Cox regression

A manually worked out, simple example: two groups

Load libraries

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------------------------------- tidy

## v ggplot2 3.1.0     v purrr   0.3.0
## v tibble  2.0.1     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.3.1     v forcats 0.3.0

## -- Conflicts ------------------------------------------------------------------------------- tidyverse_
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(maxLik)
```

```
## Loading required package: miscTools

##
## Please cite the 'maxLik' package as:
## Henningsen, Arne and Toomet, Ott (2011). maxLik: A package for maximum likelihood estimation in R. C
##
## If you have questions, suggestions, or comments regarding the 'maxLik' package, please use a forum o
## https://r-forge.r-project.org/projects/maxlik/
```

```r
library(survival)
```

```
##
## Attaching package: 'survival'

## The following object is masked from 'package:rpart':
##
##     solder
```

Data definition

Lets enter the data in R:

```r
dat <- data.frame(ratID = paste0("rat", 1:5),
                  time = c(55, 50, 70, 120, 110),
                  failure = c(0, 1, 1, 0, 1),
                  group = c(0, 1, 0, 1, 1))
```

Total number of failures D:

```r
sum(dat$failure)
```

```
## [1] 3
```

For convenience, rename 'group' to 'x':

```r
dat <- rename(dat, x = group)
dat
```

```
##   ratID time failure x
## 1  rat1   55       0 0
## 2  rat2   50       1 1
## 3  rat3   70       1 0
## 4  rat4  120       0 1
## 5  rat5  110       1 1
```

We also define an auxiliary data.frame containing events only:

```r
dat.events <- subset(dat, failure == 1)
dat.events
```

```
##   ratID time failure x
## 2  rat2   50       1 1
## 3  rat3   70       1 0
## 5  rat5  110       1 1
```

Partial log-likelihood function

Lets define the partial (log-)likelihood function

```r
pLogLik <- function(beta) {
  numerator <- with(dat.events, x * beta)
  denominator <- rep(NA_real_, length(numerator))
  for(j in seq_along(denominator))
  {
    t_j <- dat.events[j, "time"]
    risk_set <- subset(dat, time >= t_j)
    theta_j <- with(risk_set, exp(x * beta))
    denominator[j] <- log(sum(theta_j))
  }
  return(sum(numerator - denominator))
}
```

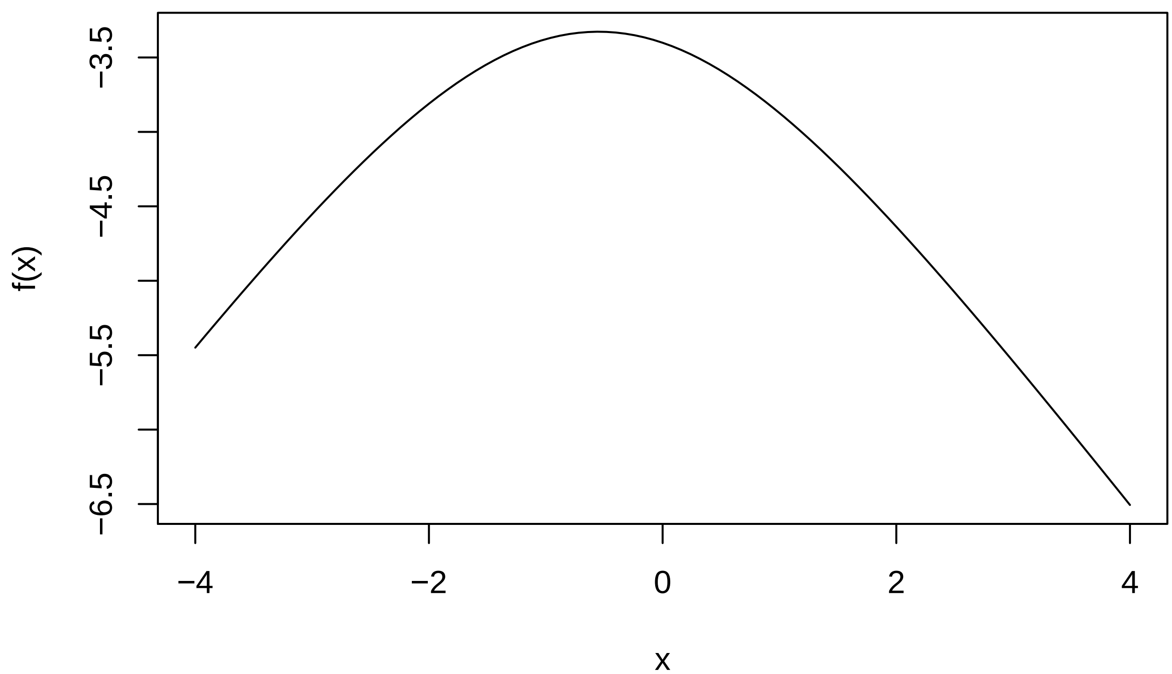
```r
debugonce(pLogLik)
```

We can plot it:

