432 Class 14

https://thomaselove.github.io/432-2023/

2023-03-02

Today's Agenda

1. Discussion of Quiz 1
2. Introduction to Time-to-Event Data

- The Survival Function, S(t)
    - Kaplan-Meier Estimation of the Survival Function
    - Creating Survival Objects in R
    - Drawing a Survival Curve
    - Comparing Survival Curves with log-rank tests

Today's R Setup

```
knitr::opts_chunk$set(comment = NA)

library(janitor)
library(broom)
library(knitr)
library(mosaic)
library(rms)
library(survival) ## new today
library(survminer) ## new today
library(tidyverse)

theme_set(theme_bw())
```

The **survex** Data Set

```
survex <- read_csv("c14/data/survex.csv",
                   show_col_types = FALSE) |>
  type.convert(as.is = FALSE)

head(survex)
```

```
# A tibble: 6 x 5
  sub_id   age grp   study_yrs death
   <int> <dbl> <fct>     <dbl> <int>
1      1  60.6 B          3.10     1
2      2  42.1 B          1.57     0
3      3  54.9 B          3.24     0
4      4  55.8 B         12.5      0
5      5  52.5 A          3.25     0
6      6  46.1 B          2.84     0
```

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 our website is motivated by a similar 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…

- `sub_id` = patient ID (1-1000)
- `age` = patient's age at study entry, years
- `grp` = patient's group (A or B)
- `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 100 observations

```r
set.seed(4322020)
ex100 <- sample_n(survex, 100, replace = F)
ex100 |> select(sub_id, study_yrs, death) |> summary()
```

```
     sub_id         study_yrs          death     
 Min.   : 23.0   Min.   : 0.175   Min.   :0.00  
 1st Qu.:258.2   1st Qu.: 2.122   1st Qu.:0.00  
 Median :468.0   Median : 4.864   Median :0.00  
 Mean   :479.1   Mean   : 6.007   Mean   :0.17  
 3rd Qu.:710.0   3rd Qu.: 9.759   3rd Qu.:0.00  
 Max.   :938.0   Max.   :14.817   Max.   :1.00  
```

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

Relationship between death and study_yrs?

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

```
favstats(study_yrs ~ death, data = ex100) |>
  kable(digits = 2)
```

death	min	Q1	median	Q3	max	mean	sd	n	missing
0	0.17	2.48	5.27	10.23	14.82	6.37	4.46	83	0
1	0.64	1.85	2.64	4.82	13.75	4.21	3.78	17	0

Building a Survival Object

```
surv_100 <- Surv(time = ex100$study_yrs, event = ex100$death)

head(surv_100, 3)
```

```
[1] 3.047  9.454+ 4.023+
```

- Subject 1 survived 3.047 years and then died.
- Subject 2 survived 9.454 years before being censored.
- Subject 3 survived 4.023 years before being censored.

Remember that 17 of these 100 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 "event" 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

Remember that `surv_100` is the survival object we created.

```
km_100 <- survfit(surv_100 ~ 1)

print(km_100, print.rmean = TRUE)
```

```
Call: survfit(formula = surv_100 ~ 1)

       n events rmean* se(rmean) median 0.95LCL 0.95UCL
[1,] 100     17   12.2     0.567     NA    13.7      NA
    * restricted mean with upper limit =  14.8
```

- 17 events (deaths) occurred in 100 subjects.
- Restricted mean survival time is 12.16 years (upper limit 14.8?)
- Median survival time is NA (why?) but has a lower bound for 95% CI.

