HW3

108070023

Q1

1. Given the critical DOI score that Google uses to detect malicious apps (-3.7), what is the probability that a randomly chosen app from Google’s app store will turn off the Verify security feature? (report a precise decimal fraction, not a percentage)

> #1-a

> pnorm(-3.7)

[1] 0.0001077997

b. Assuming there were [~2.2 million apps](https://www.statista.com/statistics/263795/number-of-available-apps-in-the-apple-app-store/) when the article was written, what number of apps on the Play Store did Google expect would maliciously turn off the Verify feature once installed?

> #1-b

> 2500000\*pnorm(-3.7)

[1] 269.4993

Q2

a. The Null distribution of t-values:

i. Visualize the distribution of Verizon’s repair times, marking the mean with a vertical line

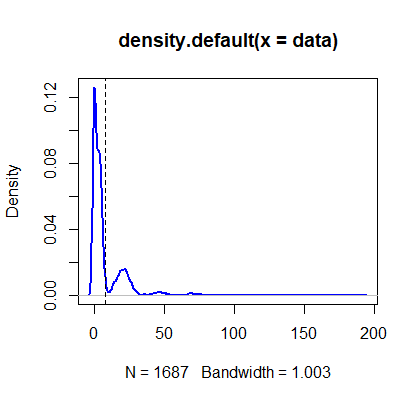
> #(i)

> data<-read.csv("verizon.csv")

> data<-data$Time

> plot(density(data),lwd=2,col="blue")

> abline(v=mean(data),lty="dashed")



> #(iii)

> mean<-mean(data)

> mean

 [1] 8.522009

 > se<-sd(data)/sqrt(length(data))

> interval\_99<-mean+2.58\*c(-se,se)

> interval\_99

[1] 7.593073 9.450946

> #(iv)

> ver\_hyp<-7.6

> mean\_sample<-mean(data)

> mean\_sample

[1] 8.522009

> sd\_sample<-sd(data)

> t\_stat<-(mean\_sample-ver\_hyp)/(sd\_sample/sqrt(length(data)))

> t\_stat

[1] 2.560762

> p\_val<-1-pt(t\_stat,df=length(data)-1)

> p\_val

[1] 0.005265342

vi. What is your conclusion about the advertising claim from this t-statistic, *and why*?

Ans: Since the p-value,which means the probability that the distribution of mean lay outside the 95% confidence interval, is higher than 5%,we can say the claim by Verizon is wrong

(By T-statistics, we can only prove something is not right.)

b. Let’s use bootstrapping on the sample data to examine this problem:

*i. Bootstrapped Percentile:* Estimate the bootstrapped 99% CI of the mean

1. > #2-b
2. > num\_boots<-2000
3. > #(i)
4. > boot<-function(sample){
5. + resample<-sample(sample,length(sample),replace = TRUE)
6. + return (mean(resample))
7. + }
8. > boot<-replicate(num\_boots,boot(data))
9. > interval\_99<-quantile(boot,c(0.025,0.975))
10. > interval\_99
11. 2.5% 97.5%
12. 7.876655 9.243322

ii.*Bootstrapped Difference of Means:*   
What is the 99% CI of the bootstrapped difference between the population mean and the hypothesized mean?

> #(ii)

> ver\_hyp<-7.6

> boot<-function(sample,mean\_hyp){

+ resample<-sample(sample,length(sample),replace = TRUE)

+ return (mean(resample)-mean\_hyp)

+ }

> boot\_diff<-replicate(num\_boots,boot(data,ver\_hyp))

> diff\_interval\_99<-quantile(boot\_diff,prob=c(0.025,0.975))

> diff\_interval\_99

2.5% 97.5%

0.2170448 1.6592330

iii.Bootstrapped t-Interval:

What is 99% CI of the bootstrapped t-statistic?

#(iii)

> boot\_t<-function(sample,mean\_hyp){

+ resample<-sample(sample,length(sample),replace = TRUE)

+ diff<-mean(resample)-mean\_hyp

+ se<-sd(resample)/sqrt(length(resample))

+ return(diff/se)

+ }

> boot\_t<-replicate(num\_boots,boot\_t(data,ver\_hyp))

> t\_interval\_99<-quantile(boot\_t,probs=c(0.025,0.975))

> t\_interval\_99

2.5% 97.5%

0.6631325 4.1348011

iv.Plot separate distributions of all three bootstraps above  
(for ii and iii make sure to include zero on the x-axis)

> #(iv)

> plot(density(boot),col="blue",lwd=2)

> abline(v=mean(boot))

> abline(v=interval\_99,lty="dashed")

> plot(density(boot\_diff),col="blue",lwd=2)

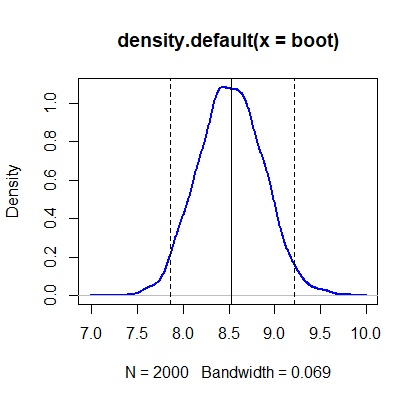
> abline(v=mean(boot\_diff))

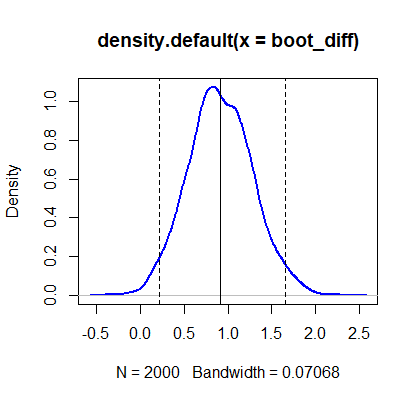
> abline(v=diff\_interval\_99,lty="dashed")

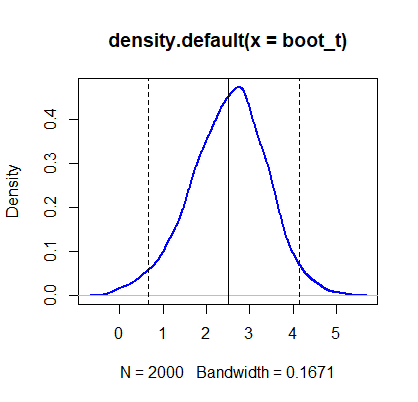
> plot(density(boot\_t),col="blue",lwd=2)

> abline(v=mean(boot\_t))

> abline(v=t\_interval\_99,lty="dashed")>







c. Do the four methods (traditional test, bootstrapped percentile, bootstrapped difference of means, bootstrapped t-Interval) agree with each other on the test?

Ans:Yes,they all prove that the claim by Verizon is wrong. By method(i), the 99% CI didin’t contain 7.6,and by method(ii)(iii), both of their 99% CI didn’t contain 0. Therefore,from the above results, we can overthrow the data claim by Verizon.