Cancer Survival Analysis

PM

2023-08-21

Contents

[R Markdown 2](#_Toc145599638)

[Important packages 2](#_Toc145599639)

[Kaplan Meire Analysis 2](#_Toc145599640)

[Examine the data 2](#_Toc145599641)

[Fit the model 3](#_Toc145599642)

[Standard survival object 3](#_Toc145599643)

[Three knot spline model to the data 3](#_Toc145599644)

[Log cumulative hazard at time t 3](#_Toc145599645)

[Generate the simulated survival data. 4](#_Toc145599646)

[Fit the Kaplan-Meier curves 4](#_Toc145599647)

[Plotting the curves 4](#_Toc145599648)

[Produce the estimates 5](#_Toc145599649)

[KM Treatment 6](#_Toc145599650)

[Survival by age (data munging) 7](#_Toc145599651)

[Cox proportional hazards model 8](#_Toc145599652)

[Fit the model 9](#_Toc145599653)

[Plot 9](#_Toc145599654)

[Covariates 11](#_Toc145599655)

[Random forest model 12](#_Toc145599656)

[Fit the model 12](#_Toc145599657)

[Plot (formatting a plots) 12](#_Toc145599658)

[Rank variable importance 13](#_Toc145599659)

[Harrell’s c-index 13](#_Toc145599660)

[Comparing survival curves 13](#_Toc145599661)

# R Markdown

The objective of this markdown is to learn survival curves in R. The reference document for project can be found here: <https://rviews.rstudio.com/2017/09/25/survival-analysis-with-r/>.

# Important packages

library(survival)  
library(ranger)  
library(ggplot2)  
library(dplyr)

library(ggfortify)  
library(survsim)

library(simsurv)  
library(flexsurv)

library(survminer)

library(readr)  
library(pec)

library(party)

print("done")

## [1] "done"

# Kaplan Meire Analysis

## Examine the data

data("brcancer")  
brcancer$trt[sample(nrow(brcancer), 686)] <- 1 # add random data and a treatment col  
brcancer$trt[sample(nrow(brcancer), 300)] <- 0  
brcancer$status[sample(nrow(brcancer), 686)] <- 1 # add random data and a status col  
brcancer$status[sample(nrow(brcancer), 300)] <- 0  
head(brcancer)

## id hormon rectime censrec trt status  
## 1 1 0 1814 1 0 0  
## 2 2 1 2018 1 0 1  
## 3 3 1 712 1 1 0  
## 4 4 1 1807 1 1 0  
## 5 5 0 772 1 1 1  
## 6 6 0 448 1 1 1

str(brcancer)

## 'data.frame': 686 obs. of 6 variables:  
## $ id : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ hormon : int 0 1 1 1 0 0 1 0 0 0 ...  
## $ rectime: int 1814 2018 712 1807 772 448 2172 2161 471 2014 ...  
## $ censrec: int 1 1 1 1 1 1 0 0 1 0 ...  
## $ trt : num 0 0 1 1 1 1 1 0 0 0 ...  
## $ status : num 0 1 0 0 1 1 0 0 1 1 ...

sapply(brcancer, function(x) sum(is.na(x))) # missing values

## id hormon rectime censrec trt status   
## 0 0 0 0 0 0

## Fit the model

### Standard survival object

km <- with(brcancer, Surv(rectime, status))  
head(km, 80)

## [1] 1814+ 2018 712+ 1807+ 772 448 2172+ 2161+ 471 2014 577 184   
## [13] 1840 1842+ 1821+ 1371+ 707 1743+ 1781 865 1684 1701 1701 1693   
## [25] 379 1105+ 548 1296+ 1483+ 1570 1469+ 1472 1342+ 1349 1162 1342+  
## [37] 797+ 1232 1230 1205+ 1090 1095+ 449+ 972 825 2438 2233 286   
## [49] 1861+ 1080+ 1521+ 1693 1528+ 169 272 731+ 2059+ 1853+ 1854 1645   
## [61] 544 1666+ 353 1791 1685 191 370+ 173+ 242+ 420 438+ 1624+  
## [73] 1036 359 171 959 1351+ 486+ 525 762

### Three knot spline model to the data

true\_mod <- flexsurv::flexsurvspline(Surv(rectime, censrec) ~ hormon,   
 data = brcancer, k = 3)

