Modestly-weighted logrank test: basic tutorial

Dominic Magirr

2018-07-31

Installation

You can use `devtools::install_github()` to get the package from GitHub:

```
install.packages("devtools")
library(devtools)
install_github("dominicmagirr/modestWLRT")
```

Load packages

```
library(dplyr)
library(ggplot2)
library(modestWLRT)
```

Simulate example data set

You can use the function `delayed_effect_sim` to simulate an example data set from a 2-arm RCT. Survival times on the control arm are exponentially distributed with median `med_c`. Survival times on the experimental arm follow a 2-piece exponential distibution: from time zero up to time `delay` the event rate is `rate_e_1`; thereafter the event rate is `rate_e_2`. Patient recruitment times follow a simple power distribution:

pr(recruited before t) = (t / `rec_period`)^`rec_power`, for t in (0, `rec_period`).

Data cut-off happens at time `max_cal_t`, and any patients still alive have their survival time censored.

```
example_data = delayed_effect_sim(n_c = 10,
                                  n_e = 10,
                                  rec_period = 12,
                                  rec_power = 1,
                                  med_c = 15,
                                  rate_e_1 = log(2) / 15,
                                  rate_e_2 = 0.03,
                                  delay = 6,
                                  max_cal_t  = 36)

example_data
#>     time event       group
#> 1   9.09  TRUE     control
#> 2  12.64  TRUE     control
#> 3  34.41 FALSE     control
#> 4  25.68 FALSE     control
#> 5  13.94  TRUE     control
#> 6   3.64  TRUE     control
#> 7   5.09  TRUE     control
#> 8   4.78  TRUE     control
#> 9   5.41  TRUE      control
#> 10  0.85  TRUE      control
#> 11 18.32  TRUE experimental
#> 12 17.57  TRUE experimental
#> 13 31.10  TRUE experimental
#> 14 23.95  TRUE experimental
#> 15 30.48 FALSE experimental
#> 16 33.42 FALSE experimental
#> 17  6.10  TRUE experimental
#> 18  4.31  TRUE experimental
#> 19 20.57  TRUE experimental
#> 20 25.20 FALSE experimental
```

Risk table

The function `get_risk_table` takes a data frame produced from `delayed_effect_sim` (or a data frame of the same form) and turns it into a risk table. This tells you how many patients were at risk / had an event / censored on each arm, at each event time.

```
example_risk_table = get_risk_table(example_data)

example_risk_table
#>        t n_e d_e n_c d_c  n d l l_c l_e
#> 1   0.85  10   0  10   1 20 1 0   0   0
#> 2   3.64  10   0   9   1 19 1 0   0   0
#> 3   4.31  10   1   8   0 18 1 0   0   0
#> 4   4.78   9   0   8   1 17 1 0   0   0
#> 5   5.09   9   0   7   1 16 1 0   0   0
#> 6   5.41   9   0   6   1 15 1 0   0   0
#> 7   6.10   9   1   5   0 14 1 0   0   0
#> 8   9.09   8   0   5   1 13 1 0   0   0
#> 9  12.64   8   0   4   1 12 1 0   0   0
#> 10 13.94   8   0   3   1 11 1 0   0   0
#> 11 17.57   8   1   2   0 10 1 0   0   0
#> 12 18.32   7   1   2   0  9 1 0   0   0
#> 13 20.57   6   1   2   0  8 1 0   0   0
#> 14 23.95   5   1   2   0  7 1 3   1   2
#> 18 31.10   2   1   1   0  3 1 2   1   1
```

Calculate weights

From the risk table, you can calculate the scores / weights from a modestWLRT. The argument `delay` is used to specify how long the scores are kept constant. See the paper

http://arxiv.org/abs/1807.11097

for details.

```
modest_weights = add_weights(example_risk_table,
                             method = "fixed_c",
                             delay = 12,
                             plot_weights = TRUE)
```

