Bayesian Survival Analysis

Shannon Houser

10/11/2021

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.3     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(survival)
library(survminer)
```

```
## Warning: package 'survminer' was built under R version 4.1.1

## Loading required package: ggpubr

## Warning: package 'ggpubr' was built under R version 4.1.1
```

```
library(rstanarm)
```

```
## Warning: package 'rstanarm' was built under R version 4.1.1

## Loading required package: Rcpp

## This is rstanarm version 2.21.1

## - See https://mc-stan.org/rstanarm/articles/priors for changes to default priors!

## - Default priors may change, so it's safest to specify priors, even if equivalent to the defaults.

## - For execution on a local, multicore CPU with excess RAM we recommend calling

##   options(mc.cores = parallel::detectCores())
```

```
library(reshape2)
```

```
## Warning: package 'reshape2' was built under R version 4.1.1
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
got <- read_csv("210927_got.csv")
```

```
## Rows: 368 Columns: 68
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (15): character_name, gender, house, spouse, parents, siblings, parent_o...
## dbl (53): royal, kingsguard, s1_episodes, s1_screenTime, s1_numOfCharactersI...
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
got2 <- got %>%
  select(duration_in_episodes, is_dead, character_name, royal, house, gender)
```

Does there appear to be evidence of differential survival depending on certain characteristics (gender, royalty status, etc.)? What about potential confounders? For this lab, use a Bayesian approach. How do your conclusions match or differ from your analysis on the previous lab?

```
stan_surv <- function(formula01,
                      formula12,
                      formula02,
                      data,
                      basehaz         = "ms",
                      basehaz_ops,
                      qnodes          = 15,
                      prior           = normal(),
                      prior_intercept = normal(),
                      prior_aux       = normal(),
                      prior_smooth    = exponential(autoscale = FALSE),
                      prior_PD        = FALSE,
                      algorithm       = c("sampling", "meanfield", "fullrank"),
                      adapt_delta     = 0.95, ...) {
  #-----------------------------
  # Pre-processing of arguments
  #-----------------------------

  if (!requireNamespace("survival"))
```

```
    stop("the 'survival' package must be installed to use this function.")

  if (missing(basehaz_ops))
    basehaz_ops <- NULL
  if (missing(data) || !inherits(data, "data.frame"))
    stop("'data' must be a data frame.")

  dots      <- list(...)
  algorithm <- match.arg(algorithm)

  formula01   <- reshape2::parse_formula(formula01, data)
  formula12   <- reshape2::parse_formula(formula12, data)
  formula02   <- reshape2::parse_formula(formula02, data)
  data      <- make_model_data(formula$tf_form, data) # row subsetting etc.
  #-----------------
  # Construct data
  #-----------------

  #----- model frame stuff

  mf_stuff01 <- make_model_frame(formula01$tf_form, data)
  mf_stuff12 <- make_model_frame(formula12$tf_form, data)
  mf_stuff02 <- make_model_frame(formula02$tf_form, data)

  mf01 <- mf_stuff01$mf # model frame
  mf12 <- mf_stuff12$mf # model frame
  mf02 <- mf_stuff02$mf # model frame

  mt01 <- mf_stuff01$mt # model terms
  mt12 <- mf_stuff12$mt # model terms
  mt02 <- mf_stuff02$mt # model terms

  #----- dimensions and response vectors

  # entry and exit times for each row of data
  t_beg01 <- make_t(mf01, type = "beg") # entry time
  t_end12 <- make_t(mf12, type = "end") # exit  time
  t_upp02 <- make_t(mf02, type = "upp") # upper time for interval censoring
  # ensure no event or censoring times are zero (leads to degenerate
  # estimate for log hazard for most baseline hazards, due to log(0))
  check1 <- any(t_end <= 0, na.rm = TRUE)
  check2 <- any(t_upp <= 0, na.rm = TRUE)
  if (check1 || check2)
    stop2("All event and censoring times must be greater than 0.")

}
```

```
m1 <- stan_surv(Surv(duration_in_episodes, is_dead) ~ royal + house + gender,
          data = got2,
          prior = normal(autoscale = T),
          prior_intercept = normal(autoscale = T),
          prior_aux = cauchy(0,5,autoscale = T),
          chains = 2, iter = 2000, seed = 123,
```

```
          prior_PD = F)

rstanarm::stan_surv()
```

Could not get rstanarm to work after several hours of removing and installing in different ways.