MATH 189 HW2

Zijian Su
Zelong Zhou
Xiangyi Lin

Last Updated: January 27, 2023

Concrete contributions

All problems were done by Zijian Su, Zelong Zhou, Xiangyi Lin. All contributing equally to this assignment. Everyone put in enough effort.

Packages

```
#install.packages("rmarkdown")
#tinytex::install_tinytex()
```

Overview problem 1

The USDA Women's Health Survey dataset (nutrient.txt) contains five types of women's nutrient intakes which were measured from a random sample of 737 women aged 25-50 years in United States. Analyze the dataset according to the following steps:

Question 1.1

Calculate sample mean and sample standard deviation ofeach variable.

```
#load the data
nutrient <- read.table("./nutrient.txt")
nutrient_ <- nutrient[,-1]
```

Answer:

Means:

```
means = apply(nutrient_,2,mean)
cat(paste(" Variable 1(Calcium)  :  ",means[1],"\n",
          "Variable 2(Iron)     :  ",means[2],"\n",
          "Variable 3(Protein)  :  ",means[3],"\n",
          "Variable 4(Vitamin A):  ",means[4],"\n",
          "Variable 5(Vitamin C):  ",means[5]))
```

```
##  Variable 1(Calcium)  :   624.049253731343
##  Variable 2(Iron)     :   11.1298995929444
##  Variable 3(Protein)  :   65.8034409769335
##  Variable 4(Vitamin A):   839.635345997286
##  Variable 5(Vitamin C):   78.9284464043419
```

Standard deviation:

```
sds = apply(nutrient_,2,sd)
cat(paste(" Variable 1(Calcium)  :  ",sds[1],"\n",
          "Variable 2(Iron)     :  ",sds[2],"\n",
          "Variable 3(Protein)  :  ",sds[3],"\n",
          "Variable 4(Vitamin A):  ",sds[4],"\n",
          "Variable 5(Vitamin C):  ",sds[5]))
```

```
##  Variable 1(Calcium)  :   397.277540103266
##  Variable 2(Iron)     :   5.98419047008833
##  Variable 3(Protein)  :   30.5757564314087
##  Variable 4(Vitamin A):   1633.53982830006
##  Variable 5(Vitamin C):   73.59527211824
```

Question 1.2

The recommend intake amount of each nutrient is given in the following table. For each nutrient, apply a univariate t-test to test if the population mean of that variable equals the recommended value. Set the significance level at a = 0.05

Answer:

Calcium T-test:

```
t.test(nutrient_[1],mu=1000)
```

```
##
##  One Sample t-test
##
## data:  nutrient_[1]
## t = -25.69, df = 736, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 1000
## 95 percent confidence interval:
##  595.3201 652.7784
## sample estimates:
## mean of x
##  624.0493
```

The confidence interval(a=0.05) is (595.3201, 652.7784). Therefore, there is a high probability that the population mean is **not equal** to the recommended value(1000).

Iron T-test:

```
t.test(nutrient_[2],mu=15)
```

```
##
##  One Sample t-test
##
## data:  nutrient_[2]
## t = -17.557, df = 736, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 15
## 95 percent confidence interval:
##  10.69715 11.56265
## sample estimates:
## mean of x
##   11.1299
```

The confidence interval(a=0.05) is (10.69715, 11.56265). Therefore, there is a high probability that the population mean is **not equal** to the recommended value(15).

Protein T-test:

```
t.test(nutrient_[3],mu=60)
```

```
##
##  One Sample t-test
```

```
##
## data:  nutrient_[3]
## t = 5.1528, df = 736, p-value = 3.3e-07
## alternative hypothesis: true mean is not equal to 60
## 95 percent confidence interval:
##  63.59235 68.01453
## sample estimates:
## mean of x
##  65.80344
```

The confidence interval(a=0.05) is (63.59235, 68.01453). Therefore, there is a high probability that the population mean is **not equal** to the recommended value(60).

Vitamin A T-test:

```
t.test(nutrient_[4],mu=800)
```

```
##
##  One Sample t-test
##
## data:  nutrient_[4]
## t = 0.6587, df = 736, p-value = 0.5103
## alternative hypothesis: true mean is not equal to 800
## 95 percent confidence interval:
##  721.5057 957.7650
## sample estimates:
## mean of x
##  839.6353
```

The confidence interval(a=0.05) is (721.5057, 957.7650). And the p-value > 0.05. Therefore, the population mean may be equal to or close to the recommended value(800). Can be considered equal

Vitamin C T-test:

```
t.test(nutrient_[5],mu=75)
```

```
##
##  One Sample t-test
##
## data:  nutrient_[5]
## t = 1.4491, df = 736, p-value = 0.1477
## alternative hypothesis: true mean is not equal to 75
## 95 percent confidence interval:
##  73.6064 84.2505
## sample estimates:
## mean of x
##  78.92845
```

The confidence interval(a=0.05) is (73.6064, 84.2505). And the p-value > 0.05. Therefore, the population mean may be equal to or close to the recommended value(75). Can be considered equal

Question 1.3

Based on the results you obtained in step 2, how would you interpret your test results? Do you think the US Women meet the recommended nutrient intake amount? If not, what would you suggest to the public?

