STAT641.HMW01

Abbas Jalili

2/14/2022

```
#loading the libararies
pacman::p_load(bootstrap, tidyverse, boot, gtools, knitr)
```

Q.1

Aflatoxin residues in peanut butter: In actual testing, 12 lots of peanut butter had aflatoxin residues in parts per billion of 4.94, 5.06, 4.53, 5.07, 4.99, 5.16, 4.38, 4.43, 4.93, 4.72, 4.92,and 4.96.

```
#loading the data:
aflatoxin <- c(4.94, 5.06, 4.53, 5.07, 4.99, 5.16, 4.38, 4.43, 4.93, 4.72, 4.92, 4.96)
n <- length(aflatoxin)
```

1.a):

How many possible bootstrap resamples of these data are there?

To calculate the total possible amount of bootstrap reasample of size n, we can use the combination():

```
samples <- combinations(12, 12, aflatoxin, repeats.allowed = TRUE)
dim(samples)
```

```
## [1] 1352078      12
```

There are 1,352,078 possible bootstrap resample of these data with size of n = 12.

1.b):

Using R and the sample() function, or a random number generator, generate five resamples of the integers from 1 to 12.

```
set.seed(1234)
boot_samp <- lapply(1:5, function(x) sample(1:12, 12, replace = TRUE))
boot_samp
```

```
## [[1]]
##  [1] 12 10  6  5 12  9  5  6  4  2  7  6
##
## [[2]]
##  [1] 10  6  4  8  4  4  5  8  4  8  3  4
##
## [[3]]
##  [1] 10  5  2  8 11  4 12  3  7  9  3  6
##
## [[4]]
##  [1]  4  8 10 11  2  5  6  1  6  8  3  6
##
## [[5]]
##  [1]  1  1  9  8 10  1 11  8 10  6  3  9
```

- I generate the five resamples of the integers from 1 to 12 with sample() as you can see above.

1.c):

For each of the resamples in b, find the mean of the corresponding elements of the aflatoxin data set. Print out the 5 bootstrap means.

```
resamp_mean <- sapply(boot_samp, function(x) mean(aflatoxin[x]))
resamp_mean
```

```
## [1] 4.961667 4.836667 4.806667 4.880833 4.799167
```

- Now I got the mean for each resamples to the corresponding value from the original dataset.

1.d):

Find the mean of the resample means. Compare this with the mean of the original data set.

```
mean(resamp_mean)
```

```
## [1] 4.857
```

```
mean(aflatoxin)
```

```
## [1] 4.840833
```

- Next comparing the mean value of the five resamples and the original data set. As we can see, both mean values are pretty close to each others.

1.e):

Find the minimum and the maximum of the five resample means. This is a crude bootstrap confidence interval on the mean. (If you had used 1000 resamples, and used the 25th and 975th largest means, this would have given a reasonable 95% confidence interval.)

```
summary(resamp_mean)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.799   4.807   4.837   4.857   4.881   4.962
```

- The 1st and 3rd quantile of the five resample means are 4.807 and 4.881. Also the min and max of the five resample means are 4.799 and 4.962.

Q.2:

Airline accidents: According to the U.S. National Transportation Safety Board, the number of airline accidents by year from 1983 to 2006 were:

```
#loading the data:

airline <- c(23, 16, 21, 24, 34, 30, 28, 24, 26, 18, 23, 23, 36,
37, 49, 50, 51, 56, 46, 41, 54, 30, 40, 31)
```

2.a):

For the sample data, compute the mean and its standard error (use sd()), and the median.

```
print(paste("The mean of the sample data is:", round(mean(airline), 2)))
```

```
## [1] "The mean of the sample data is: 33.79"
```

```
print(paste("The median of the sample data is:",round(median(airline), 2)))
```

```
## [1] "The median of the sample data is: 30.5"
```

```
print(paste("The standard error of the sample data is:",round(sd(airline), 2)))
```

```
## [1] "The standard error of the sample data is: 12.06"
```

2.b):

