Notes of Survival Analysis in R for Public Health - Coursera

Di Zhen

Feb, 2020

Certifications

- Certification for the course:

https://coursera.org/share/b3b835c7d607fb5675a4faf191d14c83

- Certification for the specialization:

https://coursera.org/share/6b9da130ae9b966c9e2957bc214474ab

Week 1. The Kaplan-Meier Plot and log-rank test

1. Key Concepts

- The Kaplan-Meier plot:
  - The Kaplan-Meier plot estimates the probability of surviving at least to any given time.
  - The Kaplan-Meier table and associated plot is the simplest (but not the only) way of estimating the survival time when you have drop-outs.
- Survival function
- Log-rank test:

  The associated test to compare the survival curves
- Censored:

'Censored' means patients who are dropped out of the study and you don't know whether they're still alive after the dropped out.

2. Life Tables

- Life tables are used to measure the probability of death at a given age and the life expectancy at varying ages
- Two types:
  - Cohort or generational life tables
    - take an actual set of people born at the same time and follow them up for their whole lives
  - Current or period life tables
    - take a hypothetical cohort of people born at the same time and uses the assumption that they are subject to the age-specific mortality rates of a region or country.
    - these rates are often calculated using census data as the base population and actual age-specific death rates during the census year (and typically also one year either side).
- Used in cohort study
  - a set or cohort of patients are enrolled at time zero and then followed up to see who gets the outcome of interest, such as death, and when they get it
- Example:
  - This assumes that everybody enters the study at the same time, t=0, and no one leaves it except by death.
  - And in this case, we ignore the more realistic case when people drop out or are “lost to follow-up” (censored)
  - The probability of surviving at least to time t=0 is 1, or 100%
  - However, the Kaplan-Meier table and associated plot is the simplest (but not the only) way of estimating the survival time when you have drop-outs.

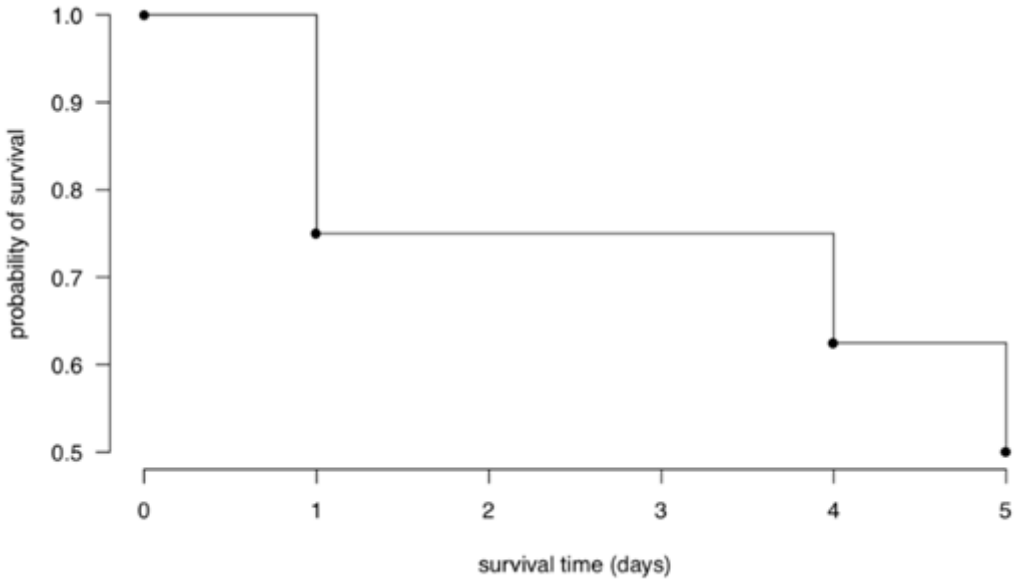
Time (t) in days	**Number of patients alive at time t**	**Number of patients who died at time t**	**Probability of survival past time t**
0 (study start)	100	0	1
1	100	2	0.98
2	98	??	??
3..	??	??	??

