432 Class 19 Slides

github.com/THOMASELOVE/2019-432

2019-04-09

Preliminaries

```r
library(skimr)
library(rms)
library(survival)
library(OIsurv)
library(survminer)
library(tidyverse)

survex <- read.csv("data/survex.csv") %>% tbl_df
```

Today's Agenda

- LTE: Project 2 Advice
- Data Visualization
- Time-to-event data
  - The Survival Function, S(t)
  - Kaplan-Meier Estimation of the Survival Function
  - Creating Survival Objects in R
  - Drawing Survival Curves
  - Testing the difference between Survival Curves
  - The Hazard Function and its Estimation

Logic, Theory and Prior Empirical Evidence

1. In thinking about which of your available variables to collect, or include in modeling, there is no good substitute for logic, theory and prior empirical evidence, and making these decisions without reference to those concerns would be disastrous.

2. If you are going to use a more algorithmic or statistical approach to help you make decisions about subtler questions like whether a non-linear treatment of some information will be helpful in building a predictive model, you want to make use of multiple and modern tools to help you do so, rather than relying on p values or summaries like R^2 or even stepwise approaches generated for the original data, without cross-validating your results.

3. Whenever possible, logic, theory and prior empirical evidence should play the most important role in developing prediction models or models for causal inference. It is rare that those concerns alone will be **sufficient** in complex real-world scenarios, but they are always **necessary**.

Today's Visualization

- What's Warming the World? (bloomberg.com, 2015)
- Income Mobility (New York Times, 2018-03-27)

Why are these compelling? What about the design of these graphics helps draw you in?

From Twitter...

Julia Silge
@juliasilge

Follow

At today's @UtahRUG we are hearing from @ucdlevy about exploratory data analysis with the tidyverse #rstats

"If visualization isn't part of your exploratory process, you're doing it wrong" 💁

Link

Time-to-Event / Survival Data: An Introduction

Working with Time to Event Data

In many medical studies, the main outcome variable is the time to the occurrence of a particular event.

- In a randomized controlled trial of cancer, for instance, surgery, radiation, and chemotherapy might be compared with respect to time from randomization and the start of therapy until death.
    - In this case, the event of interest is the death of a patient, but in other situations it might be remission from a disease, relief from symptoms or the recurrence of a particular condition.
    - Such observations are generally referred to by the generic term survival data even when the endpoint or event being considered is not death but something else.

What Do We Study in a Time-to-Event Study?

Survival analysis is concerned with prospective studies. We start with a cohort of patients and follow them forwards in time to determine some clinical outcome.

- Follow-up continues until either some event of interest occurs, the study ends, or further observation becomes impossible.

The outcomes in a survival analysis consist of the patient's **fate** and **length of follow-up** at the end of the study.

- For some patients, the outcome of interest may not occur during follow-up.
- For such patients, whose follow-up time is *censored*, we know only that this event did not occur while the patient was being followed. We do not know whether or not it will occur at some later time.

Problems with Time to Event Data

The primary problems are *non-normality* and *censoring*. . .

1. Survival data are not symmetrically distributed. They will often appear positively skewed, with a few people surviving a very long time compared with the majority; so assuming a normal distribution will not be reasonable.

2. At the completion of the study, some patients may not have reached the endpoint of interest (death, relapse, etc.). Consequently, the exact survival times are not known.

   - All that is known is that the survival times are greater than the amount of time the individual has been in the study.
   - The survival times of these individuals are said to be **censored** (precisely, they are right-censored).

Next, we'll define some special functions to build models that address these concerns.

The Survival Function, $S(t)$

The **survival function**, $S(t)$ (sometimes called the survivor function) is the probability that the survival time, T, is greater than or equal to a particular time, t.

- $S(t) =$ proportion of people surviving to time t or beyond

If there's no censoring, the survival function is easy to estimate

When there is no censoring, this function is easily estimated as . . .

$$\hat{S}(t) = \frac{\# \text{ of subjects with survival times } \geq t}{n}$$

but this won't work if there is censoring.

Understanding the Kaplan-Meier Estimator

The survival function $S(t)$ is the probability of surviving until at least time t. It is essentially estimated by the number of patients alive at time t divided by the total number of study subjects remaining at that time.

The Kaplan-Meier estimator first orders the (unique) survival times from smallest to largest, then estimates the survival function at each unique survival time.

