**Background**

In healthcare, we deal with a lot of binary outcomes. Death yes/no, disease recurrence yes/no, for instance. These outcomes are often easily analysed using binary logistic regression via finalfit().

When the time taken for the outcome to occur is important, we need a different approach. For instance, in patients with cancer, the time taken until recurrence of the cancer is often just as important as the fact it has recurred.

Finalfit wraps a number of functions to make these analyses easy to perform and output into PDFs and Word documents.

**Installation**

# Make sure finalfit is up-to-date

install.packages("finalfit")

**Dataset**

We’ll use the classic “Survival from Malignant Melanoma” dataset from the boot package to illustrate. The data consist of measurements made on patients with malignant melanoma. Each patient had their tumour removed by surgery at the Department of Plastic Surgery, University Hospital of Odense, Denmark during the period 1962 to 1977.

For the purposes of demonstration, we are interested in the association between tumour ulceration and survival after surgery.

**Get data and check**

library(finalfit)

melanoma = boot::melanoma #F1 here for help page with data dictionary

ff\_glimpse(melanoma)

#> Continuous

#> label var\_type n missing\_n missing\_percent mean sd

#> time time 205 0 0.0 2152.8 1122.1

#> status status 205 0 0.0 1.8 0.6

#> sex sex 205 0 0.0 0.4 0.5

#> age age 205 0 0.0 52.5 16.7

#> year year 205 0 0.0 1969.9 2.6

#> thickness thickness 205 0 0.0 2.9 3.0

#> ulcer ulcer 205 0 0.0 0.4 0.5

#> min quartile\_25 median quartile\_75 max

#> time 10.0 1525.0 2005.0 3042.0 5565.0

#> status 1.0 1.0 2.0 2.0 3.0

#> sex 0.0 0.0 0.0 1.0 1.0

#> age 4.0 42.0 54.0 65.0 95.0

#> year 1962.0 1968.0 1970.0 1972.0 1977.0

#> thickness 0.1 1.0 1.9 3.6 17.4

#> ulcer 0.0 0.0 0.0 1.0 1.0

#>

#> Categorical

#> data frame with 0 columns and 205 rows

As can be seen, all variables are coded as numeric and some need recoding to factors.

**Death status**

status is the the patients status at the end of the study.

* 1 indicates that they had died from melanoma;
* 2 indicates that they were still alive and;
* 3 indicates that they had died from causes unrelated to their melanoma.

There are three options for coding this.

* Overall survival: considering all-cause mortality, comparing 2 (alive) with 1 (died melanoma)/3 (died other);
* Cause-specific survival: considering disease-specific mortality comparing 2 (alive)/3 (died other) with 1 (died melanoma);
* Competing risks: comparing 2 (alive) with 1 (died melanoma) accounting for 3 (died other); see more below.

**Time and censoring**

time is the number of days from surgery until either the occurrence of the event (death) or the last time the patient was known to be alive. For instance, if a patient had surgery and was seen to be well in a clinic 30 days later, but there had been no contact since, then the patient’s status would be considered 30 days. This patient is censored from the analysis at day 30, an important feature of time-to-event analyses.

**Recode**

library(dplyr)

library(forcats)

melanoma = melanoma %>%

mutate(

# Overall survival

status\_os = case\_when(

status == 2 ~ 0, # "still alive"

TRUE ~ 1), # "died melanoma" or "died other causes"

# Diease-specific survival

status\_dss = case\_when(

status == 2 ~ 0, # "still alive"

status == 1 ~ 1, # "died of melanoma"

status == 3 ~ 0), # "died of other causes is censored"

# Competing risks regression

status\_crr = case\_when(

status == 2 ~ 0, # "still alive"

status == 1 ~ 1, # "died of melanoma"

status == 3 ~ 2), # "died of other causes"

