CGD analysis

```
library(purrr)
library(reshape2)
library(dplyr)
library(rstan)
library(dplyr)
library(glue)
library(abind)
library(survival)
library(multimcm)
library(frailtyHL)

library(multimcm)

# rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores() - 1)
```

Introduction

The analysis is a comparison with a frequentist analysis in Lai and Yau (2009).

Data

The data is chronic granulomatous disease (CGD) data available as part of the `{frailtyHL}` package.

```
data(cgd)
head(cgd)
#>   id            center     random   treat    sex age height weight   inherit
#> 1  1 Scripps Institute 1989-06-07  rIFN-g female  12    147   62.0 autosomal
#> 2  1 Scripps Institute 1989-06-07  rIFN-g female  12    147   62.0 autosomal
#> 3  1 Scripps Institute 1989-06-07  rIFN-g female  12    147   62.0 autosomal
#> 4  2 Scripps Institute 1989-06-07 placebo   male  15    159   47.5 autosomal
#> 5  2 Scripps Institute 1989-06-07 placebo   male  15    159   47.5 autosomal
#> 6  2 Scripps Institute 1989-06-07 placebo   male  15    159   47.5 autosomal
#>   steroids propylac  hos.cat tstart enum tstop status
#> 1        0        0 US:other      0    1   219      1
#> 2        0        0 US:other    219    2   373      1
#> 3        0        0 US:other    373    3   414      0
#> 4        0        1 US:other      0    1     8      1
#> 5        0        1 US:other      8    2    26      1
#> 6        0        1 US:other     26    3   152      1
```

Create a new variable for the duration until the event.

```
cgd <- mutate(cgd, time = tstop - tstart)
# center_id = as.numeric(as.factor(center))
```

The data has repeat measurements and 13 hospitals so for simplicity we'll remove some of the data in the initial analyses. Take the first event for each individual by start time.

```
cgd_first <-
  cgd |>
  group_by(id) |>
  arrange(tstart) |>
  filter(row_number() == 1)
```

Drop any levels which only appeared in the data we've just removed and convert the variables to numeric or factor for unique and categorical data, respectively.

```
cgd_center <- cgd_first |>
  filter(center %in% c("Amsterdam", "NIH", "Univ. of Zurich", "Scripps Institute")) |>
  droplevels() |>
  mutate(center_id = as.numeric(center),
         hos.cat_id = as.numeric(hos.cat),
         sex_id = as.factor(as.numeric(sex)))
```

Next, append the background hazard rate by age, sex and country. We first need to harmonise the country names and their associated cities. Then we join the data with the background mortality data according to age, sex and country.

```
cgd_center <-
  cgd_center |>
  tidyr::separate(hos.cat, c("country", "hospital"),
                  remove = FALSE) |>
  mutate(country = toupper(country),
         age_event = round(age + time/365.25, 0),    # convert from days to years
         country = ifelse(center == "Amsterdam", "NETHERLANDS",
                          ifelse(center == "Univ. of Zurich", "SWITZERLAND",
                                 ifelse(center == "Copenhagen", "DENMARK", country)))) |>
  select(-hospital)

# background mortality data
load(here::here("../bgfscure/data/bg.mortality.RData"))

# haramonise fields
bg.mortality <- bg.mortality |>
  mutate(ACOUNTRY = toupper(ACOUNTRY),
         ACOUNTRY = ifelse(ACOUNTRY == "UNITED STATES", "US", ACOUNTRY),
         SEX = ifelse(SEX == "M", "male", "female"),
         rate = ifelse(rate == 0, 1e-10, rate)) |>  # replace so >0
  rename(country = ACOUNTRY,
         sex = SEX,
         age_event = Age)

input_data <- merge(cgd_center, bg.mortality,
                    by = c("age_event", "sex", "country"), sort = FALSE)
```

That is the last step in the data wrangling. We can now proceed to the analysis.

