Introduction to Survival Analysis in R

Lucy Cheng

Department of Statistics

Feb 9, 2010
Survival analysis in R

- Tools available in package **survival**
- `library(survival)`  # load the package
- `library(KMsurv)`  # load the package which includes data sets from Klein and Moeschberger’s book
- `library(help=survival)`  # view available functions and data sets
- `library(help=KMsurv)`  # view available data sets
- `data(aml)`  # load the data set aml
Functions of interest

- Survival object: `Surv`
- Kaplan-Meier estimates: `survfit`
- The log-rank test: `survdiff`
- The Cox proportional hazards model: `coxph`
- The Accelerated failure time model: `survreg`
Before complex functions may be performed, the data has to be put into the proper format: a survival object

\texttt{Surv(time, time2, event, type)}

Constructions are based on the data that is right-censored or left-truncated and right-censored
Survival object
Kaplan-Meier estimate
Tests for two or more samples
Cox proportional hazard model
Accelerated failure time model

Right-censored

- **Surv(time, time2)**
- `time` is a vector of event time, and `time2` is a vector of indicator (denoting if the event was observed or censored)
- > `data(aml); attach(aml)`
- > `my.surv.object <- Surv(time, status)`

```
[1]  9 13 13+ 18 23 28+ 31 34 45+ 48 161+ ...
```
Left-truncated and right-censored

- `Surv(time, time2, event)`
- **time** is the left-truncation time (observations never result in values below this time point)
- **time2** is the event time (or censoring time)
- **event** is the indicator variable
Left-truncated and right-censored

`> data(psych); attach(psych)

age  time  death
  1   51    1    1
  2   58    1    1
  3   55    2    1
  4   28   22    1
  5   21   30    0
   .   .   .   .
 25   36   40    1
 26   32   39    0`

`> my.surv.object <- Surv(age, age + time, death)
> my.surv.object
[1] (51,52] (58,59] (55,57] (28,50] (21,51+] ... (36,76] (32,71+]`
Kaplan-Meier estimate

- Kaplan-Meier estimator of survival is a nonparametric method of inference concerning the survivor function \( S = P_r(T > t) \).

- \( \texttt{> survfit(formula, conf.int = 0.95, conf.type = "log")} \)
  - \texttt{formula} is a survival object
  - \texttt{conf.int} is the confidence interval level and ranges between 0 and 1, with the default 0.95
  - \texttt{conf.type} is ’log’ by default, specifying the transformation used to construct the confidence interval.
Confidence bounds for Kaplan-Meier estimate

- The **default** intervals in survfit are called ”log” and the formula is:

\[ \exp(\log \hat{S}(t) \pm 1.96 \cdot s.e. (\hat{H}(t))) \]

- A linear confidence interval created by using `conf.type='plain'`
Cumulative Hazard

- relationship: \( S(t) = \exp(-H(t)) \)
- estimation: \( \hat{H}(t) = -\log(\hat{S}(t)) \)
- to estimate \( \hat{S}(t) \), using output from survfit()
  - \(< my.surv < -Surv(time, status)\)
  - \(< my.fit < -summary(survfit(my.surv))\)
  - \(< S.hat < -my.fit$surv\)
  - \(< H.hat < - - \log(S.hat)\)
Tests for two or more samples

- `survdiff(formula, rho=0)`

- `formula` is a survival object against a categorical covariate variable.

- `rho` is a scalar parameter that controls the type of test.

- With default `rho=0`, this is the log-rank or Mantel-Haenszel test.

- With `rho=1`, it is equivalent to the Peto and Peto modification of the Gehan-Wilcoxon test.
Tests for two or more samples

- When the sample size is rather limited (say less than 10), it might be dangerous to rely on asymptotic tests.
- The function `survtest` from package `coin` can be used to compute an exact Log-rank test.

```r
> library(coin).
> survtest(Surv(time, event) ~ group, distribution = "exact").
```
Cox proportional hazard model

Cox Proportional Hazard Model

- The most popular model for survival analysis because of its simplicity and no assumption about survival distribution.
- A semi-parametric model

\[ h_i(t) = h_0(t) \exp(\beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_k x_{ik}) \]

- the baseline hazard \( h_0(t) \) can take any form, the covariates enter the model linearly.
- The hazard ratio for these two observations, \( \frac{h_i(t)}{h_j(t)} = \frac{h_0(t)e^{\theta_i}}{h_0(t)e^{\theta_j}} = \frac{e^{\theta_i}}{e^{\theta_j}} \) is independent of time \( t \). This defines the “proportional hazards property”.
> coxph(formula, method)

**formula** is linear model with a survival object as the response variable

**method** is used to specify how to handle ties. The default is ‘efron’. Other options are ‘breslow’ and ‘exact’.

> cox.fit <- coxph(Surv(time, status) ~ x + y + z, method = 'breslow')

to obtain the baseline survival function

> my.survfit.object <- survfit(coxph.fit)
Cox PH model with time-dependent covariates

- Creating a new time-dependent covariate of the form $Z_{i,new}(t) = Z_i \ast transformation(t)$, from a covariate $Z_i$.
- A common transformation is log.
- A function to construct a time-dependent Cox PH model matrix and also run the model
  
