PH 716 Applied Survival Analysis

Part 3: Comparing Multiple Survival Functions

Zhiyang Zhou (zhou67@uwm.edu, zhiyanggeezhou.github.io)

2026/02/17 10:56:32

A motivating real-world study: Acute Myelogenous Leukemia (AML)

We consider data from a clinical study of patients with AML. After achieving remission, patients were assigned to one of two groups: maintained remission therapy and no maintenance therapy. The outcome of interest is the time (in weeks) to relapse or death. Some patients did not relapse during follow-up, resulting in right-censored observations.

```
head(survival::aml)
```

```
##   time status          x
## 1    9      1 Maintained
## 2   13      1 Maintained
## 3   13      0 Maintained
## 4   18      1 Maintained
## 5   23      1 Maintained
## 6   28      0 Maintained
```

From this study, clinicians naturally ask:

- Do patients receiving maintenance therapy remain relapse-free longer?
- Are the survival experiences between the two groups different?
- Is any observed difference statistically significant?

Adapting the workflow in the last lecture

```
library(survival)
library(survminer)
data_Maintained <- aml[aml$x == 'Maintained',]
data_Nonmaintained <- aml[aml$x == 'Nonmaintained',]
km_Maintained <- survfit(
  formula = Surv(time, status) ~ 1
  ,data = data_Maintained,
  ,conf.type = "log-log"
)
km_Nonmaintained <- survfit(
  formula = Surv(time, status) ~ 1
  ,data = data_Nonmaintained,
  ,conf.type = "log-log"
)
survminer::ggsurvplot(
  fit = list("Maintained" = km_Maintained, "Nonmaintained" = km_Nonmaintained),
  xlab = "Time",
```

```
  conf.int = TRUE,
  conf.int.style = "step",
  censor = TRUE,
  legend.labs = c("Maintained", "Nonmaintained"),
  risk.table = FALSE,
  cumevents = FALSE,
  tables.height = 0.15,
  combine = TRUE # two curves in one plot
)
```

Updated workflow (more concise)

```
library(survival)
library(survminer)
km_aml <- survfit(
  formula = Surv(time, status) ~ x
  ,data = aml,
  ,conf.type = "log-log"
)
summary(km_aml)
survminer::ggsurvplot(
  fit = km_aml,
  xlab = "Time",
  conf.int = TRUE,
  conf.int.style = "step",
  censor = TRUE,
  risk.table = FALSE,
  cumevents = FALSE,
  tables.height = 0.15
)
```

Assumptions for the log-rank test

- Independent survival times across subjects.
- Independent and non-informative right censoring: $C_i \perp T_i \mid$ group.
- Fixed group membership.
- Within each group k, the survival times are identically distributed with hazard function $\lambda_k(t)$.

Hypotheses to be tested

- Null hypothesis $H_0 : \lambda_1(t) = \lambda_2(t) = \lambda(t)$ for all t
  - $\lambda(t)$ is the hazard function for the combined population of the two groups
- Alternative hypothesis H_1 could be:
  - One-sided $H_1 : \lambda_1(t) \geq \lambda_2(t)$ for all t and $\lambda_1(t) > \lambda_2(t)$ for some t
  - One-sided $H_1 : \lambda_1(t) \leq \lambda_2(t)$ for all t and $\lambda_1(t) < \lambda_2(t)$ for some t
  - Two-sided $H_1 : \lambda_1(t) \neq \lambda_2(t)$ for some t

Two-sample log-rank test

- Basic idea: comparing individual estimated survival to the pooled estimated survival
  - Under H_0, the two groups are from the same population, so their estimated survival curves should be close to each other and to the pooled estimated survival curve.

```r
library(survival)
library(survminer)
data_Maintained <- aml[aml$x == 'Maintained',]
data_Nonmaintained <- aml[aml$x == 'Nonmaintained',]
km_Maintained <- survfit(
  formula = Surv(time, status) ~ 1
  ,data = data_Maintained,
  ,conf.type = "log-log"
)
km_Nonmaintained <- survfit(
  formula = Surv(time, status) ~ 1
  ,data = data_Nonmaintained,
  ,conf.type = "log-log"
)
km_Pooled <- survfit(
  formula = Surv(time, status) ~ 1
  ,data = aml,
  ,conf.type = "log-log"
)
survminer::ggsurvplot(
  fit = list("Maintained" = km_Maintained, "Nonmaintained" = km_Nonmaintained, 'Pooled'= km_Pooled),
  xlab = "Time",
  conf.int = F,
  conf.int.style = "step",
  censor = TRUE,
  legend.labs = c("Maintained", "Nonmaintained", 'Pooled'),
  risk.table = FALSE,
  cumevents = FALSE,
  tables.height = 0.15,
  combine = TRUE
)
```

