Introduction to Survival Analysis

Solution to Exercises

**This document is part of a resource ‘Advanced Topics in Survival Analysis’ on** [**https://www.ncrm.ac.uk/resources/online/all/?id=20860**](https://www.ncrm.ac.uk/resources/online/all/?id=20860)

The Dataset

The dataset for these exercises is provided with the online material and it is called:

**df\_exercise.RData**

This is a simulated dataset I have created for these exercises. It includes 500 cases. In creating this dataset, I imagined a fictional setting of a study where children who had not yet uttered a sentence (subject – verb – object) are followed up for a year: the occurrence of their first sentence is recorded in days within the year. The main variables in the dataset are:

|  |
| --- |
| **ID**: The adolescent ID in the study. |
| **time**: The timing of first sentence utterance in days or the timing (in days) when participants left the study, if they are censored. |
| **censor**: A dummy variable to indicate whether participants are censored (censor==1) or not (censor==0) |
| **male**: A dummy variable for sex, where male==1 indicates male children and male==0 indicates female children (in this fictional example, only two options were provided) |
| **ses**: An indicator of Socio-Economic Status, ranging from 1 (High SES) to 3 (Deprived SES) |

The Scripts:

The files:

exercises n1 adv sa.R

exercises n2 adv sa.R

provide the full R scripts to run the solutions to these exercises.

Exercise Set #1

1. Create a survival object using the library “survival”.

**Solution**

You should have run a similar command:

surv\_obj <- Surv(dfex$time, dfex$event)

# double check that the Surv is reading the file correctly:

Surv(dfex$time, dfex$event)[1:10]



This shows that the first case experienced the event at 309 days, the second case was censored (“+” sign”) and left the study at 186 days without experiencing the event, and so on. Check this output with the dataset to make sure the software is reading the dataset correctly.

1. Extract and plot the Kaplan-Meier **survivor function** with 95% Confidence Intervals (CIs), and estimate the sample **median survival time**.

**Solution**

The Kaplan-Meier estimates are obtained by the “survfit” command run on the survival object created, for example:

km\_fit <- survfit(Surv(time, event) ~ 1, data = dfex)

The following:

survfit(Surv(time, event) ~ 1, data = dfex)

provides the sample median survival time, which is 160 days:

A screen shot of a computer code

AI-generated content may be incorrect.

Using the “survminer” library it is possible to create customised survivor function plots and other similar plots, using the statistics extract by “survfit”:

library(survminer)

# Plot survival with custom CI color and line size

gsurv<-ggsurvplot(

km\_fit,

conf.int = TRUE,

conf.int.fill = "#FFB347", # Your custom CI color (e.g., orange)

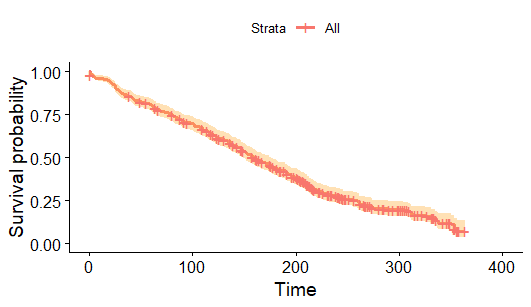
conf.int.alpha = 0.4, # Transparency of CI

palette = "#2E9Fdfex", # Line color (e.g., blue)

size = 1.2 # Line thickness

)

print(gsurv)



1. Extract and plot the **cumulative hazard function** using Nelson-Aalen method; include 95% CIs.

**Solution**

The Nelson-Aalen estimates are stored in the “km\_fit” object created in the previous step: they are dubbed “cumhaz”. We can thus use “survminer” to extract and plot them, including their 95% CIs:

# Plot cumulative hazard with custom CI color and line size

gcumhaz<-ggsurvplot(

km\_fit,

**fun = "cumhaz",**

conf.int = TRUE,

conf.int.fill = "#FFB347", # Your custom CI colour

conf.int.alpha = 0.4, # Transparency of CI

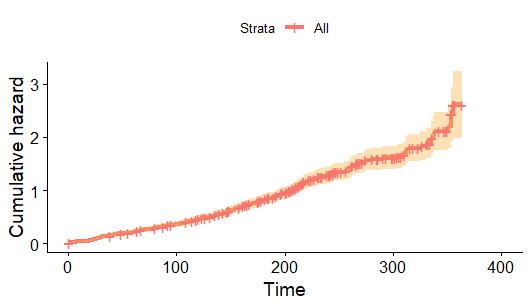
palette = "#2E9Fdfex", # Line colour (e.g., blue)

size = 1.5 # Line thickness

)

print(gcumhaz)

Note that to plot the cumulative hazard function, you need to add the option “fun” to the ggsurvplot command (see option in bold above).



1. Estimate and plot the **hazard function using kernel-smoothing**: I suggest you try three different bandwidths: 15, 30, and 90 days.