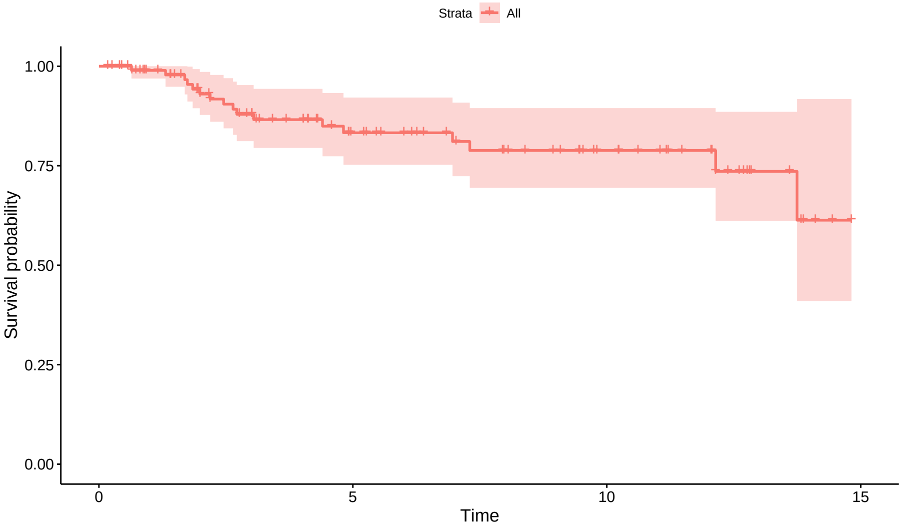
Summary of the Kaplan-Meier Estimate

- Up to 0.641 years, no one died, but five people were censored (so 95 were at risk at that time). (Estimated survival probability = 0.989)
- By the time of the next death at 1.312 years, only 87 people were still at risk. (Estimated Pr(survival) now 0.978)

```
summary(km_100)
```

```
Call: survfit(formula = surv_100 ~ 1)

  time n.risk n.event survival std.err lower 95% CI upper 95%
 0.641     95       1    0.989  0.0105        0.969         1.
 1.312     87       1    0.978  0.0153        0.949         1.
 1.690     82       1    0.966  0.0192        0.929         1.
 1.742     81       1    0.954  0.0224        0.911         0.
 1.846     80       1    0.942  0.0251        0.894         0.
 1.987     77       1    0.930  0.0276        0.878         0.
 2.190     74       1    0.918  0.0299        0.861         0.
 2.455     72       1    0.905  0.0321        0.844         0.
```

Kaplan-Meier Plot, via `survminer`



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

```
ggsurvplot(km_100, data = ex100)
```

- 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.

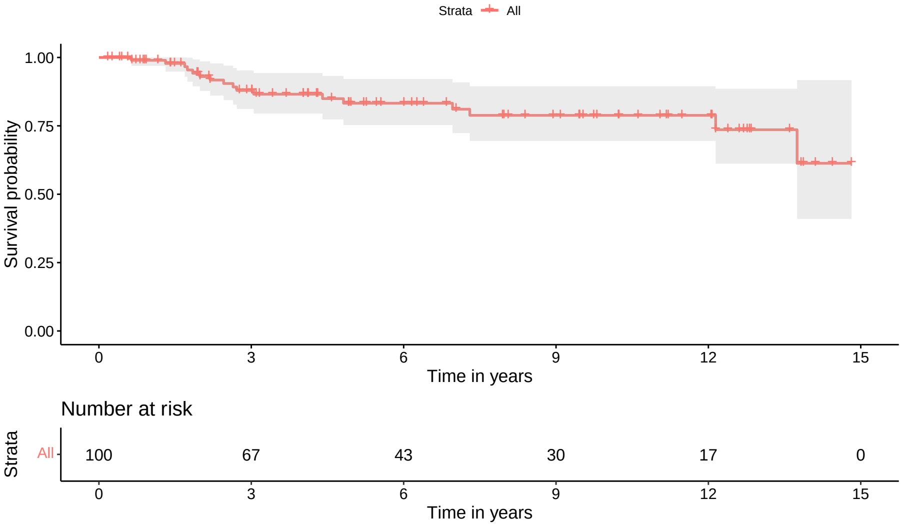
Where We Are So Far

- Created a small (n = 100) simulated data frame, called `ex100`.
- Observed 17 deaths, and 83 subjects censored before death.
- Survival object (containing time and fate) called `surv_100`
- Created Kaplan-Meier estimate of survival function, called `km_100`.
- Plotted the Kaplan-Meier estimate with `ggsurvplot()`.

Next steps:

1. Add a number at risk table to our Kaplan-Meier curve.
2. Consider potential predictors (age and group) of our time-to-event outcome.

Adding a Number at Risk Table



Adding a Number at Risk Table (code)

```
ggsurvplot(km_100, data = ex100,
   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 Group

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

- So, for instance, in our sample, 8 of 32 in group A and 9 of 68 in group B had the event (died).

