R Homework 9

Fisher Ankney

November 13th, 2018

Statistics 5193

Question 1a

Functions:

- Write a function called boot.mean that takes a sample from a vector called my.datawith replacement and returns the trimmed mean from the sampled vector. For full credit
- The inputs for the function are my.data and trim.perecent.
- my.data should have no default value.
- strim.percent has default value of 0.05 and is the percent trimmed from each tail

```r
library(readxl)
StudentData <- read_excel("~/Documents/data_science/r_stat_5193/data/StudentData.xlsx")
```

```r
boot.mean <- function(my.data, trim.percent = 0.05) {
  dat.sample <- sample(my.data, replace = T)
  mean(dat.sample, trim = trim.percent)
}
```

Question 1b

Get a bootstrapped sample 5% and 10% trimmed mean for the HSClass variable in the Social Media data set.

```r
# the use of attach is not recommended due to the convoluted nature of the
# function, it reduces reproducibility, therefore I will reference the
# dataframe in the suggested code

set.seed(1)
boot.mean(StudentData$HSClass)
```

```
## [1] 309
```

```r
boot.mean(StudentData$HSClass, trim.percent = .10)
```

```
## [1] 344
```

Question 2a

For loop:

- Write a for loop that runs boot.mean(HSClass) 10,000 times and save the means in a vector called my.boot.sample. Run the code set.seed(1) before your loop and print the mean and standard deviation of the vector to the console.

```r
my.boot.sample <- vector('double', length = 10000)

set.seed(1)
for (i in 1:10000) {
  my.boot.sample[i] <- boot.mean(StudentData$HSClass)
}

mean(my.boot.sample)
```

```
## [1] 310.1323
```

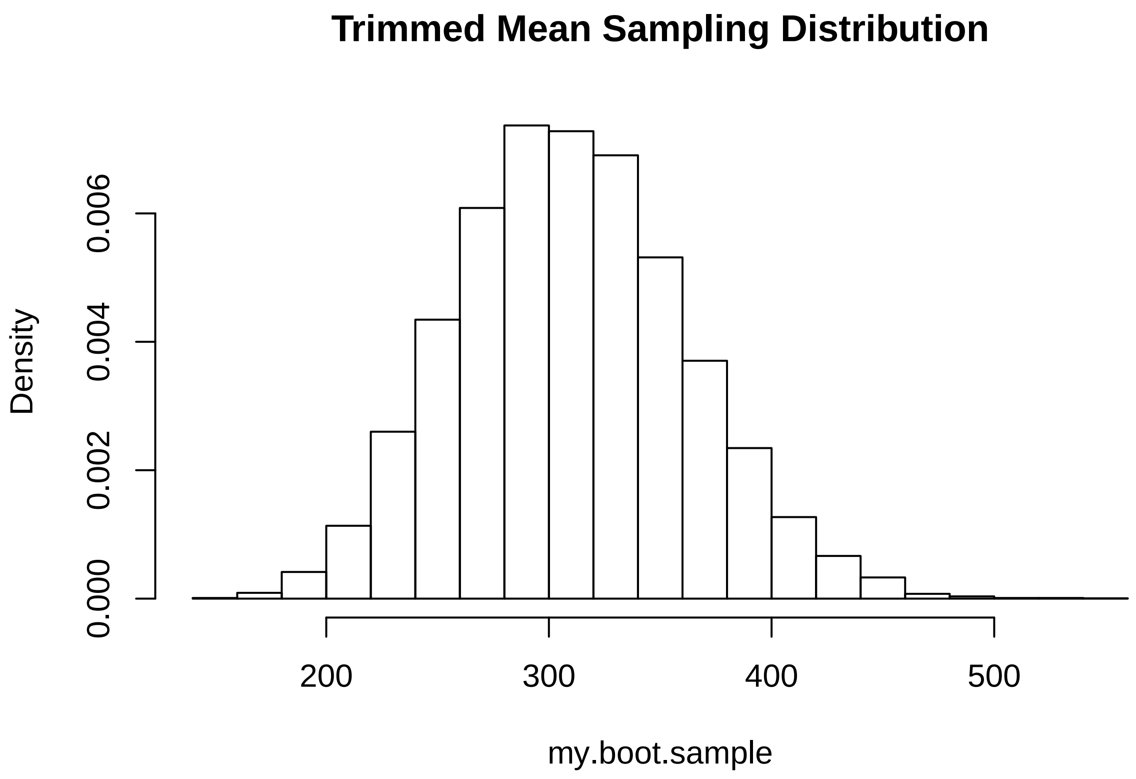
```r
sd(my.boot.sample)
```

```
## [1] 51.98851
```

