Lab Assignment #3

Math 437 - Modern Data Analysis

Due February 15, 2023

Instructions

There are two purposes to this lab. First, you will get comfortable with the family-wise error rate and false discovery rate. Then, we will get more practice with coding a nonparametric bootstrap estimate of standard error.

```
library(ISLR2)
library(ggplot2)
library(dplyr)
```

This lab assignment is worth a total of **15 points**.

Problem 1: Family-Wise Error Rate

Part a (Code: 1 pt; Computation and Explanation: 0.5 pts)

Run the code in ISLR Labs 13.6.1 and 13.6.2. Put each chunk from the textbook in its own chunk.

```
set.seed (6)
x <- matrix ( rnorm (10 * 100), 10, 100)
x[, 1:50] <- x[, 1:50] + 0.5
```

```
t.test(x[, 1], mu = 0)
```

```
##
##  One Sample t-test
##
## data:  x[, 1]
## t = 2.0841, df = 9, p-value = 0.06682
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.05171076  1.26242719
## sample estimates:
## mean of x
## 0.6053582
```

```
p.values <- rep (0, 100)
 for (i in 1:100){
 p.values[i] <- t.test (x[, i], mu = 0)$p.value
 decision <- rep ("Do not reject H0", 100)
 decision[p.values <= .05] <- " Reject H0"
 }
```

```
table(decision,c( rep ("H0 is False ", 50), rep ("H0 is True ", 50)))
```

```
##
## decision            H0 is False  H0 is True
##    Reject H0                  10           3
##   Do not reject H0            40          47
```

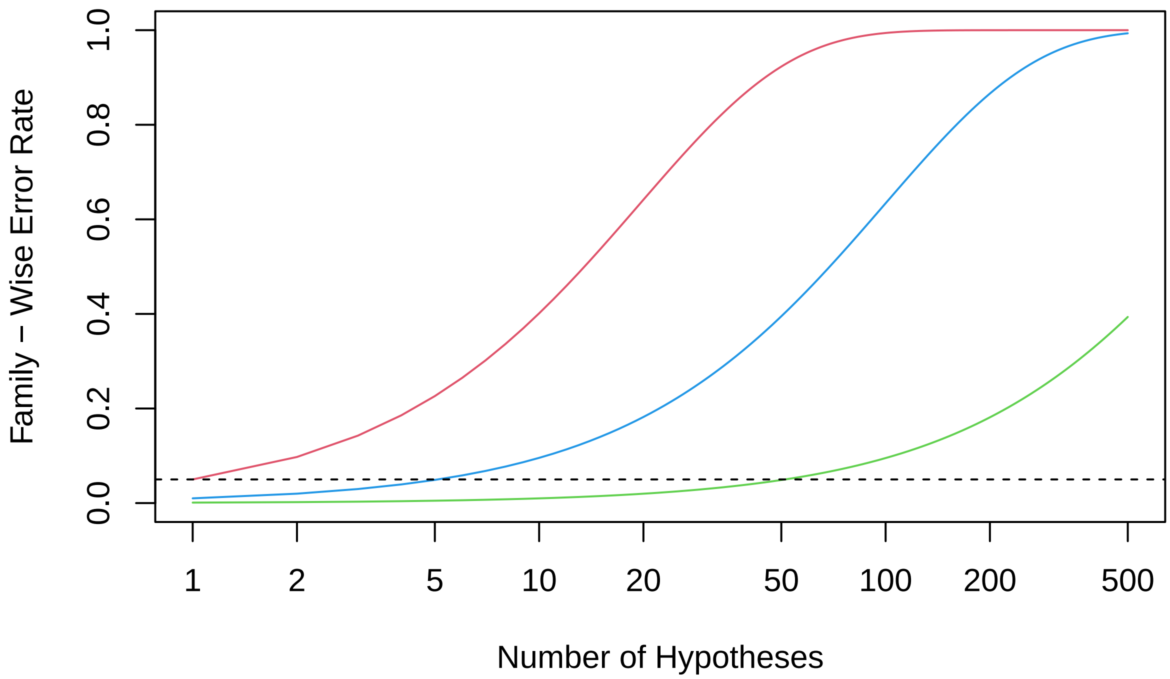
```
x <- matrix ( rnorm (10 * 100), 10, 100)
x[, 1:50] <- x[, 1:50] + 1
for (i in 1:100){
  p.values[i] <- t.test(x[,i], mu = 0)$p.value}
 decision <- rep ("Do not reject H0", 100)
 decision[p.values <= .05] <- " Reject H0"
 table (decision,c( rep ("H0 is False ", 50), rep ("H0 is True ", 50)))
```

```
##
## decision            H0 is False  H0 is True
##    Reject H0                  41           1
##   Do not reject H0             9          49
```