```r
f <- Vectorize(pLogLik)
curve(f, from = -4, to = 4)
```

```
## debugging in: (function (beta)
## {
##     numerator <- with(dat.events, x * beta)
##     denominator <- rep(NA_real_, length(numerator))
##     for (j in seq_along(denominator)) {
##         t_j <- dat.events[j, "time"]
##         risk_set <- subset(dat, time >= t_j)
##         theta_j <- with(risk_set, exp(x * beta))
##         denominator[j] <- log(sum(theta_j))
##     }
##     return(sum(numerator - denominator))
## })(beta = dots[[1L]][[1L]])
## debug at <text>#1: {
##     numerator <- with(dat.events, x * beta)
##     denominator <- rep(NA_real_, length(numerator))
##     for (j in seq_along(denominator)) {
##         t_j <- dat.events[j, "time"]
##         risk_set <- subset(dat, time >= t_j)
##         theta_j <- with(risk_set, exp(x * beta))
##         denominator[j] <- log(sum(theta_j))
##     }
##     return(sum(numerator - denominator))
## }
## debug at <text>#2: numerator <- with(dat.events, x * beta)
## debug at <text>#3: denominator <- rep(NA_real_, length(numerator))
## debug at <text>#4: for (j in seq_along(denominator)) {
##     t_j <- dat.events[j, "time"]
##     risk_set <- subset(dat, time >= t_j)
##     theta_j <- with(risk_set, exp(x * beta))
##     denominator[j] <- log(sum(theta_j))
## }
## debug at <text>#6: t_j <- dat.events[j, "time"]
## debug at <text>#7: risk_set <- subset(dat, time >= t_j)
## debug at <text>#8: theta_j <- with(risk_set, exp(x * beta))
## debug at <text>#9: denominator[j] <- log(sum(theta_j))
## debug at <text>#6: t_j <- dat.events[j, "time"]
## debug at <text>#7: risk_set <- subset(dat, time >= t_j)
## debug at <text>#8: theta_j <- with(risk_set, exp(x * beta))
## debug at <text>#9: denominator[j] <- log(sum(theta_j))
## debug at <text>#6: t_j <- dat.events[j, "time"]
## debug at <text>#7: risk_set <- subset(dat, time >= t_j)
## debug at <text>#8: theta_j <- with(risk_set, exp(x * beta))
## debug at <text>#9: denominator[j] <- log(sum(theta_j))
## debug at <text>#11: return(sum(numerator - denominator))
## exiting from: (function (beta)
## {
##     numerator <- with(dat.events, x * beta)
##     denominator <- rep(NA_real_, length(numerator))
##     for (j in seq_along(denominator)) {
##         t_j <- dat.events[j, "time"]
```

```
##         risk_set <- subset(dat, time >= t_j)
##         theta_j <- with(risk_set, exp(x * beta))
##         denominator[j] <- log(sum(theta_j))
##     }
##     return(sum(numerator - denominator))
## })(beta = dots[[1L]][[1L]])
```



Maximum partial-Likelihood estimation

```r
fit.ML <- maxLik(pLogLik, start = c(beta = 0))
summary(fit.ML)
```

```
## --------------------------------------------
## Maximum Likelihood estimation
## Newton-Raphson maximisation, 2 iterations
## Return code 1: gradient close to zero
## Log-Likelihood: -3.327063
## 1  free parameters
## Estimates:
##      Estimate Std. error t value Pr(> t)
## beta  -0.5493     1.4179  -0.387   0.698
## --------------------------------------------
```

Beta is not significate different from 0 and 1

With the `coxph` function:

```
fit.cph <- coxph(Surv(time, failure) ~ x, data = dat)
summary(fit.cph)
```

```
## Call:
## coxph(formula = Surv(time, failure) ~ x, data = dat)
##
##   n= 5, number of events= 3
##
##      coef exp(coef) se(coef)      z Pr(>|z|)
## x -0.5493    0.5774   1.4179 -0.387    0.698
##
##   exp(coef) exp(-coef) lower .95 upper .95
## x    0.5774      1.732   0.03585     9.297
##
## Concordance= 0.5  (se = 0.202 )
## Rsquare= 0.029   (max possible= 0.743 )
## Likelihood ratio test= 0.15  on 1 df,   p=0.7
## Wald test            = 0.15  on 1 df,   p=0.7
## Score (logrank) test = 0.15  on 1 df,   p=0.7
```

Lower risk means longer time Beta = -0.549 +- 1.418 * 1.96 exp(Beta) = 0.577

```
confint(fit.cph)
```

```
##       2.5 %   97.5 %
## x -3.328286 2.229673
```

We can reproduce the Likelihood-ratio test:

```
LRT <- 2 * (fit.ML$maximum - pLogLik(0))
data.frame(LRT = LRT,
           pvalue = pchisq(LRT, df = 1, lower.tail = FALSE))
```

```
##         LRT    pvalue
## 1 0.1482688 0.7001953
```

The Wald test is already in the `maxLik` summary output.

A manually worked out, simple example: one continuous covariate

```
dat <- data.frame(time = c(6, 7, 10, 15, 19, 25),
                  event = c(1, 0, 1, 1, 0, 1),
                  age = c(67, 62, 34, 41, 46, 28))
```

```r
fit <- coxph(Surv(time, event) ~ age, data = dat)
summary(fit)
```

```
## Call:
## coxph(formula = Surv(time, event) ~ age, data = dat)
##
##   n= 6, number of events= 4
##
##        coef exp(coef) se(coef)    z Pr(>|z|)
## age 0.07606   1.07903  0.07316 1.04    0.298
##
##     exp(coef) exp(-coef) lower .95 upper .95
## age     1.079     0.9268    0.9349     1.245
##
## Concordance= 0.7  (se = 0.237 )
## Rsquare= 0.209   (max possible= 0.76 )
## Likelihood ratio test= 1.41  on 1 df,   p=0.2
## Wald test            = 1.08  on 1 df,   p=0.3
## Score (logrank) test = 1.33  on 1 df,   p=0.2
```

We might express age in decades:

```r
dat <- mutate(dat, age_dec = age / 10)
summary(coxph(Surv(time, event) ~ age_dec, data = dat))
```

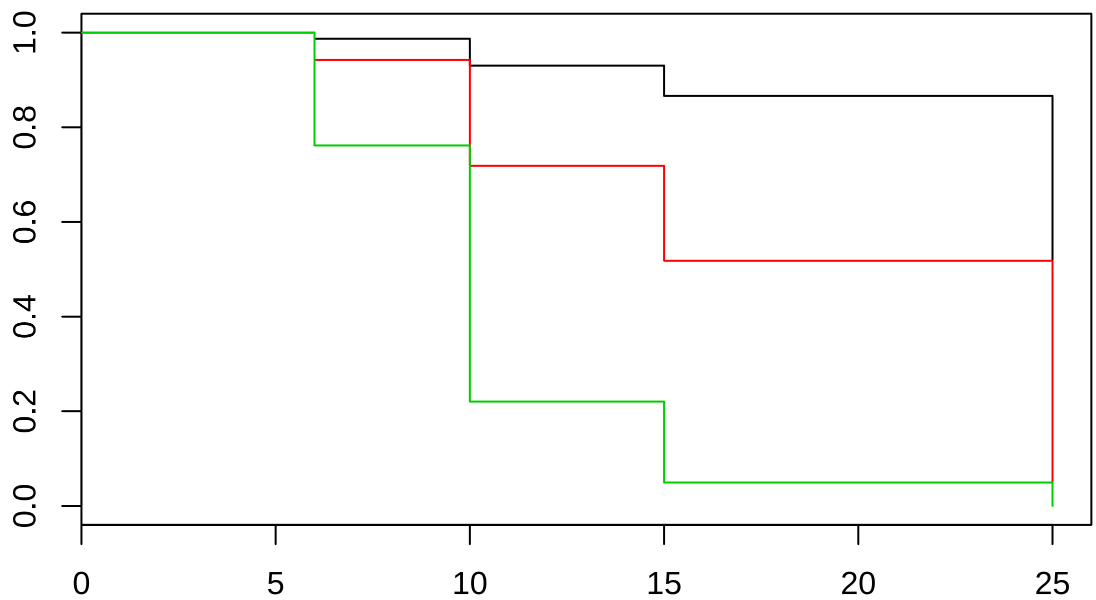
```
## Call:
## coxph(formula = Surv(time, event) ~ age_dec, data = dat)
##
##   n= 6, number of events= 4
##
##           coef exp(coef) se(coef)    z Pr(>|z|)
## age_dec 0.7606    2.1397   0.7316 1.04    0.298
##
##         exp(coef) exp(-coef) lower .95 upper .95
## age_dec      2.14     0.4674      0.51     8.976
##
## Concordance= 0.7  (se = 0.237 )
## Rsquare= 0.209   (max possible= 0.76 )
## Likelihood ratio test= 1.41  on 1 df,   p=0.2
## Wald test            = 1.08  on 1 df,   p=0.3
## Score (logrank) test = 1.33  on 1 df,   p=0.2
```

Try to do some predictions