**Solution:**

In this case we may opt to use the “muhaz” library: the following command extracts the number of events and the number at risks within intervals from the dataset, while the band-width method is set to “global” and the band-width grid is set to different time lengths of time, from 15 to 90 days:

haz\_fit\_15 <- muhaz(dfex$time, dfex$event, bw.method = "global", bw.grid = 15)

haz\_fit\_30 <- muhaz(dfex$time, dfex$event, bw.method = "global", bw.grid = 30)

haz\_fit\_90 <- muhaz(dfex$time, dfex$event, bw.method = "global", bw.grid = 90)

The information from these functions is then put into respective data frames:

haz\_fit\_15 <- muhaz(dfex$time, dfex$event, bw.method = "global", bw.grid = 15)

haz\_fit\_30 <- muhaz(dfex$time, dfex$event, bw.method = "global", bw.grid = 30)

haz\_fit\_90 <- muhaz(dfex$time, dfex$event, bw.method = "global", bw.grid = 90)

and then is combined in a single dataframe:

haz\_df <- bind\_rows(dfex\_15, dfex\_30, dfex\_90 )

The latter data frame is used to plot the different estimates (in this case I also used “viridis” colour scheme):

library(viridis)

ggplot(haz\_df, aes(x = time, y = hazard, color = bandwidth)) +

geom\_line(size = 1.2, alpha = .70) +

labs(

x = "Time (days)",

y = "Hazard",

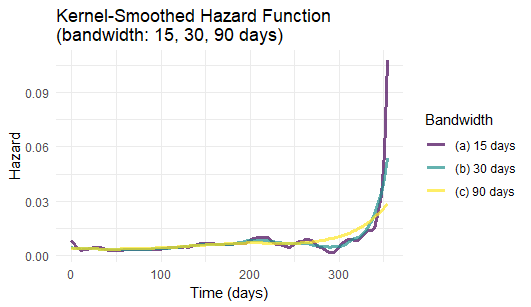
title = "Kernel-Smoothed Hazard Function\n(bandwidth: 15, 30, 90 days)",

color = "Bandwidth"

) +

scale\_color\_viridis\_d(option = "D") + # Use the discrete viridis palette

theme\_minimal()



Exercise Set #2

1. Explore differences between males and females, firstly by plotting their respective Kaplan-Meier survivor functions, and their respective median survival time.

Solution:

Use the “survfit” option to use the Kaplan-Meier method, and then invoke a table from the object you have just created:

km\_fit <- survfit(Surv(time, event) ~ male, data = dfex)

# Display respective median lifetimes

summary(km\_fit)$table

A black background with white text

AI-generated content may be incorrect.

The table shows the median lifetime for females (male=0) is 148 days (*95% CI* 126 to 174), while the median lifetime for males (male=1) is 166 days (*95% CI* 152 to 195). Therefore, the central tendency is for males to utter a sentence for the first time 18 days later than females.

I can use the “survminer” library to plot the estimated Kaplan-Meier survivor functions for females and males:

# Create a customized plot

g1<-ggsurvplot(

km\_fit,

data = dfex,

conf.int = FALSE, # DO not Show 95% confidence intervals

palette = c("#E7B800", "#2E9FDF"), # Custom colors for lines

size = 1.2, # Change line thickness

xlab = "Days",

ylab = "Survival probability",

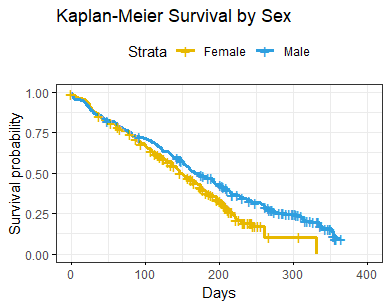
legend.labs = c("Female", "Male"), # Adjust labels as appropriate

ggtheme = theme\_bw(), # Clean background

title = "Kaplan-Meier Survival by Sex"

)

g1



I have not plotted the confidence intervals in the plot above, but the “ggsurvplot” options allow you to add them by using this option:

conf.int = **TRUE**,

The other options would also allow you change the colours and the transparency (“alpha”) of the confidence intervals shaded areas.

1. Plot sex differences in the log of the negative log of the survivor function.

**Solution**:

The log of the negative log of the survivor function is important insofar is the basis of the Cox regression model.

We can start by calculating the negative log of the survivor function, which is an alternative way to estimate the cumulative hazard function (alternative to the Nelson-Aalen method).