16.2 Section

```
m <- 1:500
fwe1 <- 1 - (1 - 0.05)^m
fwe2 <- 1 - (1 - 0.01)^m
fwe3 <- 1 - (1 - 0.001)^m
```

```
par (mfrow = c(1, 1))
plot (m, fwe1 , type = "l", log = "x", ylim = c(0, 1), col = 2,
      ylab = " Family - Wise Error Rate ",
      xlab = " Number of Hypotheses ")
lines (m, fwe2 , col = 4)
lines (m, fwe3 , col = 3)
abline (h = 0.05, lty = 2)
```



```
library (ISLR2)
fund.mini <- Fund[, 1:5]
t.test (fund.mini[, 1], mu = 0)
```

```
##
##  One Sample t-test
##
## data:  fund.mini[, 1]
## t = 2.8604, df = 49, p-value = 0.006202
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  0.8923397 5.1076603
## sample estimates:
## mean of x
##         3
```

```
fund.pvalue <- rep (0, 5)
for (i in 1:5){fund.pvalue[i] <- t.test(fund.mini[, i], mu = 0)$p.value}
fund.pvalue
```

```
## [1] 0.006202355 0.918271152 0.011600983 0.600539601 0.755781508
```

```
p.adjust(fund.pvalue , method = "bonferroni")
```

```
## [1] 0.03101178 1.00000000 0.05800491 1.00000000 1.00000000
```

```
pmin (fund.pvalue * 5, 1)
```

```
## [1] 0.03101178 1.00000000 0.05800491 1.00000000 1.00000000
```

```
p.adjust (fund.pvalue , method = "holm")
```

```
## [1] 0.03101178 1.00000000 0.04640393 1.00000000 1.00000000
```

```
apply(fund.mini , 2, mean)
```

```
## Manager1 Manager2 Manager3 Manager4 Manager5
##      3.0     -0.1      2.8      0.5      0.3
```

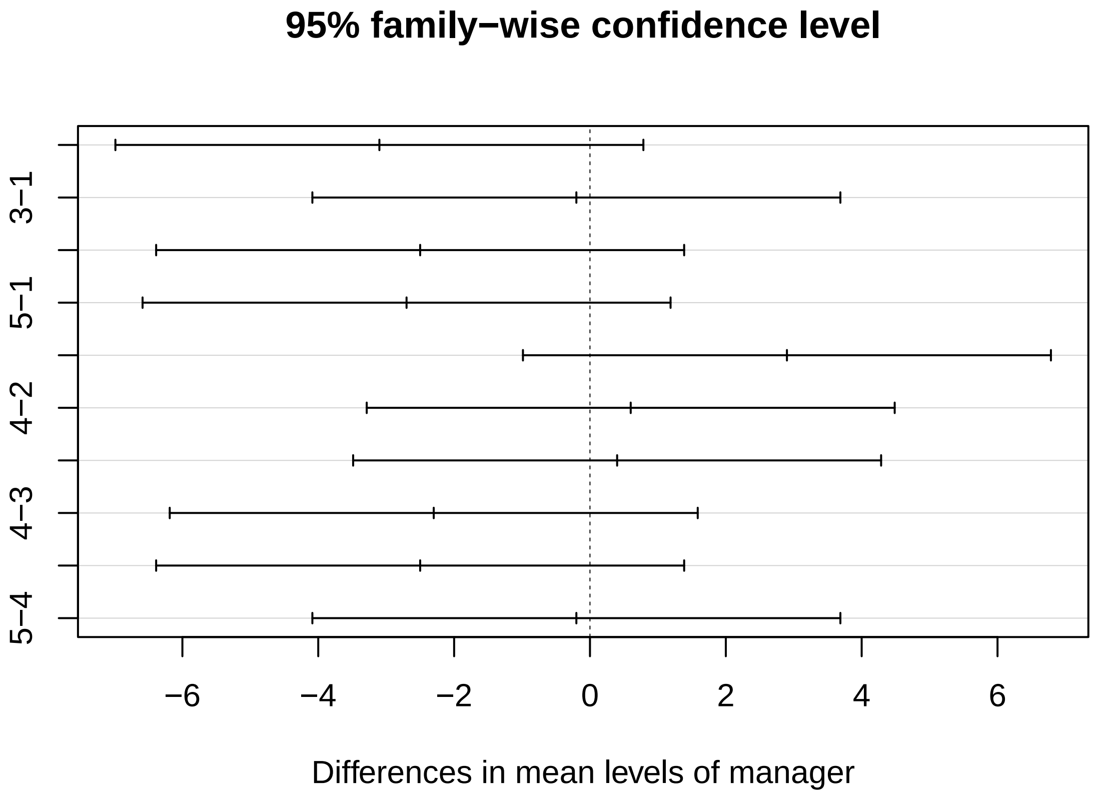
```
t.test(fund.mini[, 1], fund.mini[, 2], paired = T)
```

```
##
##  Paired t-test
##
## data:  fund.mini[, 1] and fund.mini[, 2]
## t = 2.128, df = 49, p-value = 0.03839
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  0.1725378 6.0274622
## sample estimates:
## mean difference
##             3.1
```

```
returns <- as.vector ( as.matrix (fund.mini))
manager <- rep (c("1", "2", "3", "4", "5") , rep (50, 5))
a1 <- aov(returns ~ manager)
TukeyHSD (x = a1)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = returns ~ manager)
##
## $manager
##     diff        lwr       upr     p adj
## 2-1 -3.1 -6.9865435 0.7865435 0.1861585
## 3-1 -0.2 -4.0865435 3.6865435 0.9999095
## 4-1 -2.5 -6.3865435 1.3865435 0.3948292
## 5-1 -2.7 -6.5865435 1.1865435 0.3151702
## 3-2  2.9 -0.9865435 6.7865435 0.2452611
## 4-2  0.6 -3.2865435 4.4865435 0.9932010
## 5-2  0.4 -3.4865435 4.2865435 0.9985924
## 4-3 -2.3 -6.1865435 1.5865435 0.4819994
## 5-3 -2.5 -6.3865435 1.3865435 0.3948292
## 5-4 -0.2 -4.0865435 3.6865435 0.9999095
```

```
plot(TukeyHSD (x = a1))
```