```r
pred <- survfit(fit, newdata = data.frame(age = c(20, 40, 60)))
pred
```

```
## Call: survfit(formula = fit, newdata = data.frame(age = c(20, 40, 60)))
##
##   n events median 0.95LCL 0.95UCL
## 1 6      4     NA      25      NA
## 2 6      4     25      10      NA
## 3 6      4     10       6      NA
```

```r
plot(pred, col = 1:3)
```



Case study: the pharmacoSmoking dataset, DAP (Data Analysis Plan)

1. MOdel for TTR given: TRT, Age, EMPL, Gender, Race

- Table of estimates
- With commentary

2. Risk stratification/ Data segmentation:

- Based on the model indentify subject. At high risk Top 10

3. Predictions:

- Median TTR
- Survival ("6 months"/ covariates)

4. Compare TRT effficacy in FT vs NON-FT employment

Load the data

```
library(asaur)
d_raw <- pharmacoSmoking
head(d_raw)
```

```
##    id ttr relapse         grp age gender     race employment yearsSmoking
## 1  21 182       0   patchOnly  36   Male    white         ft           26
## 2 113  14       1   patchOnly  41   Male    white      other           27
## 3  39   5       1 combination  25 Female    white      other           12
## 4  80  16       1 combination  54   Male    white         ft           39
## 5  87   0       1 combination  45   Male    white      other           30
## 6  29 182       0 combination  43   Male hispanic         ft           30
##   levelSmoking ageGroup2 ageGroup4 priorAttempts longestNoSmoke
## 1        heavy     21-49     35-49             0              0
## 2        heavy     21-49     35-49             3             90
## 3        heavy     21-49     21-34             3             21
## 4        heavy       50+     50-64             0              0
## 5        heavy     21-49     35-49             0              0
## 6        heavy     21-49     35-49             2           1825
```

```
table(d_raw$employment, useNA = "always")
```

```
##
##    ft other    pt  <NA>
##    72    39    14     0
```

```
dat <- mutate(d_raw,
              employment = ifelse(employment == "ft", "ft", "other"),
              grp = relevel(grp, ref = "patchOnly"),
              race = ifelse(race == "white", "white",
                     ifelse(race == "black", "black", "other")))
table(dat$employment)
```

```
##
##    ft other
##    72    53
```

```
table(dat$race)
```

```
##
## black other white
##    38    10    77
```

Fit the Cox model

```
fit <- coxph(Surv(ttr, relapse) ~ grp + age + gender + employment + race, data = dat)
summary(fit)
```