```
ex100 |> tabyl(death, grp) |> adorn_totals()
```

```
 death  A  B
     0 24 59
     1  8  9
 Total 32 68
```

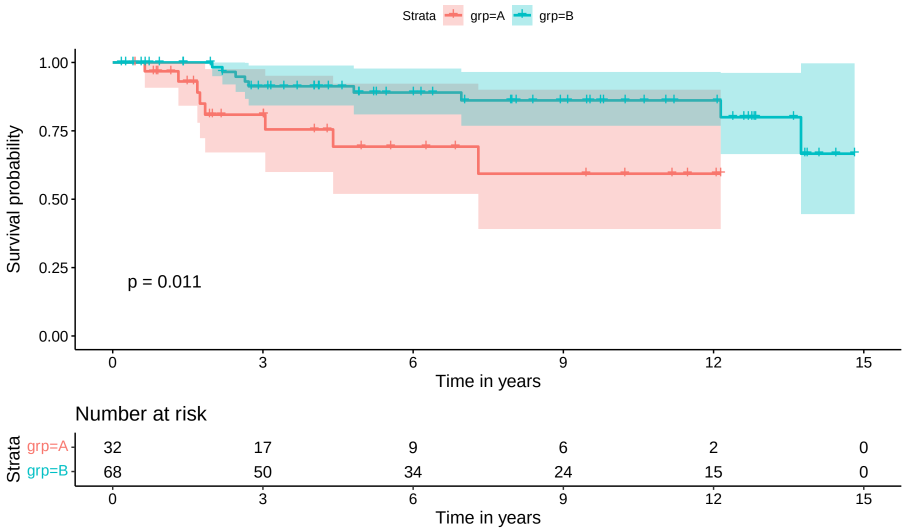
Estimated Survival Function, by Group

```
km_100_grp <- survfit(surv_100 ~ ex100$grp)

print(km_100_grp, print.rmean = TRUE)
```

```
Call: survfit(formula = surv_100 ~ ex100$grp)

             n events rmean* se(rmean) median 0.95LCL 0.95UCL
ex100$grp=A 32      8   10.2     1.325     NA     7.3      NA
ex100$grp=B 68      9   13.0     0.561     NA    13.7      NA
    * restricted mean with upper limit =  14.8
```

- 8 of 32 group A subjects died; estimated restricted mean survival time is 10.2 years.
- 9 of 68 in group B died, est. restricted mean survival = 13.0 years.

Kaplan-Meier Survival Function Estimates, by Group



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

```
ggsurvplot(km_100_grp, data = ex100,
           conf.int = TRUE,
           xlab = "Time in years",
           break.time.by = 3,
           risk.table = TRUE,
           risk.table.height = 0.25,
           pval = TRUE)
```

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_100 ~ ex100$grp)
```

```
Call:
survdiff(formula = surv_100 ~ ex100$grp)

             N Observed Expected (O-E)^2/E (O-E)^2/V
ex100$grp=A 32        8     3.75      4.81      6.39
ex100$grp=B 68        9    13.25      1.36      6.39

 Chisq= 6.4  on 1 degrees of freedom, p= 0.01
```

When comparing the survival curves stratified by group, the test gives p = 0.01

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_100 ~ ex100$grp, rho = 1)
```

```
Call:
survdiff(formula = surv_100 ~ ex100$grp, rho = 1)

             N Observed Expected (O-E)^2/E (O-E)^2/V
ex100$grp=A 32     7.44     3.45      4.62       6.7
ex100$grp=B 68     7.79    11.79      1.35       6.7

 Chisq= 6.7  on 1 degrees of freedom, p= 0.01
```

Alternative log rank tests

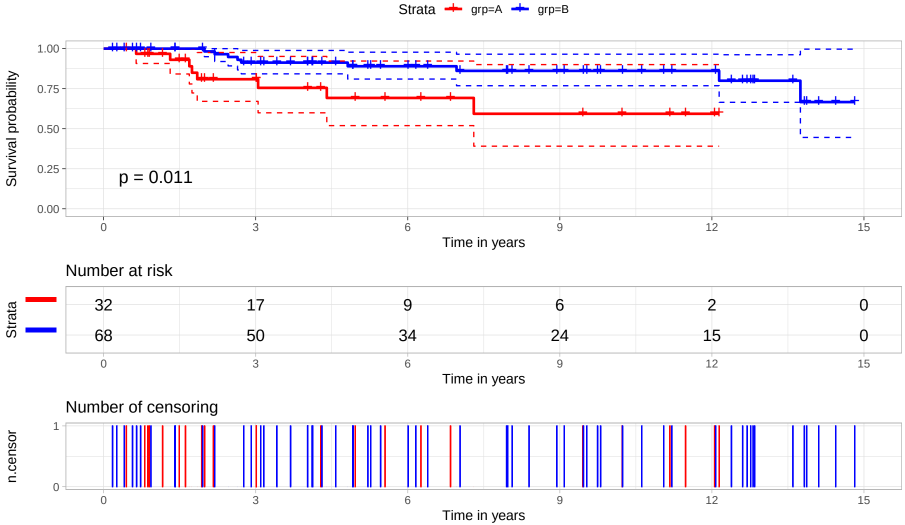
- As compared to the log rank test, this Peto-Peto modification (and others using `rho > 0`) give greater weight to the left hand (earlier) side of the survival curves.
- To obtain chi-square tests that give greater weight to the right hand (later) side of the survival curves than the log rank test,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

The log rank test generalizes to permit survival comparisons across more 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 (Code)