Analysis

By center

No covariates in the latent model and the incidence (cure) model as

$$T \sim \text{Exp}(\lambda)\pi_i = \text{logit}^{-1}(\alpha + \beta_{\text{treat}[i]} + \gamma_{\text{center}[i]})\text{center}[i] \sim N(\mu_{\text{center}}, \sigma^2_{\text{center}})$$

```
out <-
  bmcm_stan(
    input_data = input_data,
    formula = "Surv(time=time, event=status) ~ 1",
    cureformula = "~ treat + (1 | center_id)",
    family_latent = "exponential",
    centre_coefs = TRUE,
    bg_model = "bg_fixed",
    bg_varname = "rate",
    bg_hr = 1,
    t_max = 400)
#>
#> SAMPLING FOR MODEL 'bmcm_stan_exp_exp_exp_exp' NOW (CHAIN 1).
#> Chain 1:
#> Chain 1: Gradient evaluation took 0.000209 seconds
#> Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 2.09 seconds.
#> Chain 1: Adjust your expectations accordingly!
#> Chain 1:
#> Chain 1:
#> Chain 1: WARNING: There aren't enough warmup iterations to fit the
#> Chain 1:          three stages of adaptation as currently configured.
#> Chain 1:          Reducing each adaptation stage to 15%/75%/10% of
#> Chain 1:          the given number of warmup iterations:
#> Chain 1:            init_buffer = 15
#> Chain 1:            adapt_window = 75
#> Chain 1:            term_buffer = 10
#> Chain 1:
#> Chain 1: Iteration:   1 / 500 [  0%]  (Warmup)
#> Chain 1: Iteration:  50 / 500 [ 10%]  (Warmup)
#> Chain 1: Iteration: 100 / 500 [ 20%]  (Warmup)
#> Chain 1: Iteration: 101 / 500 [ 20%]  (Sampling)
#> Chain 1: Iteration: 150 / 500 [ 30%]  (Sampling)
#> Chain 1: Iteration: 200 / 500 [ 40%]  (Sampling)
#> Chain 1: Iteration: 250 / 500 [ 50%]  (Sampling)
#> Chain 1: Iteration: 300 / 500 [ 60%]  (Sampling)
#> Chain 1: Iteration: 350 / 500 [ 70%]  (Sampling)
#> Chain 1: Iteration: 400 / 500 [ 80%]  (Sampling)
#> Chain 1: Iteration: 450 / 500 [ 90%]  (Sampling)
#> Chain 1: Iteration: 500 / 500 [100%]  (Sampling)
#> Chain 1:
#> Chain 1:  Elapsed Time: 0.761 seconds (Warm-up)
#> Chain 1:                2.307 seconds (Sampling)
#> Chain 1:                3.068 seconds (Total)
#> Chain 1:
#> Warning in validityMethod(object): The following variables have undefined
#> values: log_lik,The following variables have undefined values: log_lik_1,The
```

```
#> following variables have undefined values: log_lik_2,The following variables
#> have undefined values: log_lik_3,The following variables have undefined values:
#> log_lik_4. Many subsequent functions will not work correctly.
#> Warning: There were 5 divergent transitions after warmup. See
#> https://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
#> to find out why this is a problem and how to eliminate them.
#> Warning: Examine the pairs() plot to diagnose sampling problems
#> Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
#> Running the chains for more iterations may help. See
#> https://mc-stan.org/misc/warnings.html#bulk-ess
#> Warning: Tail Effective Samples Size (ESS) is too low, indicating posterior variances and tail quant
#> Running the chains for more iterations may help. See
#> https://mc-stan.org/misc/warnings.html#tail-ess
```

```
    # t_max = 365)
```