  ```r
time.dep.coxph(d.f, col.time, col.delta, col.cov, td.cov, transform = log, method = 'efron', output.model = TRUE, output.data.frame = FALSE, verbose = TRUE)
  ```
- This function may be loaded using
  
  ```r
  ```
Cox PH model with time-dependent covariates

Arguments:

- **d.f**: A data frame containing the event/censoring times, a column indicating whether the event was observed, and covariates.
- **col.time**: The column name or number in d.f that designates the event/censoring times.
- **col.delta**: The column name or number in d.f that indicates what times were observed.
- **col.cov**: A vector of the column names or numbers in d.f designating the covariates to be included in the model.
- **td.cov**: The column name or number in d.f that indicates the covariate to be made time-dependent (the $Z_i$).
- All left arguments have default.
Model diagnostic for the Cox PH model

Three kinds of diagnostic:

- Testing the proportional hazard assumption.
- Examining influential observations.
- Detecting nonlinearity in relationship between the log hazard and the covariates.
Testing PH assumption

- R function `cox.zph` calculates tests of the proportional hazards assumption for each covariate.
- Graphical diagnostic can be done by `plot(cox.zph)`. Systematic departures from a horizontal line are indicative of non-proportional hazards, since PH assumes that estimates $\beta_1, \beta_2, \ldots, \beta_p$ do not vary much over time.
As an example, the following is the model I fit in one of my projects:

```r
cox <- coxph(Surv(mor) ~ factor(knot) + factor(offg) + moe)
```

- `cox.zph(cox)` computes a test for each covariate, along with a global test for the model as a whole:

<table>
<thead>
<tr>
<th></th>
<th>rho</th>
<th>chisq</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>factor(knot)1</td>
<td>-0.0875</td>
<td>0.694</td>
<td>0.4048</td>
</tr>
<tr>
<td>factor(knot)2</td>
<td>0.1215</td>
<td>1.491</td>
<td>0.2220</td>
</tr>
<tr>
<td>offg</td>
<td>0.0439</td>
<td>0.190</td>
<td>0.6633</td>
</tr>
<tr>
<td>moe</td>
<td>-0.0524</td>
<td>0.319</td>
<td>0.5722</td>
</tr>
<tr>
<td>GLOBAL</td>
<td>NA</td>
<td>9.179</td>
<td>0.0568</td>
</tr>
</tbody>
</table>

Therefore, there is no statistically significant evidence of non-proportional hazards for any of the covariates, and the global test is also not quite statistically significant.
Moreover, we may plot(cox.zph(cox))

- **knot**
- **offgrad**
- **moe**
Testing influential observations

- \(<\text{resid(cox, type = ”dfbeta”)}\>$ $\text{coef}$ gives the change in each regression coefficient when each observation is removed from the data (influence statistics). Plot influence statistics:
The **martingale residuals** may be plotted against covariates to detect nonlinearity, and may also be used to form component-plus-residual (or partial residual) plots, again in the manner of linear and generalized linear models.

\[
> \text{residuals(cox, type="martingale")}
\]

Nonlinearity is not an issue for categorical variables, so we only examine plots of martingale residuals and partial residuals against the variable **moe**.
Testing Nonlinearity

Nonlinearity, it appears, is slight here.
Accelerated failure time model

- Parametric model needs to make assumptions about the distribution of survival data.

- `> survreg(formula, dist = 'weibull')`

- `dist` has several options: weibull, exponential, gaussian, logistic, lognormal and loglogistic.

- need to check the consistency with parametric distributions beforehand.
Several parametric distributions are used to describe time to event data.

Each parametric distribution is defined by a different hazard function.

Table: Parametric survival distributions commonly used in epidemiology

<table>
<thead>
<tr>
<th>Distribution</th>
<th>h(t)</th>
<th>H(t)</th>
<th>S(t)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exponential</td>
<td>$\lambda$</td>
<td>$\lambda t$</td>
<td>$\exp(-\lambda t)$</td>
</tr>
<tr>
<td>Weibull</td>
<td>$\lambda p t^{p-1}$</td>
<td>$\lambda t^p$</td>
<td>$\exp(-\lambda t^p)$</td>
</tr>
<tr>
<td>Gompertz</td>
<td>$a \exp(bt)$</td>
<td>$\frac{a}{b} (\exp(bt) - 1)$</td>
<td>$\exp(-\frac{a}{b} (\exp(bt) - 1))$</td>
</tr>
<tr>
<td>Log-logistic</td>
<td>$\frac{abt^{b-1}}{1+at^b}$</td>
<td>log$(1 + at^b)$</td>
<td>$(1 + at^b)^{-1}$</td>
</tr>
</tbody>
</table>
Survival object Kaplan-Meier estimate Tests for two or more samples Cox proportional hazard model Accelerated failure time model

Parametric survival distributions

Exponential and Weibull distributions

Checking for consistency with exponential and weibull distributions by plotting their hazard functions:
Parametric survival distributions

The End

Thank you!