HW2

Runye Hu A13410385
Yijie Fan A13485989
Xinyi He A13561164
Yifan Wu A14060535
Yiwen Cai A13530685

Problem1

Nutrient data

```
nutrient=read.table("https://www.math.ucsd.edu/~wez243/data/nutrient.txt")
nutrient$V1=NULL
colnames(nutrient)=c("Calcium", "Iron", "Protein", "Vitamin A", "Vitamin C")
```

Q1:

```
means <- colMeans(nutrient, na.rm = TRUE)
print(means)
```

```
##   Calcium      Iron   Protein Vitamin A Vitamin C
## 624.04925  11.12990  65.80344 839.63535  78.92845
```

```
sd <- apply(nutrient, 2, function(x) sd(x, na.rm = TRUE))
print(sd)
```

```
##    Calcium       Iron    Protein  Vitamin A  Vitamin C
##  397.27754    5.98419   30.57576 1633.53983   73.59527
```

Q2:

```
intake = c(1000,15,60,800,75)
p_val = c();
for (i in 1:ncol(nutrient)){
  p = t.test(nutrient[i],alternative = c("two.sided"),mu = intake[i], paired = FALSE,con
f.level = 0.95)$p.value
  p_val = c(p_val,p)
}
p_val
```

```
## [1] 2.108727e-104  6.654391e-58  3.299706e-07  5.102954e-01  1.477297e-01
```

```
ans = p_val > matrix(0.05,1,ncol(nutrient))
ans
```

```
##       [,1]  [,2]  [,3] [,4] [,5]
## [1,] FALSE FALSE FALSE TRUE TRUE
```

Q3:

For Calcium, Iron, Protein we reject the hypotheses that the population mean of that variable equals the recommended values because their p-value are less than significance level.
For VitaminA and VitaminC, we don't have enough evidence to reject because their p-value are larger than significance level. Therefore, we conclude that the US women don't meet the recommended nutrient intake amount.
My recommendation is that US women should intake more food with rich Calcium and Iron while intaking a little bit less protein.

Problem2

Multiple data

```
multiple = read.table("~/Downloads/multiple.txt");
```

Q1:

```
p_val = c()
ans = c()
for (i in 1:50){
  p = t.test(multiple[i], alternative = c("two.sided"),mu = 0,paired = FALSE,conf.level
 = 0.9)$p.value
  p_val = c(p_val,p)
}
p_val
```

```
##  [1] 7.477468e-34 1.030644e-32 1.613953e-33 5.107798e-37 4.299681e-35
##  [6] 4.181053e-33 1.782424e-39 2.281744e-34 4.767758e-34 9.020768e-42
## [11] 8.501800e-01 3.387655e-01 8.433591e-01 2.353081e-01 8.346443e-01
## [16] 3.410084e-01 4.495285e-01 8.988573e-01 4.855360e-01 8.298508e-02
## [21] 7.435778e-01 4.344605e-01 5.521855e-01 4.775194e-01 7.573900e-01
## [26] 1.354197e-01 9.927262e-01 2.944327e-01 6.783256e-01 1.898939e-01
## [31] 4.716828e-01 9.302681e-02 4.736660e-01 9.189993e-01 2.138515e-01
## [36] 6.519921e-01 3.408716e-01 2.211123e-01 8.806238e-01 7.315268e-02
## [41] 7.171567e-01 6.472996e-01 6.915055e-01 2.475067e-02 2.191321e-01
## [46] 7.678119e-01 3.489369e-01 1.367531e-01 8.886784e-01 7.316609e-01
```

```
ans = p_val > matrix(0.1,1,50)
ans
```

```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11]
## [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
##      [,12] [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
##      [,23] [,24] [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
##      [,34] [,35] [,36] [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
##      [,45] [,46] [,47] [,48] [,49] [,50]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

So the population mean vector doesn't equal to a vector whose elements are all zeros.

Q2:

```
type1 = 4
type1
```

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

```
type2 = 0
type2
```

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

```
FWER = 1-(1-0.1)^50
FWER
```

```
## [1] 0.9948462
```

```
FDP = type1/(50-sum(ans))
FDP
```

```
## [1] 0.2857143
```

Number of Type 1 Error: 4
Number of Type 2 Error: 0
FWER: 0.9948462
FDP: 0.2857143

Q3:

