Divyansh Tiwari (2020111002)

2023-02-16

# Question 1

The following is the code for the 1st question.

p\_values <- c(0.0050, 0.0010, 0.0100, 0.0005, 0.0009, 0.0400, 0.0560, 0.0500, 0.0480, 0.0130, 0.0370, 0.0430, 0.0020, 0.0250, 0.1100, 0.0700, 0.0800)  
  
p\_values <- sort(p\_values)  
  
p\_bonferroni <- p.adjust(p\_values, method = "bonferroni", n = length(p\_values))  
  
p\_hochberg <- p.adjust(p\_values, method = "hochberg", n = length(p\_values))  
  
  
print(p\_values)

## [1] 0.0005 0.0009 0.0010 0.0020 0.0050 0.0100 0.0130 0.0250 0.0370 0.0400  
## [11] 0.0430 0.0480 0.0500 0.0560 0.0700 0.0800 0.1100

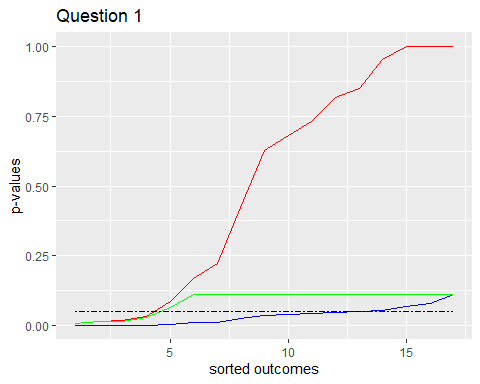
print(p\_bonferroni)

## [1] 0.0085 0.0153 0.0170 0.0340 0.0850 0.1700 0.2210 0.4250 0.6290 0.6800  
## [11] 0.7310 0.8160 0.8500 0.9520 1.0000 1.0000 1.0000

print(p\_hochberg)

## [1] 0.0085 0.0144 0.0150 0.0280 0.0650 0.1100 0.1100 0.1100 0.1100 0.1100  
## [11] 0.1100 0.1100 0.1100 0.1100 0.1100 0.1100 0.1100

library(reshape2)  
library(ggplot2)  
  
data <- data.frame(x = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17), y1 = p\_values, y2 = p\_bonferroni, y3 = p\_hochberg, y4 = c(0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500, 0.0500))  
  
gfg\_plot <- ggplot(data, aes(x)) +   
 geom\_line(aes(y = y1), color = "blue") +  
 geom\_line(aes(y = y2), color = "red") +  
 geom\_line(aes(y = y3), color = "green")+  
 geom\_line(aes(y = y4), color = "black", linetype = "twodash")+ labs(title="Question 1", x = "sorted outcomes", y = "p-values")  
  
gfg\_plot



#+geom\_line(aes(color=c("Unadjusted", "Bonferroni", "Hochberg")))

The following is the inference from the graph:

High adjusted p values for the earlier experiments show that there is not enough data to conclusively demonstrate an effect. The more severe Bonferroni correction produces bigger adjusted p values. It is less rigorous since the Hochberg adjustment correlates to lower adjusted p values for the experiment than our alpha. The bonferroni test is a conservative test since it has a high probability of rejecting the null hypothesis.

# Question 2

The following is the code for the second question.

## Part A

library(readxl)  
mydataq2 <- read\_excel("./BRSM\_Results\_Visualization.xlsx", 1)  
  
print(mydataq2)

## # A tibble: 43 × 2  
## Group `No. of Mosquitoes`  
## <chr> <dbl>  
## 1 Beer 27  
## 2 Beer 19  
## 3 Beer 20  
## 4 Beer 20  
## 5 Beer 23  
## 6 Beer 17  
## 7 Beer 21  
## 8 Beer 24  
## 9 Beer 31  
## 10 Beer 26  
## # … with 33 more rows

beer <- mydataq2[mydataq2$Group == "Beer", ]  
print(beer)

## # A tibble: 25 × 2  
## Group `No. of Mosquitoes`  
## <chr> <dbl>  
## 1 Beer 27  
## 2 Beer 19  
## 3 Beer 20  
## 4 Beer 20  
## 5 Beer 23  
## 6 Beer 17  
## 7 Beer 21  
## 8 Beer 24  
## 9 Beer 31  
## 10 Beer 26  
## # … with 15 more rows

water <- mydataq2[mydataq2$Group == "Water", ]  
print(water$`No. of Mosquitoes`)

## [1] 21 19 13 22 15 22 15 22 20 12 24 24 21 19 18 16 23 20

beer\_mean <- mean(beer$`No. of Mosquitoes`)  
print(beer\_mean)

## [1] 23.6

water\_mean <- mean(water$`No. of Mosquitoes`)  
print(water\_mean)

## [1] 19.22222

test.stat.1 <- abs(water\_mean - beer\_mean)  
  
beer\_median <- median(beer$`No. of Mosquitoes`)  
water\_median <- median(water$`No. of Mosquitoes`)  
test.stat.2 <- abs(beer\_median - water\_median)  
print(test.stat.2)

## [1] 4

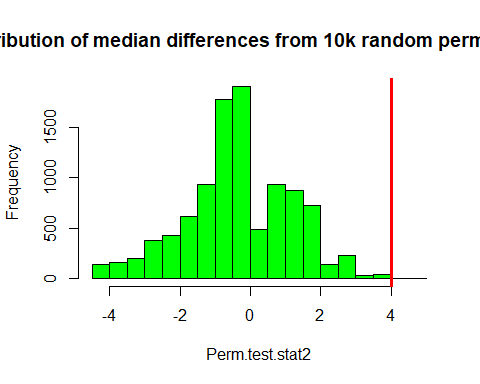
cat("The difference between the group medians is", test.stat.2)

