Exercise for survival analysis

Alessio Crippa

February 28, 2018

Survival analysis, Exercises

Consider now the Whitehall study, a large prospective cohort of 17,260 male British Civil Servants. *Lancet*, Volume 323, Issue 8384, 1984, Pages 1003–1006. During 10 years follow-up (165,612 person-years) we observed 1,670 deaths.

Questions

Data inspection

1. Read the **wh** data available at http://alecri.github.io/downloads/data/whitehall.csv

```
wh = read.csv("http://alecri.github.io/downloads/data/whitehall.csv")
```

2. Get familiar with the data. How many observations and variables (which type) are in the dataset?

```
# number of rows and columns
dim(wh)
```

```
## [1] 17260    18
```

```
# first observations
head(wh)
```

```
##   id all10 pyall10 chd   pyar jobgrade age sysbp       map     ht     chol
## 1  1     0  9.9999   1 24.665 Clerical  46   121  97.00000 154.94 6.201550
## 2  2     0  9.9999   1 17.832     Prof  55   135  97.66666 179.07 4.909561
## 3  3     0  9.9999   0 27.157     Prof  43   106  82.00000 173.99 4.754522
## 4  4     0  9.9999   0 11.135     Prof  56   160 111.33334 168.91 8.785530
## 5  5     0  9.9999   0 26.344     Prof  44   119  87.66666 158.75 5.788114
## 6  6     0  9.9999   0 27.121     Prof  48   133 106.33334 177.80 4.392765
##   agecat      bmi cigs diasbp    wt smoke        bmic
## 1  40-49 41.37230    0     85 99.32     0   (26.6,42]
## 2  50-59 21.07212    0     79 67.57     0   (14,22.5]
## 3  40-49 22.76982    0     70 68.93     0 (22.5,26.6]
## 4  50-59 27.50031    0     87 78.46     0   (26.6,42]
## 5  40-49 21.05425    0     72 53.06     0   (14,22.5]
## 6  40-49 29.40894    0     93 92.97     0   (26.6,42]
```

```
# names of variables
colnames(wh)
```

```
##  [1] "id"       "all10"    "pyall10"  "chd"      "pyar"     "jobgrade"
##  [7] "age"      "sysbp"    "map"      "ht"       "chol"     "agecat"
## [13] "bmi"      "cigs"     "diasbp"   "wt"       "smoke"    "bmic"
```

```
# structure of the data
str(wh)
```

```
## 'data.frame':    17260 obs. of  18 variables:
```

```
##  $ id      : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ all10   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pyall10 : num  10 10 10 10 10 ...
##  $ chd     : int  1 1 0 0 0 0 0 0 0 0 ...
##  $ pyar    : num  24.7 17.8 27.2 11.1 26.3 ...
##  $ jobgrade: Factor w/ 4 levels "Admin","Clerical",..: 2 4 4 4 4 4 2 4 4 2 ...
##  $ age     : int  46 55 43 56 44 48 42 46 48 46 ...
##  $ sysbp   : int  121 135 106 160 119 133 110 151 125 164 ...
##  $ map     : num  97 97.7 82 111.3 87.7 ...
##  $ ht      : num  155 179 174 169 159 ...
##  $ chol    : num  6.2 4.91 4.75 8.79 5.79 ...