- The survival function at the second death time, $t_{(2)}$ is equal to the estimated probability of not dying at time $t_{(2)}$ conditional on the individual being still at risk at time $t_{(2)}$.

The Kaplan-Meier Estimator

1. Order the survival times from smallest to largest, where t_{(j)} is the jth largest unique survival time, so we have. . .

$$t_{(1)} \leq t_{(2)} \leq t_{(3)} \leq ... t_{(n)}$$

2. The Kaplan-Meier estimate of the survival function is

$$\hat{S}(t) = \prod_{j:t_{(j)} \leq t} (1 - \frac{d_j}{r_j})$$

where r_j is the number of people at risk just before $t_{(j)}$, including those censored at time $t_{(j)}$, and d_j is the number of people who experience the event at time $t_{(j)}$.

Creating a Survival Object in R

The Surv function, part of the survival package in R, will create a
survival object from two arguments:

1. time = follow-up time
2. event = a status indicator, where
   - event = 1 or TRUE means the event was observed (for instance, the patient died)
   - event = 0 or FALSE means the follow-up time was censored

The `survex` data frame

The `survex.csv` file on the course website is essentially the same as a file simulated by Frank Harrell and his team[1] to introduce some of the key results from the `cph` function, which is part of the `rms` package in R.

The `survex` data includes 1,000 subjects...

- `id` = patient ID (1-1000)
- `age` = patient's age at study entry, years
- `sex` = patient's sex (Male or Female)
- `study.yrs` = patient's years of observed time in study until death or censoring
- `death` = 1 if patient died, 0 if censored.

---

[1] see the rms package documentation

A first example: Looking at just 50 observations

```
set.seed(432); ex50 <- sample_n(survex, 50, replace = F)
ex50 %>% select(id, study.yrs, death) %>% summary()
```

```
      id            study.yrs          death
 Min.   : 41.0   Min.   : 0.613   Min.   :0.00
 1st Qu.:250.0   1st Qu.: 3.825   1st Qu.:0.00
 Median :593.5   Median : 6.424   Median :0.00
 Mean   :525.0   Mean   : 7.025   Mean   :0.16
 3rd Qu.:725.5   3rd Qu.:10.106   3rd Qu.:0.00
 Max.   :998.0   Max.   :14.589   Max.   :1.00
```

For a moment, let's focus on developing a survival object in this setting.

ex50 %>% skim(study.yrs, death)

```
Variable type: integer
 variable missing complete  n mean   sd p0 p25 median p75 p100      hist
    death       0        50 50 0.16 0.37  0   0      0   0    1 ▇▁▁▁▁▁▁▂

Variable type: numeric
  variable missing complete  n mean   sd   p0  p25 median  p75  p100     hist
 study.yrs       0        50 50 7.02 3.87 0.61 3.82   6.42 10.11 14.59 ▃▆▇▃▅▃▂▁
```

- `study.yrs` here is follow-up time, in years
- `death = 1` if subject had the event (death), 0 if not.

```
ex50 %>% count(death)
```

```
# A tibble: 2 x 2
  death     n
  <int> <int>
1     0    42
2     1     8
```

Building a Survival Object

```
surv_50 <- Surv(time = ex50$study.yrs, event = ex50$death)
head(surv_50, 3)
```

```
[1] 5.098+ 2.975  5.842+
```

- Subject 1 survived 5.098 years before being censored.
- Subject 2 survived 2.975 years, and then died.

Remember that 8 of these 50 subjects died, the rest were censored at the latest time where they were seen for follow-up.

On dealing with time-to-event data

You have these three subjects.

1. Alice died in the hospital after staying for 20 days.
2. Betty died at home on the 20th day after study enrollment, after staying in the hospital for the first ten days.
3. Carol left the hospital after 20 days, but was then lost to follow up.

Suppose you plan a time-to-event analysis.

- How should you code "time" and "status" to produce a "time-to-event" object you can model if . . .
  - **death** is your primary outcome
  - **length of hospital stay** is your primary outcome?