Question 2b

Construct a histogram of the 10,000 trimmed sample means with title "Trimmed Mean Sampling Distribution". Use the probability=T option in the histogram.

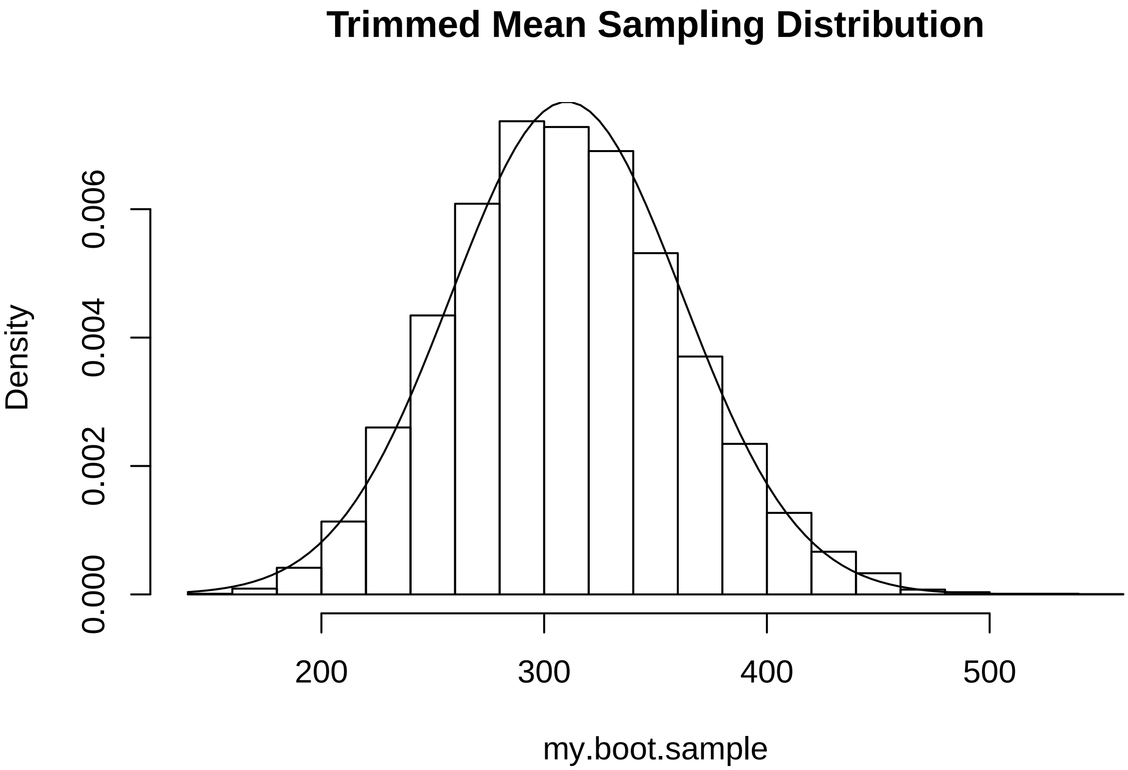
```r
hist(my.boot.sample,
     probability = T,
     main = "Trimmed Mean Sampling Distribution")
```


Trimmed Mean Sampling Distribution
Density
0.000
0.002
0.004
0.006
200
300
400
500
my.boot.sample