Answer:

From the results in Q2 we can draw a simple conclusion:
Calcium's population mean between (595.3201, 652.7784), p-value < 0.05, can be considered lower than the recommended value(1000)
Iron's population mean between (10.69715, 11.56265), p-value < 0.05, can be considered lower than the recommended value(15)
Protein's population mean between (63.59235, 68.01453), p-value < 0.05, can be considered higher than the recommended value(60)
Vitamin A's population mean between (721.5057, 957.7650), p-value > 0.05, can be considered equal to the recommended value(800)
Vitamin C's population mean between (73.6064, 84.2505), p-value > 0.05, can be considered equal to the recommended value(75)
There is a 95% probability that the above conclusion is correct.

Therefore, the calcium, iron, and protein intake of US women do not meet the recommended values. Vitamin A and vitamin C, may meet the recommended value.

Thus, we think US women need to supplement more calcium and iron and reduce protein intake. For vitamins A and C, no action is required.

Overview problem 2

The Multiple Testing dataset (multiple.txt) is a simulated dataset which contains 50 variables and 100 observations per variable. Suppose we know that the first 10 variables have mean equal to 2 and the rest of them have mean equal to 0. Analyze the dataset according to the following steps:

```
#load the data
multiple <- read.table("./multiple.txt")
```

Question 2.1

Perform multiple testing to the population mean vector to test if it equals to a vector whose elements are all zeros. Set the significance level at a= 0.1.

Answer:

```
index = 0
for (i in colnames(multiple)){
    print(i)
    print(t.test(multiple[i],conf.level = 0.90))
  index = index +1
}
```

The above code can do a t-test for all variables. Since a large amount of text will be output, the output results are not displayed here.

ALL mean-values:

```
mean_list <- apply(multiple,2,mean)
mean_list
```

```
##            V1            V2            V3            V4            V5
##  1.8940678549  1.9495588509  1.8472232644  2.1064240800  2.1382300739
##            V6            V7            V8            V9           V10
##  1.9636644072  1.8994955068  1.9480038358  2.0026202907  2.0543524682
##           V11           V12           V13           V14           V15
##  0.0186284017  0.1064941850  0.0208598431 -0.1192291949 -0.0217248105
##           V16           V17           V18           V19           V20
##  0.0923767217  0.0757773622  0.0122809168  0.0706703475 -0.1695070469
##           V21           V22           V23           V24           V25
##  0.0330970401 -0.0811682370 -0.0625056984  0.0668571760  0.0308819015
##           V26           V27           V28           V29           V30
##  0.1440159315  0.0007501295  0.1067662912 -0.0422424251  0.1427961826
##           V31           V32           V33           V34           V35
## -0.0707237934 -0.1693982915 -0.0724560040 -0.0094649133  0.1450184074
##           V36           V37           V38           V39           V40
## -0.0430598023  0.0980891644  0.1239572231  0.0151652240 -0.1775869937
##           V41           V42           V43           V44           V45
##  0.0362143684  0.0439564207 -0.0383438084 -0.2212801400  0.1361946588
##           V46           V47           V48           V49           V50
## -0.0290113821  0.0915312195  0.1534932454  0.0141569258 -0.0296165144
```

ALL p-values:

```r
p_list = c()
for (i in colnames(multiple)){
  p_list <- c(p_list, t.test(multiple[i],conf.level = 0.90)$p.value)
}
p_list
```

```
##  [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
```

According to the results of T-test, Ignore the first 10 p-values, we found that there are several variables with P-value less than 0.1. They are v20(p=0.08299), v32(p=0.09303), v40(p=0.07315), v44(p=0.02475).

In the case of significance level at a =0.1,
there is a 90% probability that the mean of these variables is not 0. This conflicts with the information we got from the question.

Question 2.2

Based on the test results in step 1, calculate the following quantities: number of type I errors, FWER and FDP.

Answer:

Number of type I errors:

```
Number_of_type_I_errors = sum(p_list[11:50] <= 0.1)
Number_of_type_I_errors
```

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

FWER:

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

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

FDP:

```
Number_of_total_rejec <- sum(p_list[1:50] <= 0.1)
FDP <- Number_of_type_I_errors/Number_of_total_rejec
FDP
```

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

Question 2.3

Redo the multiple testing in step 1 with Bonferroni correction (set a = 0.1). Calculate the FWER of your new test results.

Answer:

New FWER with Bonferroni correction

```
FWER_Bonferroni_Correction <- 1-(1-(0.1/50))^50  #new a = a/m
FWER_Bonferroni_Correction
```

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

Question 2.4

Redo the multiple testing in step 1 with BH procedure (set a = 0.1). Calculate the FDP and FWER of your new test results. How does the results compared with the ones you obtained in step 2 and step 3?

Answer:

```
p_sorted <- sort(p_list)
for (i in 1:length(p_sorted)){
  if (p_sorted[i] <= i / 50 * 0.1){
    k <- i
  }
}

BH_nmber_of_type_I <- sum(p_list[11:50] <= p_sorted[k])
BH_number_of_correct <- sum(p_list[1:10] <= p_sorted[k])
BH_nmber_of_type_I
```

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

```
BH_number_of_correct
```

```
## [1] 10
```

FWER:

```
FWER_BH <- FWER <- 1-(1-p_sorted[k])^50
FWER_BH
```

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

FDP:

```
FDP <- BH_nmber_of_type_I / (BH_nmber_of_type_I + BH_number_of_correct)
FDP
```

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

FWER in step 2 is 0.9948462, FWER in step 3 is 0.09525318. After using Bonferroni correction, FWER became smaller.

Also, after using BH procedure, we can see from the results that both FWER and FDP are smaller.