HW2

Bulun Te

2024-01-26

```r
library(dplyr)
library(survival)
library(kableExtra)
library(knitr)
library(ggplot2)

# Reading the data

data = read.table("Breast_cancer_Table_1_2_Collet.txt")

# Changing the column names

data = data %>% rename(
  id = V1,
  x = V2,
  delta = V3,
  z = V4
)

data %>% head()
```

```
##   id   x delta z
## 1  1 224     0 0
## 2  2 212     0 0
## 3  3 208     0 0
## 4  4 198     0 0
## 5  5 181     1 0
## 6  6 148     1 0
```

```r
data %>% psych::describe()
```

```
##       vars  n  mean    sd median trimmed   mad min max range  skew kurtosis
## id       1 45 23.00 13.13     23   23.00 16.31   1  45    44  0.00    -1.28
## x        2 45 96.22 69.35     71   92.32 66.72   5 225   220  0.51    -1.09
## delta    3 45  0.58  0.50      1    0.59  0.00   0   1     1 -0.30    -1.95
## z        4 45  0.71  0.46      1    0.76  0.00   0   1     1 -0.90    -1.21
##          se
## id     1.96
## x     10.34
## delta  0.07
## z      0.07
```

Answer without using survival package

Question 1(a)

$\hat{\Lambda}_{NA}(t) = \sum_{t_j<t} \frac{D_j}{Y_j}$

where D_j is the number of events at time t_j and Y_j is the number of individuals at risk at time t_j.

```
# sort data with x column with aescending order

data_sorted = data %>% arrange(x)

data_sorted
```

```
##    id   x delta z
## 1  14   5     1 1
## 2  15   8     1 1
## 3  16  10     1 1
## 4  17  13     1 1
## 5  18  18     1 1
## 6  13  23     1 0
## 7  19  24     1 1
## 8  20  25     1 1
## 9  21  26     1 1
## 10 22  31     1 1
## 11 23  35     1 1
## 12 24  40     1 1
## 13 25  41     1 1
## 14 12  47     1 0
## 15 26  48     1 1
## 16 27  50     1 1
## 17 28  59     1 1
## 18 29  61     1 1
## 19 30  68     1 1
## 20 11  69     1 0
## 21 10  70     0 0
## 22  9  71     0 0
## 23 31  71     1 1
## 24 32  76     0 1
## 25  8 100     0 0
## 26  7 101     0 0
## 27 33 105     0 1
## 28 34 107     0 1
## 29 35 109     0 1
## 30 36 113     1 1
## 31 37 116     0 1
## 32 38 118     1 1
## 33 39 143     1 1
## 34  6 148     1 0
## 35 40 154     0 1
## 36 41 162     0 1
## 37  5 181     1 0
## 38 42 188     0 1
```

```
## 39   4 198     0 0
## 40   3 208     0 0
## 41   2 212     0 0
## 42  43 212     0 1
## 43  44 217     0 1
## 44   1 224     0 0
## 45  45 225     0 1
```

```
data_sorted$y = sapply(data_sorted$x,function(u){sum(data_sorted$x >= u)})

data_sorted$lambda = data_sorted$delta / data_sorted$y

data_sorted$cumulative_hazard = cumsum(data_sorted$lambda)

cumulative_hazard = data_sorted %>%
  select(x,delta,y,lambda,cumulative_hazard) %>%
  filter(delta ==1) %>%
  select(cumulative_hazard)

ppl_at_risk = data_sorted %>%
  select(x,delta,y,lambda,cumulative_hazard) %>%
  filter(delta >=1) %>% select(x,y)

data2 = data.frame(time = ppl_at_risk$x,
                    Y = ppl_at_risk$y,
                   Nelson_Alan_Cumaltive= cumulative_hazard)

data2
```

```
##    time  Y cumulative_hazard
## 1     5 45        0.02222222
## 2     8 44        0.04494949
## 3    10 43        0.06820531
## 4    13 42        0.09201483
## 5    18 41        0.11640508
## 6    23 40        0.14140508
## 7    24 39        0.16704610
## 8    25 38        0.19336189
## 9    26 37        0.22038892
## 10   31 36        0.24816670
## 11   35 35        0.27673813
## 12   40 34        0.30614989
## 13   41 33        0.33645292
## 14   47 32        0.36770292
## 15   48 31        0.39996098
## 16   50 30        0.43329432
## 17   59 29        0.46777708
## 18   61 28        0.50349136
## 19   68 27        0.54052840
## 20   69 26        0.57898994
## 21   71 24        0.62065660
## 22  113 16        0.68315660
## 23  118 14        0.75458518
## 24  143 13        0.83150825
```

```
## 25   148  12         0.91484159
## 26   181   9         1.02595270
```

