Workbook for Introduction to TTE modeling

Relationship between functions

2023-07-11

Contents

Preliminaries for R examples

```
library(tidyverse)
library(stringr)
library(survival)
library(survminer)
library(texreg)
library(mgcv)
library(flexsurv)
library(muhaz)
library(Hmisc)


theme_set(theme_bw())
```

Plotting hazard and survival functions over time for some common distributions

To better understand the relationship between the hazard, cumulative hazard, survival and density functions, let's plot them for a three commonly used parametric distributions in TTE modeling: exponential, Weibull, and log-normal.

Exponential distribution

The hazard is constant as a function of time: $h(t) = \lambda$.

From first principles: * h(t) = lambda * H(t) = lambda * t * S(t) = exp(-lambda * t) * f(t) = h(t) * S(t) = lambda * exp(-lambda * t)

```
plot_data <- crossing(lambda = c(0.5,1,2), times = seq(0,5,length=100)) %>%
  mutate(hazard = lambda,
         cumulative_hazard = lambda * times,
         survival = exp(-cumulative_hazard),
```
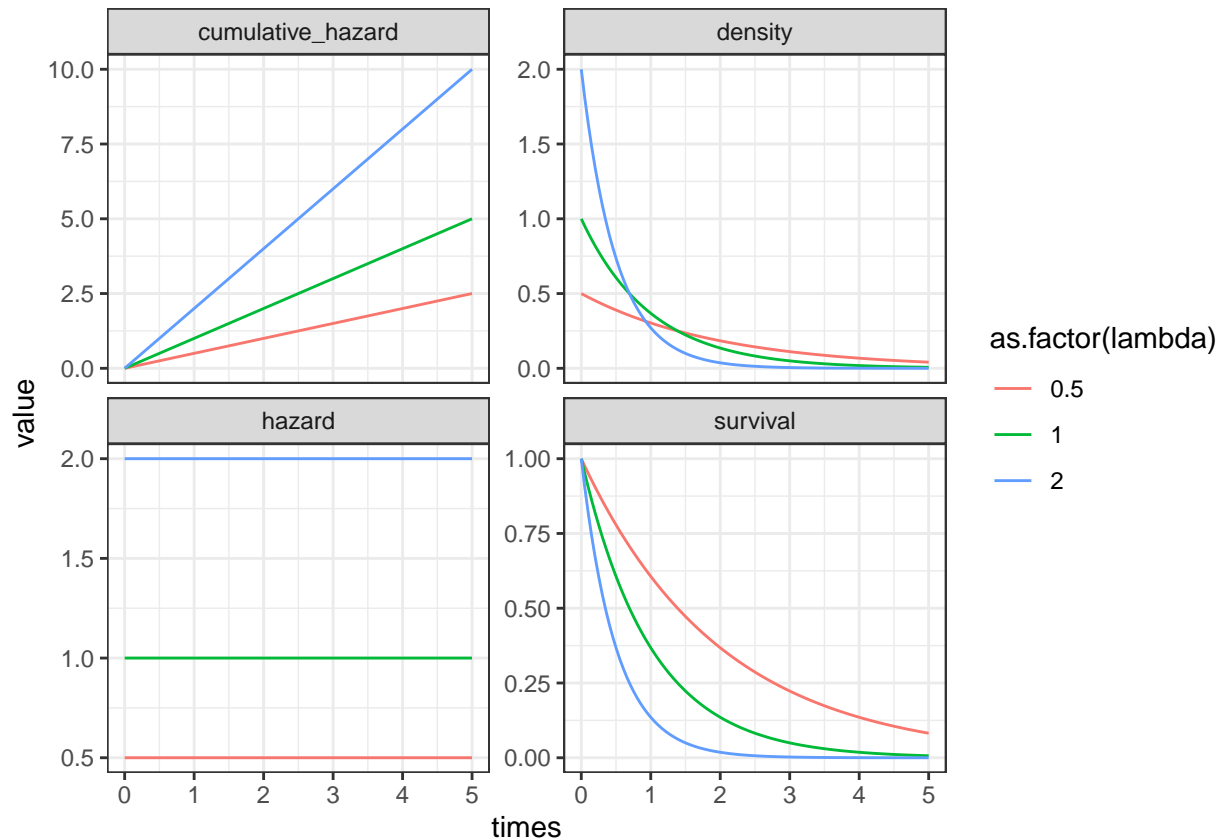
```
            density = hazard * survival)

plot_data %>%
  pivot_longer(cols=hazard:density, names_to = 'type', values_to='value') %>%
  ggplot(aes(x=times, y=value, col=as.factor(lambda), group=lambda)) +
  geom_line() +
  facet_wrap(~type, scales = "free_y")
```


Weibull

The Weibull distribution has two parameters: lambda and gamma

The log hazard is linear in the log of time:

$$h(t) = \gamma\lambda t^{\gamma-1} \iff \log h(t) = \log\gamma + \log\lambda + (\gamma - 1)\log t$$