## The difference between the group medians is 4

set.seed(1979)  
  
n <- length(mydataq2$Group)  
P <- 10000  
var <- mydataq2$`No. of Mosquitoes`  
  
PermSamples <- matrix(0, nrow = n, ncol = P)  
  
for(i in 1:P)  
{  
 PermSamples[, i] <- sample(var, size = n, replace = FALSE)  
}  
  
  
Perm.test.stat1 <- rep(0, P)  
Perm.test.stat2 <- rep(0, P)  
  
for (i in 1:P)  
{  
 Perm.test.stat1[i] <- mean(PermSamples[mydataq2$Group == "Beer", i]) - mean(PermSamples[mydataq2$Group == "Water", i])  
   
 Perm.test.stat2[i] <- median(PermSamples[mydataq2$Group == "Beer", i]) - median(PermSamples[mydataq2$Group == "Water", i])  
   
}  
  
print(mean(Perm.test.stat2 >= test.stat.2))

## [1] 0.0043

hist(Perm.test.stat2, col = "green", main="Distribution of median differences from 10k random permutations")  
abline(v=test.stat.2, lwd=3, col="red")



p\_value = sum(Perm.test.stat2 >= test.stat.2)/P  
print(p\_value)

## [1] 0.0043

cat("The p\_value obtained is ", p\_value)

## The p\_value obtained is 0.0043

We observe that the p-value is less than 0.05 which brings to us a very high possibility of the alternate directional hypothesis to be true and the observed statistic is significant.

## Part B

t <- t.test(beer$`No. of Mosquitoes`, water$`No. of Mosquitoes`, var.equal = TRUE)  
t <- t$statistic[['t']]  
cat("The initial t-score is ", t)

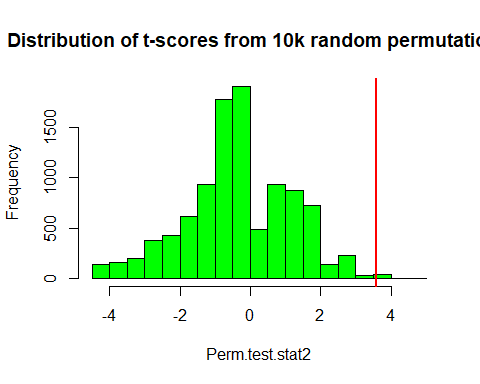
## The initial t-score is 3.586984

p2\_perm = rep(0, P)  
  
for(i in 1:P)  
{  
 temp\_t <- t.test(head(PermSamples[, i], 25), tail(PermSamples[,i], 18), var.equal = TRUE)  
 p2\_perm[i] <- temp\_t$statistic[['t']]  
}  
  
p\_value\_2 = sum(p2\_perm >= t)/P  
cat("The observed p\_value is ", p\_value\_2)

## The observed p\_value is 3e-04

The observed p\_value is less than 0.05 which confirms that the alternate hypothesis is true and the observed statistic is significant.

hist(Perm.test.stat2, col="green", main = "Distribution of t-scores from 10k random permutations")  
abline(v=t, lwd=2, col="red")



## Part C

Since, we are calculating the p-value assuming that the hypothesis is non-directional, therefore we need to calculate the p-value with the absolute value of the differences of the medians.

p\_value\_3 = sum(abs(Perm.test.stat2) >= test.stat.2)/P  
cat("The p-value is this case is ", p\_value\_3)

## The p-value is this case is 0.0181

The p-value is observed to be less than 0.05 which implies that the non-directional hypothesis is true and the observed statistic is significant.

# Question 3

Following is the code for the third question.

mydataq3 <- read\_excel("./IQ.xlsx", 1)

## New names:  
## • `` -> `...1`

q3\_p <- cor(mydataq3$IQ, mydataq3$TESTSCORE)  
print("The intial correlation value is ")

## [1] "The intial correlation value is "

print(q3\_p)

## [1] 0.4931479

n <- length(mydataq3$IQ)  
P <- 10000  
iq <- mydataq3$IQ  
test\_scores <- mydataq3$TESTSCORE  
  
iq\_samples <- matrix(0, nrow = n, ncol = P)  
test\_scores\_samples <- matrix(0, nrow = n, ncol = P)  
  
for(i in 1:P)  
{  
 iq\_samples[, i] <- sample(iq, size = n, replace = FALSE)  
 test\_scores\_samples[, i] <- sample(test\_scores, size = n, replace = FALSE)  
}  
  
p3\_perm = rep(0, P)  
  
for(i in 1:P)  
{  
   
 p3\_perm[i] = cor(iq\_samples[, i], test\_scores\_samples[, i])  
}  
   
p\_value\_4 <- sum(p3\_perm >= q3\_p)/P  
print(p\_value\_4)

## [1] 0

print(paste("The observed p-value is ", p\_value\_4))

## [1] "The observed p-value is 0"

The p-value that is being obtained as I run the code above oscillates between 0 and 1e-04. At the very moment, the p-value from the code above is 1e-04, but for some reason it is getting rendered as 0 in the pdf.

We observe that the p-value is much less than 0.05 based on our observed statistic. Therefore, we can safely assume that we can reject the null hypothesis, since there exists a correlation between IQ and Test scores.

hist(p3\_perm, col="green", main = "Distribution of correlation values from 10k random permutations", xlim = c(-0.7, 0.7))  
abline(v=q3\_p, lwd=2, col="red")

