PH 716 Applied Survival Analysis Part IV: Competing risks

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

2024/May/06 11:16:42

Competing risks

- $K (\geq 2)$ (mutually exclusive) events of interest
 - Occurrence of one of these events precluding us from observing the other event on this subject
 - Observation for each subject terminated if
 - * encountering one (and only one) of these K events or
 - * censoring
 - E.g., death from different causes: natural causes, accidental death, homicide, suicide, etc.
 - E.g., disease-free survival from multiple conditions: heart disease, cancer, chronic respiratory diseases, stoke, diabetes, kidney diseases, etc.
- Notations
 - -i: subject index, $i = 1, \ldots, n$
 - $-\widetilde{T}_i = \min(T_i, C_i)$: observed survival time for subject *i*
 - * T_i : authentic survival time for subject i
 - * C_i : censoring time for subject i
 - $-\Delta_i$: the (re-defined) event indicator for subject i
 - * $\Delta_i = k, k = 1, \dots, K$: $T_i = \widetilde{T}_i$ and the event label is k

*
$$\Delta_i = 0$$
: $T_i = C_i$

 $-x_{i1},\ldots,x_{ip}$: values of covariates for subject i

Recall functions characterizing the survival distribution

- Limited to continuous T_i
- Hazard function

$$\lambda_{T_i}(t) = \lim_{\delta \to 0^+} \frac{\Pr(t \le T_i < t + \delta \mid T_i \ge t)}{\delta}$$

- The instantaneous risk of experiencing one event at time t, assuming the subject has survived up to t

• Cumulative hazard function $\Lambda_{T_i}(t) = \int_0^t \lambda_{T_i}(u) du$

- Survival function $S_{T_i}(t) = \Pr(T_i > t)$
- (Cumulative) distribution function $F_{T_i}(t) = \Pr(T_i \le t) = 1 S_{T_i}(t)$
- Probability density function $f_{T_i}(t) = dF_{T_i}(t)/dt$
- Interaction among the above functions

$$- \lambda_{T_i}(t) = -d \ln S_{T_i}(t)/dt = -d \ln\{1 - F_{T_i}(t)\}/dt - \Lambda_{T_i}(t) = -\ln S_{T_i}(t) - S_{T_i}(t) = \exp\{-\Lambda_{T_i}(t)\} = \exp\{-\int_0^t \lambda_{T_i}(u)du\} - f_{T_i}(t) = -dS_{T_i}(t)/dt = S_{T_i}(t)\lambda_{T_i}(t)$$

Motivation to consider the event type k

- May sacrifice valuable information by ignoring the event label (i.e., merging all the K events together) - E.g., $\lambda_{T_i}(t) = \lambda_0(t) \exp(\sum_{j=1}^p x_{ij}\beta_j)$ * β_1 potentially insignificant in general with x_{i1} as a strong predictor for certain specific event

Cause-specific functions characterizing the survival distribution

• Cause-specific hazard function

$$\lambda_{T_i}^{(k)}(t) = \lim_{\delta \to 0^+} \frac{\Pr(t \le T_i < t + \delta, \Delta_i = k \mid T_i \ge t)}{\delta}, \quad k = 1, \dots, K$$

- The instantaneous risk of experiencing event k at time t, assuming the subject has survived up to t $-\sum_{k=1}^{K} \lambda_{T_i}^{(k)}(t) = \lambda_{T_i}(t)$
- Cumulative cause-specific hazard function $\Lambda_{T_i}^{(k)}(t) = \int_0^t \lambda_{T_i}^{(k)}(u) du$

$$-\sum_{k=1}^{K} \Lambda_{T_i}^{(k)}(t) = \Lambda_{T_i}(t)$$

- Sub-distribution function/cumulative incidence function (CIF) $F_{T_i}^{(k)}(t) = \Pr(T_i \leq t, \Delta_i = k)$
 - NOT a (cumulative) distribution function
 - The probability of dying from event k up to time t, while acknowledging that the subject may die
- of other K-1 causes first $-\sum_{k=1}^{K} F_{T_i}^{(k)}(t) = F_{T_i}(t) \Rightarrow S_{T_i}(t) = 1 \sum_{k=1}^{K} F_{T_i}^{(k)}(t)$ Sub-distribution hazard function

$$\bar{\lambda}_{T_i}^{(k)}(t) = -\frac{\mathrm{d}\ln\{1 - F_{T_i}^{(k)}(t)\}}{\mathrm{d}t} = \frac{\mathrm{d}F_{T_i}^{(k)}(t)/\mathrm{d}t}{1 - F_{T_i}^{(k)}(t)}, \quad k = 1, \dots, K$$