3. Kaplan-Meier Table and Plot

Time (t) in days	Event
0 (study start)	8 patients recruited
1	2 patients die
4	1 patient dies
5.	1 patient dies
etc	etc

Time (t) in days	Number of patients alive at time t	Number of patients who died at time t	Proportion of patients surviving past time t	Probability of survival past time t
0 (study start)	8	0	(8-0)/8=1	1
1	8	2	(8-2)/8=0.75	1 * 0.75 = 0.75
4	6	1	(6-1)/6=0.83	0.75*0.83 = 0.623
5	5	1	(5-1)/5=0.8	0.623*0.8 = 0.498

- To construct a Kaplan-Meier table (above)
  - The Kaplan-Meier method can be used to estimate this curve from the observed survival times without the assumption of an underlying probability distribution.
  - Basic idea: the probability of surviving past day t is simply the probability of surviving past day $t-1$ times the proportion of patients that survive on day t.



- Survival curve for Kaplan-Meier example
  - The plot of the survival function versus time is called the survival curve.

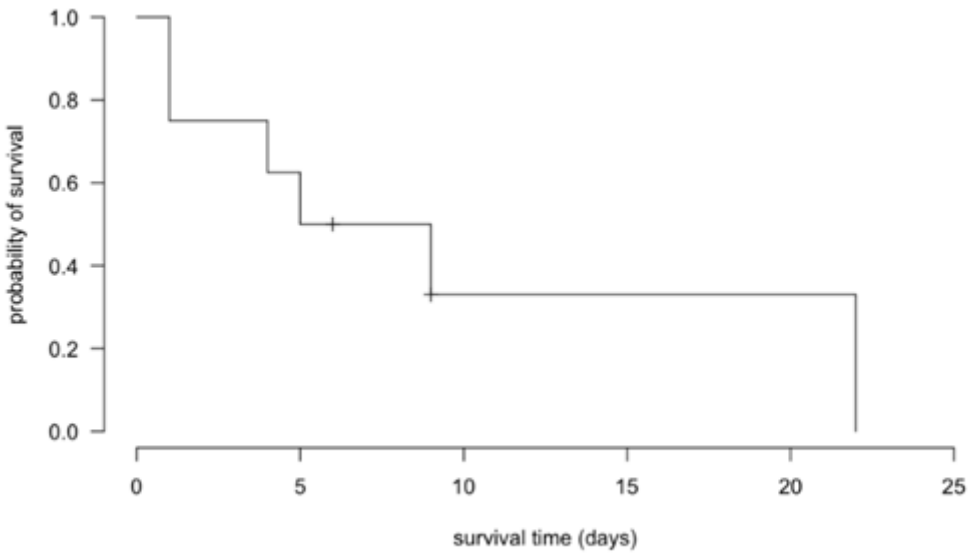
- The “steps” involve a horizontal line followed by a vertical line, because the probability is assumed to be the same until the next death occurs

Time (t) in days	Event
0 (study start)	8 patients recruited
1	2 patients die
4	1 patient dies
5.	1 patient dies
6	1 patient drop out
9	1 patient dies and 1 drops out
22	1 patient dies

Time (t) in days	Number of patients alive at time t	Number of patients who died at time t	Proportion of patients surviving past time t	Probability of survival past time t
0 (study start)	8	0	1	1
1	8	2	0.75	0.75
4	6	1	0.83	0.75*0.83 = 0.623
5	5	1	0.8	0.623*0.8 = 0.498

Time (t) in days	Number of patients alive at time t	Number of patients who died at time t	Proportion of patients surviving past time t	Probability of survival past time t
6+	4	0	(4-0)/4 = 1	0.498*1 = 0.498
9	3	1	(3-1)/3=0.667	0.498*0.667 = 0.332
9+	2	0	(2-0)/2 = 1	0.332*1 = 0.332
22	1	1	(1-1)/1 = 0	0.332*0 = 0

- Censored patients
  - classified neither as 'survived' nor as 'died' on any given day
  - Then we add a new line, mark it with a little '+' right after the time count and denote the censored patient(s) by taking them off the count of patients alive at time t.