Using the output of Lab 13.6.1, estimate the power of this test to detect each of the two alternative hypotheses $\mu = 0.5$ and $\mu = 1$.

```
power = 41/50
```

The power of the test is 0.82.

Part b (Explanation: 1 pt)

What does the `p.adjust` function do exactly when `method = bonferroni`? What about when `method = holm`? Why does it make more sense for R to adjust the p-values rather than the significance level when controlling the FWER?

`p.adjust` will adjust the p-values using Bonferroni correction where raw p-values are multiplied by the number of comparisons (the total number of hypotheses). When the Holm method is used, ascending raw p-values are multiplied by the total number of hypotheses decreased by one for each subsequent p-value. It makes more sense for R to adjust the p-values rather than the significance level when controlling the FWER in order to avoid having to set a very low significance level and therefore making more Type II errors.

Part c (Explanation: 1 pt)

Consider the second-to-last chunk in ISLR Lab 13.6.2 (the one using `TukeyHSD`). Is it appropriate to do a one-way ANOVA with this data? Explain why or why not. (You may want to produce some graphs and/or numerical summaries to support your answer.)

It is not appropriate to do a one-way ANOVA test with this data because our data is made up of repeated measures (the monthly excess returns) for each manager in the same month and therefore the samples are not independent.

Problem 2: False Discovery Rate

Part a (Code: 0.5 pts)

Run the code in ISLR Lab 13.6.3. Put each chunk from the textbook in its own chunk.

```
fund.pvalues <- rep (0, 2000)
for (i in 1:2000){
  fund.pvalues[i] <- t.test(Fund[, i], mu = 0)$p.value
  }
```

```
q.values.BH <- p.adjust (fund.pvalues , method = "BH")
q.values.BH[1:10]
```

```
##  [1] 0.08988921 0.99149100 0.12211561 0.92342997 0.95603587 0.07513802
##  [7] 0.07670150 0.07513802 0.07513802 0.07513802
```

```
sum(q.values.BH <= .1)
```