By hospital category

No covariates in the latent model and the incidence (cure) model as

$$T \sim \text{Exp}(\lambda)\pi_i = \text{logit}^{-1}(\alpha + \beta_{\text{treat}[i]} + \gamma_{\text{cat}[i]})\text{cat}[i] \sim N(\mu_{\text{cat}}, \sigma^2_{\text{cat}})$$

```
out_hos.cat <-
  bmcm_stan(
    input_data = input_data,
    formula = "Surv(time=time, event=status) ~ 1",
    cureformula = "~ treat + (1 | hos.cat_id)",
    family_latent = "exponential",
    centre_coefs = TRUE,
    bg_model = "bg_fixed",
    bg_varname = "rate",
    bg_hr = 1,
    t_max = 365)
#>
#> SAMPLING FOR MODEL 'bmcm_stan_exp_exp_exp_exp' NOW (CHAIN 1).
#> Chain 1:
#> Chain 1: Gradient evaluation took 0.000205 seconds
#> Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 2.05 seconds.
#> Chain 1: Adjust your expectations accordingly!
#> Chain 1:
#> Chain 1:
#> Chain 1: WARNING: There aren't enough warmup iterations to fit the
#> Chain 1:          three stages of adaptation as currently configured.
#> Chain 1:          Reducing each adaptation stage to 15%/75%/10% of
#> Chain 1:          the given number of warmup iterations:
#> Chain 1:            init_buffer = 15
#> Chain 1:            adapt_window = 75
#> Chain 1:            term_buffer = 10
#> Chain 1:
#> Chain 1: Iteration:   1 / 500 [  0%]  (Warmup)
#> Chain 1: Iteration:  50 / 500 [ 10%]  (Warmup)
#> Chain 1: Iteration: 100 / 500 [ 20%]  (Warmup)
```

```
#> Chain 1: Iteration: 101 / 500 [ 20%]  (Sampling)
#> Chain 1: Iteration: 150 / 500 [ 30%]  (Sampling)
#> Chain 1: Iteration: 200 / 500 [ 40%]  (Sampling)
#> Chain 1: Iteration: 250 / 500 [ 50%]  (Sampling)
#> Chain 1: Iteration: 300 / 500 [ 60%]  (Sampling)
#> Chain 1: Iteration: 350 / 500 [ 70%]  (Sampling)
#> Chain 1: Iteration: 400 / 500 [ 80%]  (Sampling)
#> Chain 1: Iteration: 450 / 500 [ 90%]  (Sampling)
#> Chain 1: Iteration: 500 / 500 [100%]  (Sampling)
#> Chain 1:
#> Chain 1:  Elapsed Time: 0.8 seconds (Warm-up)
#> Chain 1:                3.255 seconds (Sampling)
#> Chain 1:                4.055 seconds (Total)
#> Chain 1:
#> Warning in validityMethod(object): The following variables have undefined
#> values: log_lik,The following variables have undefined values: log_lik_1,The
#> following variables have undefined values: log_lik_2,The following variables
#> have undefined values: log_lik_3,The following variables have undefined values:
#> log_lik_4. Many subsequent functions will not work correctly.
#> Warning: There were 3 divergent transitions after warmup. See
#> https://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
#> to find out why this is a problem and how to eliminate them.
#> Warning: Examine the pairs() plot to diagnose sampling problems
#> Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
#> Running the chains for more iterations may help. See
#> https://mc-stan.org/misc/warnings.html#bulk-ess
#> Warning: Tail Effective Samples Size (ESS) is too low, indicating posterior variances and tail quant
#> Running the chains for more iterations may help. See
#> https://mc-stan.org/misc/warnings.html#tail-ess
```