##  $ agecat  : Factor w/ 3 levels "40-49","50-59",..: 1 2 1 2 1 1 1 1 1 1 ...
##  $ bmi     : num  41.4 21.1 22.8 27.5 21.1 ...
##  $ cigs    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ diasbp  : num  85 79 70 87 72 93 75 98 95 88 ...
##  $ wt      : num  99.3 67.6 68.9 78.5 53.1 ...
##  $ smoke   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ bmic    : Factor w/ 3 levels "(14,22.5]","(22.5,26.6]",..: 3 1 2 3 1 3 2 2 2 3 ...
```

```r
# summary of the data
summary(wh)
```

```
##        id            all10            pyall10            chd        
##  Min.   :    1   Min.   :0.00000   Min.   : 0.008   Min.   :0.0000  
##  1st Qu.: 4316   1st Qu.:0.00000   1st Qu.:10.000   1st Qu.:0.0000  
##  Median : 8630   Median :0.00000   Median :10.000   Median :0.0000  
##  Mean   : 8630   Mean   :0.09676   Mean   : 9.595   Mean   :0.1492  
##  3rd Qu.:12945   3rd Qu.:0.00000   3rd Qu.:10.000   3rd Qu.:0.0000  
##  Max.   :17260   Max.   :1.00000   Max.   :10.000   Max.   :1.0000  
##       pyar            jobgrade          age            sysbp      
##  Min.   : 0.008   Admin   :  948   Min.   :40.0   Min.   : 85.0  
##  1st Qu.:18.702   Clerical: 2712   1st Qu.:47.0   1st Qu.:121.0  
##  Median :25.602   Other   : 1583   Median :51.0   Median :133.0  
##  Mean   :21.807   Prof    :12017   Mean   :51.6   Mean   :136.1  
##  3rd Qu.:26.355                    3rd Qu.:57.0   3rd Qu.:148.0  
##  Max.   :27.381                    Max.   :64.0   Max.   :280.0  
##       map               ht             chol          agecat    
##  Min.   : 51.67   Min.   :134.6   Min.   : 1.034   40-49:7210  
##  1st Qu.: 91.33   1st Qu.:171.4   1st Qu.: 4.315   50-59:7772  
##  Median :100.00   Median :175.3   Median : 5.039   60-64:2278  
##  Mean   :101.68   Mean   :175.8   Mean   : 5.108               
##  3rd Qu.:109.67   3rd Qu.:180.3   3rd Qu.: 5.814               
##  Max.   :209.33   Max.   :203.2   Max.   :13.230               
##       bmi             cigs           diasbp             wt        
##  Min.   :14.36   Min.   : 0.00   Min.   :  5.00   Min.   : 36.73  
##  1st Qu.:22.78   1st Qu.: 0.00   1st Qu.: 75.00   1st Qu.: 69.39  
##  Median :24.64   Median : 0.00   Median : 83.00   Median : 76.19  
##  Mean   :24.73   Mean   : 6.65   Mean   : 84.47   Mean   : 76.47  
##  3rd Qu.:26.50   3rd Qu.:13.00   3rd Qu.: 92.00   3rd Qu.: 82.54  
##  Max.   :41.65   Max.   :60.00   Max.   :201.00   Max.   :136.05  
##      smoke                 bmic     
##  Min.   :0.0000   (14,22.5]  :3807  
##  1st Qu.:0.0000   (22.5,26.6]:9307  
##  Median :0.0000   (26.6,42]  :4146  
##  Mean   :0.4147
```

```
##  3rd Qu.:1.0000
##  Max.   :1.0000
```