- The instantaneous risk at time t of experiencing event k, assuming the subject has survived from event k up to t
- NOT the cause-specific hazard function: $\bar{\lambda}_{T_i}^{(k)}(t) \leq \lambda_{T_i}^{(k)}(t)$
- Interaction among the above functions

$$- \lambda_{T_{i}}^{(k)}(t) = \frac{\mathrm{d}F_{T_{i}}^{(k)}(t)/\mathrm{d}t}{S_{T_{i}}(t)}$$

$$* \operatorname{Proof:} \lambda_{T_{i}}^{(k)}(t) = \lim_{\delta \to 0^{+}} \frac{\operatorname{Pr}(t \leq T_{i} \leq t + \delta, \Delta_{i} = k, T_{i} \geq t)}{\delta \operatorname{Pr}(T_{i} \geq t)} = \lim_{\delta \to 0^{+}} \frac{\operatorname{Pr}(t \leq T_{i} < t + \delta, \Delta_{i} = k)}{\delta S_{T_{i}}(t)} = \frac{\mathrm{d}F_{T_{i}}^{(k)}(t)/\mathrm{d}t}{\delta S_{T_{i}}(t)}$$

$$- F_{T_{i}}^{(k)}(t) = \int_{0}^{t} \lambda_{T_{i}}^{(k)}(u)S_{T_{i}}(u)\mathrm{d}u = 1 - \exp\{-\int_{0}^{t} \bar{\lambda}_{T_{i}}^{(k)}(u)\mathrm{d}u\}$$

Naive KM estimator [DM, Sec. 9.2.1]

- Assuming that
 - $-T_i$ iid across *i*, i.e., $T_i \stackrel{\text{iid}}{\sim} T$
 - $-T_i$ independent of C_i given covariates (if any)
 - Times to different events are independent (typically violated in medical cases)
 - * Implying that at each time point the hazard of each event is the same for subjects at risk as for subjects that have experienced other competing events by that time
- Estimation procedure
 - Take the event k as the event of interest with other events considered as censored
 - Apply KM estimator to the resulting binary setting and then estimate the probability of survival from one event (in the absence of other causes) by $\prod_{j:t_j \leq t} \{1 - \hat{\lambda}_T^{(k)}(t_j)\}\$
 - * $0 = t_0 < t_1 < \cdots < t_J$: unique failure times
 - * $\hat{\lambda}_T^{(k)}(t_j) = d_{kj}/r_j$: an estimate of the cause-specific hazard function
 - d_{kj} : # of event k that happened exactly at time t_j
 - $\cdot r_i$: # of individuals at risk up to time t_i
- Underestimating the survival probability (i.e., overestimating the failure probability)
 - Potentially treating subjects that will never fail as if they could fail
 - The bias inflated when the competition when the hazards of competing events are larger

Ex. 9.1 High risk population in asaur::prostateSurvival

- Dataset asaur::prostateSurvival involves covariates as below.
 - grade: a factor with levels mode (moderately differentiated) and poor (poorly differentiated)
 - stage: a factor with levels T1ab (Stage T1, clinically diagnosed), T1c (Stage T1, diagnosed via a PSA test), and T2 (Stage T2)
 - ageGroup: a factor with levels 66-69, 70-74, 75-79, & 80+
 - survTime: the survival time from diagnosis to death (from prostate cancer or other causes) or last date known alive
 - status: a censoring variable, 0 (censored), 1 (death from prostate cancer), and 2 (death from other causes)
- Consider the high risk population (i.e.,grade="poor", stage="T2" & ageGroup="80+").

```
options(digits=4)
library(asaur)
library(survival)
sapply(asaur::prostateSurvival, class)
data.ex91 = asaur::prostateSurvival[
  asaur::prostateSurvival$grade == "poor" &
  asaur::prostateSurvival$stage == "T2" &
  asaur::prostateSurvival$ageGroup == "80+"
٦
km.prost.naive = survfit(
  Surv(survTime, event=(data.ex91$status==1)) ~ 1,
  data=data.ex91
)
km.other.naive = survfit(
  Surv(survTime, event=(data.ex91$status==2)) ~ 1,
  data=data.ex91
)
plot(
  km.prost.naive$surv ~ km.prost.naive$time, type="s", ylim=c(0,1), lwd=2, col="blue",
  xlab="Months from prostate cancer diagnosis",
  ylab='Estimated survival probability',
)
lines(km.other.naive$surv ~ km.other.naive$time, type="s", col="green", lwd=2)
legend(
  "topright",
  c(
    "Prostate",
    "Other"
  ),
  col=c('blue','green'), lwd=2
)
```

