PH 716 Applied Survival Analysis

Part II: Nonparametric survival curve estimation

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

2024/Mar/03 21:21:45

Notations

- i : subject index, $i = 1, \dots, n$
- T_i : (authentic) survival time for subject i
- C_i : censoring time for subject i
- $\widetilde{T}_i = \min(T_i, C_i)$: observed survival time for subject i
- Δ_i: event indicator for subject i; $= 1$ if $\widetilde{T}_i = T_i$; $= 0$ if $\widetilde{T}_i = C_i$

Assumptions

- T_i is iid across i, i.e., $T_i \sim T$ for all i
- T_i is independent of C_i given covariates (if any)

Kaplan-Meier (KM) estimator

- To estimate $S_T(t)$ ($= S_{T_i}(t)$ for all i) nonparametrically
- Observed distinct authentic survival times: $t_1 < t_2 < \cdots < t_{n_D}$
  - n_D: # of distinct time points at which events are observed
- Recall for discrete survival time
  - $S_T(t) = \prod_{j:t_j \le t}\{1 - \lambda_T(t_j)\}$
- KM estimator
  - $\widehat{S}_{T,KM}(t) = \prod_{j:t_j \le t}\{1 - \hat{\lambda}_T(t_j)\}$
    - $\hat{\lambda}_T(t_j) = d_j/r_j$: an estimate of the (conditional) probability for an individual who survives up to time t_j experiences the event at t_i, i.e., $\Pr(\text{event occurs in } [t_j, t_{j+1}) \mid T \ge t_j)$
      - d_j: # of events that happened exactly at time t_j
      - r_j: # of individuals at risk up to time t_j (have not yet had an event or been censored prior to t_j)

- Ex. 2.1: Find the KM estimator for the data below, where the + sign denotes a right-censored subject:

i	1	2	3	4	5	6	7	8	9	10
$\widetilde{T}_i$	2	5+	8	12+	15	21+	25	29	30+	34

- Risk table

j	t_j	r_j	d_j	d_j/r_j	$\widehat{S}_{KM}(t_j)$
–	0	10	0	0	1
1	2	10	1	.1	$1 \times (1 - .1) = .9$
2	8	8	1	.125	$.9 \times (1 - .125) = .787$
3	15	6	1	.167	$.787 \times (1 - .167) = .656$
4	25	4	1	.25	$.656 \times (1 - .25) = .492$
5	29	3	1	.33	$.492 \times (1 - .33) = .328$
6	34	1	1	1	0

```r
ex21 = data.frame(
  time=c(2, 5, 8, 12, 15, 21, 25, 29, 30, 34),
  delta=c(1, 0, 1, 0, 1, 0, 1, 1, 0, 1)
)
km.ex21 = survival::survfit(
  formula=survival::Surv(time, delta)~1,
  data=ex21,
  conf.type="log-log")
summary(km.ex21)
```

- Variance of KM estimator
  - $\mathrm{var}(d_j/r_j) \approx d_j/\{r_j(r_j - d_j)\}$ (since d_j/r_j is the mle of $\lambda_T(t_j) \Rightarrow d_j/r_j \approx N(\lambda_T(t_j), \lambda_T(t_j)\{1 - \lambda_T(t_j)\}/r_j)$)
  - $\mathrm{var}\{\ln \widehat{S}_{T,KM}(t)\} \approx \sum_{j:t_j \le t} d_j/\{r_j(r_j - d_j)\}$ (the delta method)
  - $\mathrm{var}\{\widehat{S}_{T,KM}(t)\} \approx \{\widehat{S}_{T,KM}(t)\}^2 \sum_{j:t_j \le t} d_j/\{r_j(r_j - d_j)\}$ (applying the delta method twice)
  - $\mathrm{var}[\ln\{-\ln \widehat{S}_{T,KM}(t)\}] \approx \{\widehat{S}_{T,KM}(t)\}^{-2} \sum_{j:t_j \le t} d_j/\{r_j(r_j - d_j)\}$ (applying the delta method twice)
    - leading to the confidence interval of $\widehat{S}_{T,KM}(t)$ based on the log-log transformation which is guaranteed to be inside $[0, 1]$