```
## Call:
## coxph(formula = Surv(ttr, relapse) ~ grp + age + gender + employment +
##     race, data = dat)
##
##   n= 125, number of events= 89
##
##                      coef exp(coef) se(coef)      z Pr(>|z|)
## grpcombination  -0.60461   0.54629  0.21830 -2.770 0.005613 **
## age             -0.03575   0.96488  0.01069 -3.345 0.000821 ***
## genderMale      -0.04312   0.95780  0.23711 -0.182 0.855691
## employmentother  0.73991   2.09574  0.24132  3.066 0.002168 **
## raceother       -0.68519   0.50400  0.45634 -1.501 0.133232
## racewhite       -0.25824   0.77241  0.23268 -1.110 0.267065
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                 exp(coef) exp(-coef) lower .95 upper .95
## grpcombination     0.5463     1.8305    0.3561    0.8380
## age                0.9649     1.0364    0.9449    0.9853
## genderMale         0.9578     1.0441    0.6018    1.5244
## employmentother    2.0957     0.4772    1.3060    3.3631
## raceother          0.5040     1.9841    0.2061    1.2327
## racewhite          0.7724     1.2947    0.4895    1.2187
##
## Concordance= 0.649  (se = 0.03 )
## Rsquare= 0.18   (max possible= 0.998 )
## Likelihood ratio test= 24.8  on 6 df,   p=4e-04
## Wald test            = 24.37  on 6 df,   p=4e-04
## Score (logrank) test = 25.07  on 6 df,   p=3e-04
```

We can change the contrasts as we see fit:

```
dat <- mutate(dat, grp = relevel(grp, ref = "patchOnly"))
fit <- update(fit)
summary(fit)
```

```
## Call:
## coxph(formula = Surv(ttr, relapse) ~ grp + age + gender + employment +
##     race, data = dat)
##
##   n= 125, number of events= 89
##
##                      coef exp(coef) se(coef)      z Pr(>|z|)
## grpcombination  -0.60461   0.54629  0.21830 -2.770 0.005613 **
## age             -0.03575   0.96488  0.01069 -3.345 0.000821 ***
## genderMale      -0.04312   0.95780  0.23711 -0.182 0.855691
## employmentother  0.73991   2.09574  0.24132  3.066 0.002168 **
## raceother       -0.68519   0.50400  0.45634 -1.501 0.133232
## racewhite       -0.25824   0.77241  0.23268 -1.110 0.267065
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                 exp(coef) exp(-coef) lower .95 upper .95
```

```
## grpcombination      0.5463     1.8305    0.3561    0.8380
## age                 0.9649     1.0364    0.9449    0.9853
## genderMale          0.9578     1.0441    0.6018    1.5244
## employmentother     2.0957     0.4772    1.3060    3.3631
## raceother           0.5040     1.9841    0.2061    1.2327
## racewhite           0.7724     1.2947    0.4895    1.2187
##
## Concordance= 0.649  (se = 0.03 )
## Rsquare= 0.18   (max possible= 0.998 )
## Likelihood ratio test= 24.8  on 6 df,   p=4e-04
## Wald test            = 24.37  on 6 df,   p=4e-04
## Score (logrank) test = 25.07  on 6 df,   p=3e-04
```

Export the result table

```r
broom::tidy(fit) %>%
  write_csv("coefficients_table.csv")
```

Data segmentation

Remove outcomes

```r
d_new <- select(dat, -ttr, -relapse)
```

```r
d_segmented <- d_new %>% mutate(risk_score = predict(fit, newdata = d_new, type = "lp"))
head(d_segmented)
```

```
##    id         grp age gender  race employment yearsSmoking levelSmoking
## 1  21   patchOnly  36   Male white         ft           26        heavy
## 2 113   patchOnly  41   Male white      other           27        heavy
## 3  39 combination  25 Female white      other           12        heavy
## 4  80 combination  54   Male white         ft           39        heavy
## 5  87 combination  45   Male white      other           30        heavy
## 6  29 combination  43   Male other         ft           30        heavy
##   ageGroup2 ageGroup4 priorAttempts longestNoSmoke risk_score
## 1     21-49     35-49             0              0  0.3681131
## 2     21-49     35-49             3             90  0.9292519
## 3     21-49     21-34             3             21  0.9398153
## 4       50+     50-64             0              0 -0.8800627
## 5     21-49     35-49             0              0  0.1816247
## 6     21-49     35-49             2           1825 -0.9137190
```

