Assignment 6

(1) $$P(T > t_j \mid T > t_{j-1}) = \frac{P(T > t_j \cap T > t_{j-1})}{P(T > t_{j-1})}$$

$$= \frac{P(T > t_j)}{P(T > t_{j-1})} = \frac{S(t_j)}{S(t_{j-1})}$$

(2) $$\prod_{j=1}^{k} p_j = \prod_{j=1}^{k} \frac{S(t_j)}{S(t_{j-1})}$$

$$= \frac{S(\cancel{t_1})}{\underbrace{S(t_0)}_{=1}} \cdot \frac{S(\cancel{t_2})}{S(\cancel{t_1})} \cdot \frac{S(\cancel{t_3})}{S(\cancel{t_2})} \cdots \frac{S(\cancel{t_{k-1}})}{S(\cancel{t_{k-2}})} \cdot \frac{S(t_k)}{S(\cancel{t_{k-1}})}$$

$$= S(t_k)$$

(3) (a) $E(\hat{p}) = p$, $Var(\hat{p}) = \frac{p(1-p)}{n}$ $\left(= \frac{\sigma^2}{n} \right)$

(b) $\hat{p} \dot{\sim} N\left(p, \frac{p(1-p)}{n}\right)$

(4) $\hat{p}_j = 1 - \frac{d_j}{n_j}$ (one minus conditional probability of death)

$= \frac{n_j - d_j}{n_j}$ The proportion who lived

~~(5) (a) $\hat{p} \dot{\sim} N\left(p_j, \frac{p_j(1-p_j)}{n_j}\right)$~~

(5) (a) $\hat{p}_j \dot{\sim} N\left(p_j, \frac{p_j(1-p_j)}{n_j}\right)$

(5b) Using the one-variable delta method, $\hat{P}_j$ should be asymptotically normal with expected value P_j. To get the variance, with $g(x) = \log x$

$g'(x) = \frac{d}{dx} \log x = \frac{1}{x}$, so asymptotic variance is

$\frac{1}{P_j^2} \frac{P_j(1-P_j)}{n_j} = \frac{1-P_j}{n_j P_j}$, and (loosely)

$$\log \hat{P}_j \mathrel{\dot\sim} N\left(\log P_j, \frac{1-P_j}{n_j P_j}\right)$$

(6) (a) $\log \hat{S}(t) = \sum_{t_j \le t} \log \hat{P}_j$

(b) $E(\log \hat{S}(t)) \approx \sum_{t_j \le t} E(\log \hat{P}_j)$

$= \sum_{t_j \le t} \log P_j$.

(c) $Var(\log \hat{S}(t)) \approx \sum_{t_j \le t} Var(\log \hat{P}_j)$

$= \sum_{t_j \le t} \frac{1-P_j}{n_j P_j}$, so we guess

(d) $\log \hat{S}(t) \mathrel{\dot\sim} N\left(\overbrace{\sum_{t_j \le t} \log P_j}^{\text{or } \log(S(t))}, \sum_{t_j \le t} \frac{1-P_j}{n_j P_j}\right)$

(7) Have asymptotic distribution of $\log \hat{S}(t)$. Want distribution of $g(\log \hat{S}(t)) = \operatorname{Exp}\{\log \hat{S}(t)\}$. Using one-variable delta method $g'(x) = g(x)$, so

$$\hat{S}(t) \,\dot{\sim}\, N\left(S(t),\ S(t)^2 \sum_{t_j \le t} \frac{1-p_j}{n_j p_j}\right)$$

(8) Standard error is the square root of the estimated variance. Estimated variance is

$$\hat{S}(t)^2 \sum_{t_j \le t} \frac{1-\hat{p}_j}{n_j \hat{p}_j} = \hat{S}(t)^2 \sum_{t_j \le t} \frac{d_j / n_j}{\cancel{n_j} \frac{n_j - d_j}{\cancel{n_j}}}$$

$$= \hat{S}^2(t) \sum_{t_j \le t} \frac{d_j}{n_j(n_j - d_j)}$$, and standard error is

$$\hat{S}(t) \sqrt{\sum_{t_j \le t} \frac{d_j}{n_j(n_j - d_j)}}$$, where

$$\hat{S}(t) = \prod_{t_j \le t} \hat{p}_j = \prod_{t_j \le t} \left(1 - \frac{d_j}{n_j}\right)$$