- Visualization of KM estimator

```r
# A plain way
plot(km.ex21)
# A more fancy way
survminer::ggsurvplot(
  km.ex21,
  xlab="Time",
  xlim=c(0,40),
  conf.int = T,
  conf.int.style="step",
  censor=T,
  legend.labs = c("Entire Cohort"),
  risk.table = F,
  cumevents = F,
  tables.height = 0.15
)
```

- Properties of KM estimator
  - $\widehat{S}_{T,KM}(t)$ is a right-continuous step function, approximating the (likely smooth) $S_T(t)$

  - $\widehat{S}_{T,KM}(t)$ is a consistent (but typically biased) estimator of $S_T(t)$
    * As n increases, $\widehat{S}_{T,KM}(t)$ becomes less jagged
    * The bias vanishes when there is no censoring, stemming from the possibility that the last survivor becomes censored.
  - In the absence of censoring, $\widehat{S}_{T,KM}(t)$ reduces to $1 - \widehat{F}_T(t)$
    * $\widehat{F}_T(t) = \#\{i : T_i \leq t\}/n$ is the empirical cumulative distribution function (ECDF)
  - Note that $\widehat{S}_{T,KM}(t)$ has n_D jumps
    * One jump at each distinct failure time
    * There is no jump at the censored times! (why?)
  - $\widehat{S}_{T,KM}(t)$ is well-defined (it can be specified) up to the last observed time $\max\{\widetilde{T}_1, \ldots, \widetilde{T}_n\}$
    * One cannot estimate $S_T(t)$ for times $\max\{\widetilde{T}_1, \ldots, \widetilde{T}_n\}$ using the KM procedure
    * Because no data available in the sample beyond time $\max\{\widetilde{T}_1, \ldots, \widetilde{T}_n\}$
  - If last survivor is censored, KM estimator will NOT drop down to 0

- Ex. 2.2: Visualization of two KM estimators
  - This dataset is from the Mayo Clinic trial in the primary biliary cirrhosis (PBC) conducted between 1974 and 1984. A total of 424 PBC patients met eligibility criteria for the randomized placebo controlled trial of the drug D-penicillamine.

```r
head(survival::pbc[,1:4])
# Cleaning
data.ex22 = survival::pbc[complete.cases(survival::pbc[,1:4]), 1:4]
data.ex22$status = 1*(data.ex22$status %in% c(1,2)) # merging status 1 and 2
head(data.ex22)
# Fitting
km.ex22 = survival::survfit(
  formula=survival::Surv(time,status)~trt, data=data.ex22, conf.type="log-log"
)
print(km.ex22)
summary(km.ex22)
# Plotting
plot(km.ex22)
survminer::ggsurvplot(
  km.ex22,
  xlab="Time",
  conf.int = T,
  conf.int.style="step",
  censor = F,
  risk.table = F,
  cumevents = F,
  tables.height = 0.15
)
```

Nelson-Aalen(-Altschuler-Fleming-Harrington) estimator

- Estimating the cumulative hazard
  - Recall for discrete times, $\Lambda_T(t) = \sum_{j:t_j \leq t} \lambda_T(t)$
  - $\widehat{\Lambda}_{T,NA}(t) = \sum_{j:t_j \leq t} \hat{\lambda}_T(t_j) = \sum_{j:t_j \leq t} d_j/r_j$
- Estimating the survival function
  - Recall for continuous times, $S_T(t) = \exp\{-\Lambda_T(t)\}$
  - $\widehat{S}_{T,NA}(t) = \exp\{-\widehat{\Lambda}_{T,NA}(t)\} = \exp\{-\sum_{j:t_j \leq t} d_j/n_j\}$
- Asymptotically equivalent to KM

– KM and NA give the same estimator as $n \to \infty$

- Revisit Ex. 2.1: Find the NA estimator for the data below, where the + sign denotes a right-censored subject:

i	1	2	3	4	5	6	7	8	9	10
$\widetilde{T}_i$	2	5+	8	12+	15	21+	25	29	30+	34

```
ex21 = data.frame(
  time=c(2, 5, 8, 12, 15, 21, 25, 29, 30, 34),
  delta=c(1, 0, 1, 0, 1, 0, 1, 1, 0, 1)
)
na.ex21 = survival::survfit(
  formula=survival::Surv(time, delta)~1,
  data=ex21,
  conf.type="log-log",
  type = 'fh')
summary(na.ex21)
```