432 Class 21

Thomas E. Love, Ph.D.

2024-04-02

## Today’s Topic: Time-to-Event Data

* Back in Class 14, we discussed
  + Kaplan-Meier Estimation of the Survival Function
  + Creating Survival Objects, Drawing Survival Curves
  + Testing the difference between Survival Curves
* Today: start Cox Proportional Hazards Regression
  + The Hazard Function and its Estimation

See Chapters 29-31 of our Course Notes

## Setup

knitr::opts\_chunk$set(comment=NA)  
options(width = 80)  
  
library(janitor)

Attaching package: 'janitor'

The following objects are masked from 'package:stats':  
  
 chisq.test, fisher.test

library(broom)  
library(gt)  
library(rms)

Loading required package: Hmisc

Attaching package: 'Hmisc'

The following object is masked from 'package:gt':  
  
 html

The following objects are masked from 'package:base':  
  
 format.pval, units

library(survival)  
library(survminer)

Loading required package: ggplot2

Loading required package: ggpubr

Attaching package: 'survminer'

The following object is masked from 'package:survival':  
  
 myeloma

library(tidyverse)

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
✔ dplyr 1.1.4 ✔ readr 2.1.5  
✔ forcats 1.0.0 ✔ stringr 1.5.1  
✔ lubridate 1.9.3 ✔ tibble 3.2.1  
✔ purrr 1.0.2 ✔ tidyr 1.3.0

── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
✖ dplyr::filter() masks stats::filter()  
✖ dplyr::lag() masks stats::lag()  
✖ dplyr::src() masks Hmisc::src()  
✖ dplyr::summarize() masks Hmisc::summarize()  
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

theme\_set(theme\_bw())

# A Simulated Example (survex)

## The survex example (from Class 14)

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.

To start, we’ll model a survival object Surv(study\_yrs, death) using grp.

## The survex example

survex <- read\_csv("c21/data/survex.csv", show\_col\_types = FALSE) |>  
 mutate(grp = factor(grp))  
  
survex

# A tibble: 1,000 × 5  
 sub\_id age grp study\_yrs death  
 <dbl> <dbl> <fct> <dbl> <dbl>  
 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  
 7 7 61.2 B 6.06 1  
 8 8 35.4 B 14.6 0  
 9 9 59.9 B 5.03 1  
10 10 75.4 A 3.25 0  
# ℹ 990 more rows

## Create a survival object

surv\_obj <- Surv(time = survex$study\_yrs,   
 event = survex$death)  
  
head(surv\_obj)

[1] 3.095 1.574+ 3.239+ 12.496+ 3.253+ 2.837+

### Comparing Survival by Group

km\_grp <- survfit(surv\_obj ~ survex$grp)  
  
km\_grp

Call: survfit(formula = surv\_obj ~ survex$grp)  
  
 n events median 0.95LCL 0.95UCL  
survex$grp=A 380 90 NA NA NA  
survex$grp=B 620 93 NA NA NA

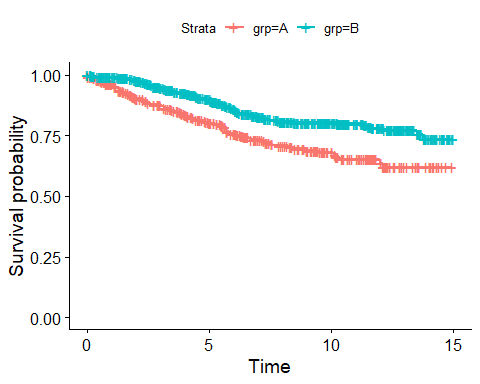
## Kaplan-Meier Estimates

summary(km\_grp)

Call: survfit(formula = surv\_obj ~ survex$grp)  
  