### Log cumulative hazard at time t

logcumhaz <- function(t, x, betas, knots) {  
   
 # Obtain the basis terms for the spline-based log  
 # cumulative hazard (evaluated at time t)  
 basis <- flexsurv::basis(knots, log(t))  
   
 # Evaluate the log cumulative hazard under the  
 # Royston and Parmar specification  
 res <-   
 betas[["gamma0"]] \* basis[[1]] +   
 betas[["gamma1"]] \* basis[[2]] +  
 betas[["gamma2"]] \* basis[[3]] +  
 betas[["gamma3"]] \* basis[[4]] +  
 betas[["gamma4"]] \* basis[[5]] +  
 betas[["hormon"]] \* x[["hormon"]]  
   
 res  
}

## Generate the simulated survival data.

covariates <- data.frame(id = 1:686, hormon = rbinom(686, 1, 0.5))  
sim\_data <- simsurv(  
 betas = true\_mod$coefficients, # "true" parameter values  
 x = covariates, # covariate data for 686 individuals  
 knots = true\_mod$knots, # knot locations for splines  
 logcumhazard = logcumhaz, # definition of log cum hazard  
 maxt = NULL, # no right-censoring  
 interval = c(1E-8,100000)) # interval for root finding  
sim\_data <- merge(covariates, sim\_data)  
head(sim\_data)

## id hormon eventtime status  
## 1 1 0 404.2340 1  
## 2 2 0 1729.8946 1  
## 3 3 1 766.1843 1  
## 4 4 0 2013.3581 1  
## 5 5 0 7234.3771 1  
## 6 6 1 239.7716 1

### Fit the Kaplan-Meier curves

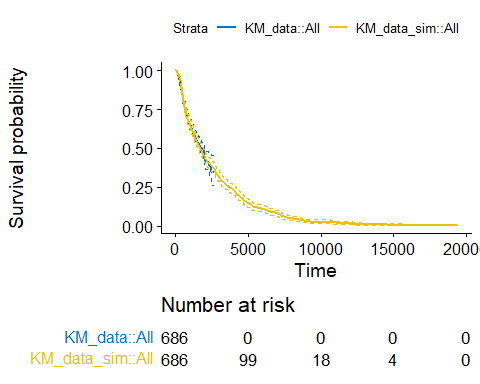
KM\_data <- survminer::surv\_fit(Surv(rectime, censrec) ~ 1, data = brcancer)  
KM\_data\_sim <- survminer::surv\_fit(Surv(eventtime, status) ~ 1, data = sim\_data)

## Plotting the curves

p <- ggsurvplot\_combine(list(KM\_data, KM\_data\_sim),  
 risk.table = TRUE,  
 conf.int = TRUE,  
 censor = FALSE,  
 conf.int.style = "step",  
 tables.theme = theme\_cleantable(),  
 palette = "jco")

## Warning: There was 1 warning in `mutate()`.  
## ℹ In argument: `survtable = purrr::map2(...)`.  
## Caused by warning:  
## ! `select\_()` was deprecated in dplyr 0.7.0.  
## ℹ Please use `select()` instead.  
## ℹ The deprecated feature was likely used in the survminer package.  
## Please report the issue at <https://github.com/kassambara/survminer/issues>.

plot.new()   
print(p,newpage = FALSE)



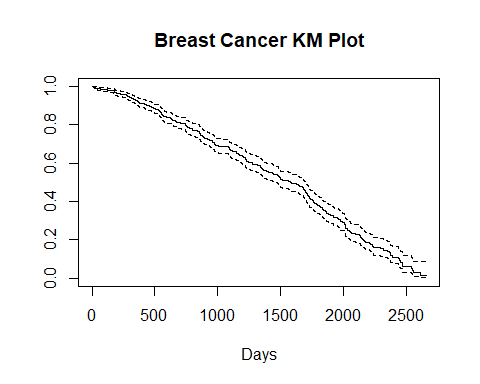
*This shows that the simulated data does appear to plausibly resemble the original data.*

## Produce the estimates

km\_fit <- survfit(formula = Surv(rectime, status) ~ 1, data = brcancer) #probability of survival over time  
summary(km\_fit, times = c(1,30,60,90\*(1:10))) # print the estimates for 1, 30, 60 and 90 days, and then every 90 days thereafter