4. Practice in R: Run a KM plot and log-rank test

Download the data from here.

```
install.packages("survival")
library(survival)
library(ggplot2)
# Import data
g <- read.csv(file = "simulated HF mort data for GMPH (1K) final.csv",
header=TRUE, sep=',')
# Variables
gender <- as.factor(g[,"gender"])
fu_time <- g[,"fu_time"]
death <- g[,"death"]
# Run an overall Kaplan-Meier plot
km_fit <- survfit(Surv(fu_time, death) ~ 1)
plot(km_fit)
```

- "Survfit" fits a simple survival model that depends only on gender in terms of predictors: in this case there aren't any predictors, so the model just has the intercept, denoted by "1". The two arguments used by "Surv" are the follow-up time for each patient and whether they died
- The survfit() function produces the Kaplan-Meier estimates of the probability of survival over time that are used by "plot" to produce the Kaplan-Meier curve above

```
# Estimate
summary(km_fit, times = c(1:7,30,60,90*(1:10)))
```

- The "times" argument gives us control over what time periods we want to see.

```
# Extend this by splitting the curve by gender
km_gender_fit <- survfit(Surv(fu_time, death) ~ gender)
plot(km_gender_fit)
# Run a logrank test
survdiff(Surv(fu_time, death) ~ gender, rho=0)
```

- With rho = 0, which is the default, it yields the log-rank or Mantel-Haenszel test
- The log-rank test compares the survival time by gender. We can know both genders will have similar survival rate or not based on p-value.
- The **log-rank test** is based on a comparison of the observed numbers of deaths and the numbers of deaths expected if in fact there were no difference in the probability of death between the groups (genders in this case) and uses a chi-squared test.

	N	Observed	Expected	$(O-E)^2/E$	$(O-E)^2/V$
gender=1	548	268	271	0.0365	0.082
gender=2	452	224	221	0.0448	0.082

Chisq=0.1 on 1 degrees of freedom, p=0.8

- Interpretation: The 548 male patients (gender = 1) had 268 observed deaths, but you would expect 271 under the null hypothesis of no difference in survival times by gender group. And the female group also had roughly the same number of deaths as expected under the null, which confirms that there is no difference (p = 0.8) of survival after hospital admission between two genders.

Week 2. The simple Cox model

1. Key Concepts

- Cox proportional hazards model:
  - Cox proportional hazards model can be used to compare the survival of multiple groups of patients at the same time.
  - The hazards are assumed by the model to be proportional
  - Hazard: The risk of death at a given moment in time
  - Hazard function/rate: The way the hazard changes overtime
  - Hazard ratio
- The hazard function

  https://data.princeton.edu/wws509/notes/c7s1

    - The hazard function $h(t)$ is the probability of the event happening at time t, given that it has not yet happened, or the probability of dying at time t having survived up to time t.
- Risk set
    - The number of patients that are subjected to the risk change over time as people die or drop out
    - The risk set at time t is defined as the set of patients at time t that are at risk of experiencing the event
    - Survival analysis consists of a family of methods, and one way that they differ is in their handling of drop-outs and other issues when they define the risk set.

2. Practice in R: Run a simple Cox model

```
install.packages("survminer")
library(survminer)
# variables
ethnicgroup <- factor(g[,"ethnicgroup"])
fu_time <- g[,"fu_time"]
death <- g[,"death"]
# Run the Cox model
cox <- coxph(Surv(fu_time, death) ~ ethnicgroup)
summary(cox)
```

- Notice the missing data

```
levels(ethnicgroup)<-c(levels(ethnicgroup),"8") # add level 8 to the factor
ethnicgroup[is.na(ethnicgroup)] <- "8" # Change NA to "None"
cox <- coxph(Surv(fu_time, death) ~ ethnicgroup)
summary(cox)
```

- Check the standard errors
- The hazard for one variable is the hazard ratio (exp(coefficient)) times reference category.
    - If the hazard is above 1, it means that group has an increased risk compared with reference category. For example, if the hazard is 1.15, it is 15% higher than reference category.
- If the p-value is small, there is a statistically significant difference in favor of these patients compared with the reference category.