Question 2c

Add a normal curve to the histogram that has mean equal to the mean and standard deviation computed in part a.

```
hist(my.boot.sample,
     probability = T,
     main = "Trimmed Mean Sampling Distribution")

curve(dnorm(x, mean = mean(my.boot.sample), sd = sd(my.boot.sample)),
      add = T)
```



Question 2d

Get the 2.5th percentile and the 97.5th percentile of my.boot.sample using the quantile function. Note: this is a 95% bootstrapped confidence interval of the 5% trimmed mean for HS Class. Take STAT 5093 for more on the theory of bootstrapping.

```
quantile(my.boot.sample, c(0.025, 0.975))
```

```
##     2.5%    97.5%
## 214.9068 417.5470
```

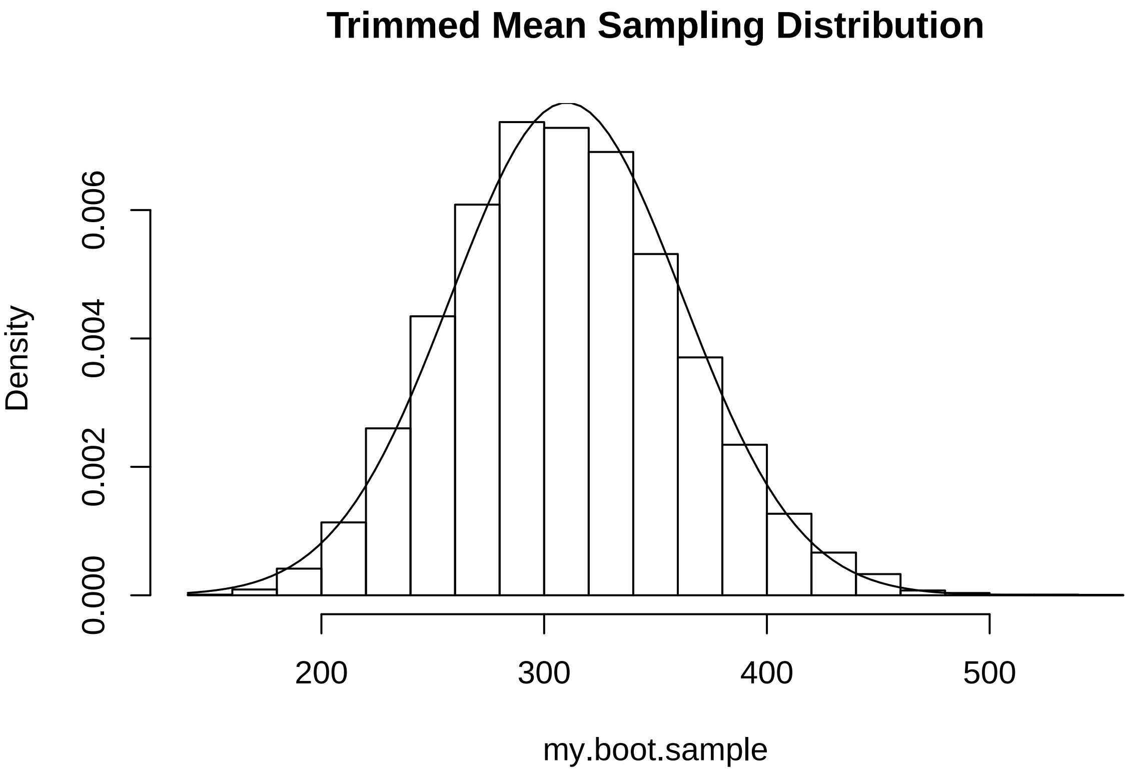
Question 3a

```r
boot.ci <- function(my.data, trim.percent = 0.05, plot.it = T) {

# recreate boot.mean function
  boot.mean <- function(my.data, trim.percent = trim.percent) {
    dat.sample <- sample(my.data, replace = T)
    mean(dat.sample, trim = trim.percent)
  }

# create empty vector to store results
  my.boot.sample <- vector('double', length = 10000)

# loop
  for (i in 1:10000) {
    my.boot.sample[i] <- boot.mean(my.data, trim.percent = trim.percent)
  }

# save mean and sd for plotting
data.mean <- mean(my.boot.sample)
data.sd <- sd(my.boot.sample)

# return confidence interval
  print(quantile(my.boot.sample, c(0.025, 0.975)))

# conditional plot.it
  if (plot.it == T) {
     hist(my.boot.sample,
           probability = T,
           main = "Trimmed Mean Sampling Distribution")

     curve(dnorm(x, mean = data.mean, sd = data.sd), add = T)

  }
}
```