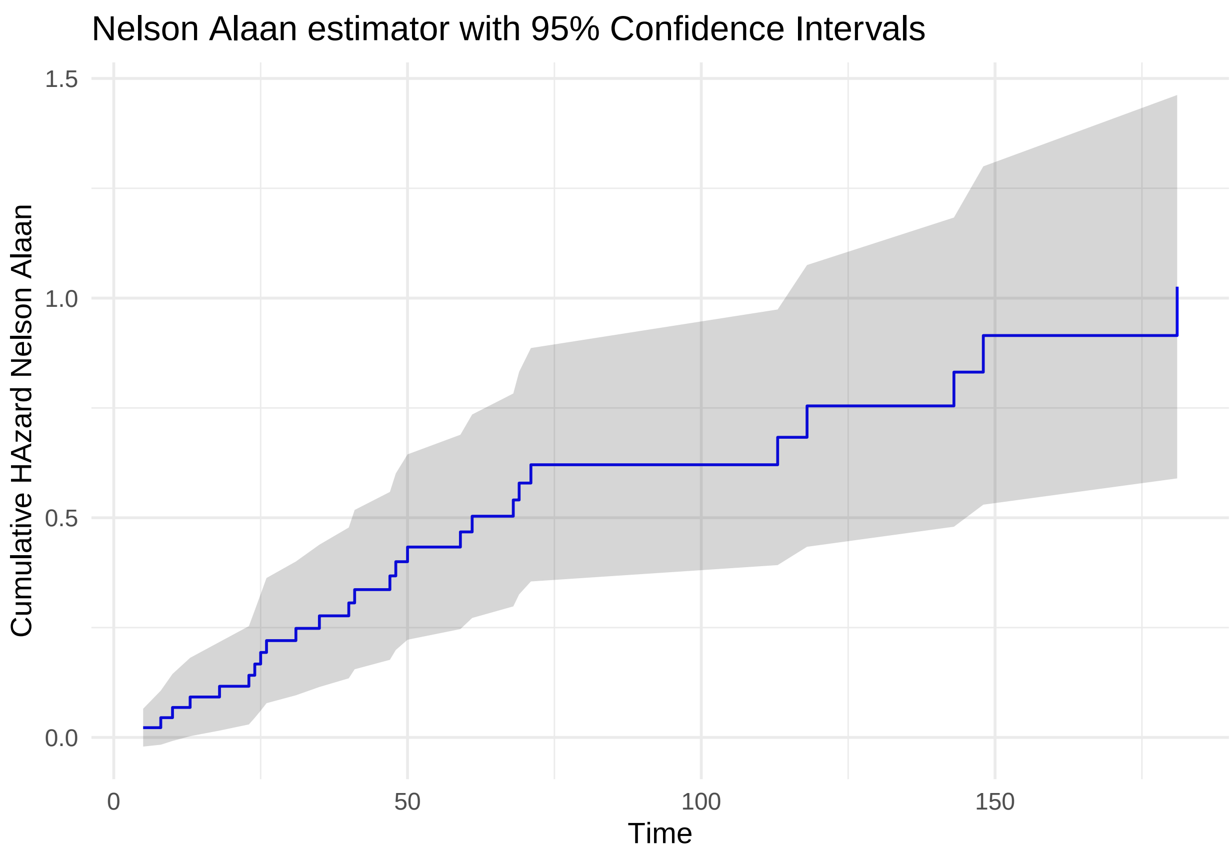
Question 1(b)