3. Missing Data

- Three types
    - 'missing completely at random' (MCAR)
        - One patient is just as likely to have missing values as any other patient: males just as likely as females, older patients just as likely as younger ones etc.
    - 'missing at random' (MAR)
        - Missingness can be explained by other variables for which there is full information
    - 'missing not at random' (MNAR)
        - when the missingness is specifically related to what's missing and so the probability of the value being missing depends on unobserved variables, i.e., variables not in your data set. This is generally the most problematic type.
- Problem:

    - If the data are MCAR, this will produce unbiased estimates as long as the sample size is still sufficiently large.
    - If the data are MAR or MNAR, the estimates will be biased.
- Solution:
    - Mean substitution (or mean imputation):
        - This has the advantage of not changing the overall mean for that variable. However, it artificially decreases the estimated variation. It also makes it difficult to detect correlations between the imputed variable and other variables. Hence mean substitution always gives biased results and is not recommended.
    - Multiple imputation:
        - Missing variables are assumed to be MAR (or MCAR) and are imputed by drawing from a distribution. This is done multiple times and yields multiple different completed datasets. Each of these datasets is analyzed, and the results are combined into a single overall result. Multiple imputation has been shown to yield unbiased results for MAR or MCAR data.
    - Maximum likelihood:
        - This approach also gives unbiased results for MAR (or MCAR) data. Data are assumed to be normally distributed with a certain (multivariate) mean and variance.

Week 3. The Multiple Cox Model

1. Practice in R: Run multiple Cox model

- Each association is adjusted for the other predictors in the model

```
cox <- coxph(Surv(fu_time, death) ~ age + gender + copd + prior_dnas +
ethnicgroup)
summary(cox)
```

2. Non-convergence Problem

- Problem:
    - Very high standard errors
    - Infinitely wide CI
- Reason:
    - Extremely data in reference category chose by R by default
- Solution:
    1. Change the reference category

    ```
    quintile <- relevel(quintile, ref = 2) # quintile 1 as the ref cat

    cox <- coxph(Surv(fu_time, death) ~ age + gender + copd + quintile +
    ethnicgroup)
    summary(cox)
    ```

    2. Combine categories

        Not always a good idea, because two group are different.

```
quintile_5groups <- g[,'quintile'] # best start with the original data set,
not from "quintile"
quintile_5groups[quintile_5groups==0] <- 5
quintile_5groups <- factor(quintile_5groups)
table(quintile_5groups, exclude=NULL)

cox <- coxph(Surv(fu_time, death) ~ age + gender + copd + quintile +
ethnicgroup)
summary(cox)
```

3. Exclude the patients

if combining categories doesn't make sense and there are only a few patients in the problematic category

4. Drop the offending variable

if combining categories doesn't make sense and if there are too many few patients in the problematic category for us to be comfortable dropping them all.

```
quintile_5groups <- g[,'quintile']
quintile_5groups[quintile_5groups==0] <- NA # set the zeroes to missing
quintile_5groups <- factor(quintile_5groups)
table(quintile_5groups, exclude=NULL)

cox <- coxph(Surv(fu_time, death) ~ age + gender + copd + ethnicgroup)
summary(cox)
```

3. Alternative to Cox Model

- Cox doesn't care about the distribution of survival times or what the hazard function looks like. This is why it's called "semi-parametric": it has some parameters – those of the predictors – but it has no parameters to describe the hazard function for patients with a value of zero for the predictors. The Cox model was developed to look at the effect of covariates on the hazard function rather than to estimate survival times.
- The Kaplan-Meier estimates are examples of non-parametric survival analysis
- However, making assumptions about the shape of the hazard function – adding parameters to the model to describe the shape, making the model "fully parametric" – can lead to better prediction.
- There are several such fully parametric models such as Weibull, exponential, log-normal, and log-logistic models, where hazard function has to be specified.
- Survival analysis can also be run in a Bayesian framework.

Week 4. Testing model assumptions and choosing predictors

1. Checking the proportionality assumption

- An informal visual check:
  - When you have a predictor with few categories, you can also use the Kaplan-Meier plot. The lines give the survival probability for each gender at each time point.
  - If the predictor satisfies the proportional hazard assumption, then the graph of the survival function versus the survival time should yield parallel curves.