 survex$grp=A   
 time n.risk n.event survival std.err lower 95% CI upper 95% CI  
 0.013 378 1 0.997 0.00264 0.992 1.000  
 0.034 377 1 0.995 0.00373 0.987 1.000  
 0.047 376 1 0.992 0.00456 0.983 1.000  
 0.240 370 1 0.989 0.00528 0.979 1.000  
 0.261 369 1 0.987 0.00591 0.975 0.998  
 0.348 365 1 0.984 0.00648 0.971 0.997  
 0.411 362 1 0.981 0.00701 0.968 0.995  
 0.480 356 1 0.979 0.00751 0.964 0.993  
 0.552 353 1 0.976 0.00799 0.960 0.992  
 0.611 350 1 0.973 0.00844 0.957 0.990  
 0.641 348 1 0.970 0.00886 0.953 0.988  
 0.789 343 1 0.967 0.00928 0.949 0.986  
 0.797 342 1 0.965 0.00967 0.946 0.984  
 1.088 331 1 0.962 0.01007 0.942 0.982  
 1.116 330 1 0.959 0.01045 0.938 0.979  
 1.118 329 1 0.956 0.01082 0.935 0.977  
 1.133 328 1 0.953 0.01117 0.931 0.975  
 1.154 326 1 0.950 0.01152 0.928 0.973  
 1.242 322 1 0.947 0.01185 0.924 0.970  
 1.264 321 1 0.944 0.01218 0.920 0.968  
 1.268 320 1 0.941 0.01249 0.917 0.966  
 1.295 319 1 0.938 0.01280 0.913 0.964  
 1.312 317 1 0.935 0.01309 0.910 0.961  
 1.468 311 1 0.932 0.01339 0.906 0.959  
 1.479 310 1 0.929 0.01368 0.903 0.956  
 1.578 306 1 0.926 0.01397 0.899 0.954  
 1.690 303 1 0.923 0.01425 0.896 0.951  
 1.739 301 1 0.920 0.01453 0.892 0.949  
 1.742 300 1 0.917 0.01480 0.888 0.946  
 1.795 297 1 0.914 0.01507 0.885 0.944  
 1.846 295 1 0.911 0.01534 0.881 0.941  
 1.871 294 1 0.908 0.01559 0.878 0.939  
 1.952 289 1 0.905 0.01585 0.874 0.936  
 2.001 286 1 0.901 0.01611 0.870 0.933  
 2.284 280 1 0.898 0.01637 0.867 0.931  
 2.313 278 1 0.895 0.01663 0.863 0.928  
 2.437 274 1 0.892 0.01689 0.859 0.925  
 2.442 273 1 0.888 0.01714 0.855 0.923  
 2.513 272 1 0.885 0.01738 0.852 0.920  
 2.599 270 1 0.882 0.01762 0.848 0.917  
 2.624 269 1 0.879 0.01786 0.844 0.914  
 3.035 259 1 0.875 0.01811 0.840 0.911  
 3.047 258 1 0.872 0.01836 0.837 0.908  
 3.062 257 1 0.868 0.01859 0.833 0.906  
 3.107 256 1 0.865 0.01883 0.829 0.903  
 3.209 253 1 0.862 0.01906 0.825 0.900  
 3.345 249 1 0.858 0.01930 0.821 0.897  
 3.476 246 1 0.855 0.01953 0.817 0.894  
 3.528 243 1 0.851 0.01977 0.813 0.891  
 3.722 239 1 0.848 0.02000 0.809 0.888  
 3.820 234 1 0.844 0.02024 0.805 0.885  
 3.901 230 1 0.840 0.02048 0.801 0.881  
 3.968 227 1 0.837 0.02072 0.797 0.878  
 4.066 223 1 0.833 0.02097 0.793 0.875  
 4.143 220 1 0.829 0.02121 0.788 0.872  
 4.248 218 1 0.825 0.02145 0.784 0.868  
 4.304 215 1 0.821 0.02169 0.780 0.865  
 4.402 212 1 0.817 0.02193 0.776 0.862  
 4.418 211 1 0.814 0.02217 0.771 0.858  
 4.813 204 1 0.810 0.02242 0.767 0.855  
 4.854 202 1 0.806 0.02266 0.762 0.851  
 4.979 198 1 0.802 0.02291 0.758 0.848  
 5.234 192 1 0.797 0.02317 0.753 0.844  
 5.474 188 1 0.793 0.02343 0.749 0.840  
 5.529 185 1 0.789 0.02369 0.744 0.837  
 5.588 180 1 0.784 0.02396 0.739 0.833  
 5.675 176 1 0.780 0.02424 0.734 0.829  
 5.676 175 2 0.771 0.02477 0.724 0.821  
 5.747 172 1 0.767 0.02502 0.719 0.817  
 5.823 169 1 0.762 0.02528 0.714 0.813  
 5.866 168 1 0.758 0.02554 0.709 0.809  
 6.128 162 1 0.753 0.02580 0.704 0.805  
 6.289 156 1 0.748 0.02609 0.699 0.801  
 6.412 150 1 0.743 0.02638 0.693 0.797  
 6.631 142 1 0.738 0.02671 0.687 0.792  
 6.644 141 1 0.733 0.02703 0.681 0.788  
 7.167 130 1 0.727 0.02740 0.675 0.783  
 7.302 127 1 0.721 0.02778 0.669 0.778  
 7.538 120 1 0.715 0.02819 0.662 0.773  
 7.606 116 1 0.709 0.02861 0.655 0.767  
 8.253 110 1 0.703 0.02907 0.648 0.762  
 8.468 106 1 0.696 0.02954 0.640 0.756  
 8.968 98 1 0.689 0.03008 0.632 0.750  
 9.469 87 1 0.681 0.03076 0.623 0.744  
 10.064 76 1 0.672 0.03163 0.613 0.737  
 10.152 74 1 0.663 0.03248 0.602 0.730  
 10.319 70 1 0.653 0.03337 0.591 0.722  
 11.994 39 1 0.637 0.03648 0.569 0.712  
 12.142 36 1 0.619 0.03952 0.546 0.702  
  