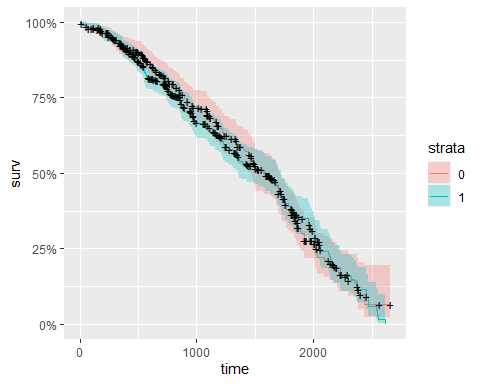
## Call: survfit(formula = Surv(rectime, status) ~ 1, data = brcancer)  
##   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 1 686 0 1.000 0.00000 1.000 1.000  
## 30 679 6 0.991 0.00356 0.984 0.998  
## 60 676 2 0.988 0.00410 0.980 0.996  
## 90 671 4 0.982 0.00501 0.973 0.992  
## 180 657 8 0.971 0.00646 0.958 0.983  
## 270 638 10 0.956 0.00789 0.940 0.971  
## 360 603 22 0.923 0.01033 0.902 0.943  
## 450 573 15 0.899 0.01168 0.877 0.922  
## 540 538 18 0.870 0.01313 0.845 0.897  
## 630 490 26 0.828 0.01494 0.799 0.857  
## 720 465 12 0.807 0.01571 0.777 0.838  
## 810 417 20 0.771 0.01695 0.738 0.805  
## 900 383 21 0.731 0.01815 0.697 0.768

plot(km\_fit, xlab = "Days", main = 'Breast Cancer KM Plot')



## KM Treatment

km\_trt\_fit <- survfit(Surv(rectime, status) ~ trt, data= brcancer)  
autoplot(km\_trt\_fit)



*Since I randomized the treatment data for this population the results are that both treatments have similar effectiveness. However, treatment 0 was more effective.*

## Survival by age (data munging)

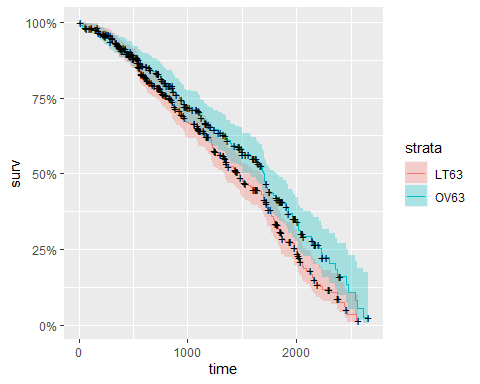
# create mock ages and add random data and a age col  
brcancer$age[sample(nrow(brcancer), 686)] <- 33:85

## Warning in brcancer$age[sample(nrow(brcancer), 686)] <- 33:85: number of items  
## to replace is not a multiple of replacement length

head(brcancer)

## id hormon rectime censrec trt status age  
## 1 1 0 1814 1 0 0 74  
## 2 2 1 2018 1 0 1 60  
## 3 3 1 712 1 1 0 70  
## 4 4 1 1807 1 1 0 53  
## 5 5 0 772 1 1 1 73  
## 6 6 0 448 1 1 1 63

brcancer\_age <- mutate(brcancer, AG = ifelse((age < 63), "LT63", "OV63"),  
 AG = factor(AG),  
 trt = factor(trt,labels=c("standard","test")),  
 censrec = factor(censrec,labels=c("N0","Yes")))  
  
km\_AG\_fit <- survfit(Surv(rectime, status) ~ AG, data = brcancer\_age )  
autoplot(km\_AG\_fit)



*As expected, the survival curve has a similar result as the treatment curve because of the randomized mock data. Patients over 63yo had a higher survival rate.*

# Cox proportional hazards model

# modify the df to include treatment, age, and status data  
brcancer$age[sample(nrow(brcancer), 686)] <- 33:85

## Warning in brcancer$age[sample(nrow(brcancer), 686)] <- 33:85: number of items  
## to replace is not a multiple of replacement length

brcancer$trt[sample(nrow(brcancer), 686)] <- 1 # add random data and a treatment col  
brcancer$trt[sample(nrow(brcancer), 300)] <- 0  
brcancer$status[sample(nrow(brcancer), 686)] <- 1 # add random data and a status col  
brcancer$status[sample(nrow(brcancer), 300)] <- 0  
knitr::opts\_chunk$set(echo = TRUE)  
head(brcancer\_age)