3. `all10` and `pyall10` are the variables indicating if a person died (`all10 = 1`) and the corresponding follow-up time. Describe the two variables. What is the mortality rate (x 10000)?

```r
tab = table(wh$all10)
tab
```

```
##
##     0     1
## 15590  1670
```

```r
prop.table(tab)
```

```
##
##         0         1
## 0.9032445 0.0967555
```

```r
summary(wh$pyall10)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.008  10.000  10.000   9.595  10.000  10.000
```

```r
library(tidyverse)
summarise(wh, 10000*sum(all10)/sum(pyall10))
```

```
##   10000 * sum(all10)/sum(pyall10)
## 1                        100.8383
```

4. Create a survival object. Display the first 10 observation?

```r
library(survival)
all = Surv(wh$pyall10, wh$all10)
head(all, n = 10)
```

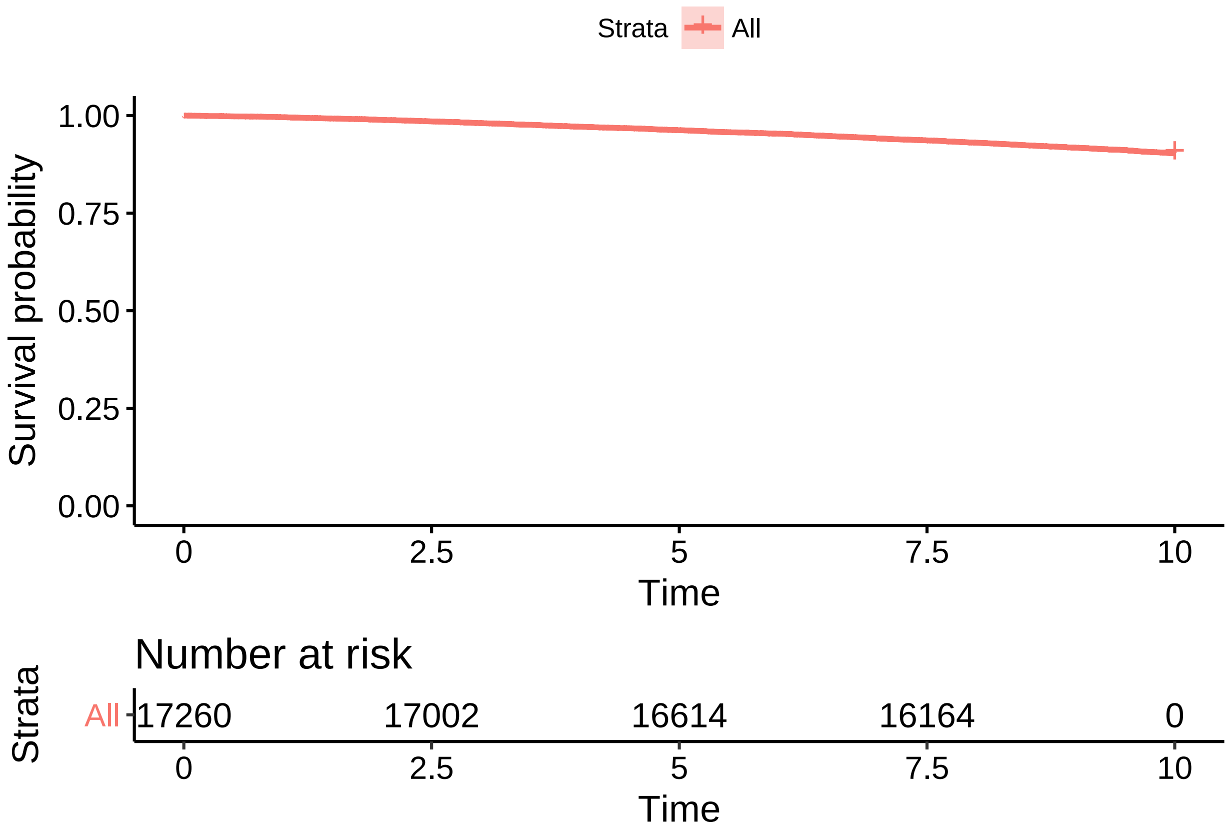
```
##  [1] 9.9999+ 9.9999+ 9.9999+ 9.9999+ 9.9999+ 9.9999+ 9.9999+ 9.9999+
##  [9] 9.9999+ 9.9999+
```

5. Estimate the survival function using the Kaplan–Meier method. Why there is no information about the survival time?

```r
fitkm = survfit(all ~ 1, data = wh)
fitkm
```

```
## Call: survfit(formula = all ~ 1, data = wh)
##
##        n  events  median 0.95LCL 0.95UCL
##    17260    1670      NA      NA      NA
```

```r
library(survminer)
ggsurvplot(fitkm, risk.table = T)
```



6. Estimate the 1th and 5th percentiles of survival times and interpret the results.

```
quantile(fitkm, c(.01, .05))
```

```
## $quantile
##     1     5
## 1.878 6.301
##
## $lower
##     1     5
## 1.670 6.067
##
## $upper
##     1     5
## 2.089 6.585
```