With sex covariate in latent model

No covariates in the latent model and the incidence (cure) model as

$$T \sim \text{Exp}(\lambda_i)\lambda_i = \alpha_\lambda + \beta_{\text{sex}[i]}\pi_i = \text{logit}^{-1}(\alpha_\pi + \beta_{\text{treat}[i]} + \gamma_{\text{center}[i]})\text{center}[i] \sim N(\mu_{\text{center}}tyg, \sigma^2_{\text{center}})$$

```
out_sex <-
  bmcm_stan(
    input_data = input_data,
    formula = "Surv(time=time, event=status) ~ sex",
    cureformula = "~ treat + (1 | hos.cat_id)",
    family_latent = "exponential",
    bg_model = "bg_fixed",
    bg_varname = "rate",
    bg_hr = 1,
    t_max = 365)
#>
#> SAMPLING FOR MODEL 'bmcm_stan_exp_exp_exp_exp' NOW (CHAIN 1).
#> Chain 1:
#> Chain 1: Gradient evaluation took 0.000213 seconds
#> Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 2.13 seconds.
#> Chain 1: Adjust your expectations accordingly!
#> Chain 1:
```

```
#> Chain 1:
#> Chain 1: WARNING: There aren't enough warmup iterations to fit the
#> Chain 1:          three stages of adaptation as currently configured.
#> Chain 1:          Reducing each adaptation stage to 15%/75%/10% of
#> Chain 1:          the given number of warmup iterations:
#> Chain 1:            init_buffer = 15
#> Chain 1:            adapt_window = 75
#> Chain 1:            term_buffer = 10
#> Chain 1:
#> Chain 1: Iteration:   1 / 500 [  0%]  (Warmup)
#> Chain 1: Iteration:  50 / 500 [ 10%]  (Warmup)
#> Chain 1: Iteration: 100 / 500 [ 20%]  (Warmup)
#> Chain 1: Iteration: 101 / 500 [ 20%]  (Sampling)
#> Chain 1: Iteration: 150 / 500 [ 30%]  (Sampling)
#> Chain 1: Iteration: 200 / 500 [ 40%]  (Sampling)
#> Chain 1: Iteration: 250 / 500 [ 50%]  (Sampling)
#> Chain 1: Iteration: 300 / 500 [ 60%]  (Sampling)
#> Chain 1: Iteration: 350 / 500 [ 70%]  (Sampling)
#> Chain 1: Iteration: 400 / 500 [ 80%]  (Sampling)
#> Chain 1: Iteration: 450 / 500 [ 90%]  (Sampling)
#> Chain 1: Iteration: 500 / 500 [100%]  (Sampling)
#> Chain 1:
#> Chain 1:  Elapsed Time: 8.899 seconds (Warm-up)
#> Chain 1:                32.175 seconds (Sampling)
#> Chain 1:                41.074 seconds (Total)
#> Chain 1:
#> Warning in validityMethod(object): The following variables have undefined
#> values: log_lik,The following variables have undefined values: log_lik_1,The
#> following variables have undefined values: log_lik_2,The following variables
#> have undefined values: log_lik_3,The following variables have undefined values:
#> log_lik_4. Many subsequent functions will not work correctly.
#> Warning: There were 1 divergent transitions after warmup. See
#> https://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
#> to find out why this is a problem and how to eliminate them.
#> Warning: Examine the pairs() plot to diagnose sampling problems
#> Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
#> Running the chains for more iterations may help. See
#> https://mc-stan.org/misc/warnings.html#bulk-ess
#> Warning: Tail Effective Samples Size (ESS) is too low, indicating posterior variances and tail quant
#> Running the chains for more iterations may help. See
#> https://mc-stan.org/misc/warnings.html#tail-ess
```