 survex$grp=B   
 time n.risk n.event survival std.err lower 95% CI upper 95% CI  
 0.139 615 1 0.998 0.00162 0.995 1.000  
 0.174 613 1 0.997 0.00230 0.992 1.000  
 0.209 610 1 0.995 0.00282 0.990 1.000  
 0.314 604 1 0.993 0.00326 0.987 1.000  
 0.382 603 1 0.992 0.00364 0.985 0.999  
 1.021 570 1 0.990 0.00403 0.982 0.998  
 1.255 564 1 0.988 0.00439 0.980 0.997  
 1.562 551 1 0.987 0.00474 0.977 0.996  
 1.741 546 1 0.985 0.00506 0.975 0.995  
 1.833 543 1 0.983 0.00537 0.972 0.993  
 1.878 541 1 0.981 0.00565 0.970 0.992  
 1.959 538 1 0.979 0.00593 0.968 0.991  
 1.987 537 1 0.977 0.00619 0.965 0.990  
 2.108 530 1 0.976 0.00645 0.963 0.988  
 2.190 526 1 0.974 0.00670 0.961 0.987  
 2.230 524 1 0.972 0.00694 0.958 0.986  
 2.252 522 1 0.970 0.00717 0.956 0.984  
 2.339 518 1 0.968 0.00740 0.954 0.983  
 2.447 514 1 0.966 0.00762 0.951 0.981  
 2.455 512 1 0.964 0.00784 0.949 0.980  
 2.475 510 1 0.962 0.00805 0.947 0.978  
 2.584 504 1 0.961 0.00825 0.945 0.977  
 2.641 501 1 0.959 0.00846 0.942 0.975  
 2.667 500 1 0.957 0.00865 0.940 0.974  
 2.674 499 1 0.955 0.00885 0.938 0.972  
 2.681 498 1 0.953 0.00903 0.935 0.971  
 2.715 497 1 0.951 0.00922 0.933 0.969  
 2.936 485 1 0.949 0.00940 0.931 0.968  
 2.975 482 1 0.947 0.00959 0.928 0.966  
 3.095 478 1 0.945 0.00977 0.926 0.964  
 3.115 476 1 0.943 0.00995 0.924 0.963  
 3.140 475 1 0.941 0.01013 0.921 0.961  
 3.189 472 1 0.939 0.01030 0.919 0.960  
 3.235 469 1 0.937 0.01047 0.917 0.958  
 3.501 447 1 0.935 0.01065 0.914 0.956  
 3.515 446 2 0.931 0.01101 0.909 0.953  
 3.632 441 1 0.929 0.01119 0.907 0.951  
 3.729 437 1 0.927 0.01136 0.905 0.949  
 3.983 431 1 0.924 0.01154 0.902 0.947  
 4.035 424 1 0.922 0.01171 0.900 0.946  
 4.198 418 1 0.920 0.01189 0.897 0.944  
 4.232 417 1 0.918 0.01207 0.895 0.942  
 4.419 408 1 0.916 0.01224 0.892 0.940  
 4.449 407 1 0.913 0.01242 0.889 0.938  
 4.467 405 1 0.911 0.01259 0.887 0.936  
 4.500 403 1 0.909 0.01276 0.884 0.934  
 4.544 401 1 0.907 0.01293 0.882 0.932  
 4.675 397 1 0.904 0.01310 0.879 0.930  
 4.815 394 1 0.902 0.01326 0.876 0.928  
 4.929 383 1 0.900 0.01344 0.874 0.926  
 5.030 380 2 0.895 0.01378 0.868 0.922  
 5.031 378 1 0.893 0.01394 0.866 0.920  
 5.114 372 1 0.890 0.01411 0.863 0.918  
 5.292 362 1 0.888 0.01428 0.860 0.916  
 5.326 360 1 0.885 0.01445 0.857 0.914  
 5.477 354 1 0.883 0.01463 0.854 0.912  
 5.481 353 1 0.880 0.01480 0.852 0.910  
 5.500 352 1 0.878 0.01497 0.849 0.908  
 5.640 347 1 0.875 0.01514 0.846 0.905  
 5.647 345 1 0.873 0.01530 0.843 0.903  
 5.708 341 1 0.870 0.01547 0.840 0.901  
 5.716 340 1 0.868 0.01564 0.837 0.899  
 5.844 335 1 0.865 0.01580 0.835 0.896  
 5.853 334 1 0.862 0.01597 0.832 0.894  
 5.929 331 1 0.860 0.01613 0.829 0.892  
 6.016 327 1 0.857 0.01629 0.826 0.890  
 6.026 326 1 0.854 0.01645 0.823 0.887  
 6.061 324 1 0.852 0.01661 0.820 0.885  
 6.104 321 1 0.849 0.01677 0.817 0.883  
 6.107 320 1 0.847 0.01693 0.814 0.880  
 6.216 316 1 0.844 0.01708 0.811 0.878  
 6.273 313 1 0.841 0.01724 0.808 0.876  
 6.533 302 1 0.838 0.01741 0.805 0.873  
 6.720 291 1 0.835 0.01758 0.802 0.871  
 6.762 290 1 0.833 0.01776 0.799 0.868  
 6.824 285 1 0.830 0.01793 0.795 0.866  
 6.962 279 1 0.827 0.01811 0.792 0.863  
 7.215 266 1 0.824 0.01831 0.788 0.860  
 7.231 265 1 0.820 0.01850 0.785 0.858  
 7.236 264 1 0.817 0.01869 0.782 0.855  
 7.505 254 1 0.814 0.01889 0.778 0.852  
 7.823 248 1 0.811 0.01910 0.774 0.849  
 7.840 246 1 0.808 0.01931 0.771 0.846  
 8.465 222 1 0.804 0.01956 0.767 0.843  
 10.244 167 1 0.799 0.02003 0.761 0.839  
 11.318 133 1 0.793 0.02076 0.753 0.835  
 11.511 126 1 0.787 0.02153 0.746 0.830  
 11.643 120 1 0.780 0.02232 0.738 0.825  
 12.143 98 1 0.772 0.02347 0.728 0.820  
 13.502 47 1 0.756 0.02814 0.703 0.813  
 13.746 37 1 0.735 0.03400 0.672 0.805

## Plotting the K-M Curves

ggsurvplot(km\_grp, data = survex)



## Comparing Survival by group

This is the log rank test.