KM estimator of CIF [DM, Sec. 9.2.2]

- Assuming that
 - $-T_i$ iid across *i*, i.e., $T_i \stackrel{\text{iid}}{\sim} T$
 - $-T_i$ independent of C_i given covariates (if any)
- Estimation procedure
 - Estimate overall survival $S_T(t)$ by $\widehat{S}_{T,KM}(t) = \prod_{j:t_j \leq t} \{1 \sum_{k=1}^K \widehat{\lambda}_T^{(k)}(t_j)\}$
 - * $0 = t_0 < t_1 < \cdots < t_J$: unique failure times

* $\hat{\lambda}_{T}^{(k)}(t_{j}) = d_{kj}/r_{j}$: an estimate of the cause-specific hazard function $\cdot d_{kj}$: # of event k that happened exactly at time t_{j} $\cdot r_{j}$: # of individuals at risk up to time t_{j} - Estimate CIF $F_{T}^{(k)}(t)$ by $\hat{F}_{T,KM}^{(k)}(t) = \sum_{j:t_{j} \leq t} \hat{\lambda}_{T}^{(k)}(t_{j}) \hat{S}_{T,KM}(t_{j}-1)$

Revisit Ex. 9.1

```
options(digits=4)
library(asaur)
library(survival)
library(mstate)
sapply(asaur::prostateSurvival, class)
data.ex91 = asaur::prostateSurvival[
  asaur::prostateSurvival$grade == "poor" &
  asaur::prostateSurvival$stage == "T2" &
  asaur::prostateSurvival$ageGroup == "80+"
]
km.cif = Cuminc(
  time = data.ex91$survTime,
  status = data.ex91$status
)
km.cif
# Plot of CIFs and the overall survival function
plot(
  km.cif$CI.1 ~ km.cif$time, type="s", ylim=c(0,1), lwd=2, col="blue",
  xlab="Months from prostate cancer diagnosis",
  ylab="Probability"
)
lines(km.cif$CI.2 ~ km.cif$time, type="s", lwd=2, col="green")
lines(km.cif$Surv ~ km.cif$time, type="s", lwd=2, col="red")
legend(
  "topright",
  <mark>c</mark>(
    "CIF (prostate)",
    "CIF (other)",
    'Overall survival'
  ),
  col=c('blue','green','red'), lwd=2
)
# Stacked plot
library(ggplot2)
cuminc_data = as.data.frame(km.cif[, c('time', 'Surv', 'CI.1', 'CI.2')])
cuminc_data = tidyr::pivot_longer(
  cuminc_data, cols = -time, names_to = "Types", values_to = "estimate")
ggplot(data = cuminc_data, aes(x = as.numeric(time), y = estimate, fill = Types)) +
  geom_area(alpha = 0.6) +
  labs(x = "Months from prostate cancer diagnosis", y = "Probability") +
  theme_minimal()
```

Regression on cause-specific hazards

- Assuming that
 - $-T_i$ independent across *i* given covariates
 - The independent and non-informative censoring
 - Cause-specific proportional hazards
 - * $\lambda_{T_i}^{(k)}(t) = \lambda_0^{(k)}(t) \exp(\sum_{j=1}^p x_{ij}\beta_j^{(k)})$
 - $\begin{array}{l} & \ddots \\ \lambda_0^{(k)}(t): \text{ baseline cause-specific hazard of event } k \\ & \ddots \\ \beta_1^{(k)}, \dots, \beta_p^{(k)}: \text{ covariate effects varying from one event to another} \\ & \text{ OR } \lambda_{T_i}^{(k)}(t) = \lambda_0^{(k)}(t) \exp(\sum_{j=1}^p x_{ij}\beta_j), \text{ i.e., } \beta_j \text{ shared by all the } K \text{ events} \end{array}$
- Estimation procedure

