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)

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

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
```

```r
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
```

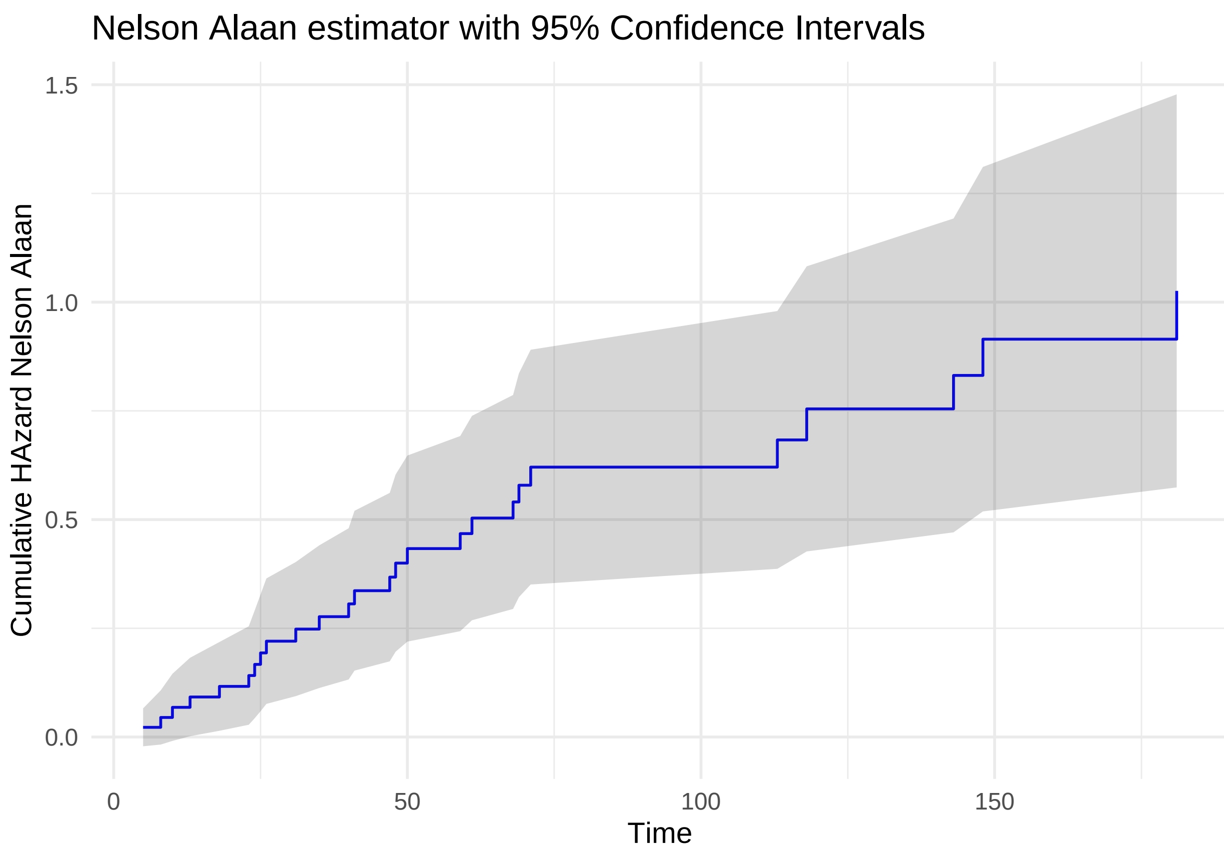
Question 1(b)

Assuming the $\hat{\Lambda}(t)$ follows normal distribution, and the variance of $\hat{\Lambda}(t)$ is estimated as $\hat{V}(t) = \sum_{t_j \le t} \frac{D_j}{Y_j^2}$, then the confidence interval is estimated as $\hat{\Lambda}(t) \pm z_{0.975}\sqrt{\hat{V}(t)}$.

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


V_estimate = ((data_sorted$delta)/(data_sorted$y^2)) %>% 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 0.065777 -0.021333
## 2    8          0.044949 0.107249 -0.017350
## 3   10          0.068205 0.145399 -0.008988
## 4   13          0.092015 0.182218  0.001812
## 5   18          0.116405 0.218492  0.014318
## 6   23          0.141405 0.254642  0.028168
## 7   24          0.167046 0.290934  0.043158
## 8   25          0.193362 0.327558  0.059166
## 9   26          0.220389 0.364662  0.076116
## 10  31          0.248167 0.402370  0.093963
## 11  35          0.276738 0.440795  0.112682
## 12  40          0.306150 0.480040  0.132260
## 13  41          0.336453 0.520206  0.152700
## 14  47          0.367703 0.561395  0.174011
## 15  48          0.399961 