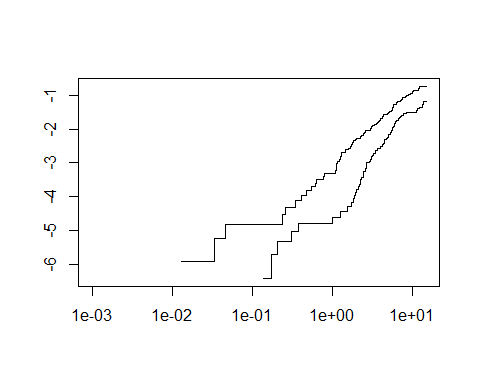
survdiff(surv\_obj ~ survex$grp)

Call:  
survdiff(formula = surv\_obj ~ 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

## Log - Log plot of K-M estimation

The two curves do not meet during the observation period, indicating the satisfaction of the proportional hazard assumption made by the log rank test.

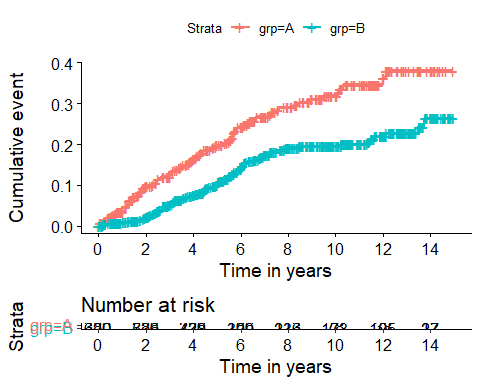
plot(survfit(surv\_obj ~ grp, data = survex), fun = "cloglog")



## Cumulative Event Rate for km\_grp

* Add fun = “event” to our ggsurvplot…

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



# The Hazard Function

## The Hazard Function

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

If is the survival function, and time is taken to be continuous, then defines .

* Note that = .

## Importance of the Hazard Function

The function is an important analytic tool.

* It is used to describe the concept of the risk of “failure” in an interval after time , conditioned on the subject having survived to time .
* is often called the *cumulative hazard function*, to emphasize the fact that its value is the “sum” of the hazard up to time .

## 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 year” conditional on the subject being alive through the first 3 years.

Let’s ignore the group information for a moment.

surv\_obj <- Surv(time = survex$study\_yrs,   
 event = survex$death)  
km1 <- survfit(surv\_obj ~ 1)

## Estimating the Hazard Function

Suppose we want to estimate across all subjects. There are several different methods, but we’ll focus on the inverse Kaplan-Meier estimator.

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

Haz1.almost <- -log(km1$surv)  
H\_est1 <- c(Haz1.almost, tail(Haz1.almost, 1))

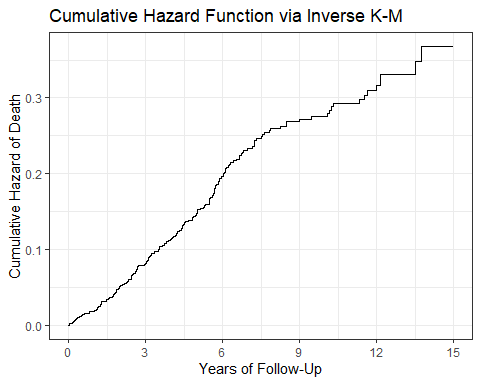
## Tibble of times and hazard estimates

haz\_frame <- tibble(  
 time = c(km1$time, tail(km1$time, 1)),  
 inverse\_KM = H\_est1)  
  
haz\_frame

# A tibble: 967 × 2  
 time inverse\_KM  
 <dbl> <dbl>  
 1 0.001 0   
 2 0.013 0.00100  
 3 0.018 0.00100  
 4 0.02 0.00100  
 5 0.034 0.00201  
 6 0.047 0.00301  
 7 0.059 0.00301  
 8 0.114 0.00301  
 9 0.121 0.00301  
10 0.135 0.00301  
# ℹ 957 more rows

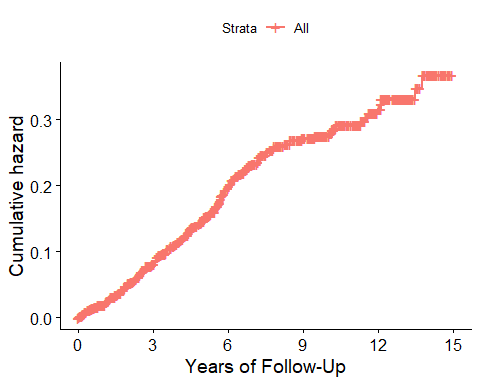
## Cumulative Hazard Function from Inverse Kaplan-Meier

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")