## id hormon rectime censrec trt status age AG  
## 1 1 0 1814 Yes standard 0 74 OV63  
## 2 2 1 2018 Yes standard 1 60 LT63  
## 3 3 1 712 Yes test 0 70 OV63  
## 4 4 1 1807 Yes test 0 53 LT63  
## 5 5 0 772 Yes test 1 73 OV63  
## 6 6 0 448 Yes test 1 63 OV63

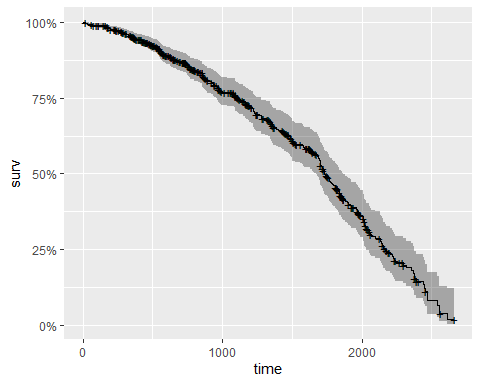
## Fit the model

cox <- coxph(Surv(rectime, status) ~ trt + hormon + censrec + age , data = brcancer\_age )  
summary(cox)

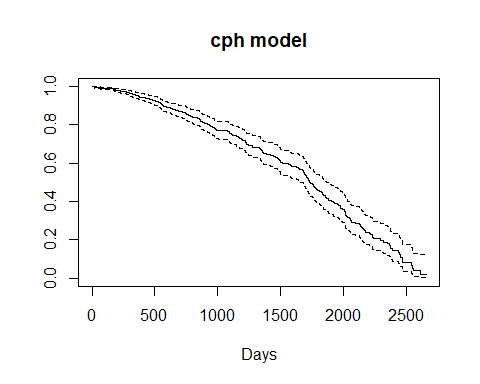
## Call:  
## coxph(formula = Surv(rectime, status) ~ trt + hormon + censrec +   
## age, data = brcancer\_age)  
##   
## n= 686, number of events= 386   
##   
## coef exp(coef) se(coef) z Pr(>|z|)   
## trttest 0.025653 1.025985 0.103170 0.249 0.8036   
## hormon -0.213075 0.808096 0.107023 -1.991 0.0465 \*   
## censrecYes 0.941968 2.565023 0.108646 8.670 <2e-16 \*\*\*  
## age -0.007308 0.992719 0.003381 -2.162 0.0306 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## exp(coef) exp(-coef) lower .95 upper .95  
## trttest 1.0260 0.9747 0.8382 1.2559  
## hormon 0.8081 1.2375 0.6552 0.9967  
## censrecYes 2.5650 0.3899 2.0731 3.1737  
## age 0.9927 1.0073 0.9862 0.9993  
##   
## Concordance= 0.646 (se = 0.016 )  
## Likelihood ratio test= 83.29 on 4 df, p=<2e-16  
## Wald test = 87.95 on 4 df, p=<2e-16  
## Score (logrank) test = 93 on 4 df, p=<2e-16

## Plot

cox\_fit <- survfit(cox)   
autoplot(cox\_fit)



plot(cox\_fit, main = "cph model", xlab="Days")

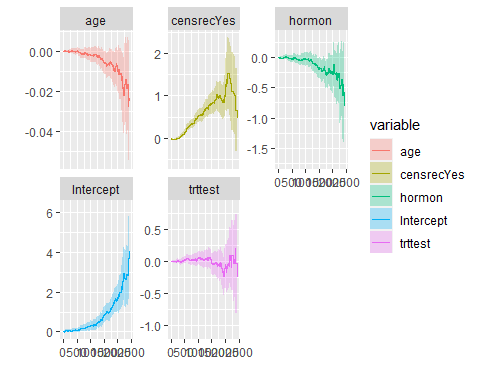


## Covariates

aa\_fit <- aareg(Surv(rectime, status) ~ trt + hormon + censrec + age , data = brcancer\_age )  
aa\_fit