From first principles: * h(t) = lambda * gamma * t^(gamma-1) * H(t) = lambda * t^gamma * S(t) = exp(-lambda * t^gamma) * f(t) = h(t) * S(t) = lambda * gamma * t^(gamma-1) * exp(-lambda * t^gamma)

```
plot_data <- crossing(lambda = c(0.5,2), gamma = c(0.5, 1, 3), times = seq(0,5,length=100)) %>%
  mutate(hazard = lambda * gamma * times^(gamma-1),
         cumulative_hazard = lambda * times^gamma,
         survival = exp(-cumulative_hazard),
         density = hazard * survival
)

plot_data %>%
```
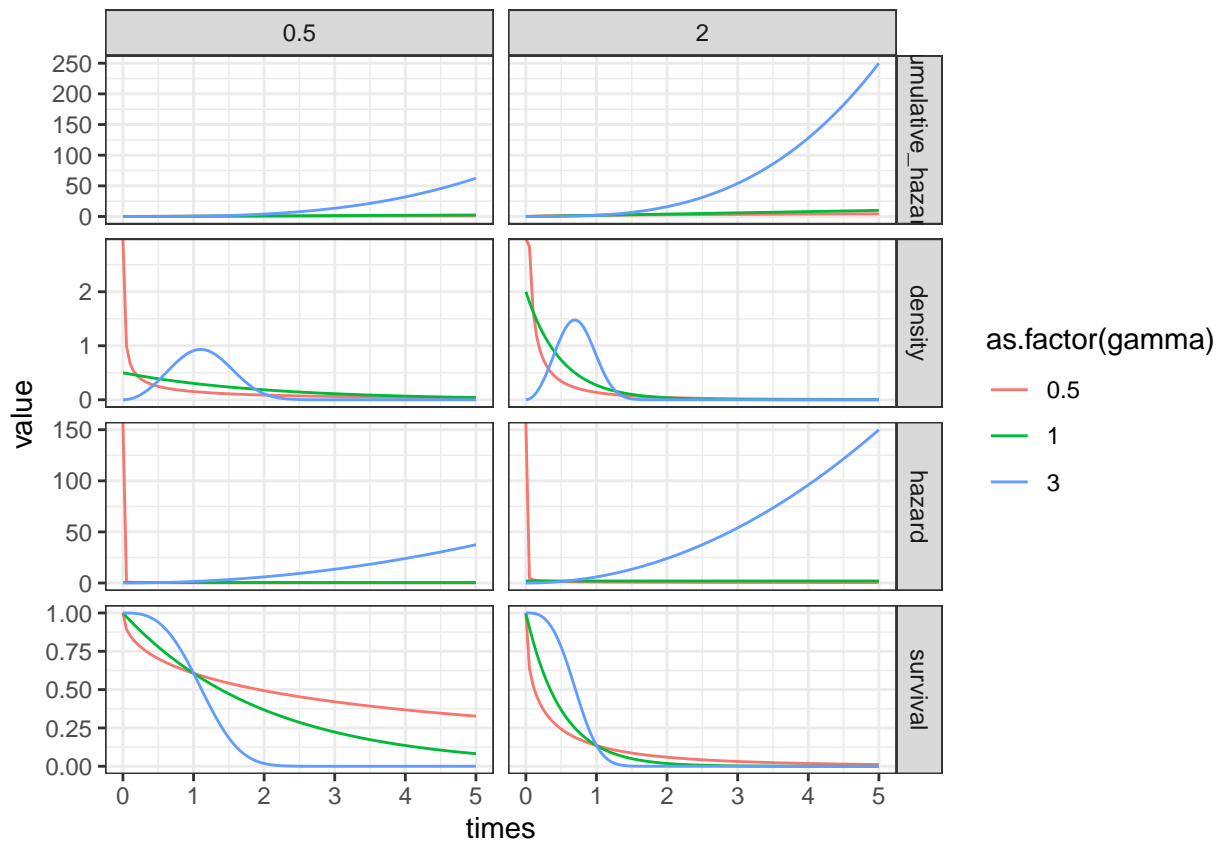
```
   pivot_longer(cols=hazard:density, names_to = 'type', values_to='value') %>%
   mutate(combination = paste(lambda,gamma,sep='-')) %>%
   ggplot(aes(x=times, y=value, col=as.factor(gamma), group=combination)) +
   geom_line() +
   facet_grid(type~lambda, scales = "free_y")
```


Log-normal

The hazard isn't easily expressed in closed form. How can you derive it from first principles? Edit the code below to make the plots.

Hint: the log normal pdf and survival function are obtained using the `dlnorm` and `plnorm` functions, respectively.

Plot the log-normal hazard function for combinations of mean (1, 2) and standard deviation (0.3,1) modifying the code below.