Using R , compute bootstrap estimates of the mean and median with estimates of their standard errors, using B = 1000. (Use set.seed(1234) to allow your results to be reproduced.) Also, compute the median of the bootstrap estimates of median.

```
set.seed(1234)

B <- 1000

boot_mean <- c()
boot_median <- c()


for (i in 1:B) {

a = sample(airline, 24, replace = TRUE)

boot_mean[i] = mean(a)
boot_median[i] = median(a)

}


print(paste("The bootstrap estimate of the mean is:",
            round(mean(boot_mean), 2)))
```

```
## [1] "The bootstrap estimate of the mean is: 33.81"
```

```
print(paste("The bootstrap estimate of the median is:",
            round(mean(boot_median), 2)))
```

```
## [1] "The bootstrap estimate of the median is: 31.51"
```

```
print(paste("The bootstrap estimates of standard errors of the mean is:",
            round(sd(boot_mean), 2)))
```

```
## [1] "The bootstrap estimates of standard errors of the mean is: 2.46"
```

```
print(paste("The bootstrap estimates of standard errors of the median is:",
            round(sd(boot_median), 2)))
```

```
## [1] "The bootstrap estimates of standard errors of the median is: 3.76"
```

```
print(paste("The median of the bootstrap estimates of median is:",
            round(median(boot_median), 2)))
```

```
## [1] "The median of the bootstrap estimates of median is: 30.5"
```

2.c):

Compare parts (a) and (b). How do the estimates compare?

Mean of the sample data and bootstrap estimate of the mean are very close (33.79167 and 33.81183). The mean of the bootstrap median is 31.5065 , but the median of the bootstrap estimates of median is 30.5 which is equal to the sample data median. The sample data has a bigger standard error value equal to 12.06497. The bootstrap estimates of standard errors of the mean is 2.463745 and the bootstrap estimates of standard errors of the median is 3.75637.

Q.3:

Input the mouse example data in R. Complete this question using the sample() function in R .

```
treat <- mouse.t
ctrl <-  mouse.c

t1 <- length(treat)
t2 <- length(ctrl)

Group <- as.factor( c(rep(1, t1), rep(2, t2)))

Group <- as.factor(ifelse(Group == 1, "Treatment", "Control"))


mouse <- data.frame(Group, Survival_time = c(treat, ctrl))

# Create an another variable called mouse1 which shuffles the original dataset.
mouse1 <- mouse %>% sample_n(size = 16)
kable(head(mouse1))
```

Group	Survival_time
Control	31
Control	146
Treatment	141
Control	104
Control	52
Control	46

```
# Exploratory data analysis
kable(mouse1 %>%
  group_by(Group) %>%
  summarise(
    count = n(),
    Mean = round(mean(Survival_time),2)
  ))
```

Group	count	Mean
Control	9	56.22
Treatment	7	86.86

*The first table shows the head() of the dataset, and the second one shows the count and mean values for each groups.

3.a):

For the treatment group alone, find the bootstrap estimates for the mean and its standard error with B = 50, 100, 200, 500, 1000 and 10,000.

```
set.seed(1234)

result <- c()

boott_mean <- c()
boott_sd <- c()


B <- c(50, 100, 200, 500, 1000, 10000)

for(j in 1:length(B)) {
  for(i in 1:B[j]) {

    result[i] = mean(sample(mouse.t, length(mouse.t), replace = TRUE))
  }
  boott_mean[j] <- mean(result)
  boott_sd[j] <- sd(result)
}

kable(round(boott_mean, 2), col.names = c("Boot_t_Mean"))
```

Boot_t_Mean
86.98
83.38
87.08
87.87
86.93
87.19

```
kable(round(boott_sd, 2), col.names = c("Boot_t_Sd"))
```

Boot_t_Sd
27.42
21.43
20.66
23.69
23.52
23.67

- Tables above are representing the bootstrap estimates for the mean and its standard errors.