Building a Kaplan-Meier Estimate (entire sample)

Remember that surv_50 is the survival object we created.

```
km_50 <- survfit(surv_50 ~ 1)

print(km_50, print.rmean = TRUE)
```

```
Call: survfit(formula = surv_50 ~ 1)

        n      events    *rmean *se(rmean)      median
    50.00        8.00     12.58       0.65          NA
  0.95LCL     0.95UCL
    13.75          NA
   * restricted mean with upper limit =  14.6
```

- 8 events (deaths) occurred in 50 subjects.
- Restricted mean survival time is 12.58 years (upper limit 14.6?)
- Median survival time is NA (why?) but has a lower bound for 95% CI.

## Summary of the Kaplan-Meier Estimate

- Up to 1.13 years, no one died, but two people were censored (so 48 were at risk at that time). (Estimated survival probability = 0.979)
- By the time of the next death at 2.44 years, only 46 people were still at risk. (Estimated Pr(survival) now 0.958)

```r
summary(km_50)
```

```
Call: survfit(formula = surv_50 ~ 1)

  time n.risk n.event survival std.err lower 95% CI
  1.13     48       1    0.979  0.0206        0.940
  2.44     46       1    0.958  0.0292        0.902
  2.62     45       1    0.937  0.0354        0.870
  2.98     44       1    0.915  0.0405        0.839
  4.20     37       1    0.891  0.0464        0.804
  5.59     30       1    0.861  0.0535        0.762
  7.23     21       1    0.820  0.0648        0.702
 13.75      5       1    0.656  0.1556        0.412
```

Kaplan-Meier Plot, via `survminer`
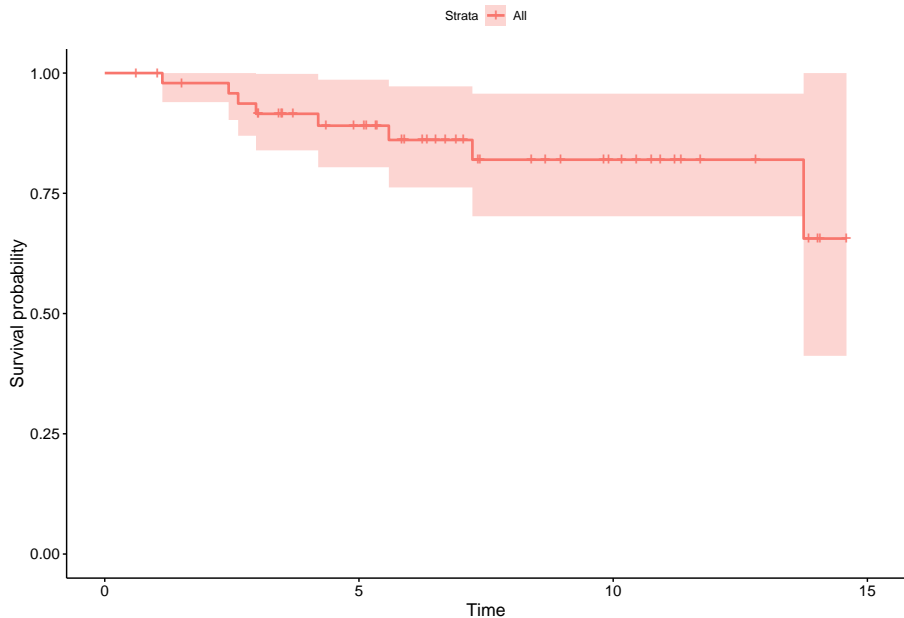
Kaplan-Meier Plot, via `survminer` (code)

```
ggsurvplot(km_50, data = ex50)
```

- The solid line indicates survival probability at each time point (in years.)
- The crosses indicate time points where censoring has occurred.
- The steps down indicate events (deaths.)
- The shading indicates (by default, 95%) pointwise confidence intervals.

For simultaneous confidence bands, visit the OpenIntro Statistics *Survival Analysis in R* materials, written by David Diez, as posted on our web site.