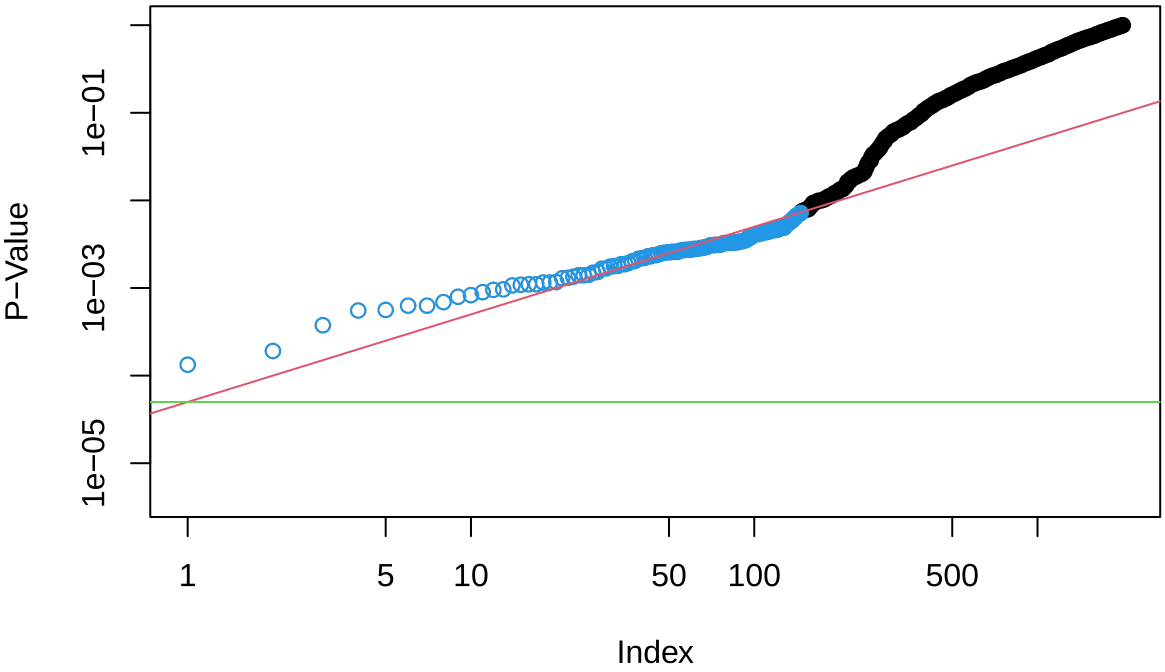
```
## [1] 146
```

```
sum(fund.pvalues <= (0.1 / 2000))
```

```
## [1] 0
```

```
ps <- sort(fund.pvalues)

m <-  length(fund.pvalues)

q <-  0.1

wh.ps <-  which(ps < q * (1:m) / m)

if(length(wh.ps) > 0) {
  wh <-  1:max(wh.ps)
} else {
  wh <- numeric(0)
}
```

```
plot(ps, log = "xy", ylim = c(4e-6, 1), ylab = "P-Value",
     xlab = "Index", main = "")

points(wh, ps[wh], col = 4)

abline(a = 0, b = (q / m), col = 2, untf = TRUE)

abline(h = 0.1 / 2000, col = 3)
```



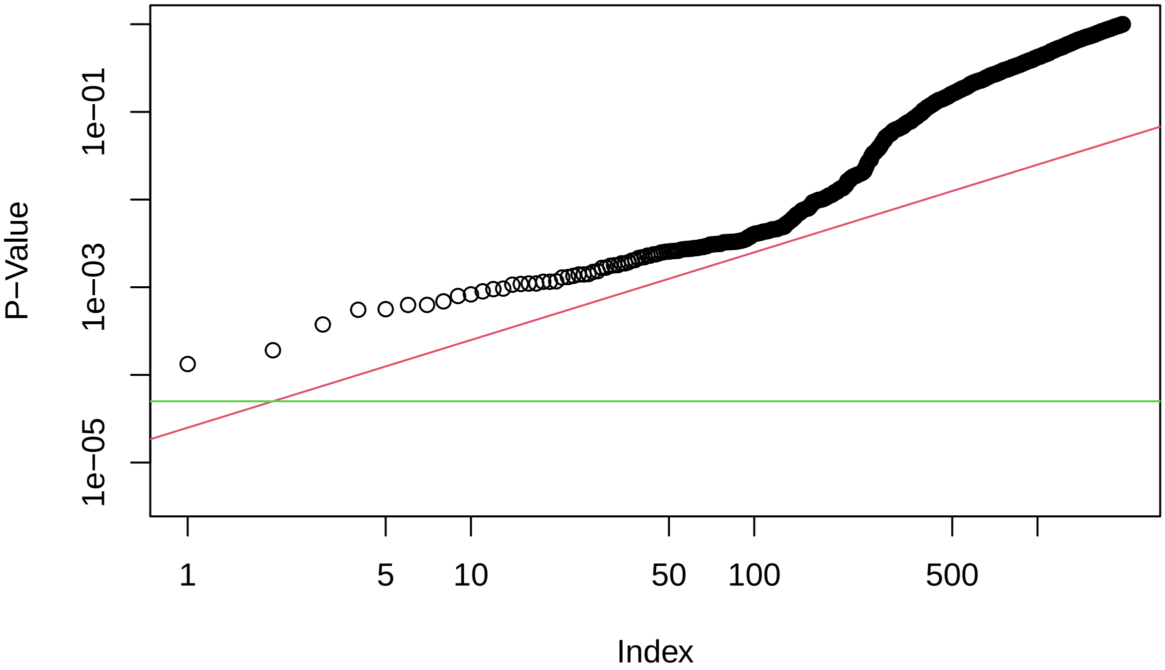
Part b (Code: 1 pt)

Finish writing the `FDR_plot` function in the code chunk below. This function should take two arguments: `p.values`, a vector of p-values, and `q`, the desired False Discovery Rate, and produce a graph like those in Figure 13.6. (Assume that on the graphs, q and α are set to the same value.)