Plots

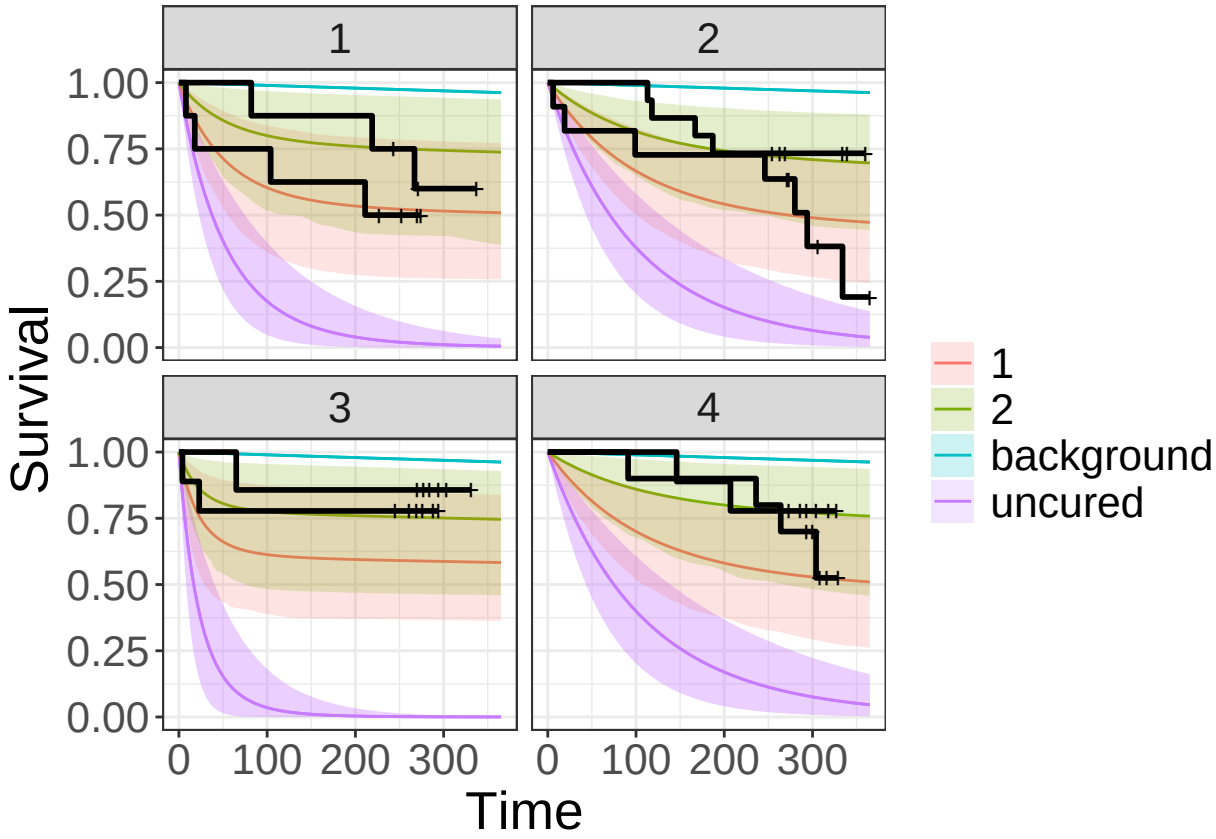
After fitting the models, we can plot the survival curves for each.

```
library(ggplot2)
#> Warning: package 'ggplot2' was built under R version 4.3.3
```