```
ggsurvplot(km_100_grp,
           data = ex100,
           palette = c("red", "blue"),
           risk.table = TRUE,
           pval = TRUE,
           conf.int = TRUE,
           xlab = "Time in years",
           break.time.by = 3,
           ggtheme = theme_light(),
           risk.table.y.text.col = T,
           risk.table.height = 0.25,
           risk.table.y.text = FALSE,
           ncensor.plot = TRUE,
           ncensor.plot.height = 0.25,
           conf.int.style = "step",
           surv.median.line = "hv")
```

A Highly Customized K-M Plot



Customizing the K-M Plot Further

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

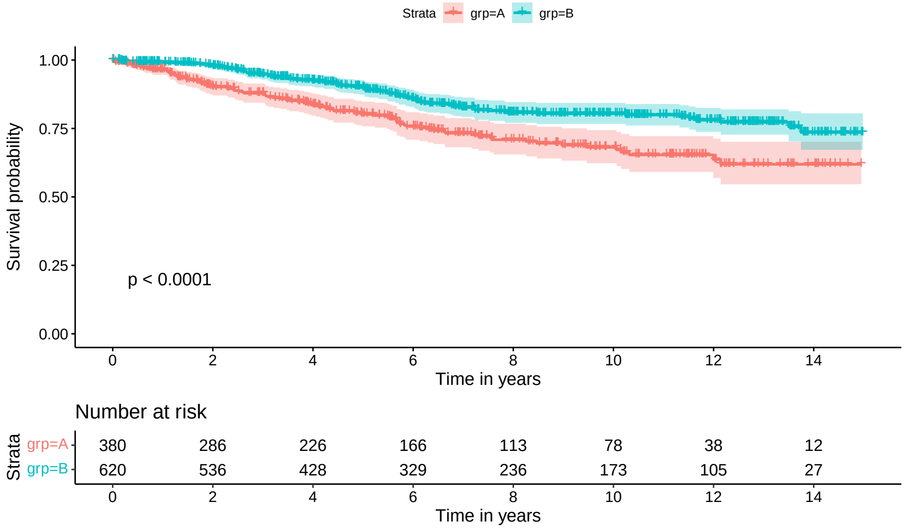
Also, consider the YouTube Video I've linked from Frank Harrell entitled "Survival Curves: Showing More by Showing Less" which highlights the value of interactive approaches.

Comparing Survival Functions, by group, n = 1000

```
surv_obj2 <- Surv(time = survex$study_yrs,
                  event = survex$death)

km_grp2 <- survfit(surv_obj2 ~ survex$grp)

survdiff(surv_obj2 ~ survex$grp)
```

```
Call:
survdiff(formula = surv_obj2 ~ survex$grp)

               N Observed Expected (O-E)^2/E (O-E)^2/V
survex$grp=A 380       90     62.7     11.85      18.1
survex$grp=B 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 Group (n = 1000)



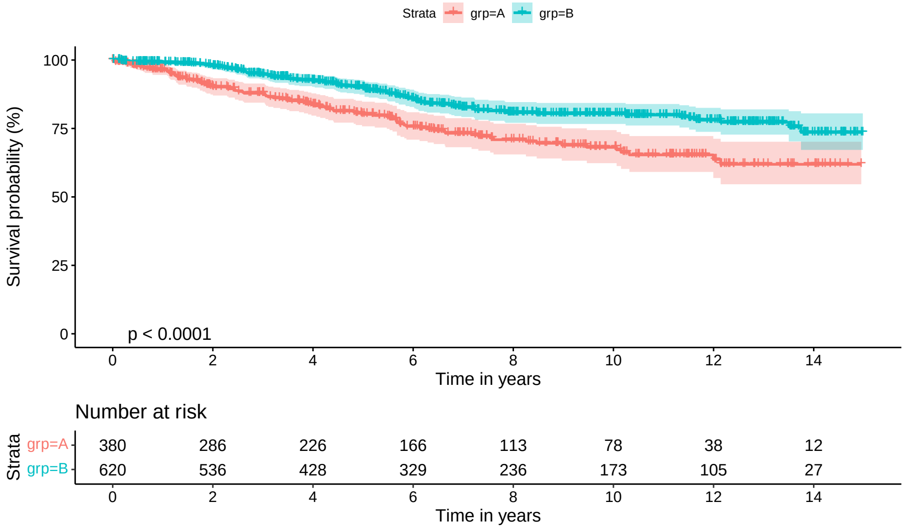
Kaplan-Meier Plot of Survival Percentage, Instead?