# Label and recode other variables

age = ff\_label(age, "Age (years)"), # table friendly labels

thickness = ff\_label(thickness, "Tumour thickness (mm)"),

sex = factor(sex) %>%

fct\_recode("Male" = "1",

"Female" = "0") %>%

ff\_label("Sex"),

ulcer = factor(ulcer) %>%

fct\_recode("No" = "0",

"Yes" = "1") %>%

ff\_label("Ulcerated tumour")

)

**Kaplan-Meier survival estimator**

We can use the excellent survival package to produce the Kaplan-Meier (KM) survival estimator. This is a non-parametric statistic used to estimate the survival function from time-to-event data. Note use of %$% to expose left-side of pipe to older-style R functions on right-hand side.

library(survival)

survival\_object = melanoma %$%

Surv(time, status\_os)

# Explore:

head(survival\_object) # + marks censoring, in this case "Alive"

#> [1] 10 30 35+ 99 185 204

# Expressing time in years

survival\_object = melanoma %$%

Surv(time/365, status\_os)

**KM analysis for whole cohort**

**Model**

The survival object is the first step to performing univariable and multivariable survival analyses.

If you want to plot survival stratified by a single grouping variable, you can substitute “survival\_object ~ 1” by “survival\_object ~ factor”

# Overall survival in whole cohort

my\_survfit = survfit(survival\_object ~ 1, data = melanoma)

my\_survfit # 205 patients, 71 events

#> Call: survfit(formula = survival\_object ~ 1, data = melanoma)

#>

#> n events median 0.95LCL 0.95UCL

#> 205.00 71.00 NA 9.15 NA

**Life table**

A life table is the tabular form of a KM plot, which you may be familiar with. It shows survival as a proportion, together with confidence limits. The whole table is shown with summary(my\_survfit).

summary(my\_survfit, times = c(0, 1, 2, 3, 4, 5))

#> Call: survfit(formula = survival\_object ~ 1, data = melanoma)

#>

#> time n.risk n.event survival std.err lower 95% CI upper 95% CI

#> 0 205 0 1.000 0.0000 1.000 1.000

#> 1 193 11 0.946 0.0158 0.916 0.978

#> 2 183 10 0.897 0.0213 0.856 0.940

#> 3 167 16 0.819 0.0270 0.767 0.873

#> 4 160 7 0.784 0.0288 0.730 0.843

#> 5 122 10 0.732 0.0313 0.673 0.796

# 5 year overall survival is 73%

**Kaplan Meier plot**

We can plot survival curves using the finalfit wrapper for the package excellent package survminer. There are numerous options available on the help page. You should always include a number-at-risk table under these plots as it is essential for interpretation.

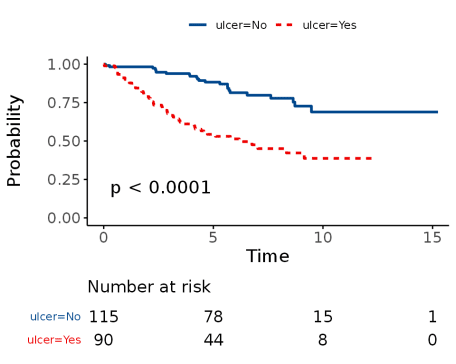
As can be seen, the probability of dying is much greater if the tumour was ulcerated, compared to those that were not ulcerated.

dependent\_os = "Surv(time/365, status\_os)"

explanatory = "ulcer"

melanoma %>%

surv\_plot(dependent\_os, explanatory, pval = TRUE)



**Cox-proportional hazards regression**

CPH regression can be performed using the all-in-one finalfit() function. It produces a table containing counts (proportions) for factors, mean (SD) for continuous variables and a univariable and multivariable CPH regression.

A hazard is the term given to the rate at which events happen.  
The probability that an event will happen over a period of time is the hazard multiplied by the time interval.   
An assumption of CPH is that hazards are constant over time (see below).

It produces a table containing counts (proportions) for factors, mean (SD) for continuous variables and a univariable and multivariable CPH regression.

