Homework 4

Connor Johnson

library(ISLR)

## Warning: package 'ISLR' was built under R version 4.0.3

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.0.4

## Registered S3 methods overwritten by 'tibble':  
## method from   
## format.tbl pillar  
## print.tbl pillar

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(MASS)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library(class)  
library(bootstrap)

## Warning: package 'bootstrap' was built under R version 4.0.3

library(boot)

## Warning: package 'boot' was built under R version 4.0.4

library(broom)

## Warning: package 'broom' was built under R version 4.0.4

##   
## Attaching package: 'broom'

## The following object is masked from 'package:bootstrap':  
##   
## bootstrap

library(coin)

## Warning: package 'coin' was built under R version 4.0.4

## Loading required package: survival

##   
## Attaching package: 'survival'

## The following object is masked from 'package:boot':  
##   
## aml

library(rcompanion)

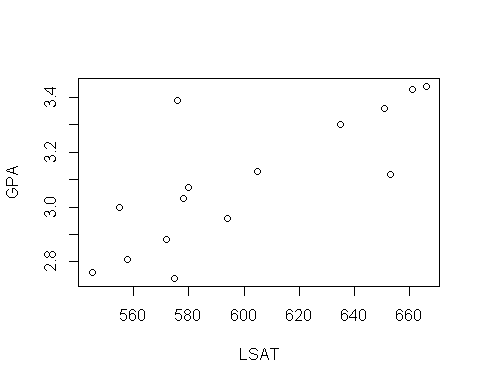
## Warning: package 'rcompanion' was built under R version 4.0.4

## Problem 2

We can use the bootstrap approach to approximate the standard deviation. The bootstrap method works by re-sampling the observations from our data a large number of times. We can then use these samples to create a distribution of estimates for our prediction. Finding the standard deviation of this distribution will be an estimate for the standard deviation of out prediction.

## Problem 3

library(bootstrap)  
data(law)  
plot(law)

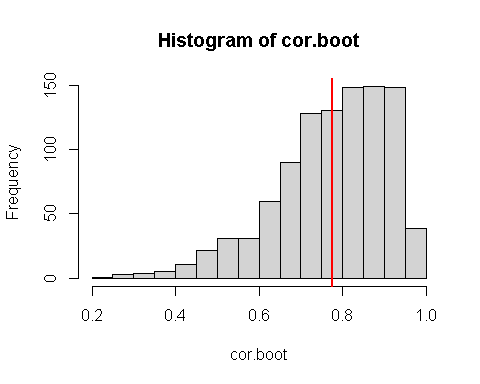


cor(law)

## LSAT GPA  
## LSAT 1.0000000 0.7763745  
## GPA 0.7763745 1.0000000

There does appear to be a strong positive linear relationship between LSAT score and GPA. The correlation between them is 0.7764

nBoot <- 1000  
data(law)  
mean.boot <- rep(0,nBoot)  
cor.boot = numeric(nBoot)  
for (i in 1:nBoot) {  
 xperm1 <- sample(1:nrow(law), size = nrow(law), replace=T)  
 cor.boot[i] = cor(law[xperm1,1],law[xperm1,2])  
}  
hist(cor.boot, breaks=20)  
set.seed(5)  
  
cor0 = cor(law$LSAT,law$GPA)  
abline(v=cor0, col="red", lwd=2)



set.seed(22)  
boot\_corr <- function(data, resample\_vector) {  
 cor(data$x[resample\_vector], data$y[resample\_vector])  
}  
ds <- data.frame(y = law$LSAT, x = law$GPA)  
boots <- boot(ds, boot\_corr, R = 1000)  
boot.ci(boots)

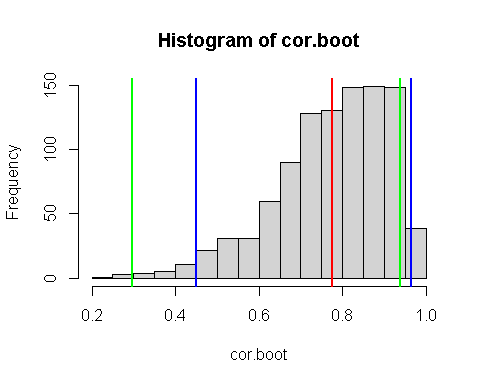
## Warning in boot.ci(boots): bootstrap variances needed for studentized intervals

## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS  
## Based on 1000 bootstrap replicates  
##   
## CALL :   
## boot.ci(boot.out = boots)  
##   
## Intervals :   
## Level Normal Basic   
## 95% ( 0.5133, 1.0480 ) ( 0.5893, 1.1038 )   
##   
## Level Percentile BCa   
## 95% ( 0.4490, 0.9635 ) ( 0.2958, 0.9383 )   
## Calculations and Intervals on Original Scale  
## Some BCa intervals may be unstable

hist(cor.boot, breaks=20)  
print("Bias = -0.004255571")