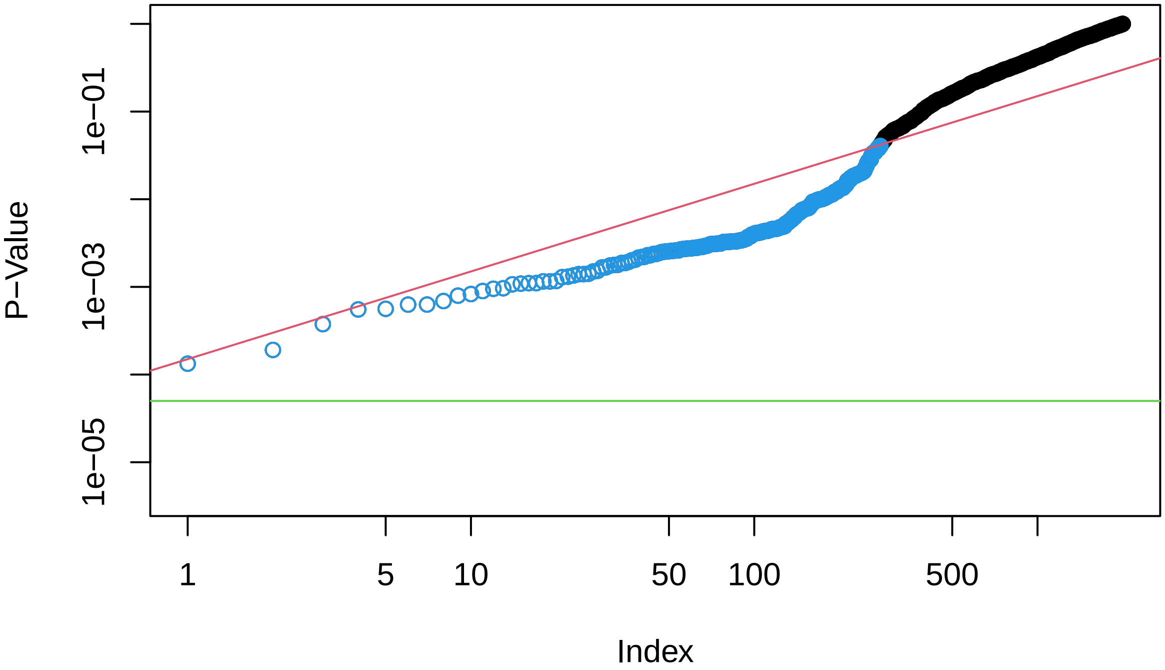
```r
FDR_plot <- function(p.values, q){
  ps <- sort(p.values)

  m <-  length(p.values)

  wh.ps <-  which(ps < q * (1:m) / m)

  if(length(wh.ps) > 0) {
    wh <-  1:max(wh.ps)
  } else {
    wh <- numeric(0)
  }

  plot(ps, log = "xy", ylim = c(4e-6, 1), ylab = "P-Value",
     xlab = "Index", main = "")

  points(wh, ps[wh], col = 4)

  abline(a = 0, b = (q / m), col = 2, untf = TRUE)

  abline(h = 0.1 / 2000, col = 3)

  invisible(wh.ps) # invisibly return the significant p-values
}
```

Test your function. First, attempt to duplicate the left and right panels of Figure 13.6.

```
FDR_plot(fund.pvalues, 0.05)
```