```r
d_segmented %>% arrange(desc(risk_score)) %>% head(10)
```

```
##     id         grp age gender  race employment yearsSmoking levelSmoking
## 1   91   patchOnly  22 Female white      other           10        heavy
## 2   95   patchOnly  22 Female white      other            9        heavy
## 3   94   patchOnly  43 Female black      other           24        light
## 4   37   patchOnly  44 Female black      other           31        heavy
## 5   25   patchOnly  37 Female white      other           23        light
## 6   88   patchOnly  41 Female white      other           26        heavy
```

```
## 7   63   patchOnly  40   Male white      other           22         heavy
## 8   62   patchOnly  49 Female black      other           35         light
## 9   39 combination  25 Female white      other           12         heavy
## 10 113   patchOnly  41   Male white      other           27         heavy
##    ageGroup2 ageGroup4 priorAttempts longestNoSmoke risk_score
## 1      21-49     21-34             3              2  1.6516891
## 2      21-49     21-34             2              3  1.6516891
## 3      21-49     35-49             2              3  1.1591080
## 4      21-49     35-49             2              2  1.1233545
## 5      21-49     35-49             5           1095  1.1153871
## 6      21-49     35-49             1              7  0.9723732
## 7      21-49     35-49             2              2  0.9650054
## 8      21-49     35-49             4            540  0.9445872
## 9      21-49     21-34             3             21  0.9398153
## 10     21-49     35-49             3             90  0.9292519
```

Predicting Median TTR and Survival (6 months | covariates)

```r
predict_survival <- survfit(fit, new_data = d_new[1, ])
summary(predict_survival)
```

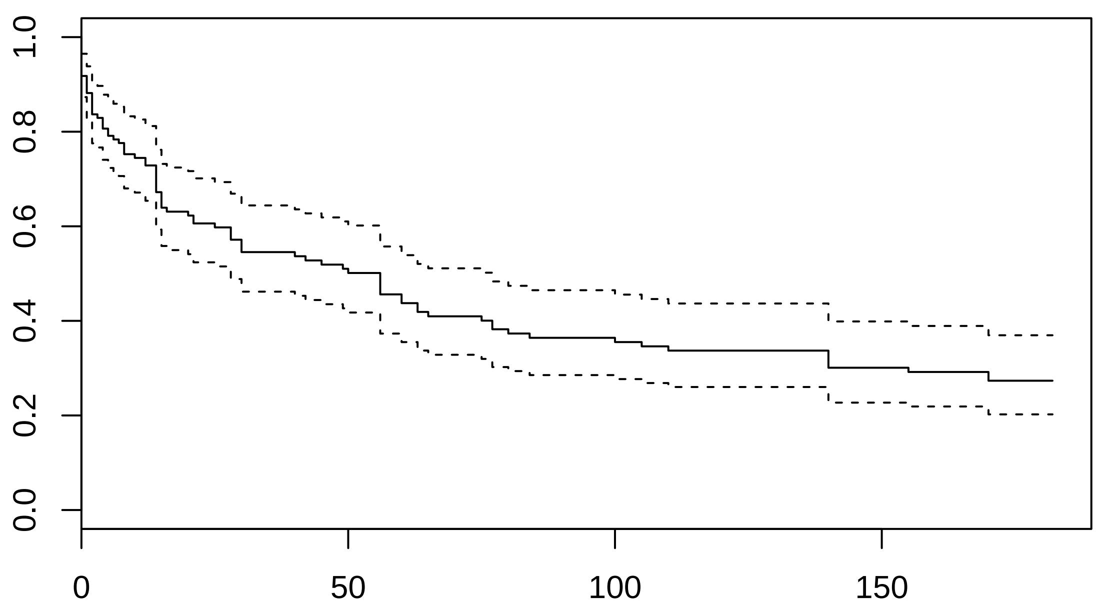
```
## Call: survfit(formula = fit, new_data = d_new[1, ])
##
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     0    125      12    0.918  0.0234        0.873        0.965
##     1    113       5    0.882  0.0279        0.829        0.938
##     2    108       6    0.837  0.0325        0.775        0.903
##     3    102       1    0.829  0.0332        0.767        0.897
##     4    101       3    0.807  0.0351        0.741        0.878
##     5     98       2    0.791  0.0362        0.723        0.866
##     6     96       1    0.784  0.0368        0.715        0.859
##     7     95       1    0.776  0.0373        0.706        0.853
##     8     94       3    0.752  0.0388        0.680        0.833
##    10     91       1    0.745  0.0393        0.671        0.826
##    12     90       2    0.729  0.0402        0.654        0.812
##    14     88       7    0.672  0.0429        0.593        0.762
##    15     81       4    0.639  0.0441        0.558        0.732
##    16     77       1    0.631  0.0444        0.550        0.724
##    20     76       1    0.623  0.0447        0.541        0.717
##    21     75       2    0.606  0.0451        0.524        0.701
##    25     73       1    0.598  0.0453        0.515        0.693
##    28     72       3    0.572  0.0459        0.488        0.669
##    30     69       3    0.545  0.0463        0.462        0.644
##    40     66       1    0.537  0.0464        0.453        0.636
##    42     65       1    0.528  0.0465        0.444        0.627
##    45     64       1    0.519  0.0466        0.435        0.619
##    49     63       1    0.510  0.0466        0.426        0.610
##    50     62       1    0.501  0.0467        0.418        0.602
##    56     61       5    0.456  0.0467        0.373        0.557
##    60     56       2    0.437  0.0466        0.355        0.539
##    63     54       2    0.419  0.0464        0.337        0.520
##    65     52       1    0.410  0.0463        0.328        0.511
```