 - For $\lambda_{T_i}^{(k)}(t) = \lambda_0^{(k)}(t) \exp(\sum_{j=1}^p x_{ij}\beta_j^{(k)})$ * Specify one event of interest and fit a Cox PH model with the remaining K-1 events treated as censoring
 - * Repeat the above step and obtain K Cox PH models For $\lambda_{T_i}^{(k)}(t) = \lambda_0^{(k)}(t) \exp(\sum_{j=1}^p x_{ij}\beta_j)$ * First reshape the data frame in the "long format" by encoding the each row with K rows
 - * Then fit a Cox PH model stratified by the encoded event lable in the long format
- When $\hat{\lambda}_{T_i}^{(k)}(t)$ is ready

$$- \widehat{S}_{T_i}(t) = \exp\{-\sum_{k=1}^K \int_0^t \hat{\lambda}_{T_i}^{(k)}(u) \mathrm{d}u\}$$

$$- \widehat{F}_{T_i}^{(k)}(t) = \int_0^t \hat{\lambda}_{T_i}^{(k)}(u) \widehat{S}_{T_i}(u) du$$

- Pros and cons
 - Easy to implement
 - Straightforward interpretation of regression coefficients in terms of hazard ratio BUT inconvenient to interpret coefficients in terms of contributions to CIFs
 - Bias induced since we treat other events as censoring

Ex. 9.2 Patients at "T2"-stage in asaur::prostateSurvival

• Consider patients with stage="T2".

```
options(digits=4)
library(asaur)
library(survival)
sapply(asaur::prostateSurvival, class)
data.ex92 = asaur::prostateSurvival[
  asaur::prostateSurvival$stage == "T2"
]
# Regression on cause-specific hazards
data.ex92$status.1 = (data.ex92$status==1)
data.ex92$status.2 = (data.ex92$status==2)
cph.prost = coxph(
  Surv(survTime, status.1)~grade + ageGroup,
  data = data.ex92
)
summary(cph.prost)
cph.other = coxph(
  Surv(survTime, status.2)~grade + ageGroup,
  data = data.ex92
)
summary(cph.other)
```

```
# Regression on cause-specific hazards with shared coefficients
## Reshape the data into the long format
data.ex92.long = NULL
K = length(unique(data.ex92$status))-1
for (i in 1:nrow(data.ex92)){
  data.curr = data.ex92[rep(i, times=K),]
  data.curr$event = c('prostate', 'other')
  data.curr$status.long=rep(0,K-1)
  if(data.ex92$status[i]>=1) {
    data.curr$status.long[which(data.curr$event==c('prostate', 'other')[data.ex92$status[i]])]=1
  }
  data.ex92.long = rbind(data.ex92.long, data.curr)
}
data.ex92.long = data.ex92.long[
  !(names(data.ex92.long) %in% c('status.1','status.2'))] # remove columns to avoid confusion
head(data.ex92)
head(data.ex92.long)
## Equivalency between two data frames
head(data.ex92)
head(data.ex92.long[data.ex92.long$event=='prostate',])
head(data.ex92.long[data.ex92.long$event=='other',])
```

```
## Cox PH model stratified by event
cph.strat = coxph(
  Surv(survTime, status.long)~grade + ageGroup+strata(event),
  data = data.ex92.long
)
summary(cph.strat)
```

Fine-Gray model

- Assuming that
 - $-T_i$ independent across *i* given covariates
 - The independent and non-informative censoring
 - $\bar{\lambda}_{T_i}^{(k)}(t) = \bar{\lambda}_0^{(k)}(t) \exp(\sum_{j=1}^p x_{ij}\beta_j^{(k)})$