Adding a Number at Risk Table
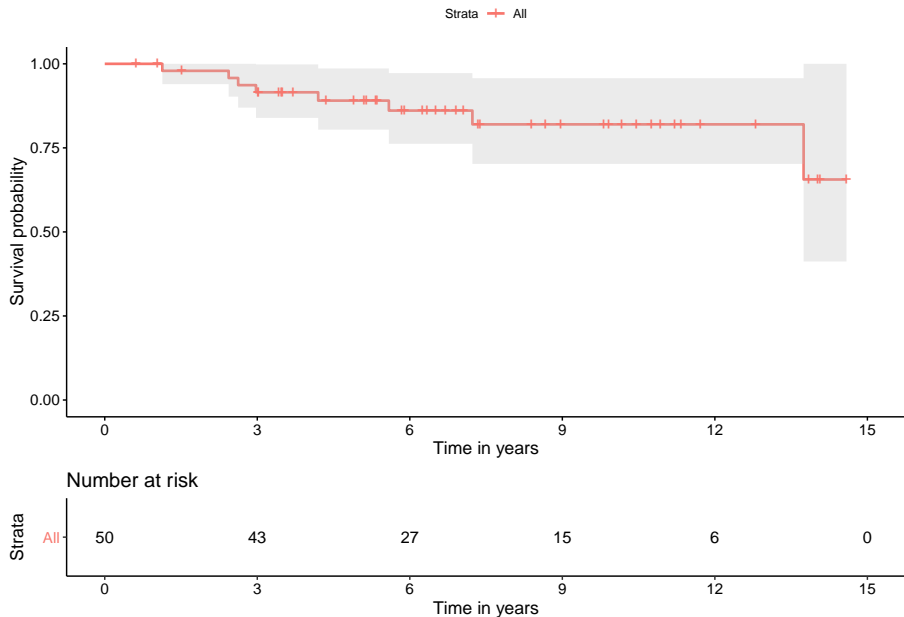
Adding a Number at Risk Table (code)

```
ggsurvplot(km_50, data = ex50,
   conf.int = TRUE,            # Add confidence interval
   risk.table = TRUE,          # Add risk table
   xlab = "Time in years",     # Adjust X axis label
   break.time.by = 3           # X ticks every 3 years
   )
```

Comparing Survival, by Sex

Suppose we want to compare the survival functions for subjects classified by their sex.

- So, for instance, in our sample, 4 of 17 females had the event (died).

```
ex50 %>% count(death, sex)
```

```
# A tibble: 4 x 3
  death sex        n
  <int> <fct>  <int>
1     0 Female    13
2     0 Male      29
3     1 Female     4
4     1 Male       4
```

Summarizing the Survival Function Estimate, by Sex

```
km_50_sex <- survfit(surv_50 ~ ex50$sex)

print(km_50_sex, print.rmean = TRUE)

Call: survfit(formula = surv_50 ~ ex50$sex)

                n events *rmean *se(rmean) median 0.95LCL
ex50$sex=Female 17      4   11.1      1.379     NA      NA
ex50$sex=Male   33      4   13.0      0.633     NA    13.7
                0.95UCL
ex50$sex=Female      NA
ex50$sex=Male        NA
    * restricted mean with upper limit =  14.3
```
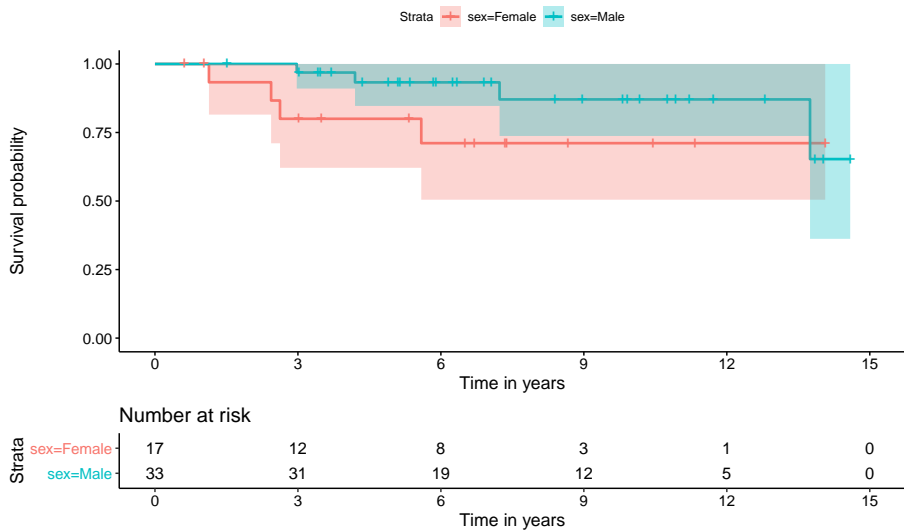
- Among females, 4 of 17 subjects died, and the estimated restricted mean survival is 11.1 years.