Assuming the $\hat{\Lambda}(t)$ follows normal distribution, and the variance of $\hat{\Lambda}(t)$ is estimated using Greenwood's formula as $\hat{V}(t) = \sum_{t_j<t} \frac{D_j(Y_j - D_j)}{Y_j^3}$, then the confidence interval is estimated as $\hat{\Lambda}(t) \pm z_{0.975}\sqrt{\hat{V}(t)}$.

```r
# Computing 95% confidence interval for the cumulative hazard assuming lambda(t) follows normal distrib


V_estimate = (data_sorted$delta*(data_sorted$y-data_sorted$delta)/(data_sorted$y^3)) %>% cumsum()

CI_upper = data_sorted$cumulative_hazard + qnorm(0.975)*sqrt(V_estimate)

CI_lower = data_sorted$cumulative_hazard - qnorm(0.975)*sqrt(V_estimate)

result_1_b = data_sorted %>% mutate(CI_upper = CI_upper, CI_lower = CI_lower)%>%
  filter(delta>=1) %>%
  select(x,cumulative_hazard,CI_upper,CI_lower) %>%
  round(.,6)

result_1_b
```

```
##       x cumulative_hazard CI_upper  CI_lower
## 1     5          0.022222 0.065290 -0.020846
## 2     8          0.044949 0.106545 -0.016646
## 3    10          0.068205 0.144516 -0.008105
## 4    13          0.092015 0.181173  0.002857
## 5    18          0.116405 0.217294  0.015516
## 6    23          0.141405 0.253296  0.029515
## 7    24          0.167046 0.289440  0.044652
## 8    25          0.193362 0.325916  0.060808
## 9    26          0.220389 0.362870  0.077908
## 10   31          0.248167 0.400425  0.095908
## 11   35          0.276738 0.438691  0.114785
## 12   40          0.306150 0.477772  0.134528
## 13   41          0.336453 0.517767  0.155139
## 14   47          0.367703 0.558776  0.176630
## 15   48          0.399961 0.600902  0.199020
## 16   50          0.433294 0.644253  0.222336
## 17   59          0.467777 0.688941  0.246613
## 18   61          0.503491 0.735091  0.271892
## 19   68          0.540528 0.782836  0.298221
## 20   69          0.578990 0.832321  0.325658
## 21   71          0.620657 0.886303  0.355010
## 22  113          0.683157 0.974079  0.392234
## 23  118          0.754585 1.075265  0.433906
## 24  143          0.831508 1.183385  0.479631
## 25  148          0.914842 1.299901  0.529782
## 26  181          1.025953 1.462332  0.589573
```

```
ggplot(result_1_b, aes(x = x)) +
  geom_step(aes(y = cumulative_hazard), direction = "hv", col = "blue") +
  geom_ribbon(aes(ymin = CI_lower, ymax = CI_upper), alpha = 0.2) +
  labs(x = "Time", y = "Cumulative HAzard Nelson Alaan") +
  ggtitle("Nelson Alaan estimator with 95% Confidence Intervals") +
  theme_minimal()
```



Question 1(c)