0.603711  0.196211
## 16  50          0.433294 0.647262  0.219327
## 17  59          0.467777 0.692165  0.243389
## 18  61          0.503491 0.738544  0.268439
## 19  68          0.540528 0.786535  0.294522
## 20  69          0.578990 0.836287  0.321693
## 21  71          0.620657 0.890603  0.350710
## 22 113          0.683157 0.979597  0.386716
## 23 118          0.754585 1.082421  0.426750
## 24 143          0.831508 1.192350  0.470667
## 25 148          0.914842 1.310927  0.518757
## 26 181          1.025953 1.477958  0.573947
```

```r
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)

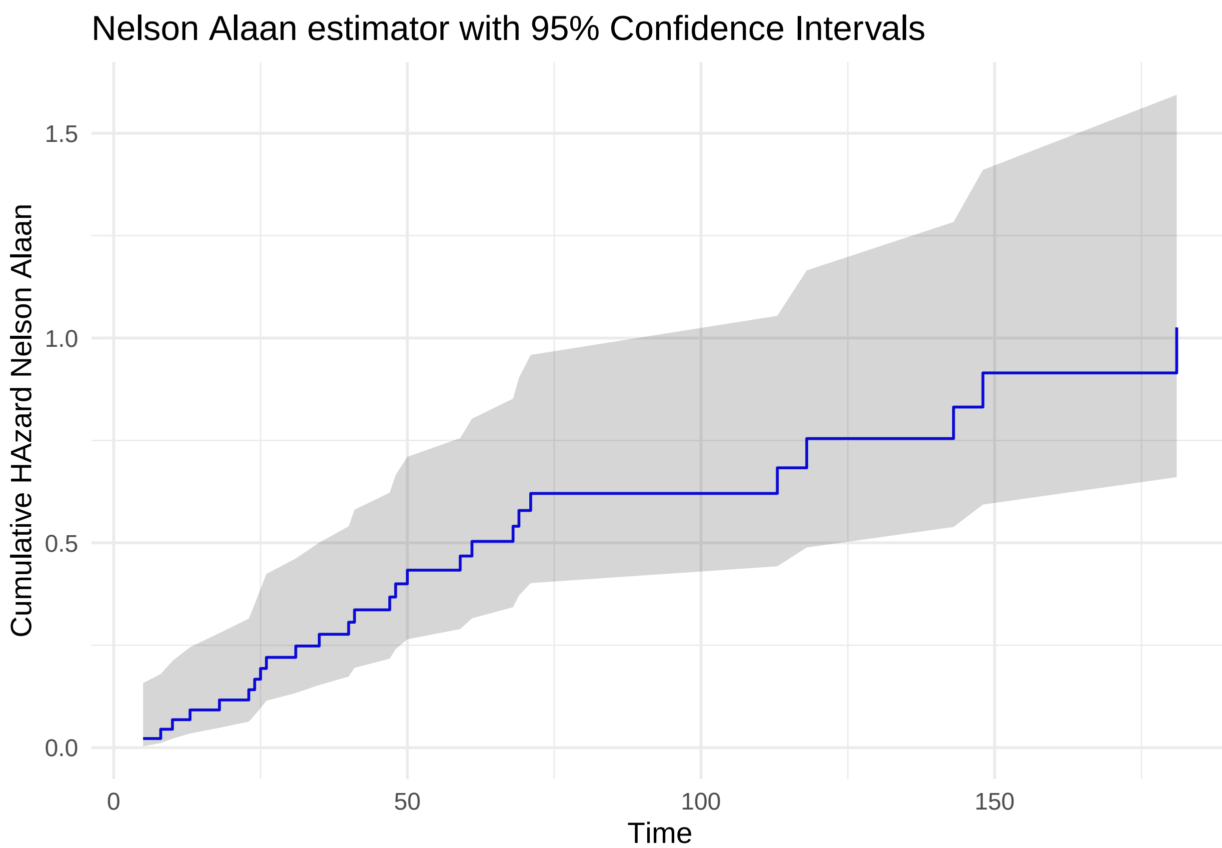
Assuming the $\hat{\Lambda}(t)$ follows normal distribution. The variance of $\Lambda\hat{(t)}$, using Delta method is estimated as $\hat{V}(\log(\hat{\Lambda}(t)) = \sum_{t_j \leq t} \frac{D_j}{Y_j^2 \hat{\Lambda}(t)^2}$, then the confidence interval is estimated as $\exp(\log(\hat{\Lambda}(t) \pm z_{0.975}\sqrt{\hat{V}(\log(\hat{\Lambda}(t)))})$.

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

# Computing using delta mathod

V_estimate = ((data_sorted$delta)/(data_sorted$y^2)) %>% cumsum()

V_estimate = V_estimate / (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 0.157757 0.003130