 - * $\bar{\lambda}_{0}^{(k)}(t)$: baseline sub-distribution hazard of event k* $\beta_{1}^{(k)}, \ldots, \beta_{p}^{(k)}$: covariate effects potentially varying from one event to another
- When $\hat{\lambda}_{T_i}^{(k)}(t)$ is ready

$$- \widehat{F}_{T_i}^{(k)}(t) = 1 - \exp\{-\int_0^t \hat{\overline{\lambda}}_{T_i}^{(k)}(u) \mathrm{d}u\}$$

- Pros and cons
 - Direct modeling of CIFs
 - Difficult to interpret regression coefficients in terms of hazard ratio, BUT may see the effect of covariates to CIFs directly
 - Might not satisfy the additive restriction $\sum_{k=1}^{K} \widehat{F}_{T_{k}}^{(k)}(t) \leq 1$

Revisit Ex. 9.2

- Poorly differentiated patients (grade=poor) have higher risk for death from both prostate and other.
- Elder patients also have higher risk for the death from both conditions.

```
options(digits=4)
library(asaur)
data.ex92 = asaur::prostateSurvival[
  asaur::prostateSurvival$stage == "T2"
٦
# Model fitting
cov1 = model.matrix(~ grade + ageGroup, data = data.ex92)[,-1]
cph.subdisthz.prost = cmprsk::crr(
 ftime = data.ex92$survTime,
 fstatus = data.ex92$status,
 cov1 = cov1,
 failcode=1
)
summary(cph.subdisthz.prost)
cph.subdisthz.other = cmprsk::crr(
 ftime = data.ex92$survTime,
 fstatus = data.ex92$status,
 cov1 = cov1,
 failcode=2
)
summary(cph.subdisthz.other)
# Predicted CIFs
cov1_new = matrix( # Specifying subjects with grade=poor and age= 70-74
  c(1, 1, 0, 0),
  byrow = T,
 ncol = 4
)
colnames(cov1_new) = colnames(cov1); cov1_new
predict.prost = predict(
  cph.subdisthz.prost,
  cov1 = cov1_new
)
head(predict.prost)
predict.other = predict(
  cph.subdisthz.other,
  cov1 = cov1_new,
  x = 1:100
)
head(predict.other)
# Create overall survival
pool.time = sort(unique(c(cph.subdisthz.prost$uftime, cph.subdisthz.other$uftime)))
cifs = data.frame(
 time = pool.time,
  surv = numeric(length(pool.time)),
 cif.prost = numeric(length(pool.time)),
  cif.other = numeric(length(pool.time))
)
for (i in 1:nrow(cifs)){
 if (!(i %in% predict.prost[,1])){
```

```
if (cifs$time[i] < predict.prost[1,1])</pre>
      cifs$cif.prost[i] = 0
    else cifs$cif.prost[i] = cifs$cif.prost[i-1]
  }else cifs$cif.prost[i] = predict.prost[,2][predict.prost[,1] == cifs$time[i]]
  if (!(i %in% predict.other[,1])){
    if (cifs$time[i] < predict.other[1,1])</pre>
      cifs$cif.other[i] = 0
    else cifs$cif.other[i] = cifs$cif.other[i-1]
  }else cifs$cif.other[i] = predict.other[,2][predict.other[,1] == cifs$time[i]]
}
cifs$surv = 1-cifs$cif.prost-cifs$cif.other
head(cifs)
# Trajectory plot of CIFs and overall survival
plot(
  cifs$cif.prost ~ cifs$time,
 type="s", ylim=c(0,1), lwd=2, col="blue",
 xlab="Months from prostate cancer diagnosis",
 ylab='Probability',
)
lines(cifs$cif.other ~ cifs$time, type="s", col="green", lwd=2)
lines(cifs$surv ~ cifs$time, type="s", col="red", lwd=2)
legend(
  "topright",
  c(
    "Prostate",
    "Other",
    'Overall Survival'
  ),
  col=c('blue','green','red'), lwd=2
)
# Stacked plot of CIFs
cifs_long = tidyr::pivot_longer(
  cifs, cols = -time, names_to = "Types", values_to = "estimate")
library(ggplot2)
ggplot(data = cifs_long, aes(x = as.numeric(time), y = estimate, fill = Types)) +
  geom_area(alpha = 0.6) +
  labs(x = "Months from prostate cancer diagnosis", y = "Probability") +
 theme_minimal()
```