(9)

t_i	n_i	d_i	$\hat{p}_i$	$\hat{S}(t_i)$
0	100	0	1	1
2	100	15	85/100	0.85
4	83	5	78/83	0.7988
5	73	10	63/73	0.6894

(10) 5 observations were censored between $t=4$ & $t=5$

(12) (a) For an exponential distribution get median by setting $S(t) = \frac{1}{2}$, so $e^{-\lambda t} = \frac{1}{2} \Rightarrow -\lambda t = \log \frac{1}{2}$

$$-\lambda t = \log(2^{-1}) = -\log 2 \Rightarrow \lambda t = \log 2 \Rightarrow t = \frac{\log 2}{\lambda}$$

And MLE of median is $\frac{\log 2}{\hat{\lambda}}$

To get a confidence interval for the median, either use the delta method or transform the confidence interval for λ.

- Delta method. From Question 1b of assignment 5,

$$\mathrm{Var}(\hat{\lambda}) \approx \frac{\lambda^2}{\sum_{i=1}^n \delta_i} = v_n \quad . \quad g(x) = (\log 2)\, x^{-1}$$

$$g'(x) = -(\log 2)\, x^{-2} = -\frac{\log 2}{x^2}$$, so asymptotic variance of median is $g'(\lambda)^2 v_n = \frac{(\log 2)^2}{\lambda^4} \frac{\lambda^2}{\sum \delta_i}$

$$= \frac{(\log 2)^2}{\lambda^2 \sum_{i=1}^n \delta_i}$$, and standard error is

$$se_{med} = \frac{\log 2}{\hat{\lambda}\sqrt{\sum \delta_i}}$$

$$se_{med} = \frac{\log 2}{\hat{\lambda}\sqrt{\sum \delta_i}}$$

- Transform confidence interval.

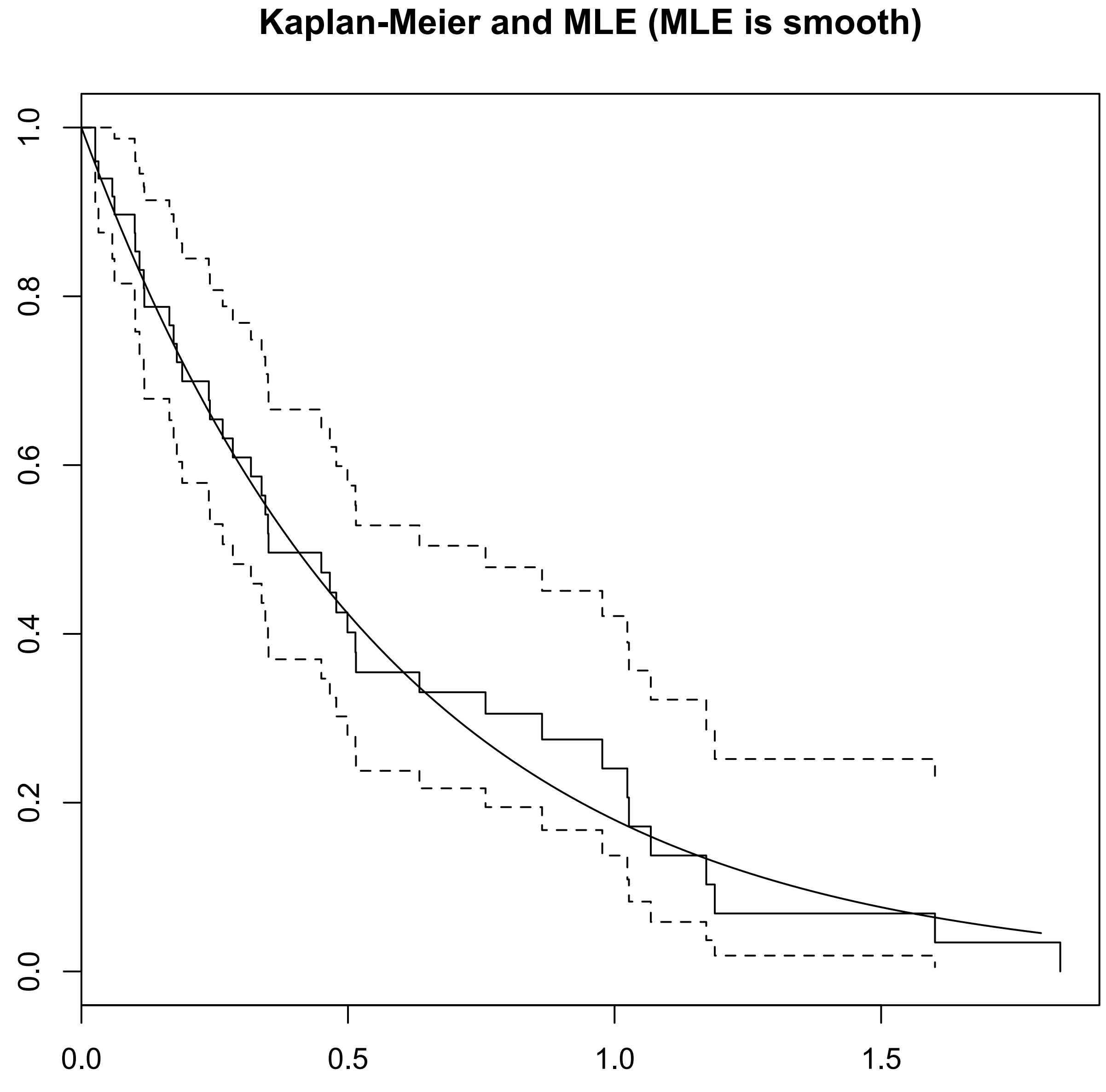
$$0.95 \approx P(A < \lambda < B) = P\left(\frac{1}{A} > \frac{1}{\lambda} > \frac{1}{B}\right)$$