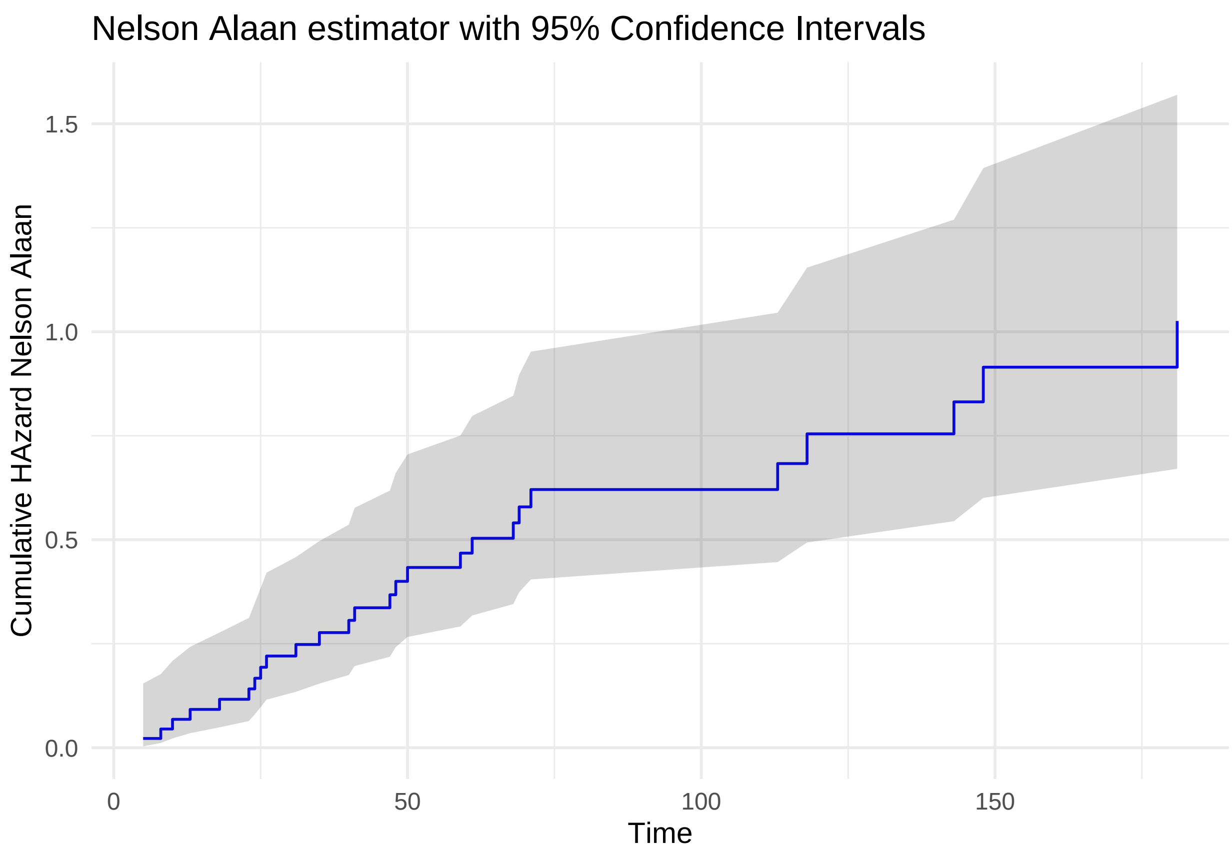
Assuming the $\hat{\Lambda}(t)$ follows normal distribution. The variance of $\hat{\Lambda(t)}$, using Delta method is estimated as $\hat{V}(\log(\hat{\Lambda}(t)) = \sum_{t_j<t} \frac{D_j(Y_j-D_j)}{Y_j^3\hat{\Lambda}(t)^2}$, then the confidence interval is estimated as $\exp(\log(\hat{\Lambda}(t)) \pm z_{0.975}\sqrt{\hat{V}(\log(\hat{\Lambda}(t)))}$.

```
# Computing 95% confidence interval for the cumulative hazard assuming
# log(lambda(t)) follows normal distribution

# Computing using delta mathod

V_estimate_temp = (data_sorted$delta*(data_sorted$y-data_sorted$delta)/(data_sorted$y^3)) %>% cumsum()

V_estimate = V_estimate_temp / (data_sorted$cumulative_hazard^2)

CI_upper = data_sorted$cumulative_hazard * exp(qnorm(0.975)*sqrt(V_estimate))
```

```r
CI_lower = data_sorted$cumulative_hazard * exp(-qnorm(0.975)*sqrt(V_estimate))

result_1_c = data_sorted %>%
  mutate(CI_upper = CI_upper, CI_lower = CI_lower)%>%
  filter(delta>=1) %>%
  select(x,cumulative_hazard,CI_upper,CI_lower) %>%
  round(.,6)

result_1_c
```

```
##      x cumulative_hazard CI_upper CI_lower
## 1    5          0.022222 0.154340 0.003200
## 2    8          0.044949 0.176949 0.011418
## 3   10          0.068205 0.208795 0.022280
## 4   13          0.092015 0.242475 0.034918
## 5   18          0.116405 0.276935 0.048929
## 6   23          0.141405 0.311969 0.064094
## 7   24          0.167046 0.347570 0.080284
## 8   25          0.193362 0.383787 0.097421
## 9   26          0.220389 0.420688 0.115457
## 10  31          0.248167 0.458350 0.134366
## 11  35          0.276738 0.496853 0.154138
## 12  40          0.306150 0.536280 0.174774
## 13  41          0.336453 0.576720 0.196283
## 14  47          0.367703 0.618264 0.218686
## 15  48          0.399961 0.661010 0.242007
## 16  50          0.433294 0.705064 0.266280
## 17  59          0.467777 0.750538 0.291545
## 18  61          0.503491 0.797558 0.317850
## 19  68          0.540528 0.846259 0.345250
## 20  69          0.578990 0.896793 0.373809
## 21  71          0.620657 0.952211 0.404547
## 22 113          0.683157 1.045839 0.446247
## 23 118          0.754585 1.154177 0.493338
## 24 143          0.831508 1.269553 0.544606
## 25 148          0.914842 1.393612 0.600551
## 26 181          1.025953 1.569822 0.670508
```

```r
ggplot(result_1_c, aes(x = x)) +
  geom_step(aes(y = cumulative_hazard), direction = "hv", col = "blue") +
  geom_ribbon(aes(ymin = CI_lower, ymax = CI_upper), alpha = 0.2) +
  labs(x = "Time", y = "Cumulative HAzard Nelson Alaan") +
  ggtitle("Nelson Alaan estimator with 95% Confidence Intervals") +
  theme_minimal()
```