**Univariable and multivariable models**

dependent\_os = "Surv(time, status\_os)"

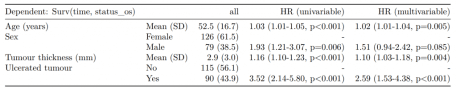
dependent\_dss = "Surv(time, status\_dss)"

dependent\_crr = "Surv(time, status\_crr)"

explanatory = c("age", "sex", "thickness", "ulcer")

melanoma %>%

finalfit(dependent\_os, explanatory)



The labelling of the final table can be easily adjusted as desired.

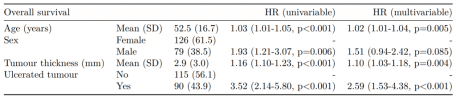
melanoma %>%

finalfit(dependent\_os, explanatory, add\_dependent\_label = FALSE) %>%

rename("Overall survival" = label) %>%

rename(" " = levels) %>%

rename(" " = all)



**Reduced model**

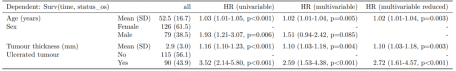
If you are using a backwards selection approach or similar, a reduced model can be directly specified and compared. The full model can be kept or dropped.

explanatory\_multi = c("age", "thickness", "ulcer")

melanoma %>%

finalfit(dependent\_os, explanatory, explanatory\_multi,

keep\_models = TRUE)



**Testing for proportional hazards**

An assumption of CPH regression is that the hazard associated with a particular variable does not change over time. For example, is the magnitude of the increase in risk of death associated with tumour ulceration the same in the early post-operative period as it is in later years.

The cox.zph() function from the survival package allows us to test this assumption for each variable. The plot of scaled Schoenfeld residuals should be a horizontal line. The included hypothesis test identifies whether the gradient differs from zero for each variable. No variable significantly differs from zero at the 5% significance level.

explanatory = c("age", "sex", "thickness", "ulcer", "year")

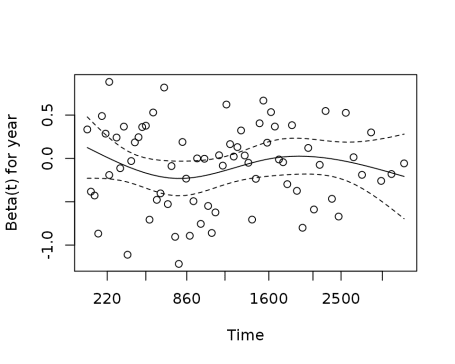
melanoma %>%

coxphmulti(dependent\_os, explanatory) %>%

cox.zph() %>%

{zph\_result <<- .} %>%

plot(var=5)



zph\_result

#> rho chisq p

#> age 0.1633 2.4544 0.1172

#> sexMale -0.0781 0.4473 0.5036

#> thickness -0.1493 1.3492 0.2454

#> ulcerYes -0.2044 2.8256 0.0928

#> year 0.0195 0.0284 0.8663

#> GLOBAL NA 8.4695 0.1322

**Stratified models**

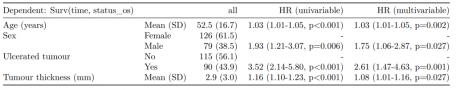
One approach to dealing with a violation of the proportional hazards assumption is to stratify by that variable. Including a strata() term will result in a separate baseline hazard function being fit for each level in the stratification variable. It will be no longer possible to make direct inference on the effect associated with that variable.

This can be incorporated directly into the explanatory variable list.

explanatory= c("age", "sex", "ulcer", "thickness", "strata(year)")

melanoma %>%

finalfit(dependent\_os, explanatory)



**Correlated groups of observations**

As a general rule, you should always try to account for any higher structure in the data within the model. For instance, patients may be clustered within particular hospitals.

There are two broad approaches to dealing with correlated groups of observations.

Including a cluster() term is akin to using generalised estimating equations (GEE). Here, a standard CPH model is fitted but the standard errors of the estimated hazard ratios are adjusted to account for correlations.

Including a frailty() term is akin to using a mixed effects model, where specific random effects term(s) are directly incorporated into the model.