We can use the “stratified” survivor functions estimated by “survfit” (saved in object “km\_fit”) first. To simplify this, I extract a summary of the estimates from the Kaplan Meier estimates in “km\_fit” into a new object I called “s”:

s <- summary(km\_fit)

I can now create a new data frame where I will also manually calculate the negative log survivor function (“nlsf”) and the log of the latter (“log\_nls”), ensuring that they are calculated by “strata”, i.e. by group:

df\_nlsf <- data.frame(

time = s$time,

surv = s$surv,

**neg\_log\_surv = -log(s$surv),**

**log\_nls = log(-log(s$surv)),**

**Group = s$strata**

)

Using ggplot, I can now plot the log log negative survivor function:

# Plot the log of neg.log.survivor

library(ggplot2)

glogH<-ggplot(df\_nlsf, aes(x = time, y = log\_nls, color = Group)) +

geom\_step(size = 1.2) +

labs(x = "Time", y = "Log H(t)",

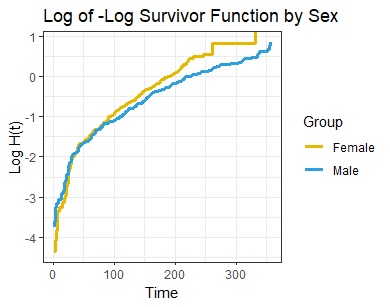
title = "Log of - Log Survivor Function by Sex") +

scale\_color\_manual(values = c("male=0" = "#E7B800", "male=1" = "#2E9FDF"),

labels = c("male=0" = "Female", "male=1" = "Male")) +

theme\_bw()

glogH



Note that one of the assumptions of the Cox regression model is that there is a constant distance between log hazard functions of different values of the predictors at every time interval. The graph above seems to indicate this assumption may have to be considered carefully in this example, or possibly relaxed. However, I will not go into this issue any further here (I refer to some of the references provided if you want to read more).

1. Run a Cox regression model including child’s sex and SES as predictors. Since SES is a categorical variable, make sure you compare children from the most deprived background to other groups. Report and interpret the results.

**Solution:**

Firstly, I will recode the SES variable into dummy variables and obtain a dummy for children from “high” SES background, one for children from “middle” SES, thus representing comparisons between these and the children from the most deprived background.

# Recode SES into two dummy coded variables that compare against more deprived

dfex$highses<-ifelse(dfex$ses==1, 1, 0)

dfex$modses<-ifelse(dfex$ses==2, 1, 0)

Now we can add these predictors and estimate Cox regression coefficients:

m2fit<-coxph(Surv(time, event) ~ **male + highses + modses**, data = dfex)

summary(m2fit)

I have also created a table to focus on the Hazard Ratios:

library(gtsummary)

# Create a summary table with HRs and 95% CIs

summary\_table <- tbl\_regression(m2fit, exponentiate = TRUE)

summary\_table

The results indicate a significant association between the event occurrence and sex: at any interval across the study males show a 15% reduction on the rate of hazard of uttering their first sentence compared to females and while controlling for children’s SES background (*HR* = 0.75, *95% CI* 0.60 to 0.93). While controlling for children’s sex, at any interval of the study children from higher SES backgrounds report a 42% increase in the rate of hazard of uttering their first sentence compared to children from the most deprived backgrounds, *HR* = 1.42 (*95% CI* 1.10 to 1.82).

1. Consider ties, and try another method such as Breslow in the Cox regression.

**Solution:**

The dataset used presents some ties. If you inspect the Kaplan-Meier table created in exercise 1, you can notice for example that as many as 9 children uttered their first sentence on the first day of the study (so there were 9 “ties”), and we observed other ties (e.g. 4 children uttered their first sentence 3 days into the study). Please refer to the references provided if you want to read more about the problems posed by ties and how to assess when the number of ties can create problems in model estimation.

The “coxph” command uses the Efron correction by default. However, other corrections are available. In this exercise, you can change the default simply by adding the option ties=”breslow” to the coxph. You can then inspect the estimates. In this example, the two methods provide very similar results.

1. Recover “baseline” functions and create functions for other “prototypical” groups: Compare a group of female from high SES backgrounds, females from deprived SES backgrounds, as well as males from high or low SES backgrounds. Recover the survivor and cumulative hazard function of these groups, and plot them.

**Solution:**

The first step is to estimate the coefficient of the model of interest, including predictors:

coxm2<-coxph(Surv(time, event) ~ male+highses+modses, data = dfex)

The coefficients of this model are now saved into object “coxm2”. We can now create some data frames that represent “prototypical” individuals.

For example, a female from a deprived background will be characterised as follows:

femaledepr<-data.frame(male=0, highses=0, modses=0)

The data frame “femaledepr” is constrained to have value of predictor “male” equal to zero (hence, female), and the values of dummies High SES and Middle SES set to zero (hence, the participant is from a deprived background).

Using the model coefficients saved in object “coxm2”, we can now “simulate” the estimate of the prototypical female from a deprived background:

s.femaledepr <- survfit(coxm2, newdata = femaledepr)

We can then extract these statistics into a dataset:

fd <- data.frame(

time = s.femaledepr$time,

surv = s.femaledepr$surv,

cumhaz = s.femaledepr$cumhaz,

group = "Female - Deprived SES"

)

Once we repeat these steps for the other prototypical cases, we can append all the estimated statistics into a single data frame I called “combineds”, and use these to create customised plots (see the scripts provided with these resources).

I created plots for the survivor and cumulative hazard functions of these prototypical cases, and put them together into a single customised graph (See overleaf).