3.b):

```
set.seed(1234)

result <- c()


boott_median <- c()
boottm_sd <- c()


B <- c(50, 100, 200, 500, 1000, 10000)

for(j in 1:length(B)) {
  for(i in 1:B[j]) {

    result[i]= median(sample(mouse.t, length(mouse.t), replace = TRUE))
  }

  boott_median[j] <- mean(result)
  boottm_sd[j] <- sd(result)
}


kable(round(boott_median, 2), col.names = c("Boot_t_Median"))
```

Boot_t_Median
82.34
80.49
81.39
80.90
79.80
79.77

```
kable(round(boottm_sd, 2), col.names = c("Boot_t_Sdm"))
```

Boot_t_Sdm
40.94
35.85
34.46
38.68
38.26
37.98

- Tables above are representing the bootstrap estimates for the median and its standard errors.

3.c):

Create a table to represent the simulation results from parts (a) and (b).

```
#creating the table of the results:

bootstrap <- data.frame(B, Bootstrap_Mean = round(boott_mean, 2) ,
                        SD_mean = round(boott_sd, 2) ,
                        Bootstrap_Median = round(boott_median, 2),
                        SD_median = round(boottm_sd, 2))

kable(bootstrap)
```

B	Bootstrap_Mean	SD_mean	Bootstrap_Median	SD_median
50	86.98	27.42	82.34	40.94
100	83.38	21.43	80.49	35.85
200	87.08	20.66	81.39	34.46
500	87.87	23.69	80.90	38.68
1000	86.93	23.52	79.80	38.26
10000	87.19	23.67	79.77	37.98

- Table above is the combination of the two previous tables.

3.d):

Now using bootstrap methods find an estimate for the median and its standard error for the control group.

```
set.seed(1234)

result <- c()


bootc_median <- c()
bootcm_sd <- c()


B <- c(50, 100, 200, 500, 1000, 10000)

for(j in 1:length(B)) {
  for(i in 1:B[j]) {

    result[i]= median(sample(mouse.c, length(mouse.c), replace = TRUE))
  }

  bootc_median[j] <- mean(result)
  bootcm_sd[j] <- sd(result)
}


kable(round(bootc_median, 2), col.names = c("Boot_c_Median"))
```

Boot_c_Median
42.98
45.78
45.15
45.73
45.86
45.52

```
kable(round(bootcm_sd, 2), col.names = c("Boot_c_Sd"))
```

Boot_c_Sd
7.29
12.73
10.31
13.18
12.30
12.32

- Table for control group bootstrap estimate the median and its standard error.

3.e):

Use results from parts (b) and (d) to find the estimated standard error for the difference between the medians.

```
bootd_median <- boott_median - bootc_median

sd(bootd_median)
```

```
## [1] 2.005117
```

- The standard error for difference between the medians is 2.005.

Q.4:

Let $X_1, X_2, ..., X_n \sim Gamma\ (2, \theta)$, where θ is unknown number. How do we estimate the standard error of the MLE estimator of θ ? Use boot package and generate your original sample with:

```
set.seed(123)

x <- rgamma(200, shape = 2, scale = 5) #original data

mle_gamma <- function(x)mean(x)/2

gboot <- function(x, mle)rgamma(length(x), shape = 2, scale = mle)

b_gamma <- boot(x, mle_gamma, R=1000, sim = "parametric", ran.gen =gboot, mle = mean(x)/2)

b_gamma
```

```
##
## PARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = x, statistic = mle_gamma, R = 1000, sim = "parametric",
##     ran.gen = gboot, mle = mean(x)/2)
##
##
## Bootstrap Statistics :
##     original      bias    std. error
## t1* 4.578726 0.005143687   0.2135533
```

- I used the boot function and Parametric bootstrap method to calculate the standard error of the MLE estimator of θ. As we know the MLE estimator for gamma distribution is :

$$X_1, X_2, ..., X_n \sim Gamma\ (k, \theta)$$

$$\hat{\theta} = \frac{\sum_{i=1}^{N} x_i}{kN}$$

where k = shape in rgamma() function.

- The standard error of the MLE is equal to 0.213.