Kaplan-Meier Survival Function Estimates, by Sex

Kaplan-Meier Survival Function Estimates, by Sex (code)

```
ggsurvplot(km_50_sex, data = ex50,
           conf.int = TRUE,
           xlab = "Time in years",
           break.time.by = 3,
           risk.table = TRUE,
           risk.table.height = 0.25)
```

Testing the difference between 2 survival curves

To obtain a significance test comparing these two survival curves, we turn to a log rank test, which tests the null hypothesis $H_0 : S_1(t) = S_2(t)$ for all t where the two exposures have survival functions $S_1(t)$ and $S_2(t)$.

```
survdiff(surv_50 ~ ex50$sex)

Call:
survdiff(formula = surv_50 ~ ex50$sex)

                  N Observed Expected (O-E)^2/E (O-E)^2/V
ex50$sex=Female  17        4     2.23     1.395      1.95
ex50$sex=Male    33        4     5.77     0.541      1.95

 Chisq= 1.9  on 1 degrees of freedom, p= 0.2
```

At usual α levels, there's no significant difference between the survival curves stratified by sex.

Alternative log rank tests

An alternative is the *Peto and Peto modification of the Gehan-Wilcoxon test*, which results from adding rho=1 to the survdiff function (rho=0, the default, yields the log rank test.)

```
survdiff(surv_50 ~ ex50$sex, rho = 1)

Call:
survdiff(formula = surv_50 ~ ex50$sex, rho = 1)

                 N Observed Expected (O-E)^2/E (O-E)^2/V
ex50$sex=Female 17     3.83     2.07     1.499      2.26
ex50$sex=Male   33     3.53     5.29     0.586      2.26

 Chisq= 2.3  on 1 degrees of freedom, p= 0.1
```

Alternative log rank tests

- As compared to the log rank test, this Peto-Peto modification (and others using `rho > 0`) give greater weight to the left hand (earlier) side of the survival curves.
- To obtain chi-square tests that give greater weight to the right hand (later) side of the survival curves than the log rank test, use `rho < 0`.

The log rank test generalizes to permit survival comparisons across mode than two groups, with the test statistic having an asymptotic chi-squared distribution with one degree of freedom less than the number of patient groups being compared.

A Highly Customized K-M Plot
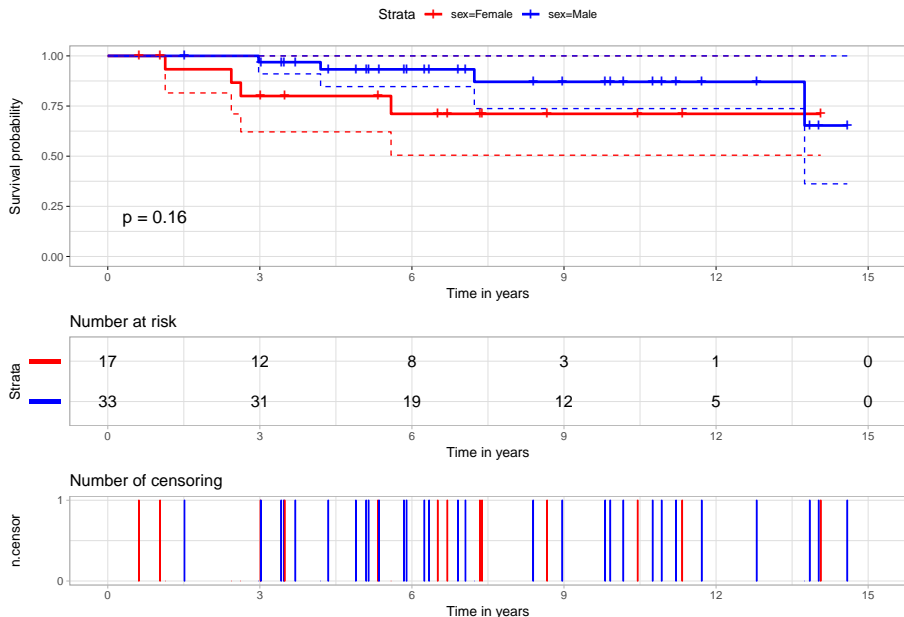
Customizing the K-M Plot Further

See https://github.com/kassambara/survminer/ for many more options.