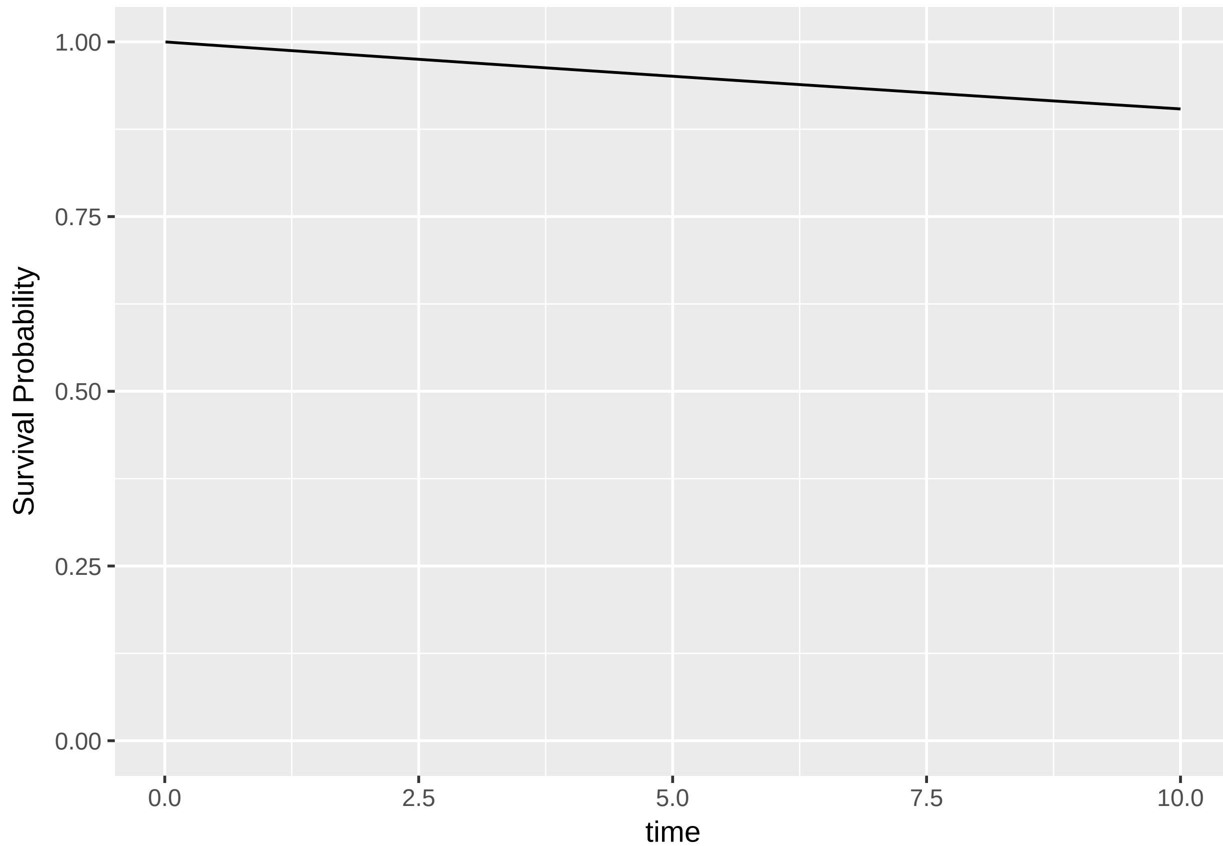
7. Assume an exponential distribution for time. Estimate the survival curve using the corresponding parametric model.

```
library(flexsurv)
fitex = flexsurvreg(all ~ 1, data = wh, dist = "exp")
fitex
```

```
## Call:
## flexsurvreg(formula = all ~ 1, data = wh, dist = "exp")
##
## Estimates:
##       est       L95%      U95%      se
## rate  0.010084  0.009612  0.010579  0.000247
##
```

```
## N = 17260,  Events: 1670,  Censored: 15590
## Total time at risk: 165611.6
## Log-likelihood = -9346.693, df = 1
## AIC = 18695.39
```

```
data.frame(summary(fitex)) %>%
  ggplot(aes(time, est)) +
  geom_line() + ylim(c(0, 1)) +
  labs(y = "Survival Probability")
```



8. Consider the possible health inequalities among british civil servants depending on the `jobgrade`. What is the mortality rate in the different jobgrade categories?

```
table(wh$jobgrade)
```

```
##
##    Admin Clerical    Other     Prof
##      948     2712     1583    12017
```

```
wh %>%
  group_by(jobgrade) %>%
  summarise(rates = 10000*sum(all10)/sum(pyall10))
```

```
## # A tibble: 4 x 2
##   jobgrade rates
##   <fct>    <dbl>
## 1 Admin     46.2
## 2 Clerical 155
## 3 Other    228
```