Question 1(d)

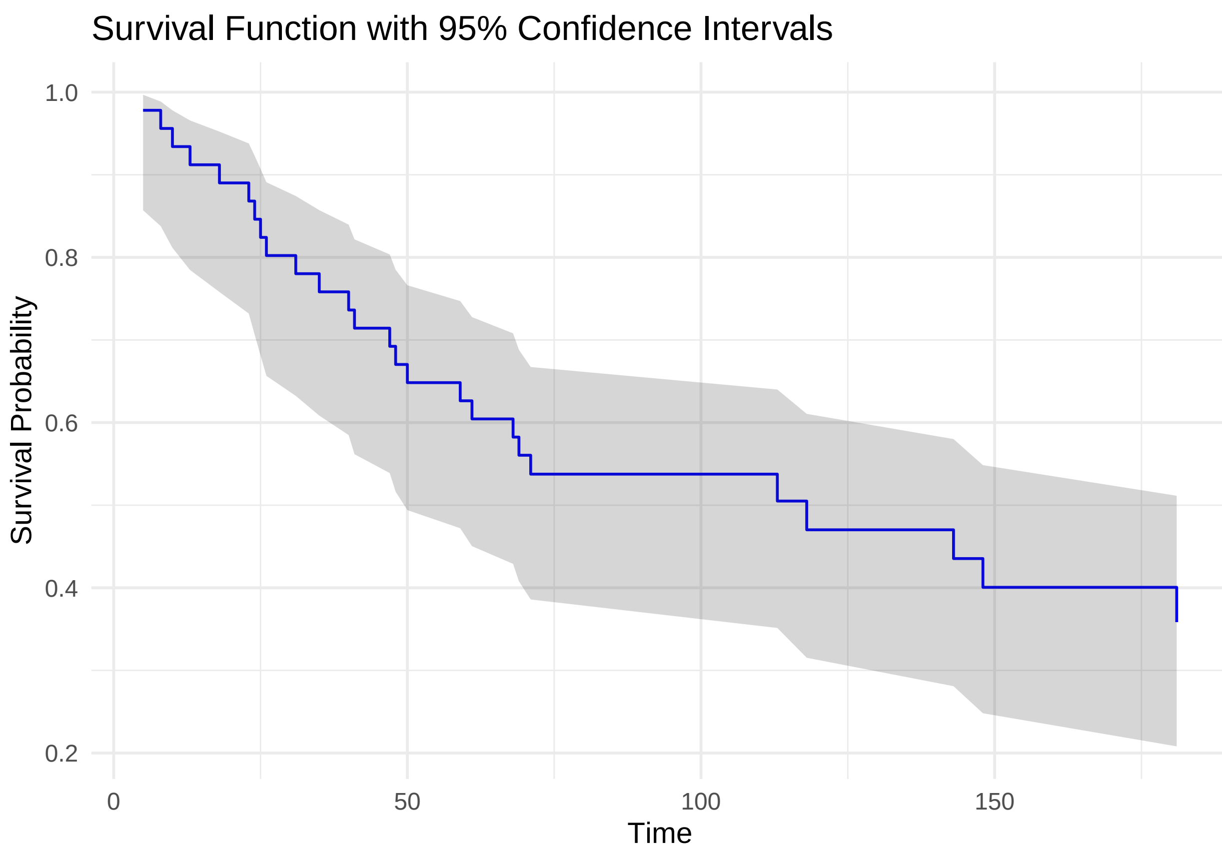
Assuming the $log(\Lambda(t))$ follows normal distribution. The variance of $log(\Lambda(t))$ is estimated as above. As $S(t) = exp(-\Lambda(t))$, thus $P\{\hat{\Lambda}(t) \in (a,b)\} = P\{\hat{S}(t) \in (e^{-b}, e^{-a})\} = 0.95$ And the CI are computed as follow:

```
V_estimate_temp = (data_sorted$delta*(data_sorted$y-data_sorted$delta)/(data_sorted$y^3)) %>% cumsum()

V_estimate = V_estimate_temp / (data_sorted$cumulative_hazard^2)

S_estimate = exp(-data_sorted$cumulative_hazard)

CI_lower = exp(-data_sorted$cumulative_hazard * exp(qnorm(0.975)*sqrt(V_estimate)))

CI_upper = exp(-data_sorted$cumulative_hazard * exp(-qnorm(0.975)*sqrt(V_estimate)))

result_1_d = data_sorted %>%
  mutate(CI_upper = CI_upper, CI_lower = CI_lower,S_estimate=S_estimate)%>%
  filter(delta>=1) %>%
  select(x,CI_upper,S_estimate,CI_lower) %>%
  round(.,6)

result_1_d
```

```
##       x CI_upper S_estimate CI_lower
```

```
## 1     5 0.996806   0.978023 0.856981
## 2     8 0.988647   0.956046 0.837822
## 3    10 0.977966   0.934069 0.811561
## 4    13 0.965685   0.912092 0.784683
## 5    18 0.952249   0.890115 0.758104
## 6    23 0.937917   0.868138 0.732004
## 7    24 0.922854   0.846161 0.706402
## 8    25 0.907174   0.824184 0.681277
## 9    26 0.890959   0.802207 0.656595
## 10   31 0.874270   0.780230 