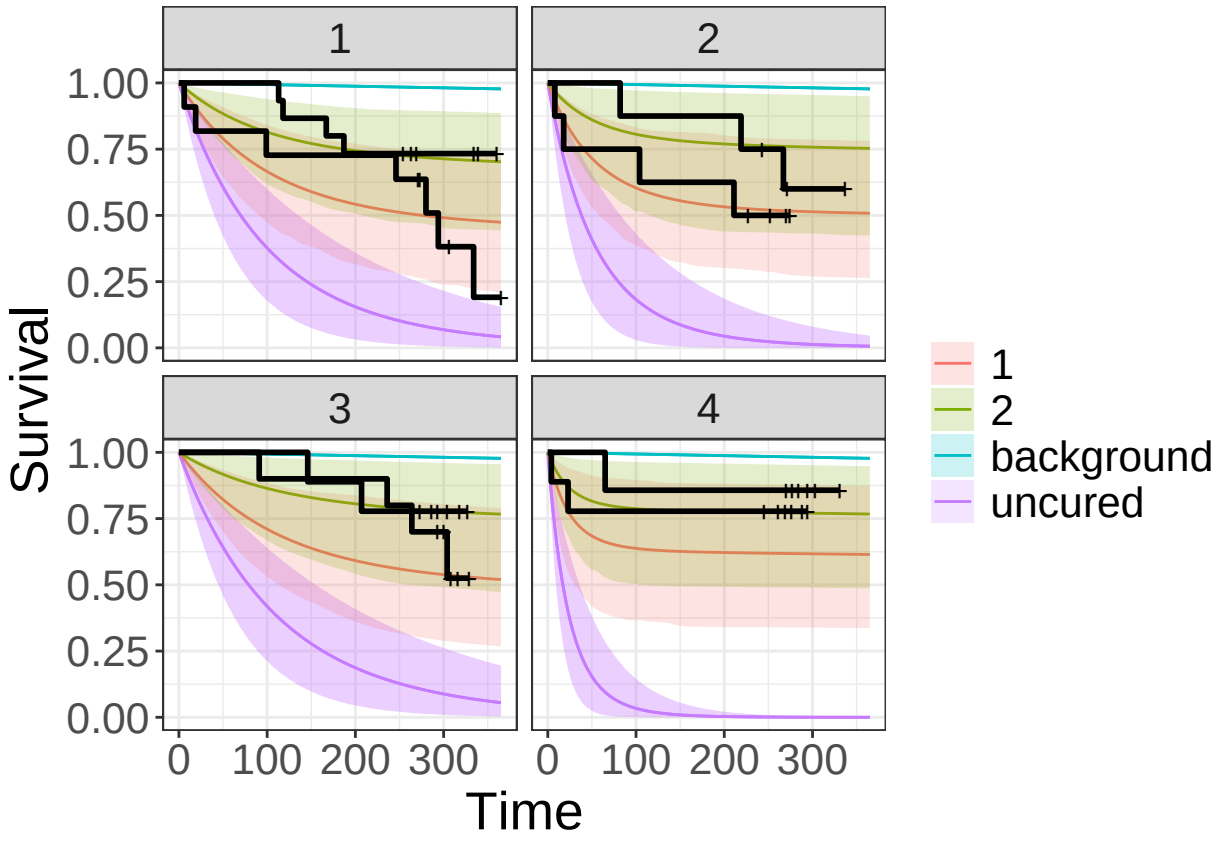
```
gg <- plot_S_joint(out,
                   add_km = TRUE,
                   annot_cf = FALSE)
```

```
#> Warning: The `x` argument of `as_tibble.matrix()` must have unique column names if
#> `.name_repair` is omitted as of tibble 2.0.0.
#> i Using compatibility `.name_repair`.
#> i The deprecated feature was likely used in the multimcm package.
#>   Please report the issue at <https://github.com/n8thangreen/multimcm/issues/>.
#> This warning is displayed once every 8 hours.
#> Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
#> generated.
```