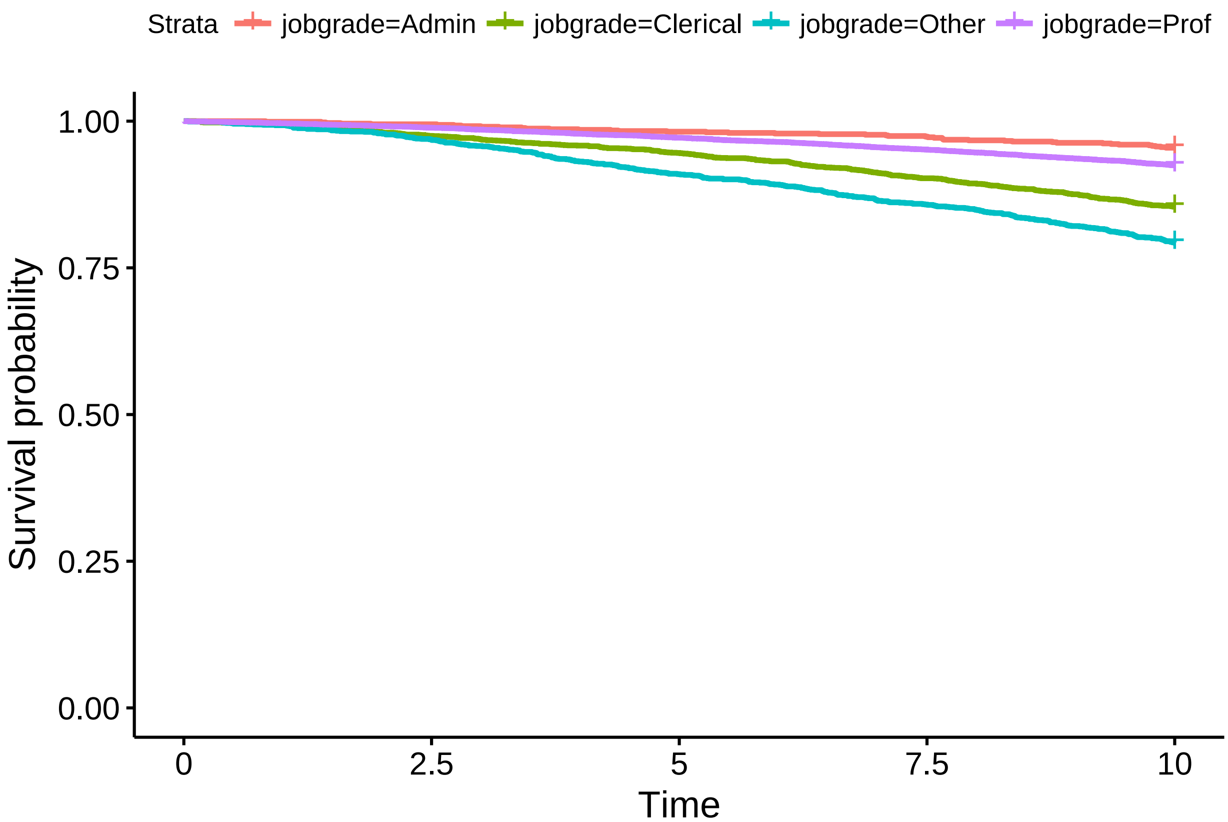
```
## 4 Prof      77.6
```

9. Estimate the survival curves and test for possible differences.

```r
fitkm_j = survfit(Surv(pyall10, all10) ~ jobgrade, data = wh)
fitkm_j
```

```
## Call: survfit(formula = Surv(pyall10, all10) ~ jobgrade, data = wh)
##
##                       n events median 0.95LCL 0.95UCL
## jobgrade=Admin      948     43     NA      NA      NA
## jobgrade=Clerical  2712    395     NA      NA      NA
## jobgrade=Other     1583    328     NA      NA      NA
## jobgrade=Prof     12017    904     NA      NA      NA
```

```r
ggsurvplot(fitkm_j)
```




```r
survdiff(Surv(pyall10, all10) ~ jobgrade, data = wh)
```

```
## Call:
## survdiff(formula = Surv(pyall10, all10) ~ jobgrade, data = wh)
##
##                       N Observed Expected (O-E)^2/E (O-E)^2/V
## jobgrade=Admin      948       43     94.3      27.9      29.6
## jobgrade=Clerical  2712      395    255.4      76.3      90.1
## jobgrade=Other     1583      328    143.5     237.4     259.7
## jobgrade=Prof     12017      904   1176.8      63.2     214.2
##
##  Chisq= 405  on 3 degrees of freedom, p= 0
```