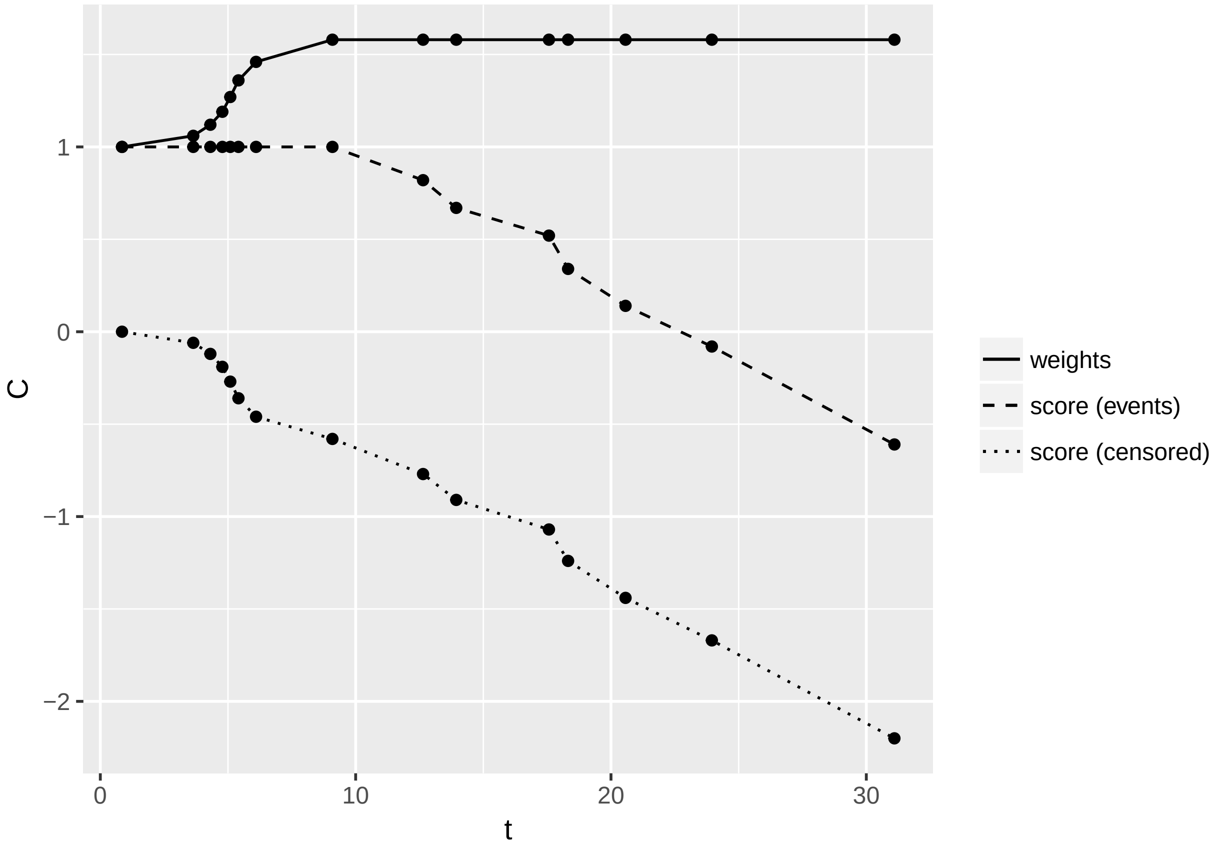

Figure 1: Scores/weights from a modestWLRT.

```
modest_weights$risk_table
#>        t n_e d_e n_c d_c  n d l l_c l_e     c    w     C
#> 1   0.85  10   0  10   1 20 1 0   0   0  1.00 1.00  0.00
#> 2   3.64  10   0   9   1 19 1 0   0   0  1.00 1.06 -0.06
#> 3   4.31  10   1   8   0 18 1 0   0   0  1.00 1.12 -0.12
#> 4   4.78   9   0   8   1 17 1 0   0   0  1.00 1.19 -0.19
#> 5   5.09   9   0   7   1 16 1 0   0   0  1.00 1.27 -0.27
#> 6   5.41   9   0   6   1 15 1 0   0   0  1.00 1.36 -0.36
#> 7   6.10   9   1   5   0 14 1 0   0   0  1.00 1.46 -0.46
#> 8   9.09   8   0   5   1 13 1 0   0   0  1.00 1.58 -0.58
#> 9  12.64   8   0   4   1 12 1 0   0   0  0.82 1.58 -0.77
#> 10 13.94   8   0   3   1 11 1 0   0   0  0.67 1.58 -0.91
#> 11 17.57   8   1   2   0 10 1 0   0   0  0.52 1.58 -1.07
#> 12 18.32   7   1   2   0  9 1 0   0   0  0.34 1.58 -1.24
#> 13 20.57   6   1   2   0  8 1 0   0   0  0.14 1.58 -1.44
#> 14 23.95   5   1   2   0  7 1 3   1   2 -0.08 1.58 -1.67
#> 18 31.10   2   1   1   0  3 1 2   1   1 -0.61 1.58 -2.20
```

```
modest_weights$p
```

Test statistics

Given the risk table with the corresponding weights, it is simple to calculate the standardized weighted logrank statistic. Larger values of Z correspond to longer survival times on the experimental arm.

```
get_zs(modest_weights)
#> [1] 1.304849
```