## Cumulative Hazard Plot via ggsurvplot

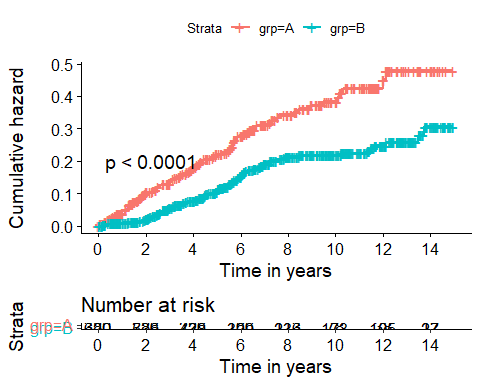
ggsurvplot(km1, data = survex, fun = "cumhaz",  
 conf.int = FALSE,  
 xlab = "Years of Follow-Up",  
 break.time.by = 3,  
 risk.table = FALSE)



## Plotting the Cumulative Hazard Function by Group

For our km\_grp fit, we’d use something like this…

ggsurvplot(km\_grp, data = survex, fun = "cumhaz",  
 xlab = "Time in years",  
 pval = TRUE,  
 break.time.by = 2,  
 risk.table = TRUE,  
 risk.table.height = 0.25)



# Fitting a Cox Proportional Hazards Regression

## Cox Proportional Hazards Regression

mod\_grp <- coxph(Surv(study\_yrs, death) ~ grp, data = survex)

The Cox proportional hazards model fits survival data with a constant (not varying over time) covariate (here, grp) to a hazard function of the form:

where we estimate the unknown value of and where is the baseline hazard which depends on but not on grp.

## Coefficients of our Cox model

mod\_grp

Call:  
coxph(formula = Surv(study\_yrs, death) ~ grp, data = survex)  
  
 coef exp(coef) se(coef) z p  
grpB -0.6195 0.5382 0.1481 -4.184 2.86e-05  
  
Likelihood ratio test=17.18 on 1 df, p=3.399e-05  
n= 1000, number of events= 183

Our hazard ratio estimate is 0.5382 for group B (vs. A)

* Hazard Ratio < 1 indicates a decrease in hazard for subjects in group B as compared to those in group A.
* Does this match our plot?

## What if we also include Age?

survex$S <- Surv(survex$study\_yrs, survex$death)  
  
mod\_age\_grp <- coxph(S ~ grp + age, data = survex)  
  
mod\_age\_grp

Call:  
coxph(formula = S ~ grp + age, data = survex)  
  
 coef exp(coef) se(coef) z p  
grpB -0.597528 0.550170 0.148207 -4.032 5.54e-05  
age 0.041920 1.042811 0.005571 7.525 5.26e-14  
  
Likelihood ratio test=69.93 on 2 df, p=6.522e-16  
n= 1000, number of events= 183

## Tidied coefficients: Age + Group model

tidy(mod\_age\_grp, exponentiate = TRUE, conf.int = T) |>  
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | p.value | conf.low | conf.high |
| --- | --- | --- | --- | --- | --- | --- |
| grpB | 0.550 | 0.148 | -4.032 | 0.000 | 0.411 | 0.736 |
| age | 1.043 | 0.006 | 7.525 | 0.000 | 1.031 | 1.054 |

* If Harry is a year older than Steve and both are in the same group, Harry’s hazard of death is 1.04 times that of Steve.
* If Harry (group B) and Sally (group A) are the same age, then Harry’s hazard of death is 0.55 times that of Sally.

## glance for this coxph model?

glance(mod\_age\_grp) |> select(3:10) |>  
 gt() |> tab\_options(table.font.size = 20) |>  
 fmt\_number(decimals = 3)

| statistic.log | p.value.log | statistic.sc | p.value.sc | statistic.wald | p.value.wald | statistic.robust | p.value.robust |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 69.932 | 0.000 | 73.331 | 0.000 | 75.830 | 0.000 | NA | NA |

glance(mod\_age\_grp) |> select(15:18) |>  
 gt() |> tab\_options(table.font.size = 20) |>  
 fmt\_number(columns = -nobs, decimals = 2)

| logLik | AIC | BIC | nobs |
| --- | --- | --- | --- |
| -1,132.83 | 2,269.67 | 2,276.08 | 1000 |

We see results from log rank, score and Wald tests, as well as the AIC and BIC. But that’s not all we can get from glance() with a Cox model…

## Concordance and

glance(mod\_age\_grp) |> select(c(1:2, 11:14)) |>  
 gt() |> tab\_options(table.font.size = 20) |>  
 fmt\_number(columns = r.squared:std.error.concordance, decimals = 3)

| n | nevent | r.squared | r.squared.max | concordance | std.error.concordance |
| --- | --- | --- | --- | --- | --- |
| 1000 | 183 | 0.068 | 0.903 | 0.688 | 0.020 |

The concordance is a goodness-of-fit measure. It describes the probability that the prediction goes in the same direction as the actual data (the fraction of concordant pairs between predictions and the data.) glance also provides a standard error for concordance.

We also get a Cox-Snell along with its maximum value (< 1.)