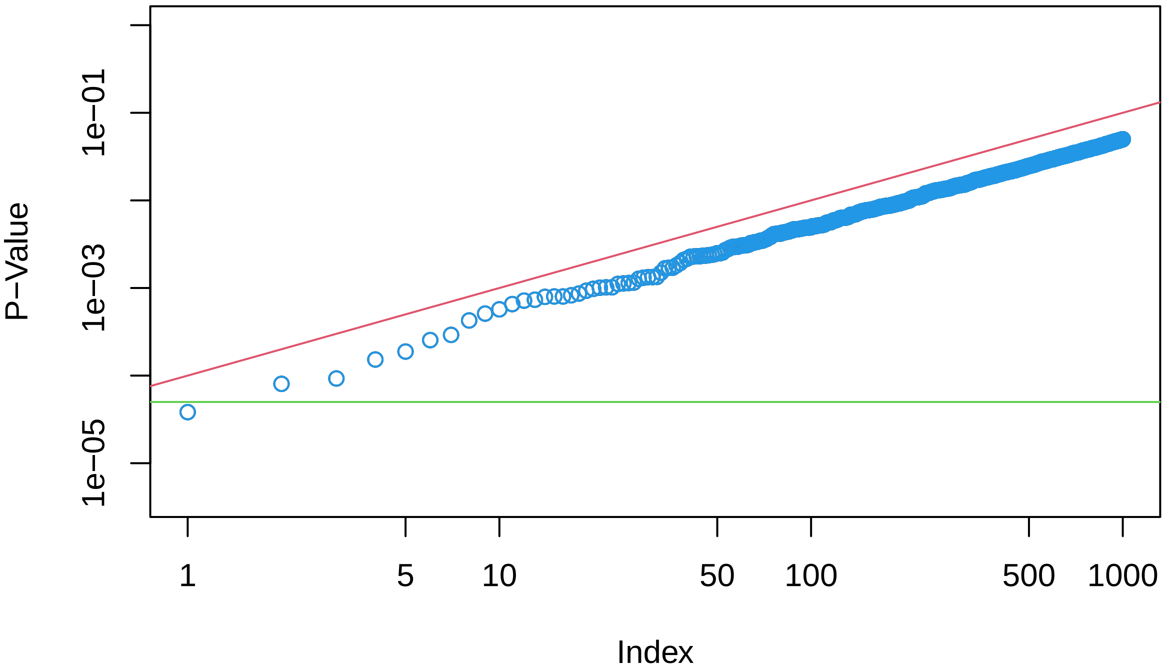


```
FDR_plot(fund.pvalues, 0.3)
```



Then, test your function on the simulated dataset in the chunk below, using $q = 0.1$.

```
set.seed(12)
sim_pvalues <- runif(1000, min = 1e-5, max = 0.05001)

FDR_plot(sim_pvalues, 0.1)
```



```
# Now put a line of code running your function on this set of "significant" p-values
```

Problem 3: Simulation Study of FWER and FDR

This problem is adapted from ISLR Chapter 13, Exercise 8.

Part a (Code: 1 pt)

Using the code in Exercise 13.7.8, create a 20 x 100 matrix where each column represents 20 random numbers from $N(0, 1)$.

```
set.seed(437)

n <- 20

m <- 100

X <- matrix(rnorm(n * m), ncol = m)
```

Then, run a t-test on each column of the matrix testing $H_0 : \mu = 0$ against $H_a : \mu \neq 0$. We are going to use the `apply` function to do this rather than adapt the `for` loop from the ISLR Labs. (Recall from the class activities that the `apply` function applies a single function to each row (`MARGIN = 1`) or column (`MARGIN = 2`) of a matrix.)

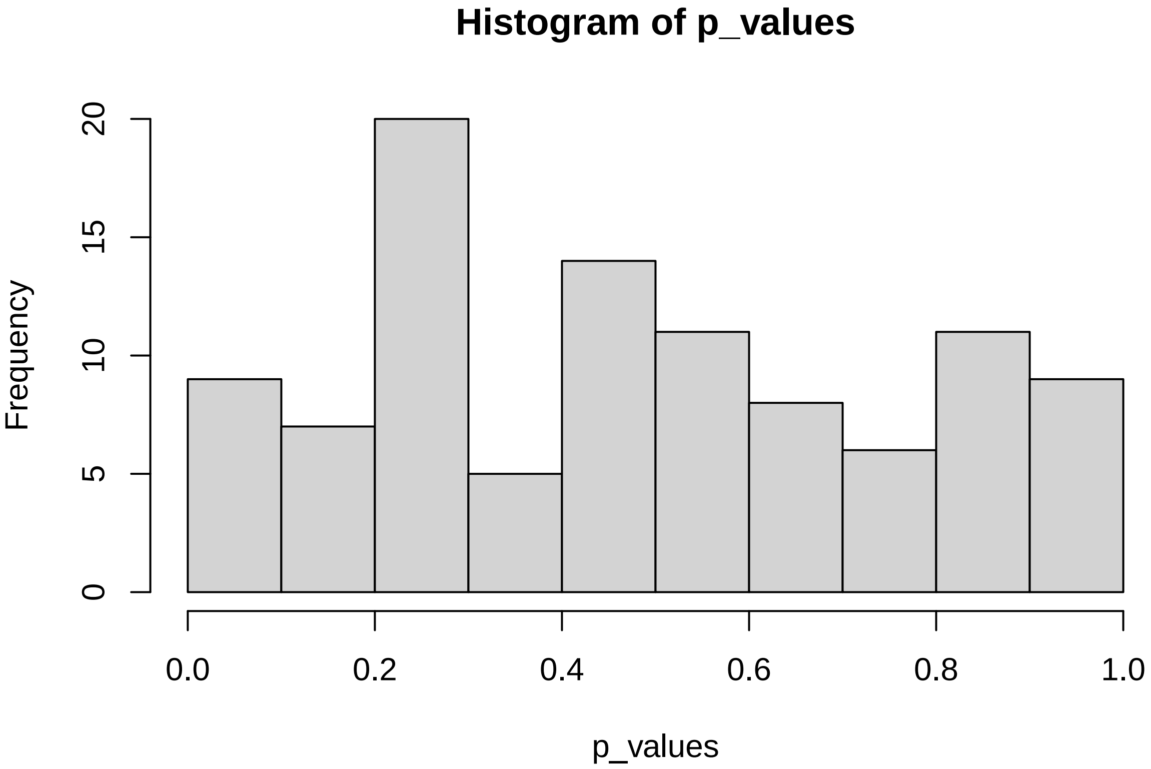
Don't forget to delete the `eval = FALSE` after you've fixed this code chunk to run properly!

```
t_test_0 <- function(x){
  # x: a vector of data

  p_value <- t.test(x, mu = 0)$p.value

  return(p_value)

}
```

```
p_values <- apply(X, 2, t_test_0)
```