## Call:  
## aareg(formula = Surv(rectime, status) ~ trt + hormon + censrec +   
## age, data = brcancer\_age)  
##   
## n= 686   
## 344 out of 352 unique event times used  
##   
## slope coef se(coef) z p  
## Intercept 1.09e-03 3.51e-03 6.54e-04 5.3700 7.87e-08  
## trttest 3.08e-05 -8.32e-06 2.93e-04 -0.0284 9.77e-01  
## hormon -1.84e-04 -6.04e-04 3.00e-04 -2.0100 4.43e-02  
## censrecYes 1.54e-03 2.76e-03 3.66e-04 7.5400 4.54e-14  
## age -6.46e-06 -2.15e-05 9.51e-06 -2.2600 2.36e-02  
##   
## Chisq=65.67 on 4 df, p=1.86e-13; test weights=aalen

autoplot(aa\_fit)



*The plots show how the effects of the covariates change over time.*

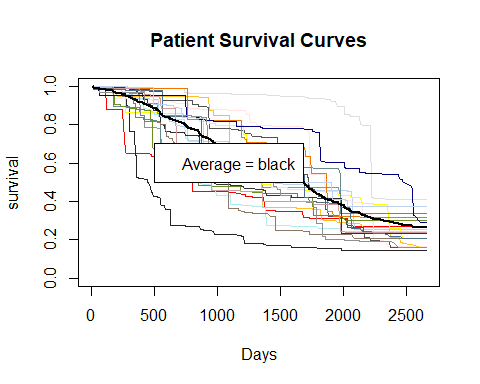
# Random forest model

## Fit the model

r\_fit <- ranger(Surv(rectime, status) ~ trt + hormon + censrec + age,  
 data = brcancer\_age,  
 mtry = 4,  
 importance = "permutation",  
 splitrule = "extratrees",  
 verbose = TRUE)  
death\_times <- r\_fit$unique.death.times   
surv\_prob <- data.frame(r\_fit$survival)  
avg\_prob <- sapply(surv\_prob,mean)

## Plot (formatting a plots)

plot(r\_fit$unique.death.times,r\_fit$survival[1,],   
 type = "l",   
 ylim = c(0,1),  
 col = "red",  
 xlab = "Days",  
 ylab = "survival",  
 main = "Patient Survival Curves")  
  
# details  
cols <- colors()  
for (n in sample(c(2:dim(brcancer\_age)[1]), 20)){  
 lines(r\_fit$unique.death.times, r\_fit$survival[n,], type = "l", col = cols[n])  
}  
lines(death\_times, avg\_prob, lwd = 2)  
legend(500, 0.7, legend = c('Average = black'))



## Rank variable importance

vi <- data.frame(sort(round(r\_fit$variable.importance, 4), decreasing = TRUE))  
names(vi) <- "importance"  
head(vi)

## importance  
## censrec 0.0686  
## age 0.0174  
## hormon 0.0051  
## trt 0.0028

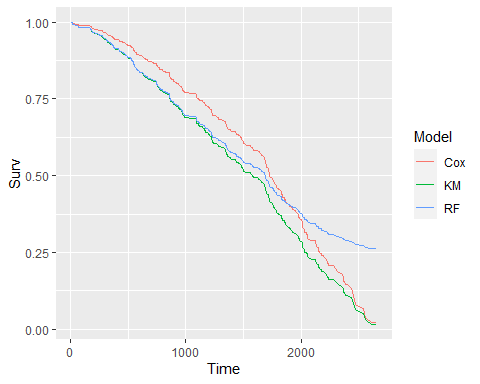
## Harrell’s c-index

cat("Prediction Error = 1 - Harrell's c-index = ", r\_fit$prediction.error)

## Prediction Error = 1 - Harrell's c-index = 0.3784513

# Comparing survival curves

kmi <- rep("KM",length(km\_fit$time))  
km\_df <- data.frame(km\_fit$time,km\_fit$surv,kmi)  
names(km\_df) <- c("Time","Surv","Model")  
  
coxi <- rep("Cox",length(cox\_fit$time))  
cox\_df <- data.frame(cox\_fit$time,cox\_fit$surv,coxi)  
names(cox\_df) <- c("Time","Surv","Model")  
  
rfi <- rep("RF",length(r\_fit$unique.death.times))  
rf\_df <- data.frame(r\_fit$unique.death.times,avg\_prob,rfi)  
names(rf\_df) <- c("Time","Surv","Model")  
  
plot\_df <- rbind(km\_df,cox\_df,rf\_df)  
  
p <- ggplot(plot\_df, aes(x = Time, y = Surv, color = Model))  
p + geom\_line()



*It is probable that the Cox model takes into account the time varying coefficients. I suspect that there are neither enough observations nor enough explanatory variables for the ranger() model to do better.*