## Impact of adding age to group?

anova(mod\_age\_grp, mod\_grp)

Warning in anova.coxphlist(object, test = test): Models with response  
"Surv(study\_yrs, death)" removed because response differs from model 1

Analysis of Deviance Table  
 Cox model: response is S  
Terms added sequentially (first to last)  
  
 loglik Chisq Df Pr(>|Chi|)   
NULL -1167.8   
grp -1159.2 17.180 1 3.399e-05 \*\*\*  
age -1132.8 52.752 1 3.785e-13 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

AIC(mod\_age\_grp, mod\_grp)

df AIC  
mod\_age\_grp 2 2269.666  
mod\_grp 1 2320.418

BIC(mod\_age\_grp, mod\_grp)

df BIC  
mod\_age\_grp 2 2276.085  
mod\_grp 1 2323.627

## Checking the Proportional Hazards Assumption

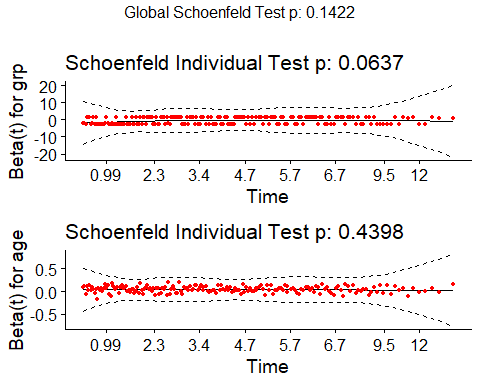
If the proportional hazards assumption is appropriate, we should see a slope of essentially zero in the residuals that are plotted against time on the next slide.

* If we see a slope that seriously different from zero, that will suggest a violation of the proportional hazards assumption.
* A hypothesis test is also performed, where a small *p* value indicates a potential problem with the assumption.

## Plot to Check Proportional Hazards

ggcoxzph(cox.zph(mod\_age\_grp))

Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):  
collapsing to unique 'x' values



## What happens if we see a violation?

We could add a non-linear predictor term or use a different kind of survival model.

* If the PH assumption fails on a categorical predictor, fit a Cox model stratified by that predictor (use strata(var) rather than var in the specification of the coxph model.)
* If the PH assumption is violated, this means the hazard isn’t constant over time, so we could fit separate Cox models for a series of time intervals.

## If we see a violation…

Another option would be to use an extension of the Cox model that permits covariates to vary over time.

Visit <https://cran.r-project.org/web/packages/survival/vignettes/timedep.pdf> for details on building the relevant data sets and models, with examples.

# A Real Data Example

## The brca trial

The brca data[[1]](#footnote-75) describes a parallel randomized trial of three treatments, adjuvant to surgery in subjects with stage-2 carcinoma of the breast.

* S\_CT = Surgery plus one year of chemotherapy
* S\_IT = Surgery plus one year of immunotherapy
* S\_Both = Surgery plus one year of chemotherapy and immunotherapy

## Variables in our brca data

The measure of efficacy was “time to death” in weeks.

In addition to treat, our variables are:

* trial\_weeks: time in the study, in weeks, to death or censoring
* last\_alive: 1 if alive at last follow-up (and thus censored), 0 if dead
* age: age in years at the start of the trial

## Ingesting the brca data

brca <- read\_csv("c21/data/brca.csv", show\_col\_types = FALSE) |>   
 mutate(across(where(is\_character), as\_factor),  
 subject = as.character(subject))

This is a typical right-censored survival data set with interest in the comparative analysis of the three treatments.

1. Does immunotherapy added to surgery plus chemotherapy improve survival? (Comparing S\_Both to S\_CT)
2. Does chemotherapy add efficacy to surgery plus immunotherapy? (S\_Both vs. S\_IT)
3. What is the effect of age on survival?

## The brca data (big problem: n = 31!)

brca

# A tibble: 31 × 5  
 subject treat trial\_weeks last\_alive age  
 <chr> <fct> <dbl> <dbl> <dbl>  
 1 A01 S\_CT 102 0 55  
 2 A02 S\_IT 192 0 62  
 3 A03 S\_Both 73 0 72  
 4 A04 S\_CT 58 1 48  
 5 A05 S\_CT 48 1 26  
 6 A06 S\_IT 182 1 52  
 7 A07 S\_IT 196 1 50  
 8 A08 S\_CT 177 1 49  
 9 A09 S\_IT 191 1 62  
10 A10 S\_Both 36 0 60  
# ℹ 21 more rows

## Create survival object

* trial\_weeks: time in the study, in weeks, to death or censoring
* last\_alive: 1 if alive at last follow-up (and thus censored), 0 if dead

So last\_alive = 0 if the event (death) occurs.

brca$S <- with(brca, Surv(trial\_weeks, last\_alive == 0))  
  
head(brca$S)

[1] 102 192 73 58+ 48+ 182+

## Build Kaplan-Meier Estimator

kmfit <- survfit(S ~ treat, dat = brca)  
  
print(kmfit, print.rmean = TRUE)

Call: survfit(formula = S ~ treat, data = brca)  
  
 n events rmean\* se(rmean) median 0.95LCL 0.95UCL  
treat=S\_CT 11 6 153 21.1 144 102 NA  
treat=S\_IT 10 5 192 19.3 192 144 NA  
treat=S\_Both 10 4 193 25.0 NA 139 NA  
 \* restricted mean with upper limit = 251

