Introduction

Slides

Chapter slides here. (To convert html to pdf, press E → Print → Destination: Save to pdf)

R code

```
######################################################################
# This code generates the numerical results in chapter 1       ##
######################################################################

library(survival)  # For standard (univariate) survival analysis
library(Wcompo)    # For weighted total events (CompoML function)
library(rmt)       # For the hfaction data set (Heart Failure ACTION trial)
library(tidyverse) # For data wrangling (dplyr, ggplot2, etc.)

# load hfaction data
data(hfaction)
head(hfaction)
#> Shows the first rows of the hfaction data frame

# convert status=1 for death, 2=hospitalization
# (Swapping the original coding of event types in hfaction)
hfaction <- hfaction |>
  mutate(
    status = case_when(
      status == 1 ~ 2, # status previously "1" becomes "2" now
      status == 2 ~ 1, # status previously "2" becomes "1" now
      status == 0 ~ 0  # status "0" remains "0" (censoring)
    )
  )

head(hfaction)
```

```r
#> Check that the status values have been reassigned correctly

# count unique patients in each arm
hfaction |>
  group_by(trt_ab) |>
  distinct(patid) |>
  count(trt_ab)
#> Summarizes how many unique patients are in each treatment arm (trt_ab)

# TFE: take the first event per patient id
# This means each patient is represented by their earliest event (or censoring)
hfaction_TFE <- hfaction |>
  arrange(patid, time) |>
  group_by(patid) |>
  slice_head() |>
  ungroup()

# ------------------------------------------------------------------------
# Mortality analysis
# ------------------------------------------------------------------------

## Get mortality data
hfaction_D <- hfaction |>
  filter(status != 2) # remove hospitalization records (now coded as "2")

## Cox model for death against trt_ab
obj_D <- coxph(Surv(time, status) ~ trt_ab, data = hfaction_D)
summary(obj_D)
#> n= 426, number of events= 93
#> coef exp(coef) se(coef)      z Pr(>|z|)
#> trt_ab -0.3973    0.6721   0.2129 -1.866   0.0621 .
# Interpretation: a hazard ratio < 1 suggests lower hazard of death in the
# training arm, though p-value is ~0.06.

# ------------------------------------------------------------------------
# TFE analysis
# ------------------------------------------------------------------------

## how many of first events are death (1) or hosp (2)
hfaction_TFE |>
  count(status)
#> Tells how many first events were coded as status=1 or status=2
```

```r
# Cox model for TFE against trt_ab
# Here "status > 0" means any event (death or hospitalization) vs. censoring
obj_TFE <- coxph(Surv(time, status > 0) ~ trt_ab, data = hfaction_TFE)
summary(obj_TFE)
#> Similar interpretation as above, now for time to first event
# (either death or hospitalization).

# ------------------------------------------------------------------------
# Mortality vs TFE
# ------------------------------------------------------------------------

library(ggsurvfit) # For Kaplan-Meier plots using ggplot2
library(patchwork) # For combining plots side by side

pD <- survfit2(Surv(time, status) ~ trt_ab, data = hfaction_D) |>
  ggsurvfit(linewidth = 1) +
  scale_ggsurvfit() +
  scale_color_discrete(labels = c("Usual care", "Training")) +
  scale_x_continuous("Time (years)", limits = c(0, 4)) +
  labs(y = "Overall survival")
#> pD is the plot of overall survival (death only) for each arm

pTFE <- survfit2(Surv(time, status > 0) ~ trt_ab, data = hfaction_TFE) |>
  ggsurvfit(linewidth = 1) +
  scale_ggsurvfit() +
  scale_color_discrete(labels = c("Usual care", "Training")) +
  scale_x_continuous("Time (years)", limits = c(0, 4)) +
  labs(y = "Hospitalization-free survival")
#> pTFE is the plot for time to first event (death or hospitalization)

# Combine pD and pTFE plots side by side, sharing legends
pD + pTFE + plot_layout(guides = "collect") &
  theme(
    legend.position = "top",
    legend.text = element_text(size = 12)
  )

# ggsave("images/intro_hfaction_unis.png", width = 8, height = 4.5)
#> Uncomment to save the plot (if needed)

# ------------------------------------------------------------------------
# Total events (proportional mean)
```

```r
# ------------------------------------------------------------------------

## fit proportional means model with death = 2 x hosp
obj_ML <- CompoML(hfaction$patid, hfaction$time, hfaction$status,
                  hfaction$trt_ab, w = c(2, 1))
#> This model weights death as "2" and hospitalization as "1".
#> CompoML fits a marginal proportional means model for these outcomes.

## summary results
obj_ML
#> Provides estimates of the weighted event rates in each arm, etc.

## plot model-based mean functions
plot(obj_ML, 0, ylim= c(0, 5), xlim= c(0, 4), xlab="Time (years)",
     col = "red", lwd = 2)
plot(obj_ML, 1, add = TRUE, col = "blue", lwd=2)
legend(0, 5, col = c("red", "blue"), c("Usual care","Training"), lwd = 2)
#> Illustrates the cumulative mean count of "weighted" events over time
```