10. Specify a Cox regression model to investigate the association between jobgrade and (log) rates of death,

adjusted for age. Interpret the results.

```r
summary(wh$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    40.0    47.0    51.0    51.6    57.0    64.0
```

```r
fitc = coxph(Surv(pyall10, all10) ~ jobgrade + I(age - 50), data = wh)
summary(fitc)
```

```
## Call:
## coxph(formula = Surv(pyall10, all10) ~ jobgrade + I(age - 50),
##     data = wh)
##
##   n= 17260, number of events= 1670
##
##                     coef exp(coef) se(coef)      z Pr(>|z|)
## jobgradeClerical 0.884125  2.420866 0.161257  5.483 4.19e-08 ***
## jobgradeOther    1.085354  2.960487 0.163564  6.636 3.23e-11 ***
## jobgradeProf     0.474210  1.606745 0.156101  3.038  0.00238 **
## I(age - 50)      0.099783  1.104932 0.004355 22.915  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## jobgradeClerical     2.421     0.4131     1.765     3.321
## jobgradeOther        2.960     0.3378     2.149     4.079
## jobgradeProf         1.607     0.6224     1.183     2.182
## I(age - 50)          1.105     0.9050     1.096     1.114
##
## Concordance= 0.708  (se = 0.007 )
## Rsquare= 0.051   (max possible= 0.847 )
## Likelihood ratio test= 911.3  on 4 df,   p=0
## Wald test            = 872.5  on 4 df,   p=0
## Score (logrank) test = 966.1  on 4 df,   p=0
```

11. Assuming the effect of age on the (log) rates of death can be approximated by a quadratic curve. Estimate and present the results from the corresponding Cox model.