Plot a histogram of the p-values obtained.

```
hist(p_values)
```



Part b (Code: 1 pt, Explanation: 0.5 pts)

Without any adjustment for multiple hypothesis tests, how many null hypotheses would be rejected at $\alpha = 0.05$? We can take advantage of the fact that R implicitly converts *logical* (TRUE/FALSE) variables to *numerical* variables.

```
alpha <- 0.05
sum(p_values <= alpha)
```

```
## [1] 4
```

Without adjustment, 4 null hypotheses would be rejected at $\alpha = 0.05$.

Obtain the adjusted p-values using the Holm step-down procedure. How many null hypotheses would be rejected if we control the FWER at 0.05?

```
sum(p.adjust(p_values, method = "holm") <= alpha)
```

```
## [1] 0
```

When adjusting p-values using the Holm step-down procedure, we are rejecting zero null hypotheses.

Obtain the adjusted p-values using the Benjamini-Hochberg procedure. How many null hypotheses would be rejected if we control the FDR at 0.05?

```
sum(p.adjust(p_values, method = "BH") <= alpha)
```

```
## [1] 0
```

When adjusting p-values using the Benjamini-Hochberg procedure, we are still rejecting 0 null hypotheses.

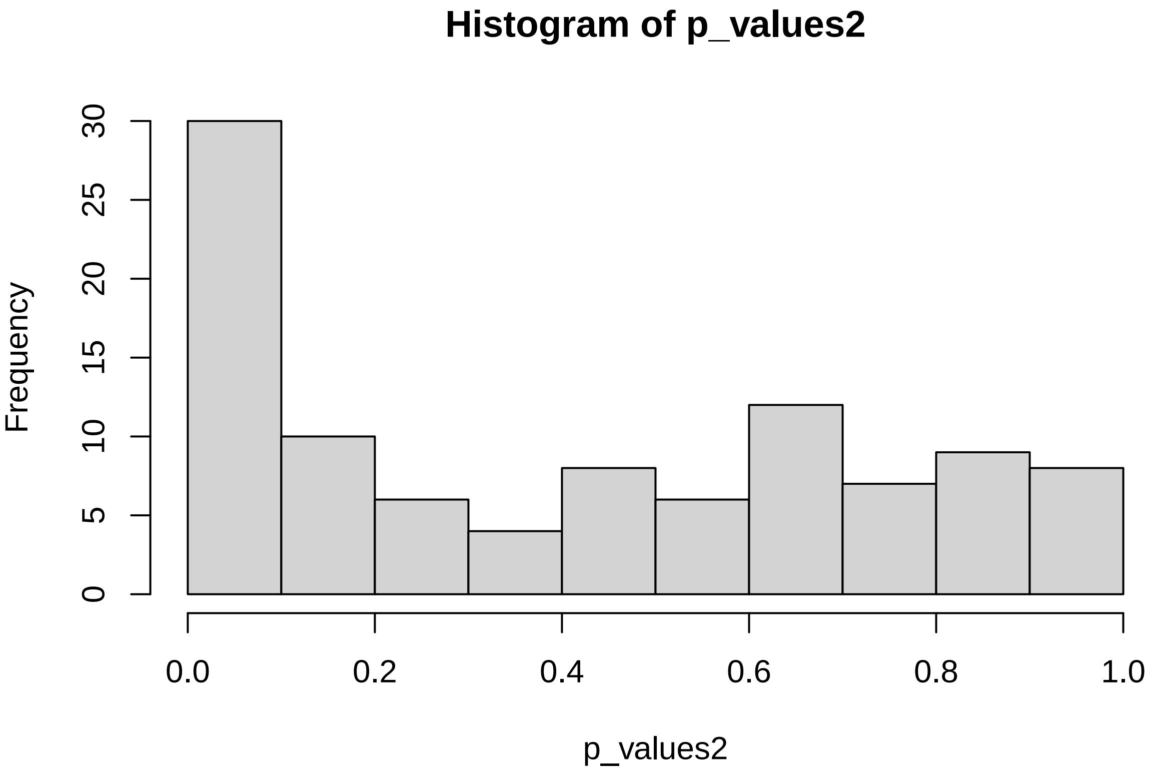
Part c (Code: 1 pt)