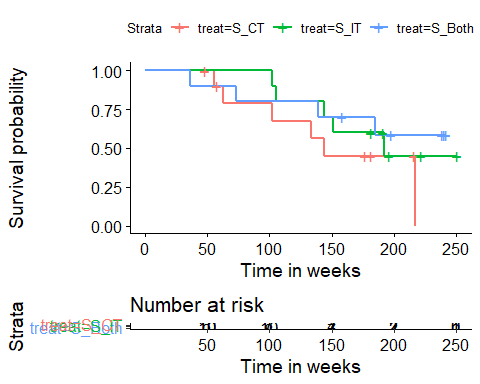
## The Kaplan-Meier estimates

summary(kmfit)

Call: survfit(formula = S ~ treat, data = brca)  
  
 treat=S\_CT   
 time n.risk n.event survival std.err lower 95% CI upper 95% CI  
 55 10 1 0.900 0.0949 0.732 1.000  
 63 8 1 0.787 0.1340 0.564 1.000  
 102 7 1 0.675 0.1551 0.430 1.000  
 133 6 1 0.562 0.1651 0.316 1.000  
 144 5 1 0.450 0.1660 0.218 0.927  
 217 1 1 0.000 NaN NA NA  
  
 treat=S\_IT   
 time n.risk n.event survival std.err lower 95% CI upper 95% CI  
 102 10 1 0.90 0.0949 0.732 1.000  
 105 9 1 0.80 0.1265 0.587 1.000  
 144 8 1 0.70 0.1449 0.467 1.000  
 151 7 1 0.60 0.1549 0.362 0.995  
 192 4 1 0.45 0.1743 0.211 0.961  
  
 treat=S\_Both   
 time n.risk n.event survival std.err lower 95% CI upper 95% CI  
 36 10 1 0.900 0.0949 0.732 1  
 73 9 1 0.800 0.1265 0.587 1  
 139 8 1 0.700 0.1449 0.467 1  
 185 6 1 0.583 0.1610 0.340 1

## K-M Plot via survminer

ggsurvplot(kmfit, data = brca, risk.table = TRUE,  
 risk.table.height = 0.25, xlab = "Time in weeks")



## Testing the difference between curves

survdiff(S ~ treat, dat = brca)

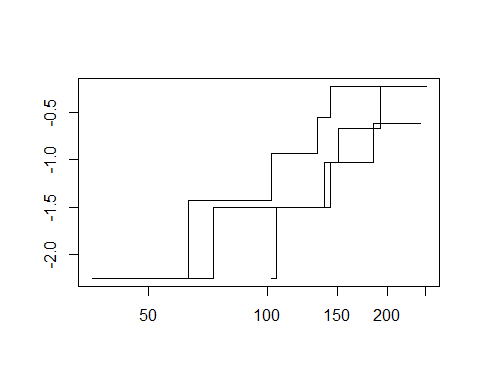
Call:  
survdiff(formula = S ~ treat, data = brca)  
  
 N Observed Expected (O-E)^2/E (O-E)^2/V  
treat=S\_CT 11 6 3.80 1.2772 1.7647  
treat=S\_IT 10 5 5.58 0.0605 0.0981  
treat=S\_Both 10 4 5.62 0.4676 0.7725  
  
 Chisq= 1.9 on 2 degrees of freedom, p= 0.4

What do we conclude?

## Log - Log plot to check PH assumption

Do we satisfy the proportional hazard assumption made by the log rank test? Do these lines for treatments cross?

plot(survfit(S ~ treat, data = brca), fun = "cloglog")



# A Cox Model for Treatment

## Fit Cox Model mod\_T: Treatment alone

mod\_T <- coxph(S ~ treat, data = brca)  
mod\_T

Call:  
coxph(formula = S ~ treat, data = brca)  
  
 coef exp(coef) se(coef) z p  
treatS\_IT -0.5832 0.5581 0.6088 -0.958 0.338  
treatS\_Both -0.8313 0.4355 0.6547 -1.270 0.204  
  
Likelihood ratio test=1.75 on 2 df, p=0.4164  
n= 31, number of events= 15

## Impact of Treatment on Survival

summary(mod\_T)

Call:  
coxph(formula = S ~ treat, data = brca)  
  
 n= 31, number of events= 15   
  
 coef exp(coef) se(coef) z Pr(>|z|)  
treatS\_IT -0.5832 0.5581 0.6088 -0.958 0.338  
treatS\_Both -0.8313 0.4355 0.6547 -1.270 0.204  
  
 exp(coef) exp(-coef) lower .95 upper .95  
treatS\_IT 0.5581 1.792 0.1692 1.840  
treatS\_Both 0.4355 2.296 0.1207 1.571  
  
Concordance= 0.577 (se = 0.083 )  
Likelihood ratio test= 1.75 on 2 df, p=0.4  
Wald test = 1.82 on 2 df, p=0.4  
Score (logrank) test = 1.89 on 2 df, p=0.4