- Distinct observed event times across the POOLED sample are $t_1 < \cdots < t_{n_D}$
  - At time t_j , there are d_{kj} events in group k, $k = 1, 2$, and $d_j = d_{1j} + d_{2j}$
  - Just prior to t_j, there are r_{kj} at risk in group k and $r_j = r_{1j} + r_{2j}$
- Test statistic
  - $U_k/\sqrt{V} \approx N(0,1)$ under H_0, $k = 1, 2$
    - $U_k = \sum_{j=1}^{n_D} r_{kj}(d_{kj}/r_{kj} - d_j/r_j) = r_{kj}\{\hat{\lambda}_1(t_j) - \hat{\lambda}(t_j)\}$
      - $\hat{\lambda}_1(t_j)$: estimated hazard rate at t_j for group k
      - $\hat{\lambda}(t_j)$: estimated hazard rate at t_j for pooled population
      - $d_{kj} = r_{kj}\hat{\lambda}_1(t_j)$: observed number of events from sample k at time t_j
      - $r_{kj}\hat{\lambda}(t_j)$: expected number of events from sample k at time t_j under H_0
    - $V = \text{var}(U_k) = \sum_{j=1}^{n_D} \frac{d_j r_{1j} r_{2j}(r_j - d_j)}{r_j^2(r_j - 1)}$
    - $U_1 = U_2$
  - The log-rank test is rank-based; one could construct the test statistic using only the order of observed event times alone.
- Rejection region
  - 2-sided: $|U_k/\sqrt{V}| > z_{1-\alpha/2}$ or equiv. $U_k^2/V > \chi^2_{1,1-\alpha}$
    - $z_{1-\alpha/2}$ is the $1 - \alpha/2$ quantile of $N(0,1)$
    - $\chi^2_{1,1-\alpha}$ is the $1 - \alpha$ quantile of $\chi^2(1)$
  - 1-sided ($H_1 : \lambda_1(t) > \lambda_2(t)$ for some t): $U_1/\sqrt{V} > z_{1-\alpha}$

  - 1-sided ($H_1 : \lambda_1(t) < \lambda_2(t)$ for some t): $-U_1/\sqrt{V} > z_{1-\alpha}$
- p-value
  - 2-sided: $p = 2\{1 - \Phi(|U_k/\sqrt{V}|)\}$
    - $\Phi(\cdot)$ is the cdf of $N(0, 1)$
  - 1-sided ($H_1 : \lambda_1(t) \geq \lambda_2(t)$ for all t and $\lambda_1(t) > \lambda_2(t)$ for some t): $p = \{1 - \Phi(U_1/\sqrt{V})\}$
  - 1-sided ($H_1 : \lambda_1(t) \leq \lambda_2(t)$ for all t and $\lambda_1(t) < \lambda_2(t)$ for some t): $p = \{1 - \Phi(-U_1/\sqrt{V})\}$

Revisit the AML data

```
library(survival)
library(survminer)
# For 2-sided H1 only
survival::survdiff(
  formula = survival::Surv(time, status) ~ x,
  data = aml
)
# OR
survminer::surv_pvalue(
  fit = survival::survfit(
    formula = survival::Surv(time, status)~x,
    data = aml
  ),
  method = 'log-rank'
)
# For 2-sided or 1-sided H1
nph::logrank.test(
  time = aml$time,
  event = aml$status,
  group = aml$x,
  alternative = 'two.sided' # 'two.sided','less','greater'
)$test
```