Just add `fun = "pct"` to the plot.

```
ggsurvplot(km_grp2, data = survex, fun = "pct",
           conf.int = TRUE,
           pval = TRUE,
           xlab = "Time in years",
           break.time.by = 2,
           risk.table = TRUE,
           risk.table.height = 0.25)
```

Result on next slide...

Kaplan-Meier Plot of Survival Percentage



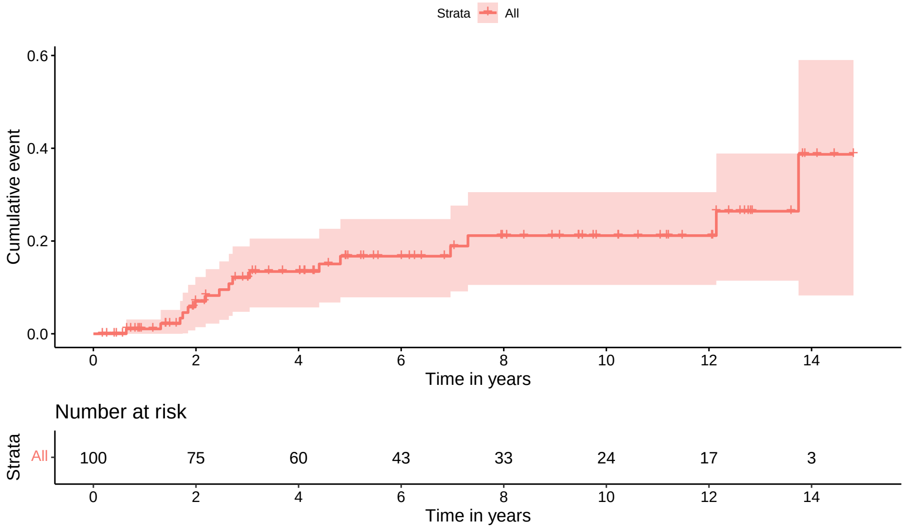
Code to plot Cumulative Event Rate

Let's look at our original `km_100` model for 100 observations.

- Add `fun = "event"` to our `ggsurvplot`.

```
ggsurvplot(km_100, data = survex, fun = "event",
           xlab = "Time in years",
           break.time.by = 2,
           risk.table = TRUE,
           risk.table.height = 0.25)
```

Can we plot the cumulative event rate instead?

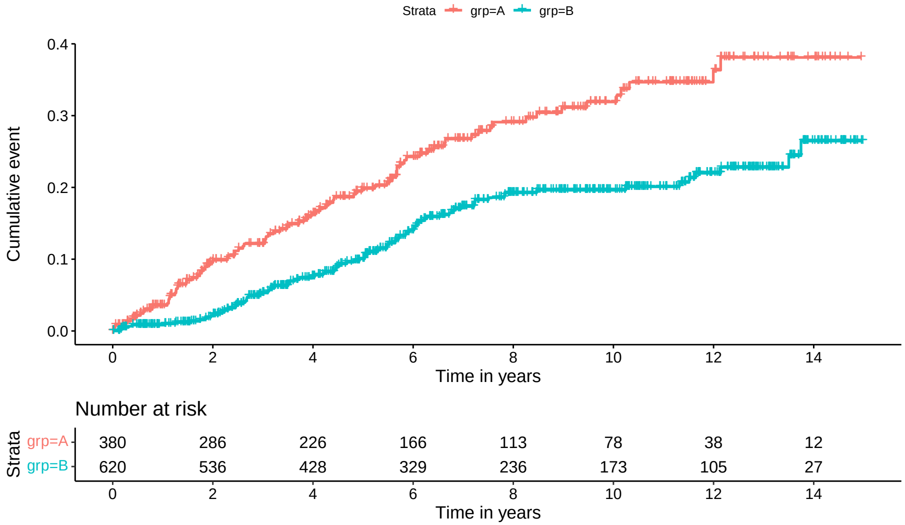


Cumulative Event Rate for `km_grp2` model

Let's look at our model for 1000 observations, that includes `grp`:

```
ggsurvplot(km_grp2, data = survex, fun = "event",
           xlab = "Time in years",
           break.time.by = 2,
           risk.table = TRUE,
           risk.table.height = 0.25)
```

Cumulative Event Rate for `km_grp2` model (Results)



More to come on Time-to-Event Data

after spring break…

Next week, we'll tackle regression on count outcomes.