## Interpreting the Summaries

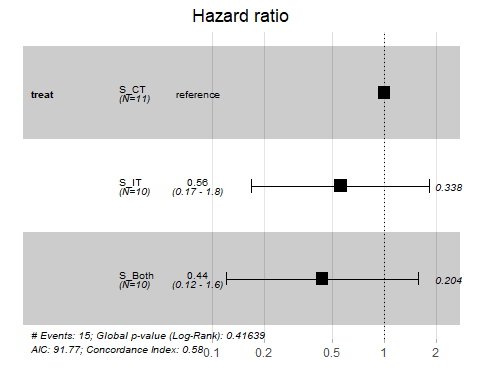
tidy(mod\_T, exponentiate = TRUE, conf.int = TRUE) |>  
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | p.value | conf.low | conf.high |
| --- | --- | --- | --- | --- | --- | --- |
| treatS\_IT | 0.558 | 0.609 | -0.958 | 0.338 | 0.169 | 1.840 |
| treatS\_Both | 0.435 | 0.655 | -1.270 | 0.204 | 0.121 | 1.571 |

* A subject treated with S\_IT is estimated to have 0.558 times the hazard (95% CI: 0.169, 1.840) of a subject treated with S\_CT (the baseline).
* A subject treated with S\_Both is estimated to have 0.435 times the hazard (95% CI 0.121, 1.571) of a subject treated with S\_CT.

## Forest Plot for Treatment model

ggforest(mod\_T, data = brca)



## glance(mod\_T) Fit Summaries

glance(mod\_T) |> select(1:8) |> gt() |>   
 fmt\_number(columns = 3:8, decimals = 3) |>  
 tab\_options(table.font.size = 24)

| n | nevent | statistic.log | p.value.log | statistic.sc | p.value.sc | statistic.wald | p.value.wald |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 31 | 15 | 1.752 | 0.416 | 1.895 | 0.388 | 1.820 | 0.403 |

shows the first eight columns.

Below we show the other eight useful columns for 432.

glance(mod\_T) |> select(11:18) |> gt() |>   
 fmt\_number(columns = 1:7, decimals = 3) |>   
 tab\_options(table.font.size = 24)

| r.squared | r.squared.max | concordance | std.error.concordance | logLik | AIC | BIC | nobs |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0.055 | 0.944 | 0.577 | 0.083 | -43.886 | 91.773 | 93.189 | 31 |

## Summing up glance(mod\_T)

glance(mod\_T) |> select(1:8) |> gt() |>   
 fmt\_number(columns = 3:8, decimals = 3) |>  
 tab\_options(table.font.size = 20)

| n | nevent | statistic.log | p.value.log | statistic.sc | p.value.sc | statistic.wald | p.value.wald |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 31 | 15 | 1.752 | 0.416 | 1.895 | 0.388 | 1.820 | 0.403 |

* n = 31 cases, with nevent = 15 events (so 16 censored)
* log rank test statistic = 1.752, *p* = 0.416
* Score test statistic = 1.895, *p* = 0.388
* Wald test statistic = 1.820, *p* = 0.403
  + Each tests : Treatment adds no value

## Summing up glance(mod\_T)

glance(mod\_T) |> select(11:18) |> gt() |>   
 fmt\_number(columns = 1:7, decimals = 3) |>   
 tab\_options(table.font.size = 20)

| r.squared | r.squared.max | concordance | std.error.concordance | logLik | AIC | BIC | nobs |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0.055 | 0.944 | 0.577 | 0.083 | -43.886 | 91.773 | 93.189 | 31 |

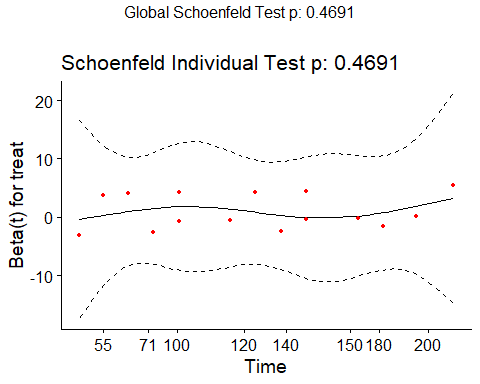
* (Cox-Snell) R-Squared = 0.055, Maximum Pseudo R-Square = 0.944
  + Cox and Snell’s pseudo- reflects the improvement of this model over the model with the intercept alone, with higher values indicating more substantial improvement over an intercept-only model.
  + Not really a percentage of anything: often the maximum value here is less than 1.

## Summing up glance(mod\_T)

* Concordance = 0.577 (standard error = 0.083)
  + Really only appropriate when we have at least one quantitative predictor in the Cox model
  + Assesses probability of agreement between survival time and the risk score generated by the predictors
  + 1 indicates perfect agreement, 0.5 indicates no better than chance
* log Likelihood = -43.886, AIC = 91.773, BIC = 93.189
  + Usual summaries, used to compare models, mostly

## Graphical PH Check for mod\_T

ggcoxzph(cox.zph(mod\_T))



## Next Time

* Fitting more complex Cox models with coxph and cph (from rms) for the brca data
* Reviewing some key ideas from the American Statistical Association’s 2019 commentary on p values and statistical significance: “[Moving to a World Beyond p < 0.05](https://www.tandfonline.com/doi/full/10.1080/00031305.2019.1583913)”

1. Chen and Peace (2011) *Clinical Trial Data Analysis Using R*, Section 5.1 [↑](#footnote-ref-75)