## [1] "Bias = -0.004255571"

abline(v=cor0, col="red", lwd=2)  
abline(v=0.4490, col="blue", lwd=2)  
abline(v=0.9635, col="blue", lwd=2)  
abline(v=0.2958, col="green", lwd=2)  
abline(v=0.9383, col="green", lwd=2)



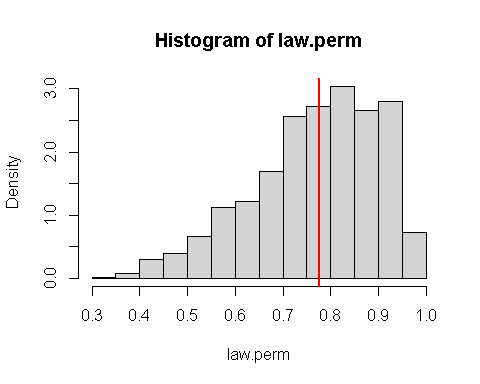
Based on the confidence interval, we can’t reject the null hypothesis that the correlation is equal to 0.5 because 0.5 is within the confidence interval

Based on the bias corrected confidence interval, we cannot reject the null hypothesis that correlation is equal to 0.5 because 0.t is within the confidence interval.

nperm <- 1000  
law.perm <- numeric(nperm)  
for (i in 1:nperm) {  
 ind <- sample(nrow(law), replace = TRUE)  
 law.perm[i] <- cor(law[ind, "LSAT"], law[ind, "GPA"])  
}  
mean(law.perm)

## [1] 0.7679349

hist(law.perm, freq = FALSE,breaks = 20)  
abline(v=cor(law$LSAT,law$GPA),col='red',lwd=2)



law.perm = sort(law.perm)  
print(paste0("95% Confidence Interval: (",law.perm[25],", ",law.perm[975],")"))

## [1] "95% Confidence Interval: (0.467025634372559, 0.953952715586109)"

The confidence interval does not contain zero so we reject the null hypothesis that correlation is equal to zero.

## Problem 4

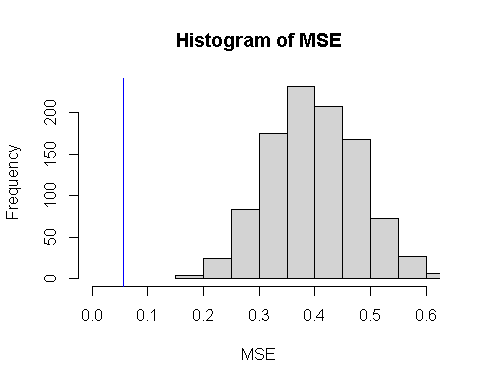
X1 = runif(50)  
X2 = runif(50)  
epsilon = rnorm(50, 0, 0.25)  
Y = X1 + X2 + epsilon

#set.seed(1)  
X1 = runif(50)  
X2 = runif(50)  
epsilon = rnorm(50, 0, 0.25)  
Y = X1 + X2 + epsilon  
reg = lm(Y~X1+X2)  
X1\_test = runif(30)  
X2\_test = runif(30)  
epsilon\_test = rnorm(30, 0, 0.25)  
Y\_test = X1\_test + X2\_test + epsilon\_test  
  
summary(reg)

##   
## Call:  
## lm(formula = Y ~ X1 + X2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.6436 -0.2155 0.0220 0.1792 0.5515   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.15191 0.09174 -1.656 0.104   
## X1 1.15754 0.13694 8.453 5.42e-11 \*\*\*  
## X2 1.17819 0.13121 8.980 9.17e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2743 on 47 degrees of freedom  
## Multiple R-squared: 0.8084, Adjusted R-squared: 0.8002   
## F-statistic: 99.15 on 2 and 47 DF, p-value: < 2.2e-16

test = data.frame(X1\_test,X2\_test)  
colnames(test) <- c("X1", "X2")  
pred = predict(reg, test)  
MSE0 = mean((pred-Y\_test)^2)

perms = 1000  
MSE = numeric(perms)  
for(i in 1:perms){  
 test$X1 = sample(test$X1)  
 test$X2 = sample(test$X2)  
 pred = predict(reg, test)  
 MSE[i] = mean((pred-Y\_test)^2)  
   
}  
hist(MSE, xlim=c(0,0.6))  
abline(v=MSE0, col="blue")

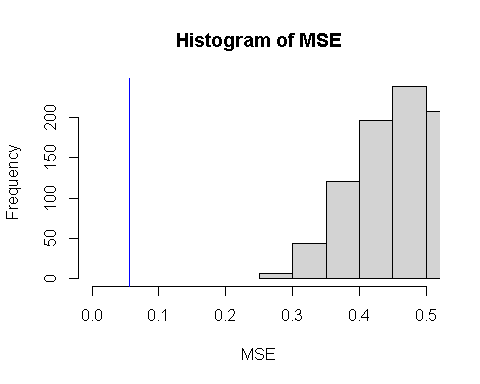


print(length(MSE[MSE < MSE0])/perms)