Question 3b

```
# again, I'd rather specify the dataset instead of use attach()

set.seed(1)
boot.ci(StudentData$HSClass)
```

```
##      2.5%     97.5%
## 214.9068 417.5470
```

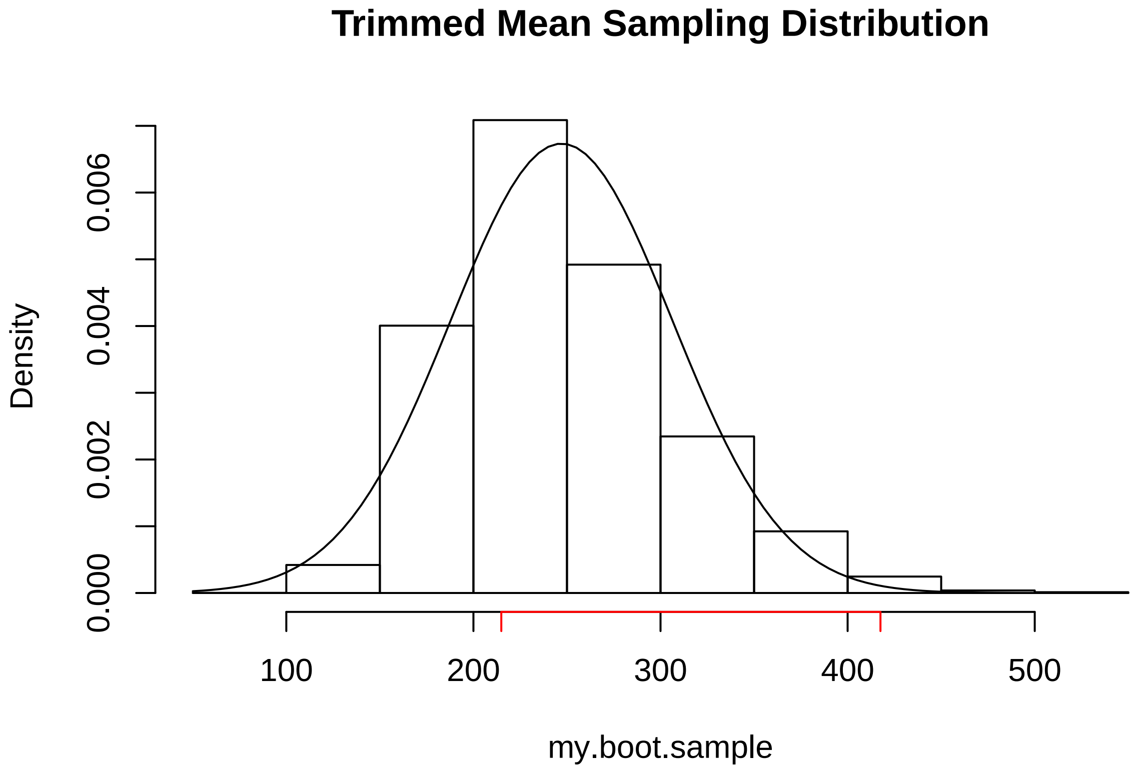


Question 3c

```
# again, I'd rather specify the dataset instead of use attach()

set.seed(1)
boot.ci(StudentData$HSClass, trim.percent=.25)
```

```
##     2.5%    97.5%
## 152.2092 382.6329
```

```
axis(side=1, at=c(214.9068,417.5470 ),
        labels=F, col='red')
```


Trimmed Mean Sampling Distribution
Density
my.boot.sample