Comparing Survival Functions, by sex, 1000 observations

```
surv_obj2 <- Surv(time = survex$study.yrs,
                  event = survex$death)


km_sex2 <- survfit(surv_obj2 ~ survex$sex)


survdiff(surv_obj2 ~ survex$sex)


Call:
survdiff(formula = surv_obj2 ~ survex$sex)

                     N Observed Expected (O-E)^2/E (O-E)^2/V
survex$sex=Female  380       90     62.7     11.85      18.1
survex$sex=Male    620       93    120.3      6.18      18.1

 Chisq= 18.1  on 1 degrees of freedom, p= 2e-05
```
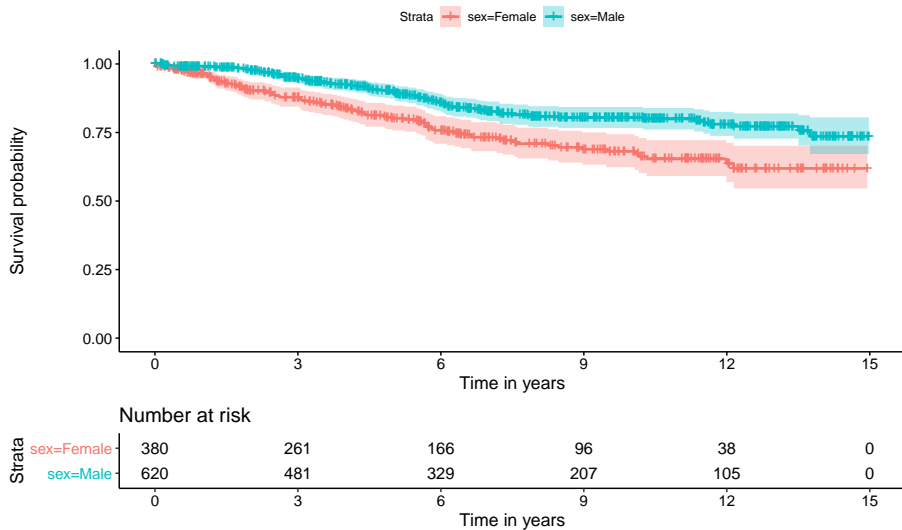
Kaplan-Meier Plot of Survival, by Sex (n = 1000)

The Hazard Function

To build regression models for time-to-event data, we will need to introduce the **hazard function**.

If $S(t)$ is the survival function, and time t is taken to be continuous, then $S(t) = e^{H(t)}$ defines the hazard function $H(t)$.

- Note that $H(t) = -ln(S(t))$.
- The function $H(t)$ is an important analytic tool.
    - It is used to describe the concept of the risk of "failure" in an interval after time t, conditioned on the subject having survived to time t.
    - It is often called the *cumulative hazard function*, to emphasize the fact that its value is the "sum" of the hazard up to time t.

Understanding the Hazard Function

Consider a subject in the survex study who has a survival time of 4 years.

- For this subject to die at 4 years, they had to survive for the first 3 years.
- The subject's hazard at 4 years is the failure rate "per years" conditional on the subject being hemorrhage-free for the first 3 years.

There are several different methods to estimate $H(t)$ and I'll discuss two now:

1. The inverse Kaplan-Meier estimator
2. The Nelson-Aalen estimator

The Inverse Kaplan-Meier Estimator of $H(t)$

I'll create something called H.est1, the inverse K-M estimate...

```
surv_50 <- Surv(ex50$study.yrs, ex50$death)
km_50 <- survfit(surv_50 ~ 1)
Haz1.almost <- -log(km_50$surv)
H_est1 <- c(Haz1.almost, tail(Haz1.almost, 1))
```
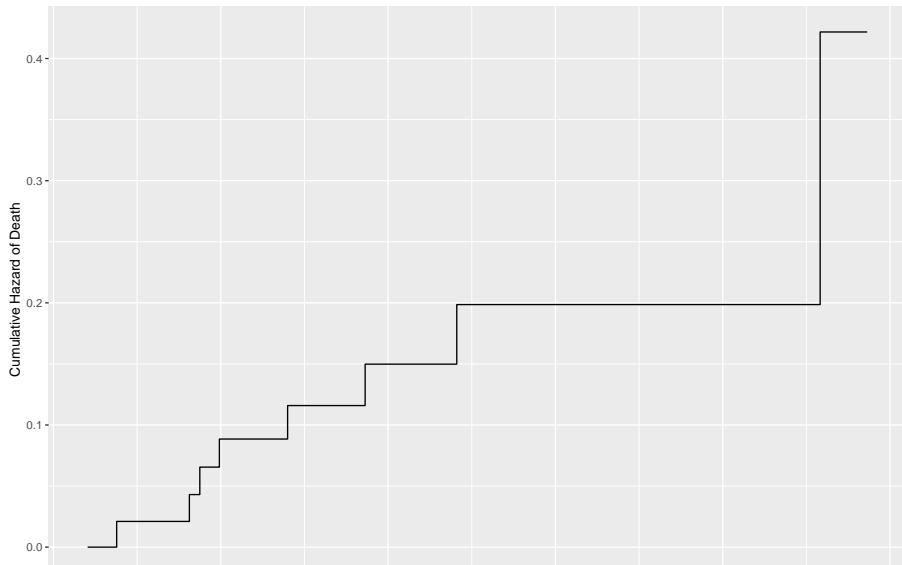
Create a data frame of the times and hazard estimates

```
haz_frame <- data_frame(
    time = c(km_50$time, tail(km_50$time, 1)),
    inverse_KM = H_est1
)

Warning: `data_frame()` is deprecated, use `tibble()`.
This warning is displayed once per session.
```

Cumulative Hazard Function from Inverse Kaplan-Meier

Cumulative Hazard Function via Inverse K–M

Cumulative Hazard Function from Inverse Kaplan-Meier (code)

```r
ggplot(haz_frame, aes(x = time, y = inverse_KM)) +
    geom_step() +
    scale_x_continuous(breaks = c(0, 3, 6, 9, 12, 15)) +
    labs(x = "Years of Follow-Up",
         y = "Cumulative Hazard of Death",
         title = "Cumulative Hazard Function via Inverse K-M")
```

Nelson-Aalen Estimator of $H(t)$

We'll create the Nelson-Aalen estimate, H_est2.

```
h.sort.of <- km_50$n.event / km_50$n.risk
Haz2.almost <- cumsum(h.sort.of)
H_est2 <- c(Haz2.almost, tail(Haz2.almost, 1))
```

Add Nelson-Aalen Estimate to our Data Frame

```
haz_frame$Nelson_Aalen <- H_est2

haz_frame %>% head(., 5)

# A tibble: 5 x 3
   time inverse_KM Nelson_Aalen
  <dbl>      <dbl>        <dbl>
1 0.613      0            0
2 1.03       0            0
3 1.13       0.0211       0.0208
4 1.51       0.0211       0.0208
5 2.44       0.0430       0.0426
```

Convert Wide Data to Long

In order to easily plot the two hazard function estimates in the same graph, we'll want to convert these data from wide format to long format, with the gather function.

```
haz_frame_long <- gather(haz_frame, key = "method",
                         value = "hazardest",
                         inverse_KM:Nelson_Aalen)

head(haz_frame_long)

# A tibble: 6 x 3
   time method      hazardest
  <dbl> <chr>           <dbl>
1 0.613 inverse_KM          0
2 1.03  inverse_KM          0
3 1.13  inverse_KM     0.0211
4 1.51  inverse_KM     0.0211
5 2.44  inverse_KM     0.0430
```