## [1] 0

We can reject the null hypothesis that and because the p-value is less than 0.05

perms = 1000  
MSE = numeric(perms)  
for(i in 1:perms){  
 test$X2 = sample(test$X2)  
 pred = predict(reg, test)  
 MSE[i] = mean((pred-Y\_test)^2)  
   
}  
hist(MSE, xlim=c(0,0.5))  
abline(v=MSE0, col="blue")



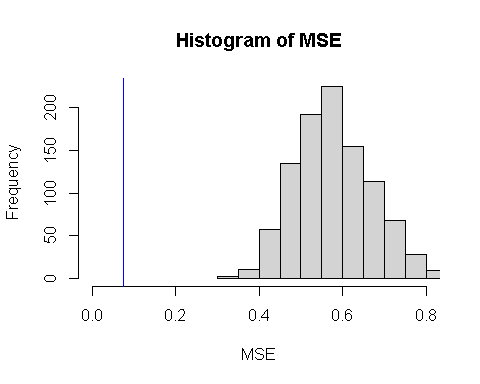
print(length(MSE[MSE < MSE0])/perms)

## [1] 0

We can reject the null hypothesis that because the p-value is less than 0.05

m <- matrix(runif(5000), ncol=10, nrow = 500)  
Xs = data.frame(m)  
colnames(Xs) <- c("X1","X2","X3","X4","X5","X6","X7","X8","X9","X10")  
epsilon\_2 = rnorm(500, 0, 0.25)  
Y = Xs$X1+Xs$X2+Xs$X3+Xs$X4+Xs$X5+Xs$X6+Xs$X7+Xs$X8+Xs$X9+Xs$X10+epsilon\_2  
  
m <- matrix(runif(500), ncol=10, nrow = 50)  
Xs\_test = data.frame(m)  
colnames(Xs) <- c("X1","X2","X3","X4","X5","X6","X7","X8","X9","X10")  
epsilon\_2\_test = rnorm(50, 0, 0.25)  
  
Y\_test = Xs\_test$X1+Xs\_test$X2+Xs\_test$X3+Xs\_test$X4+Xs\_test$X5+Xs\_test$X6+Xs\_test$X7+Xs\_test$X8+Xs\_test$X9+  
 Xs\_test$X10+epsilon\_2\_test  
  
reg = lm(Y~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10,data=Xs)  
  
pred = predict(reg, Xs\_test)  
MSE0 = mean((pred-Y\_test)^2)

X = c(Xs\_test$X8, Xs\_test$X9, Xs\_test$X10)  
perms = 1000  
MSE = numeric(perms)  
for(i in 1:perms){  
 X <- sample(X)  
 Xs\_test$X8 = sample(Xs\_test$X8)  
 Xs\_test$X9 = sample(Xs\_test$X9)  
 Xs\_test$X10 = sample(Xs\_test$X10)  
 pred = predict(reg, Xs\_test)  
 MSE[i] = mean((pred-Y\_test)^2)  
   
}  
hist(MSE, xlim=c(0,0.8))  
abline(v=MSE0, col="blue")



print(length(MSE[MSE < MSE0])/perms)

## [1] 0

We can reject the null hypothesis that , , and because the p-value is less than 0.05

## Problem 5

We would perform a z-test on the two proportions of the group with hypotheses below

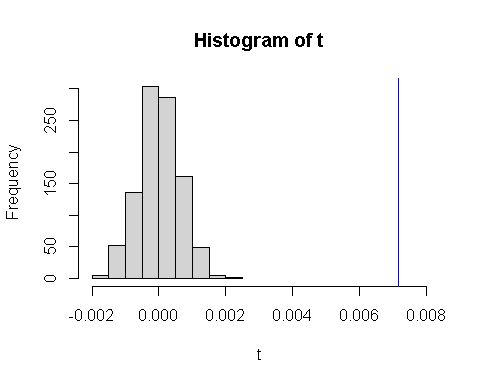
is the placebo group and is the vaccinated group. The z-stat formula is

Plugging in our values, we get with . So, we can reject the null hypothesis which shows that the vaccine is effective.

We would expect that the number of positive cases in each group would be similar to each other.

We can use the difference in proportion between the two groups as our test statistic.

vaccine = numeric(21500)  
placebo = numeric(21500)  
for (i in 1:8){  
 vaccine[i] = 1  
}  
for (i in 1:162){  
 placebo[i] = 1  
}  
t\_star = mean(placebo)-mean(vaccine)  
perms = 1000  
t = numeric(perms)  
for(i in 1:1000){  
 X = numeric(43000)  
 idx = sample(1:length(X), 170, replace=FALSE)  
 X[idx] = 1  
 vaccine = X[1:21500]  
 placebo = X[21501:43000]  
 t[i] = mean(placebo) - mean(vaccine)  
}  
hist(t, xlim=c(-0.002, 0.008))  
abline(v=t\_star, col="blue")



print(length(t[t\_star<t])/perms)

## [1] 0

None of the test values are less than the original statistic so we know that the p-value is less than <0.001.