```
p_B = c()
ans_B = c()
for (i in 1:50){
  p = t.test(multiple[i], alternative = c("two.sided"),mu = 0,paired = FALSE,conf.level
 = 0.9)$p.value
  p_B = c(p_B,p)
}
p_B
```

```
##  [1] 7.477468e-34 1.030644e-32 1.613953e-33 5.107798e-37 4.299681e-35
##  [6] 4.181053e-33 1.782424e-39 2.281744e-34 4.767758e-34 9.020768e-42
## [11] 8.501800e-01 3.387655e-01 8.433591e-01 2.353081e-01 8.346443e-01
## [16] 3.410084e-01 4.495285e-01 8.988573e-01 4.855360e-01 8.298508e-02
## [21] 7.435778e-01 4.344605e-01 5.521855e-01 4.775194e-01 7.573900e-01
## [26] 1.354197e-01 9.927262e-01 2.944327e-01 6.783256e-01 1.898939e-01
## [31] 4.716828e-01 9.302681e-02 4.736660e-01 9.189993e-01 2.138515e-01
## [36] 6.519921e-01 3.408716e-01 2.211123e-01 8.806238e-01 7.315268e-02
## [41] 7.171567e-01 6.472996e-01 6.915055e-01 2.475067e-02 2.191321e-01
## [46] 7.678119e-01 3.489369e-01 1.367531e-01 8.886784e-01 7.316609e-01
```

```
ans_B = p_B > matrix(0.1/50,1,50)
ans_B
```

```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11]
## [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
##      [,12] [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##      [,23] [,24] [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##      [,34] [,35] [,36] [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##      [,45] [,46] [,47] [,48] [,49] [,50]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

```
FWER = 1-(1-0.1/50)^50
FWER
```

```
## [1] 0.09525318
```

```
power = 10/10
power
```

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

FWER:0.09525318
power:1

Q4:

```
p_value = c()
ans_BH = c()
for (i in 1:50){
  p = t.test(multiple[i], alternative = c("two.sided"),mu = 0,paired = FALSE,conf.level
 = 0.9)$"p.value"
  p_value = c(p_value,p)
}

bh <- function(p,alpha)
{
  m <- length(p)
  l <- alpha*c(1:m)/m
  result <- matrix(TRUE,1,m)
  sort_p <- sort(p)
  set <- which(l>=sort_p)
  if(length(set)==0){
    rej <- set
  } else{
    imax <- max(set)
    threshold <- sort_p[imax]
    rej <- which(p <= threshold)
  }
  result[rej] = FALSE;

  outlist<-list(result = result)
  return(outlist)
}

bh_test <- bh(p_value,0.1)
p_value
```

```
##  [1] 7.477468e-34 1.030644e-32 1.613953e-33 5.107798e-37 4.299681e-35
##  [6] 4.181053e-33 1.782424e-39 2.281744e-34 4.767758e-34 9.020768e-42
## [11] 8.501800e-01 3.387655e-01 8.433591e-01 2.353081e-01 8.346443e-01
## [16] 3.410084e-01 4.495285e-01 8.988573e-01 4.855360e-01 8.298508e-02
## [21] 7.435778e-01 4.344605e-01 5.521855e-01 4.775194e-01 7.573900e-01
## [26] 1.354197e-01 9.927262e-01 2.944327e-01 6.783256e-01 1.898939e-01
## [31] 4.716828e-01 9.302681e-02 4.736660e-01 9.189993e-01 2.138515e-01
## [36] 6.519921e-01 3.408716e-01 2.211123e-01 8.806238e-01 7.315268e-02
## [41] 7.171567e-01 6.472996e-01 6.915055e-01 2.475067e-02 2.191321e-01
## [46] 7.678119e-01 3.489369e-01 1.367531e-01 8.886784e-01 7.316609e-01
```

```
ans_BH = bh_test$result
ans_BH
```

```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11]
## [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
##      [,12] [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##      [,23] [,24] [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##      [,34] [,35] [,36] [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##      [,45] [,46] [,47] [,48] [,49] [,50]
## [1,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

```
FDP = 0/sum(ans_BH)
FDP
```

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

```
power = 10/10
power
```

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

FDP: 0
power:1
By using BH method, we improve from the result of naive multiple testing and multiple testing with Bonferroni method.
In 2.1(naive multiple testing part), we have 4 Type I error, while we make no Type I error using BH method.
For 2.3 and 2.4, type I and type II errors are the same(we make no errors), and thus they both work good for this case.