## 2    8          0.044949 0.179743 0.011241
## 3   10          0.068205 0.211517 0.021993
## 4   13          0.092015 0.245245 0.034524
## 5   18          0.116405 0.279800 0.048428
## 6   23          0.141405 0.314955 0.063487
## 7   24          0.167046 0.350693 0.079569
## 8   25          0.193362 0.387059 0.096597
## 9   26          0.220389 0.424122 0.114522
## 10  31          0.248167 0.461957 0.133317
## 11  35          0.276738 0.500643 0.152971
## 12  40          0.306150 0.540267 0.173484
## 13  41          0.336453 0.580916 0.194866
## 14  47          0.367703 0.622683 0.217134
## 15  48          0.399961 0.665668 0.240313
## 16  50          0.433294 0.709978 0.264436
## 17  59          0.467777 0.755728 0.289542
## 18  61          0.503491 0.803046 0.315677
## 19  68          0.540528 0.852071 0.342895
## 20  69          0.578990 0.902957 0.371257
## 21  71          0.620657 0.958831 0.401754
## 22 113          0.683157 1.054320 0.442658
## 23 118          0.754585 1.165174 0.488681
## 24 143          0.831508 1.283314 0.538766
## 25 148          0.914842 1.410509 0.593357
## 26 181          1.025953 1.593914 0.660374
```

```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)

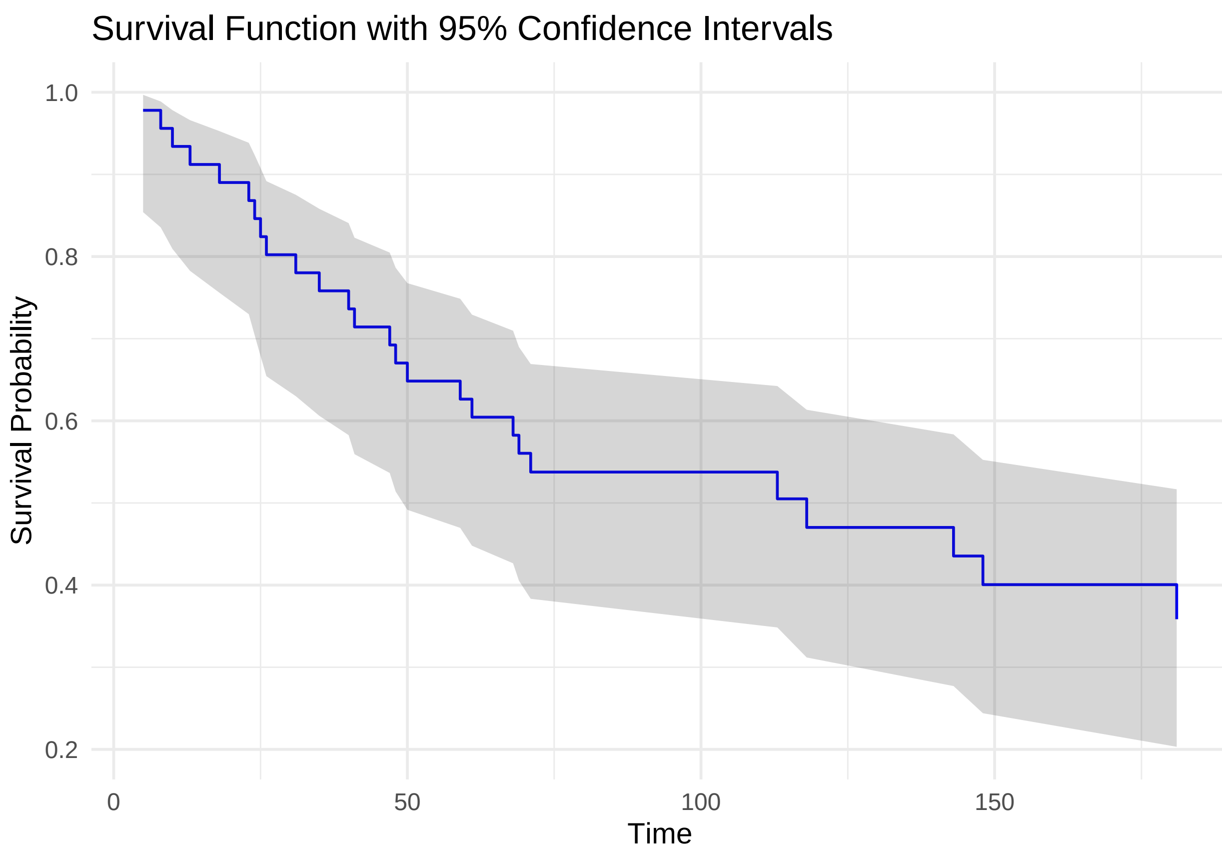
Assuming the $log(\Lambda(t))$ follows normal distribution. The variance of $log(\Lambda(t))$ is estimated as above. As $S(t) = exp(-\Lambda(t))$, thus $P\{\hat{\Lambda}(t) \in (a, b)\} = P\{\hat{S}(t) \in (e^{-b}, e^{-a})\} = 0.95$ And the CI are computed as follow:

```
V_estimate_temp = ((data_sorted$delta)/(data_sorted$y^2)) %>% 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 0.996875   0.978023 0.854057
## 2     8 0.988822   0.956046 0.835484
## 3    10 0.978247   0.934069 0.809356
## 4    13 0.966066   0.912092 0.782513
## 5    18 0.952726   0.890115 0.755935
## 6    23 0.938487   0.868138 0.729822
## 7    24 0.923514   0.846161 0.704200
## 8    25 0.907922   0.824184 0.679051
## 9    26 0.891792   0.802207 0.654344
## 10   31 0.875188   0.780230 0.630050
## 11   35 0.858155   0.758253 0.606140
## 12   40 0.840731   0.736276 0.582593
## 13   41 0.822945   0.714300 0.559386
## 14   47 0.804822   0.692323 0.536503
## 15   48 0.786382   0.670346 0.513930
## 16   50 0.767638   0.648370 0.491655
## 17   59 0.748606   0.626393 0.469669
## 18   61 0.729295   0.604417 0.447962
## 19   68 0.709713   0.582440 0.426531
## 20   69 0.689866   0.560464 0.405369
## 21   71 0.669145   0.537591 0.383341
## 22 113 0.642327   0.505020 0.348429
## 23 118 0.613435   0.470206 0.311868
## 24 143 0.583468   0.435392 0.277117
## 25 148 0.552470   0.400580 0.244019
## 26 181 0.516658   0.358455 0.203129
```

```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)

Quartile of 0.75 is unable to compute due to the estimation of survival function ends in 0.358. However, the quartiles of 0.25 and 0.5 are computed as follow:

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

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

```
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
```