```
##    75      51       1     0.400  0.0461        0.320        0.502
##    77      50       2     0.382  0.0458        0.302        0.483
##    80      48       1     0.373  0.0456        0.294        0.474
##    84      47       1     0.364  0.0453        0.285        0.465
##   100      46       1     0.355  0.0451        0.277        0.455
##   105      45       1     0.346  0.0448        0.269        0.446
##   110      44       1     0.337  0.0446        0.260        0.437
##   140      43       4     0.301  0.0432        0.227        0.399
##   155      39       1     0.292  0.0429        0.219        0.389
##   170      38       2     0.273  0.0420        0.202        0.370
```

```r
summary(predict_survival, times = 180)
```

```
## Call: survfit(formula = fit, new_data = d_new[1, ])
##
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##   180     36      89    0.273   0.042        0.202         0.37
```

```r
plot(predict_survival)
```




```r
predict_survival
```

```
## Call: survfit(formula = fit, new_data = d_new[1, ])
##
```

```
##       n  events  median 0.95LCL 0.95UCL
##     125      89      56      28      77
```

```
individual_prediction <- function(grp, age, employment, race) {
  tibble(median_ttr = ..., surv_6_months = ...)
}
```

TRT.Efficacy in FT vs other

I Part

	Beta	SE	CI	P
FT	Beta FT			
OTH	Beta OTH			

```
library(tidyverse)
d_ft <- filter(dat, employment == "ft")
d_nft <- filter(dat, employment != "ft")
head(d_ft)
```

```
##   id ttr relapse         grp age gender  race employment yearsSmoking
## 1 21 182       0   patchOnly  36   Male white         ft           26
## 2 80  16       1 combination  54   Male white         ft           39
## 3 29 182       0 combination  43   Male other         ft           30
## 4 54   2       1   patchOnly  40 Female black         ft           25
## 5 70   0       1   patchOnly  38   Male black         ft           23
## 6 85 182       0 combination  51   Male black         ft           35
##   levelSmoking ageGroup2 ageGroup4 priorAttempts longestNoSmoke
## 1        heavy     21-49     35-49             0              0
## 2        heavy       50+     50-64             0              0
## 3        heavy     21-49     35-49             2           1825
## 4        heavy     21-49     35-49             4              7
## 5        light     21-49     35-49            10             90
## 6        heavy       50+     50-64             1              7
```

```
head(d_nft)
```

```
##    id ttr relapse         grp age gender  race employment yearsSmoking
## 1 113  14       1   patchOnly  41   Male white      other           27
## 2  39   5       1 combination  25 Female white      other           12
## 3  87   0       1 combination  45   Male white      other           30
## 4  16  14       1   patchOnly  66   Male black      other           54
## 5  35  77       1   patchOnly  78 Female black      other           56
## 6  84  12       1   patchOnly  64 Female black      other           30
##   levelSmoking ageGroup2 ageGroup4 priorAttempts longestNoSmoke
## 1        heavy     21-49     35-49             3             90
## 2        heavy     21-49     21-34             3             21
## 3        heavy     21-49     35-49             0              0
## 4        heavy       50+       65+             0              0
```