0.632326
## 11   35 0.857154   0.758253 0.608443
## 12   40 0.839647   0.736276 0.584920
## 13   41 0.821779   0.714300 0.561738
## 14   47 0.803574   0.692323 0.538879
## 15   48 0.785051   0.670346 0.516330
## 16   50 0.766225   0.648370 0.494077
## 17   59 0.747109   0.626393 0.472112
## 18   61 0.727712   0.604417 0.450428
## 19   68 0.708043   0.582440 0.429017
## 20   69 0.688108   0.560464 0.407875
## 21   71 0.667279   0.537591 0.385887
## 22  113 0.640025   0.505020 0.351397
## 23  118 0.610585   0.470206 0.315317
## 24  143 0.580070   0.435392 0.280957
## 25  148 0.548509   0.400580 0.248177
## 26  181 0.511448   0.358455 0.208082
```

```r
# plotting the survival function and its confidence interval from result_1_d as step function

ggplot(result_1_d, aes(x = x)) +
  geom_step(aes(y = S_estimate), direction = "hv", col = "blue") +
  geom_ribbon(aes(ymin = CI_lower, ymax = CI_upper), alpha = 0.2) +
  labs(x = "Time", y = "Survival Probability") +
  ggtitle("Survival Function with 95% Confidence Intervals") +
  theme_minimal()
```



Question 1(e)

```r
print("quartiles of 0.25,0.5 of survival estimations are")
```

```
## [1] "quartiles of 0.25,0.5 of survival estimations are"
```

```r
c(min(result_1_d$x[which(result_1_d$S_estimate <= (1-0.25))]),
min(result_1_d$x[which(result_1_d$S_estimate <= 0.5)]))
```

```
## [1]  40 118
```