Reporting the results of a log-rank test

- Recall the motivating questions from the AML study:
  - Do patients receiving maintenance therapy remain relapse-free longer?
  - Are the survival experiences between the two groups different?
  - Is any observed difference statistically significant?
- Demo report (covering necessary components: hypotheses, the name of method, the p-value/rejection region, the significance level, and the conclusion):
  - "Testing hypotheses H_0 : ___ vs. H_1 : ___ , we carried on the ___ test."
    - "The p-value is ___ . So, at the ___ level, there was/wasn't a strong statistical evidence against H_0, i.e., we believed that ___ ."
    - OR "The value of test statistic is $T =$ ___ . Given the level ___ rejection region $T >$ ___ , there was/wasn't a strong statistical evidence against H_0, i.e., we believed that ___ ."

Ex 3.1. Breast cancer data sets used in Royston and Altman (2013)

The `survival::gbsg` data set contains patient records from a 1984–1989 trial conducted by the German Breast Cancer Study Group (GBSG). It retains 686 patients with node positive breast cancer. Are the survival experiences between the two treatment groups different?

```
gbsg_simple = survival::gbsg[
  complete.cases(survival::gbsg[, c('hormon', 'rfstime', 'status')]),
```

```
  c('hormon', 'rfstime', 'status')
]
head(gbsg_simple)
```

```
##   hormon rfstime status
## 1      0    1838      0
## 2      0     403      1
## 3      0    1603      0
## 4      0     177      0
## 5      1    1855      0
## 6      0     842      1
```

Comparing >2 survival curves

- Hypotheses to be tested
  - Null hypothesis $H_0 : \lambda_1(t) = \cdots = \lambda_K(t) = \lambda(t)$ for all t
  - Alternative hypothesis $H_1 : \lambda_{k_1}(t) \neq \lambda_{k_2}(t)$ for certain t and certain 2-tuple (k_1, k_2)

Ex. 3.2. Bladder Cancer Recurrences

A dataset on recurrences of bladder cancer. It contains three treatment arms for 118 subjects.

```
data.ex32 = survival::bladder1[
  complete.cases(survival::bladder1[,c('id', 'treatment', 'start', 'stop', 'status')]),
  c('id', 'treatment', 'start', 'stop', 'status')
]
data.ex32$status = 1*(data.ex32$status %in% c(1,2,3)) # merging status 1, 2,3
data.ex32$time = data.ex32$stop - data.ex32$start
survival::survdiff(
  formula = survival::Surv(time, status)~treatment, data=data.ex32
)
# Or
survminer::surv_pvalue(
  fit = survival::survfit(formula = survival::Surv(time, status)~treatment, data=data.ex32),
  method = 'log-rank'
)
```

Testing for trend

- Hypotheses to be tested
  - Null hypothesis $H_0 : \lambda_1(t) = \cdots = \lambda_K(t) = \lambda(t)$ for all t, $K > 2$
  - Alternative hypothesis $H_1 : \lambda_1(t) \geq \cdots \geq \lambda_K(t)$ or $\lambda_1(t) \leq \cdots \leq \lambda_K(t)$, with at least one strict inequality

Ex. 3.3. Revisit the data on bladder cancer recurrences

```
data.ex33 = survival::bladder1[
  complete.cases(survival::bladder1[,c('id', 'treatment', 'start', 'stop', 'status')]),
  c('id', 'treatment', 'start', 'stop', 'status')
]
data.ex33$status = 1*(data.ex33$status %in% c(1,2,3)) # merging status 1, 2,3
data.ex33$time = data.ex33$stop - data.ex33$start
data.ex33$treatment = factor(data.ex33$treatment, levels = c("placebo","pyridoxine","thiotepa"))
survminer::surv_pvalue(
```

```
  fit = survival::survfit(formula = survival::Surv(time, status)~treatment, data=data.ex33),
  method = 'log-rank',
  test.for.trend = T
)
# The order of treatments matters
data.ex33$treatment = factor(data.ex33$treatment, levels = c("placebo","thiotepa","pyridoxine"))
survminer::surv_pvalue(
  fit = survival::survfit(survival::Surv(time, status)~treatment, data=data.ex33),
  method = 'log-rank',
  test.for.trend = T
)
```