Both approaches achieve the same goal in different ways. Volumes have been written on GEE vs mixed effects models. We favour the latter approach because of its flexibility and our preference for mixed effects modelling in generalised linear modelling. Note cluster() and frailty() terms cannot be combined in the same model.

# Simulate random hospital identifier

melanoma = melanoma %>%

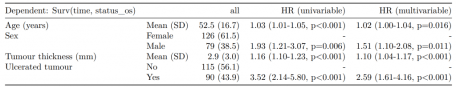
mutate(hospital\_id = c(rep(1:10, 20), rep(11, 5)))

# Cluster model

explanatory = c("age", "sex", "thickness", "ulcer", "cluster(hospital\_id)")

melanoma %>%

finalfit(dependent\_os, explanatory)

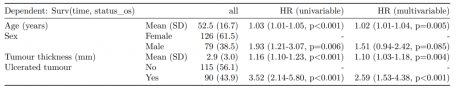


# Frailty model

explanatory = c("age", "sex", "thickness", "ulcer", "frailty(hospital\_id)")

melanoma %>%

finalfit(dependent\_os, explanatory)



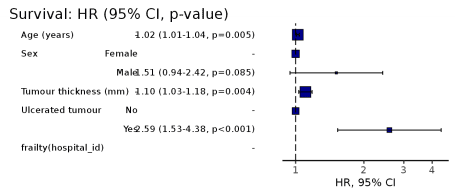
The frailty() method here is being superseded by the coxme package, and we’ll incorporate this soon.

**Hazard ratio plot**

A plot of any of the above models can be produced by passing the terms to hr\_plot().

melanoma %>%

hr\_plot(dependent\_os, explanatory)



**Competing risks regression**

Competing-risks regression is an alternative to CPH regression. It can be useful if the outcome of interest may not be able to occur because something else (like death) has happened first. For instance, in our example it is obviously not possible for a patient to die from melanoma if they have died from another disease first. By simply looking at cause-specific mortality (deaths from melanoma) and considering other deaths as censored, bias may result in estimates of the influence of predictors.

The approach by Fine and Gray is one option for dealing with this. It is implemented in the package cmprsk. The crr() syntax differs from survival::coxph() but finalfit brings these together.

It uses the finalfit::ff\_merge() function, which can join any number of models together.

explanatory = c("age", "sex", "thickness", "ulcer")

dependent\_dss = "Surv(time, status\_dss)"

dependent\_crr = "Surv(time, status\_crr)"

melanoma %>%

# Summary table

summary\_factorlist(dependent\_dss, explanatory,

column = TRUE, fit\_id = TRUE) %>%

# CPH univariable

ff\_merge(

melanoma %>%

coxphmulti(dependent\_dss, explanatory) %>%

fit2df(estimate\_suffix = " (DSS CPH univariable)")

) %>%

# CPH multivariable

ff\_merge(

melanoma %>%

coxphmulti(dependent\_dss, explanatory) %>%

fit2df(estimate\_suffix = " (DSS CPH multivariable)")

) %>%

# Fine and Gray competing risks regression

ff\_merge(

melanoma %>%

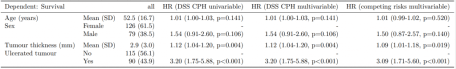
crrmulti(dependent\_crr, explanatory) %>%

fit2df(estimate\_suffix = " (competing risks multivariable)")

) %>%

select(-fit\_id, -index) %>%

dependent\_label(melanoma, "Survival")



**Summary**

So here we have various aspects of time-to-event analysis commonly used when looking at survival. There are many other applications, some which may not be obvious: for instance we use CPH for modelling length of stay in in hospital.

Stratification can be used to deal with non-proportional hazards in a particular variable.

Hierarchical structure in your data can be accommodated with cluster or frailty (random effects) terms.

Competing risks regression may be useful if your outcome is in competition with another, such as all-cause death, but is currently limited in its ability to accommodate hierarchical structures.