statistical differences between tumoral grades?

```
survdif(Surv(time, event)~as.factor(tumor.grade), datos)
Call:
survdif(formula = Surv(time, event) ~ as.factor(tumor.grade),
  data = datos)

      N Observed Expected (O-E)^2/E (O-E)^2/V
as.factor(tumor.grade)=1  81      18   42.2   13.8469   16.159
as.factor(tumor.grade)=2 444     202  198.2    0.0725    0.215
as.factor(tumor.grade)=3 161      79   58.6    7.0788    8.848

Chisq= 21.1 on 2 degrees of freedom, p= 3e-05
```

Further information about survival data analysis with R can be found in this tutorial [Tutorial Survival Analysis](#).

7 Exercise (to deliver)

Data for exercises are in the repository https://github.com/isglobal-brge/TeachingMaterials/tree/master/Longitudinal_data_analysis/data

File *pulmon.sav* contains data about a survival study about lung cancer (NOTE: data can be loaded into R by using `read.spss` function available at `foreign` library - use argument `to.data.frame=TRUE`). Columns contain this information:

- TIEMPO Supervivencia (meses)
- ESTADO: 0 VIVO, 1 MORT
- EDAD4 Age at diagnosis in years (quartiles)

Survival analysis with R (Part I)

- SEXO: HOMBRES, MUJERES
- ESTCLIN Estadio clinico: EST 0/I, EST II, EST IIIA, EST IIIB, EST IV
- IK Indice de estado general (100 estado perfecto, 0 muerte)
- CIRUGIA: 1 No operado, 2 Cirugia no radical, 3 Cirugia Radical
- QUIMIO: 1 No Quimio, 2 Platino
- RADIOTER: 1 No RT, 2 <60 Gy, 3 >60 Gy

Exercise 1: Survival function estimation

- Estimate global survival time
- Draw survival curve
- Estimate median survival time and its confidence interval
- Which is the time where 75% of people have died?
- Estimate survival curve for the covariates sex, surgery, chemotherapy and radiotherapy

Exercise 2: Comparing survival curves

- Draw survival curves of patients receiving chemotherapy and not
- Compare survival curves of patients receiving chemotherapy, radiotherapy, surgery and clinical stage by using log-rank test. Identify those variables significantly associated with survival
- Are there statistical differences depending on chemotherapy after adjusting by clinical stage?
- Is there any trend in the survival with regard to the Karnofsky index. Answer this question by visually inspecting the required plot \end{enumerate}

8 References

- The `[survival]` package (<https://cran.r-project.org/web/packages/survival/>)

9 Session information

```
R version 4.0.2 (2020-06-22)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 19041)

Matrix products: default

locale:
[1] LC_COLLATE=Spanish_Spain.1252  LC_CTYPE=Spanish_Spain.1252    LC_MONETARY=Spanish_Spain.1252
[4] LC_NUMERIC=C                   LC_TIME=Spanish_Spain.1252

attached base packages:
[1] stats      graphics  grDevices  utils      datasets  methods   base

other attached packages:
[1] survminer_0.4.8  ggpubr_0.4.0    forcats_0.5.0   stringr_1.4.0   dplyr_1.0.2
[6] purrr_0.3.4      readr_1.3.1     tidyr_1.1.2     tibble_3.0.3    ggplot2_3.3.2
[11] tidyverse_1.3.0  survival_3.2-3  knitr_1.29      BiocStyle_2.16.0
```

Survival analysis with R (Part I)

```
loaded via a namespace (and not attached):
 [1] httr_1.4.2          jsonlite_1.7.0      splines_4.0.2       carData_3.0-4
 [5] modelr_0.1.8        assertthat_0.2.1    BiocManager_1.30.10 blob_1.2.1
 [9] cellranger_1.1.0    yaml_2.2.1          pillar_1.4.6        backports_1.1.9
[13] lattice_0.20-41     glue_1.4.2          digest_0.6.25       ggsignif_0.6.0
[17] rvest_0.3.6         colorspace_1.4-1    htmltools_0.5.0     Matrix_1.2-18
[21] pkgconfig_2.0.3     broom_0.7.0         haven_2.3.1         bookdown_0.20
[25] xtable_1.8-4        scales_1.1.1        km.ci_0.5-2         openxlsx_4.1.5
[29] rio_0.5.16          KMsurv_0.1-5        farver_2.0.3        generics_0.0.2
[33] car_3.0-9           ellipsis_0.3.1      withr_2.2.0         cli_2.0.2
[37] magrittr_1.5        crayon_1.3.4        readxl_1.3.1        evaluate_0.14
[41] fs_1.5.0            fansi_0.4.1         rstatix_0.6.0       xml2_1.3.2
[45] foreign_0.8-80      tools_4.0.2         data.table_1.13.0   hms_0.5.3
[49] lifecycle_0.2.0     munsell_0.5.0       reprex_0.3.0        zip_2.1.1
[53] compiler_4.0.2      rlang_0.4.10        grid_4.0.2          rstudioapi_0.11
[57] labeling_0.3        rmarkdown_2.3       gtable_0.3.0        abind_1.4-5
[61] DBI_1.1.0           curl_4.3            R6_2.4.1            zoo_1.8-8
[65] gridExtra_2.3       lubridate_1.7.9.2   survMisc_0.5.5      utf8_1.1.4
[69] stringi_1.4.6       Rcpp_1.0.6          vctrs_0.3.3         dbplyr_1.4.4
[73] tidyselect_1.1.0    xfun_0.16
```