$$= P\left(\frac{\log 2}{B} < \frac{\log 2}{\lambda} = Med < \frac{\log 2}{A}\right)$$

```
R version 4.2.2 (2022-10-31) -- "Innocent and Trusting"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin17.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.79 (8160) x86_64-apple-darwin17.0]

[Workspace restored from /Users/brunner/.RData]
[History restored from /Users/brunner/.Rapp.history]

> # A6
>
> rm(list=ls()); options(scipen=999)
> exdata = read.table("http://www.utstat.utoronto.ca/brunner/data/legal/expo.data2.txt")
> head(exdata); Time = exdata$Time; Uncensored = exdata$Uncensored
   Time Uncensored
1 0.179          0
2 1.024          1
3 0.189          1
4 0.345          1
5 0.977          1
6 0.241          1
>
> # 11(a)
> # install.packages("survival",dependencies=TRUE) # Only need to do this once
> library(survival)
> y = Surv(Time,Uncensored)
> km = survfit(y ~ 1); km
Call: survfit(formula = y ~ 1)

      n events median 0.95LCL 0.95UCL
[1,] 50     40  0.351   0.284   0.758
> sumkm = summary(km); sumkm
Call: survfit(formula = y ~ 1)

  time n.risk n.event survival std.err lower 95% CI upper 95% CI
 0.026     50       2   0.9600  0.0277      0.90719        1.000
 0.032     47       1   0.9396  0.0338      0.87557        1.000
 0.058     44       1   0.9182  0.0392      0.84448        0.998
 0.062     43       1   0.8969  0.0437      0.81511        0.987
 0.100     41       1   0.8750  0.0478      0.78610        0.974
 0.101     40       1   0.8531  0.0514      0.75811        0.960
 0.109     39       1   0.8312  0.0545      0.73095        0.945
 0.117     38       1   0.8094  0.0573      0.70448        0.930
 0.118     37       1   0.7875  0.0598      0.67860        0.914
 0.165     36       1   0.7656  0.0620      0.65324        0.897
 0.173     35       1   0.7437  0.0640      0.62835        0.880
 0.179     34       1   0.7219  0.0657      0.60388        0.863
 0.189     32       1   0.6993  0.0674      0.57888        0.845
 0.239     31       1   0.6768  0.0689      0.55428        0.826
 0.241     30       1   0.6542  0.0702      0.53007        0.807
```

```
 0.265     29       1   0.6316  0.0713      0.50621        0.788
 0.284     28       1   0.6091  0.0723      0.48270        0.769
 0.318     27       1   0.5865  0.0730      0.45951        0.749
 0.338     26       1   0.5640  0.0736      0.43665        0.728
 0.345     25       1   0.5414  0.0741      0.41409        0.708
 0.350     24       1   0.5188  0.0743      0.39184        0.687
 0.351     23       1   0.4963  0.0744      0.36988        0.666
 0.450     21       1   0.4727  0.0745      0.34697        0.644
 0.466     20       1   0.4490  0.0745      0.32441        0.622
 0.478     19       1   0.4254  0.0742      0.30220        0.599