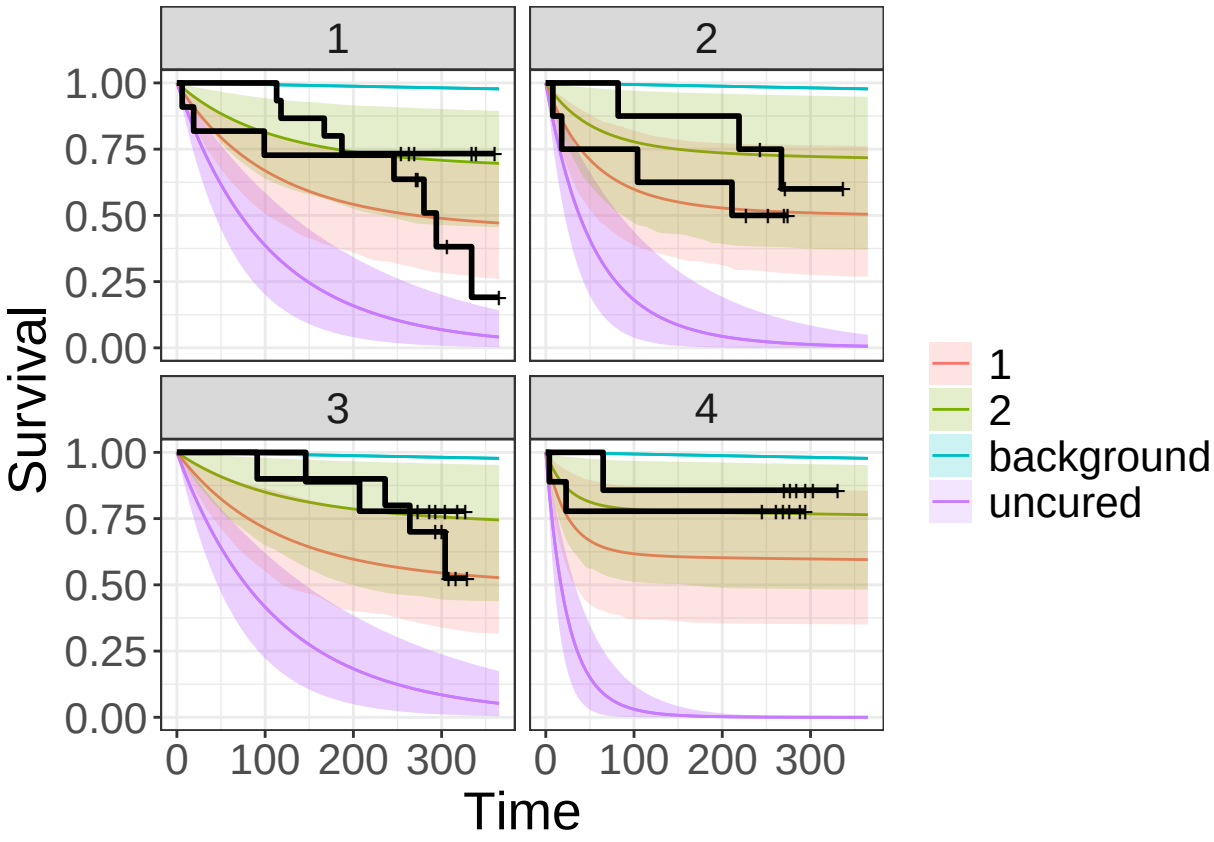
```
gg + xlim(0,365) + facet_wrap(~endpoint)
#> Scale for x is already present.
#> Adding another scale for x, which will replace the existing scale.
#> Warning: Removed 630 rows containing missing values or values outside the scale range
#> (`geom_line()`).
#> Warning: Removed 5 rows containing missing values or values outside the scale range
#> (`geom_text()`).
```




```
gg2 <- plot_S_joint(out_hos.cat,
                    add_km = TRUE,
                    annot_cf = FALSE)
gg2 + xlim(0,365) + facet_wrap(~endpoint)
#> Scale for x is already present.
#> Adding another scale for x, which will replace the existing scale.
#> Warning: Removed 5 rows containing missing values or values outside the scale range
#> (`geom_text()`).
```




```
gg3 <- plot_S_joint(out_sex,
                    add_km = TRUE,
                    annot_cf = FALSE)
gg3 + xlim(0,365) + facet_wrap(~endpoint)
#> Scale for x is already present.
#> Adding another scale for x, which will replace the existing scale.
#> Warning: Removed 5 rows containing missing values or values outside the scale range
#> (`geom_text()`).
```



References

Lai, Xin, and Kelvin K. W. Yau. 2009. "Multilevel Mixture Cure Models with Random Effects." *Biometrical Journal* 51 (3): 456–66. https://doi.org/10.1002/bimj.200800222.