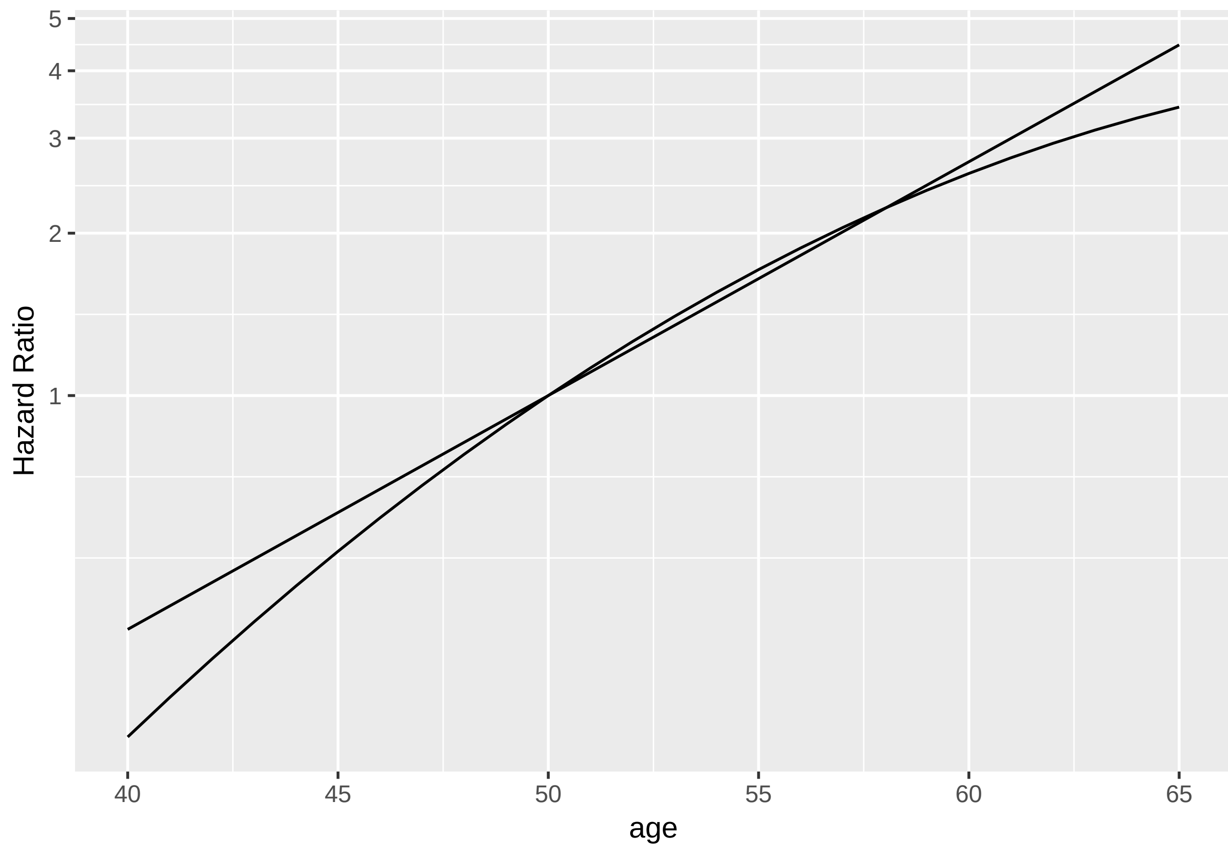
```r
fitc2 = coxph(Surv(pyall10, all10) ~ jobgrade + I(age - 50) + I((age - 50)^2), data = wh)
summary(fitc2)
```

```
## Call:
## coxph(formula = Surv(pyall10, all10) ~ jobgrade + I(age - 50) +
##     I((age - 50)^2), data = wh)
##
##   n= 17260, number of events= 1670
##
##                       coef  exp(coef)   se(coef)      z Pr(>|z|)
## jobgradeClerical  0.8972961  2.4529615  0.1612432  5.565 2.62e-08 ***
## jobgradeOther     1.1057067  3.0213588  0.1635722  6.760 1.38e-11 ***
## jobgradeProf      0.4764397  1.6103310  0.1560993  3.052 0.002272 **
## I(age - 50)       0.1202487  1.1277772  0.0073800 16.294  < 2e-16 ***
## I((age - 50)^2)  -0.0025445  0.9974587  0.0007057 -3.606 0.000311 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
##                    exp(coef) exp(-coef) lower .95 upper .95
## jobgradeClerical    2.4530     0.4077    1.7883    3.3647
## jobgradeOther       3.0214     0.3310    2.1927    4.1633
## jobgradeProf        1.6103     0.6210    1.1859    2.1867
## I(age - 50)         1.1278     0.8867    1.1116    1.1442
## I((age - 50)^2)     0.9975     1.0025    0.9961    0.9988
##
## Concordance= 0.708  (se = 0.007 )
## Rsquare= 0.052   (max possible= 0.847 )
## Likelihood ratio test= 924.8  on 5 df,   p=0
## Wald test            = 815.2  on 5 df,   p=0
## Score (logrank) test = 999.1  on 5 df,   p=0
```

```r
library(Epi)
agec = seq(40, 65, 1) - 50
hrtab1 = ci.exp(fitc, ctr.mat = cbind(0, 0, 0, agec))
hrtab2 = ci.exp(fitc2, ctr.mat = cbind(0, 0, 0, agec, agec^2))
hr = data.frame(
  age = agec + 50, lin = hrtab1, quadr = hrtab2
)

library(scales)
ggplot(hr, aes(age, lin.exp.Est..)) +
  geom_line() +
  geom_line(aes(y = quadr.exp.Est..)) +
  scale_y_continuous(trans = "log", breaks = pretty_breaks()) +
  labs(y = "Hazard Ratio")
```