 0.499     18       1   0.4018  0.0738      0.28035        0.576
 0.514     17       1   0.3781  0.0731      0.25886        0.552
 0.515     16       1   0.3545  0.0723      0.23774        0.529
 0.634     15       1   0.3309  0.0712      0.21701        0.504
 0.758     13       1   0.3054  0.0701      0.19473        0.479
 0.864     10       1   0.2749  0.0694      0.16752        0.451
 0.977      8       1   0.2405  0.0687      0.13736        0.421
 1.024      7       1   0.2061  0.0670      0.10907        0.390
 1.027      6       1   0.1718  0.0640      0.08277        0.357
 1.068      5       1   0.1374  0.0597      0.05864        0.322
 1.172      4       1   0.1031  0.0538      0.03708        0.287
 1.188      3       1   0.0687  0.0455      0.01876        0.252
 1.601      2       1   0.0344  0.0333      0.00514        0.230
 1.836      1       1   0.0000     NaN           NA           NA
>
> # (b) S-hat(0:062) = 0.8969
>
> # (c) and (d)
> # p-hat     1       2       3       4
>          48/50 * 46/47 * 43/44 * 42/43
[1] 0.8968665
>
>
> # (e) This way of doing it requires you to realize sumkm is a list.
> #     The numbers could also be entered by hand.
> #     Try sumkm[1], sumkm[2] etc. to find out.
> n = sumkm$n.risk; d = sumkm$n.event; Shat = sumkm$surv
> Shat[4] # Another way to answer (b)
[1] 0.8968665
>
> se4 = Shat[4] * sqrt(sum(d[1:4]/(n[1:4]*(n[1:4]-d[1:4])))); se4 # 0.04373656
[1] 0.04373656
>
> # This should agree with the hand" calculation. Get 0.04373819
> 0.8969 * sqrt(sum(d[1:4]/(n[1:4]*(n[1:4]-d[1:4]))))
[1] 0.04373819
>
> # (f)
> plot(km)
>
>
> # 12
> # (a)
> # First, from last week (Assignment 5),
>
> # A5 Q2a) MLE
> lambdahat = sum(Uncensored)/sum(Time);  lambdahat
[1] 1.717107
>
> # A5 Q2b Estimated asymptotic variance
> vhat = lambdahat^2 / sum(Uncensored);  vhat # Estimated asymptotic variance
[1] 0.07371138
> se = sqrt(vhat); se
[1] 0.2714984
>
```

```
> # A5 Q2c) 95% CI for lambda: 0.95 =~ P(A < lambda < B)
> A = lambdahat - 1.96*se; B = lambdahat + 1.96*se
> c(A,B)
[1] 1.184970 2.249244
>
> # Now get to 12a from A6
> # Median of an exponential is log(2)/lambda
> medhat = log(2)/lambdahat; medhat
[1] 0.4036716
>
> # There are two ways to get a CI ...
>
> # Delta method
> semed = log(2)/(lambdahat*sqrt(sum(Uncensored))); semed
[1] 0.06382608
> lower95 = medhat - 1.96*semed; upper95 = medhat + 1.96*semed
> c(lower95,upper95)
[1] 0.2785725 0.5287707
>
> # Transform CI for lambda
> c(log(2)/B, log(2)/A)
[1] 0.3081690 0.5849492
>
>
>
> # (b)  Add MLE of S(t) to plot
> t = seq(from=0,to=1.8,length=101)
> Shat = exp(-lambdahat*t)
> lines(t,Shat)
> title('Kaplan-Meier and MLE (MLE is smooth)')
>
>
```


Kaplan-Meier and MLE (MLE is smooth)
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.5
1.0
1.5