```
## 5        light       50+       65+            10              15
## 6        heavy       50+     50-64            12             365
```

```r
coxph(Surv(ttr, relapse) ~ grp, data = d_ft) %>% summary()
```

```
## Call:
## coxph(formula = Surv(ttr, relapse) ~ grp, data = d_ft)
##
##   n= 72, number of events= 49
##
##                     coef exp(coef) se(coef)      z Pr(>|z|)
## grpcombination -0.3580    0.6991   0.2891 -1.238    0.216
##
##                exp(coef) exp(-coef) lower .95 upper .95
## grpcombination    0.6991       1.43    0.3967     1.232
##
## Concordance= 0.551  (se = 0.038 )
## Rsquare= 0.021   (max possible= 0.995 )
## Likelihood ratio test= 1.55  on 1 df,   p=0.2
## Wald test            = 1.53  on 1 df,   p=0.2
## Score (logrank) test = 1.55  on 1 df,   p=0.2
```

```r
d_ft$grp <- relevel(d_ft$grp, ref = "combination")
coxph(Surv(ttr, relapse) ~ grp, data = d_ft) %>% summary()
```

```
## Call:
## coxph(formula = Surv(ttr, relapse) ~ grp, data = d_ft)
##
##   n= 72, number of events= 49
##
##                  coef exp(coef) se(coef)     z Pr(>|z|)
## grppatchOnly 0.3580    1.4304   0.2891 1.238    0.216
##
##              exp(coef) exp(-coef) lower .95 upper .95
## grppatchOnly      1.43     0.6991    0.8117     2.521
##
## Concordance= 0.551  (se = 0.038 )
## Rsquare= 0.021   (max possible= 0.995 )
## Likelihood ratio test= 1.55  on 1 df,   p=0.2
## Wald test            = 1.53  on 1 df,   p=0.2
## Score (logrank) test = 1.55  on 1 df,   p=0.2
```

```r
coxph(Surv(ttr, relapse) ~ grp, data = d_nft) %>% summary()
```

```
## Call:
## coxph(formula = Surv(ttr, relapse) ~ grp, data = d_nft)
##
##   n= 53, number of events= 40
##
##                     coef exp(coef) se(coef)      z Pr(>|z|)
## grpcombination -0.9847    0.3736   0.3288 -2.995  0.00274 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                 exp(coef) exp(-coef) lower .95 upper .95
## grpcombination     0.3736      2.677    0.1961    0.7115
##
## Concordance= 0.624  (se = 0.039 )
## Rsquare= 0.16   (max possible= 0.994 )
## Likelihood ratio test= 9.22  on 1 df,   p=0.002
## Wald test            = 8.97  on 1 df,   p=0.003
## Score (logrank) test = 9.64  on 1 df,   p=0.002
```

II Part

H0: Beta FT = Beta OTH? <-> H0: interact beta = 0 p-value = ?

Interaction Test: beta in interaction term between trt and emp

```r
fit_int <- coxph(Surv(ttr, relapse) ~ grp + employment + grp:employment, data = dat)
summary(fit_int)
```

```
## Call:
## coxph(formula = Surv(ttr, relapse) ~ grp + employment + grp:employment,
##     data = dat)
##
##   n= 125, number of events= 89
##
##                                  coef exp(coef) se(coef)      z Pr(>|z|)
## grpcombination                -0.3554    0.7009   0.2889 -1.230   0.2186
## employmentother                0.6361    1.8892   0.2802  2.270   0.0232
## grpcombination:employmentother -0.6587    0.5175   0.4343 -1.517   0.1294
##
## grpcombination
## employmentother                *
## grpcombination:employmentother
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                                exp(coef) exp(-coef) lower .95 upper .95
## grpcombination                    0.7009     1.4268    0.3978     1.235
## employmentother                   1.8892     0.5293    1.0908     3.272
## grpcombination:employmentother    0.5175     1.9323    0.2209     1.212
##
## Concordance= 0.601  (se = 0.03 )
## Rsquare= 0.099   (max possible= 0.998 )
## Likelihood ratio test= 12.99  on 3 df,   p=0.005
## Wald test            = 14.22  on 3 df,   p=0.003
## Score (logrank) test = 15.13  on 3 df,   p=0.002
```

p-value = 0.13, so it not significant different