Solution:

```
plot_data <- crossing(mean = c(1,2), sd = c(0.3, 1), times = seq(0,5,length=100)) %>%
  mutate(survival = plnorm(q = times, meanlog = mean, sdlog=sd),
         density = dlnorm(x = times, meanlog = mean, sdlog=sd),
         cumulative_hazard = -log(survival),
         hazard = density / survival
)


plot_data %>%
  pivot_longer(cols=survival:hazard, names_to = 'type', values_to='value') %>%
```

```
mutate(combination = paste(mean,sd,sep='-')) %>%
ggplot(aes(x=times, y=value, col=as.factor(sd), group=combination)) +
geom_line() +
facet_grid(type~mean, scales = "free_y")
```
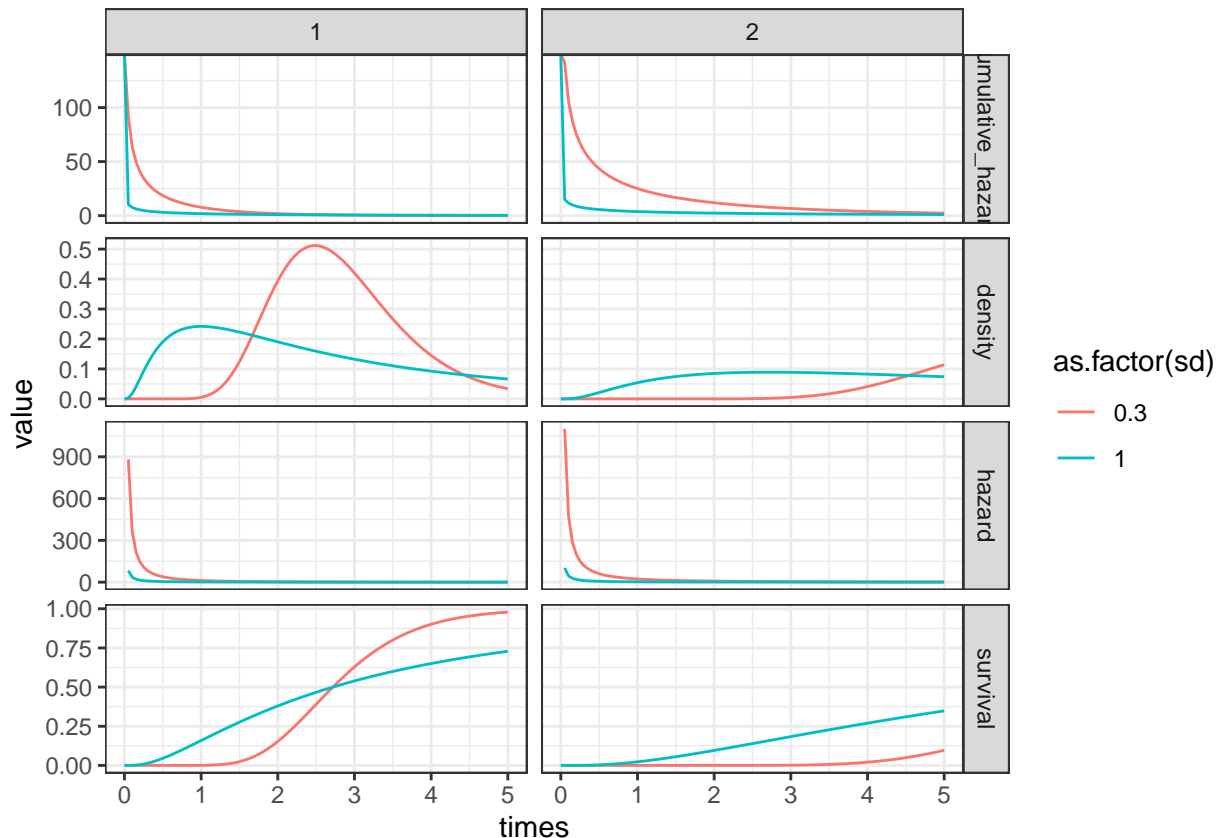

Bonus exercise

There is no reason the hazard function needs to follow one of these functions, or any function associated with a 'named' probability distribution.

Make up your own hazard function and calculate the cumulative hazard, survival and density functions.

Solution:

Let's set up a hazard function that declines exponentially from a starting value of theta0:

$$h(t) = \theta_0 * \exp(-\lambda \cdot t)$$

With this hazard function, the cumulative hazard function is

$$\int_0^t h(s) \; ds = \theta_0/\lambda * (1 - \exp(-\lambda \cdot t))$$

This is an example of a cure-rate model, where the survival function has a limiting value other than 0.

```
plot_data <- crossing(theta0 = c(1,2), lambda = c(1,2), times = seq(0,5,length=100)) %>%
  mutate(hazard = theta0 * exp(-lambda*times),
         cumulative_hazard = theta0/lambda*(1 - exp(-lambda * times)),
         survival = exp(-cumulative_hazard),
```

4

```
        density = hazard * survival
)

plot_data %>%
  pivot_longer(cols=hazard:density, names_to = 'type', values_to='value') %>%
  mutate(combination = paste(theta0,lambda,sep='-')) %>%
  ggplot(aes(x=times, y=value, col=as.factor(lambda), group=combination)) +
  geom_line() +
  facet_grid(type~theta0, scales = "free_y")
```