    - This method does not work well for continuous predictors or categorical ones with many levels because the graph becomes too “cluttered”.

```
km_fit <- survfit(Surv(fu_time, death) ~ gender)
autoplot(km_fit)
plot(km_fit, xlab = "time", ylab = "Survival probability")
```

- Using other types of residuals in Cox model
    - Deviance residuals
        - Deviance residuals are transformations of martingale residuals and help you look for outliers or influential data points. You can either examine the influence of each data point on the coefficients or plot the distribution of the residuals against the covariate.
        - Specifying the argument type = “dfbeta” plots the estimated changes in the regression coefficients on deleting each observation (patient) in turn.

```
res.cox <- coxph(Surv(fu_time, death) ~ age)
ggcoxdiagnostics(res.cox, type = "dfbeta",
                 linear.predictions = FALSE, ggtheme = theme_bw())
```

        - It’s also possible to check outliers by visualizing the deviance residuals, which are normalized transformations of the martingale residual and should be roughly symmetrically distributed about zero with a standard deviation of 1.
            - Positive values correspond to individuals that “died too soon” compared with expected survival times.
            - Negative values correspond to individuals that “lived too long” compared with expected survival times.
            - Very large or small values are outliers, which are poorly predicted by the model.

```
res.cox <- coxph(Surv(fu_time, death) ~ age)
ggcoxdiagnostics(res.cox, type = "deviance",
                 linear.predictions = FALSE, ggtheme = theme_bw())
```

    - Martingale residual
        - Test whether any continuous variables that assumed to have a linear relation with the outcome actually do have a linear relation
        - Martingale residuals may present any value between minus infinity and 1, and have a mean of zero
            - Martingale residuals near 1 represent individuals that “died too soon”
            - Large negative values correspond to individuals that “lived too long”

```
ggcoxfunctional(Surv(fu_time, death) ~ age + log(age) + sqrt(age))
```

- The function cox.zph() correlates for each predictor the corresponding set of scaled Schoenfeld residuals with time
    - If p-value is high, assumption is met.
    - If the line is pretty flat, meaning that the effect of gender changes little during the follow-up period. That’s good news.

```
fit <- coxph(Surv(fu_time, death) ~ gender) # fit the desired model
temp <- cox.zph(fit)# apply the cox.zph function to the desired model
print(temp) # display the results
plot(temp) # plot the curves
```

2. Residual for Cox model

- Schoenfeld Residuals
  - Schoenfeld Residuals test whether two hazard functions are parallel, or "proportional".
  - e.g. If residuals for gender do not correlate the follow-up time, i.e. independent of time, the assumption is valid.
- Martingale Residuals
  - Martingale Residuals test whether a continuous predictor, such as age, has a linear relationship with the outcome.
  - Martingale residuals may present any value between minus infinity and 1, and have a mean of zero
    - Martingale residuals near 1 represent individuals that “died too soon”
    - Large negative values correspond to individuals that “lived too long”
- Deviance Residuals
  - Deviance Residuals spot influential points
  - Influential points: those are unusual enough to have a big influence on the model's hazard ratio
  - In a Cox model, one unusual point can dramatically affect the size of one or more hazard ratio.

3. Model selection methods

- forwards selection - awful
- stepwise selection - awful
- backwards elimination - fine
  1. Fit the model containing all your chosen predictors – either all your a priori ones or all your available ones (if your data set isn’t too large)
  2. Store all the coefficients from that model
  3. Remove in one go all predictors whose p value is above the preset threshold, typically the usual 0.05 (in a variant of this, you remove the predictor with the highest p value and refit the model, repeating steps until all the predictors have p values above the chosen threshold)
  4. Compare the coefficients for the remaining predictors with their coefficients from the original model
     - until the coefficients haven’t changed much from the original model
     - If a predictor's coefficient has changed noticeably, find the variable(s) that's been removed that are correlated with this affected predictor
     - Anything less than 0.05, e.g. a change from HR=1.30 to HR=1.34, is not big enough
- using priori knowledge - always good