Plot Hazard Estimates and Compare

Cumulative Hazard Function
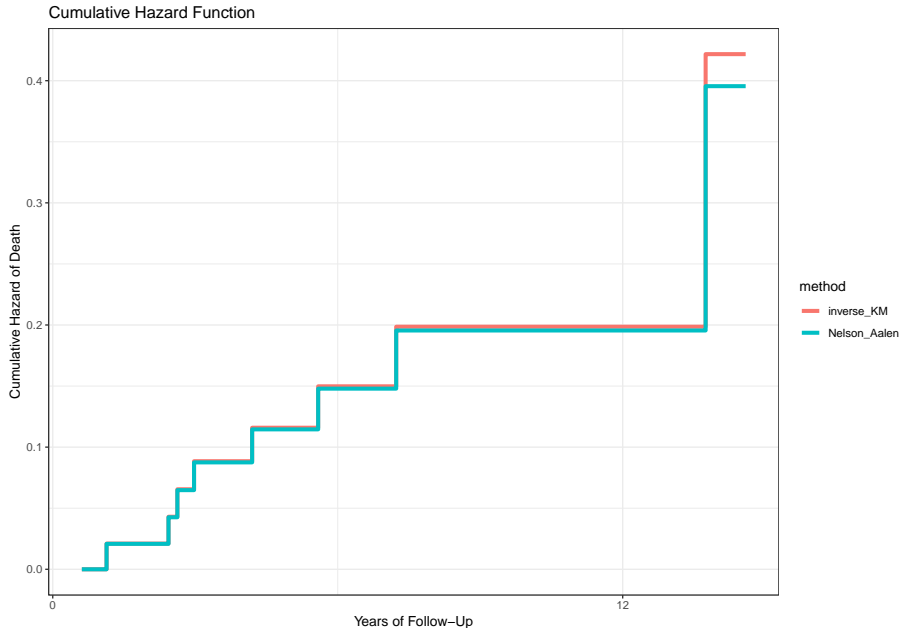
Plot Hazard Estimates and Compare (code)

```
ggplot(haz_frame_long, aes(x = time, y = hazardest,
                    col = method)) +
    geom_step(size = 1.5) +
    scale_x_continuous(breaks = c(0, 12, 24, 36, 48)) +
    labs(x = "Years of Follow-Up",
         y = "Cumulative Hazard of Death",
         title = "Cumulative Hazard Function") +
    theme_bw()
```

Estimating the Hazard Function in all 1000 observations

```
km_2 <- survfit(surv_obj2 ~ 1)
Haz1.almost2 <- -log(km_2$surv)
H_est_invKM2 <- c(Haz1.almost2, tail(Haz1.almost2, 1))

h.sort.of2 <- km_2$n.event / km_2$n.risk
Haz2.almost2 <- cumsum(h.sort.of2)
H_est_NelAal2 <- c(Haz2.almost2, tail(Haz2.almost2, 1))
```
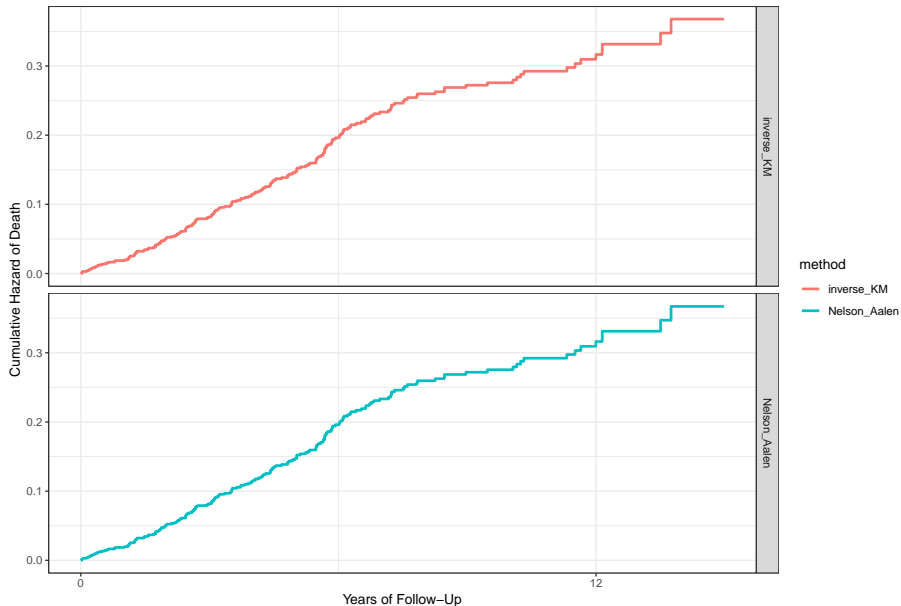
Creating the Data Frame (n = 1000 observations)

```
haz_frame2 <- data_frame(
    time = c(km_2$time, tail(km_2$time, 1)),
    inverse_KM = H_est_invKM2,
    Nelson_Aalen = H_est_NelAal2)

haz_frame_long2 <- gather(haz_frame2, key = "method",
                          value = "hazardest",
                          inverse_KM:Nelson_Aalen)
```

Plot of the resulting Hazard Estimates

Cumulative Hazard Function (based on n = 1000)

Next Time

Building a Cox Proportional Hazards Regression Model