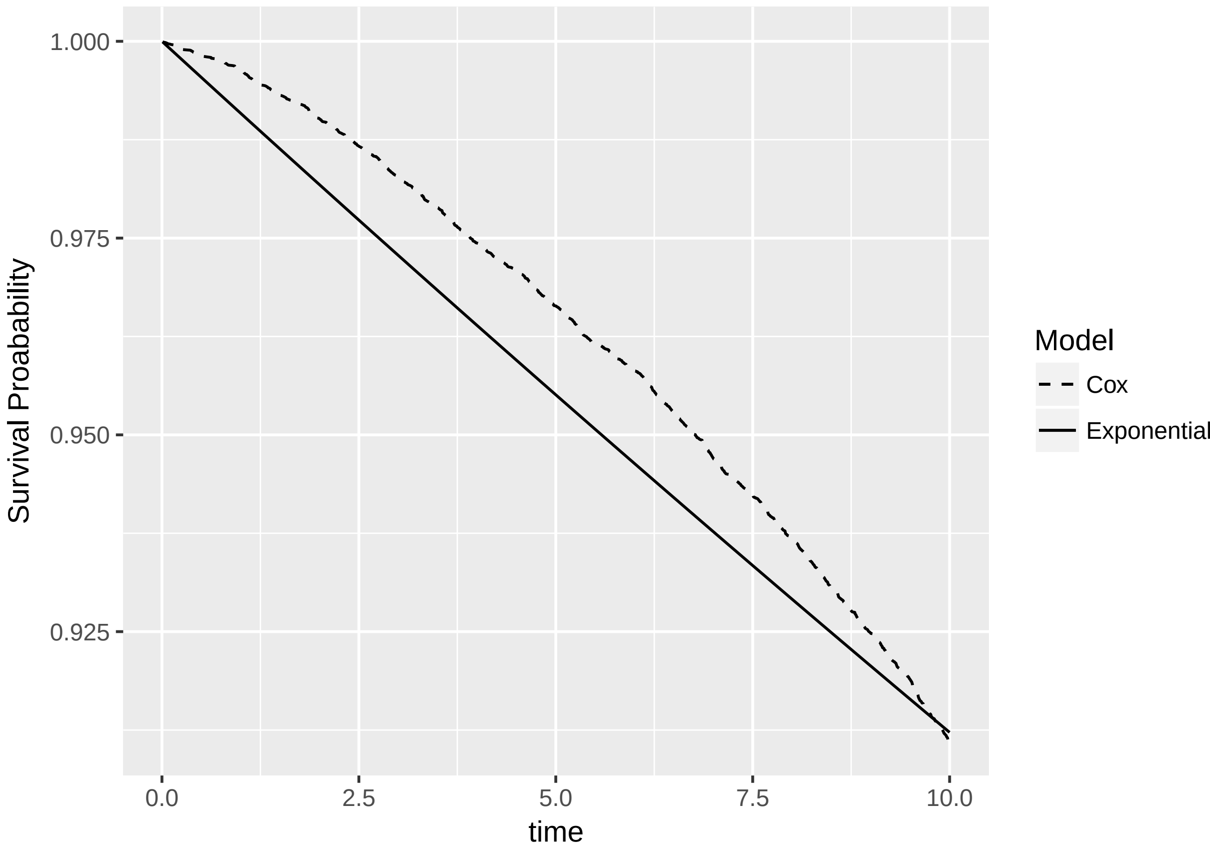
12. Run a similar analysis as in 10. assuming an exponential distribution for the survival time. Interpret the results

```
fitex2 = flexsurvreg(Surv(pyall10, all10) ~ jobgrade + I(age - 50), data = wh, dist = "exp")
fitex2
```

```
## Call:
## flexsurvreg(formula = Surv(pyall10, all10) ~ jobgrade + I(age -
##     50), data = wh, dist = "exp")
##
## Estimates:
##                   data mean  est       L95%      U95%      se
## rate                    NA   0.003836  0.002840  0.005181  0.000588
## jobgradeClerical  0.157126   0.873615  0.557537  1.189693  0.161267
## jobgradeOther     0.091715   1.070061  0.749456  1.390666  0.163577
## jobgradeProf      0.696234   0.470254  0.164301  0.776207  0.156101
## I(age - 50)       1.596582   0.098657  0.090126  0.107188  0.004352
##                   exp(est)  L95%      U95%
## rate                    NA        NA        NA
## jobgradeClerical  2.395555  1.746366  3.286073
## jobgradeOther     2.915559  2.115850  4.017526
## jobgradeProf      1.600401  1.178569  2.173214
## I(age - 50)       1.103688  1.094313  1.113143
##
## N = 17260,  Events: 1670,  Censored: 15590
## Total time at risk: 165611.6
## Log-likelihood = -8901.141, df = 5
## AIC = 17812.28
```

13. Compare the predicted survival curves based on the estimated models in 10. and 12. for a 50 years-old man with Clerical as jobgrade.

```
library(ggfortify)
newd = data.frame(age = 50, jobgrade = "Clerical")
fortify(survfit(fitc, newdata = newd)) %>%
  data.frame() %>%
  ggplot(aes(time, surv, linetype = "Cox")) +
  geom_line() +
  geom_line(data = summary(fitex2, newdata = newd, tidy = T),
            aes(y = est, linetype = "Exponential")) +
  scale_linetype_manual(values = c("dashed", "solid")) +
  labs(y = "Survival Proabability", linetype = "Model")
```



Survival Proabability
1.000
0.975
0.950
0.925
0.0
2.5
5.0
7.5
10.0
time
Model
Cox
Exponential