Create a new matrix, X2, that is exactly the same as X, except that the first 25 fund managers do actually have a slight long-term return of +1% (i.e., in rows 1-25 $\mu = 1$ not 0.01). Conduct your 100 t-tests and plot a histogram of the new p-values.

```
sub1 <- matrix(rnorm(20*25, mean = 1), ncol = 25)
sub2 <- matrix(rnorm(20*75), ncol = 75)
X2 <- cbind(sub1,sub2)
```

```
p_values2 <- apply(X2, 2, t_test_0)
```

```
hist(p_values2)
```



Part d (Code: 1 pt, Explanation: 1 pt)

Without any adjustment for multiple hypothesis tests, how many of the 75 true null hypotheses would be rejected at $\alpha = 0.05$? How many of the 25 false null hypotheses would be rejected?

```
reject.falseH0 <- sum(p_values2[1:25] <= alpha)
reject.trueH0 <- sum(p_values2[26:100] <= alpha)
c(true = reject.trueH0, false = reject.falseH0)
```

```
##  true false
##     2    25
```

Obtain the adjusted p-values using the Holm step-down procedure. How many null hypotheses of each type would be rejected if we control the FWER at 0.05?

```r
p.adj <- p.adjust(p_values2, method = "holm")
reject.falseH0 <- sum(p.adj[1:25] <= alpha)
reject.trueH0 <- sum(p.adj[26:100] <= alpha)
c(true = reject.trueH0, false = reject.falseH0)
```

```
##  true false 
##     0    16
```

Obtain the adjusted p-values using the Benjamini-Hochberg procedure. How many null hypotheses of each type would be rejected if we control the FDR at 0.05?

```r
p.adj <- p.adjust(p_values2, method = "BH")
reject.falseH0 <- sum(p.adj[1:25] <= alpha)
reject.trueH0 <- sum(p.adj[26:100] <= alpha)
c(true = reject.trueH0, false = reject.falseH0)
```

```
##  true false 
##     0    23
```

Compare your results from the three procedures. Why is it important to consider the situation where *some* null hypotheses are true, when evaluating the performance of the different procedures?

It is important to consider the situations where some null hypotheses are true as using either of these procedures impact the chances of having type I and type II errors which can only be noticed when *some* of the null hypotheses are true meaning the different procedures impact the error rates differently, and thus the power differently as well.

Problem 4: Bootstrap Estimation of Standard Error

Part a (Code: 0.5 pts)

Run the code in the first half of ISLR Lab 5.3.4, "Estimating the Accuracy of a Statistic of Interest." Put each chunk from the textbook in its own chunk.

If you are in the actuarial science concentration, you should be familiar with (or will at some point see) this formula! For the rest of us, note that X and Y are assumed to be the yearly return of two different financial assets, and α, the quantity to be estimated, is the fraction of money to be invested in X such that the variance (risk) of the total investment $\alpha X + (1 - \alpha)Y$ is minimized. In this problem α is not the significance level!

Part b (Code: 2 pts)

According to the instructions for Lab 5.3.4, "We can implement a bootstrap analysis by performing this command [alpha.fn on a bootstrap sample] many times, recording all of the corresponding estimates for α, and computing the resulting standard deviation."

Write a code chunk that performs all of those steps and prints out the standard deviation. Use 1000 bootstrap samples.

Part c (Code: 1 pt)

Replicate the center panel of textbook Figure 5.10: a histogram of the bootstrap estimates of α (from Part b) with a solid pink (or red) line at the true value of $\alpha = 0.6$. You may use either base R plotting commands (which uses `abline` to add the vertical line) or the `ggplot2` package (which adds a `geom_vline` to the plot).

Part d (Explanation: 1 pt)

Note that the distribution you graphed in Part c is a sampling distribution of $\hat{\alpha}$. Explain why it would be appropriate to use this sampling distribution to construct a confidence interval for α, but not to obtain a p-value for a hypothesis test of $H_0 : \alpha = 0.6$ against $H_a : \alpha \neq 0.6$.