CHL5224 Assignment#2

Whitney (Huei-Chen) Chiu
Student number 1004823596

Introduction

In large-scale genetic association studies, the scale of multiple hypothesis testing is often enormous [1]. To study different methodological approaches to correct for type 1 error, in this assignment, we consider an observed event $Y \sim N(170, 7^2)$ and a SNP with a minor allele frequency of 0.2. 10,000 replicates were simulated based on a sample size of 1000 and the distribution set above. Four types of genetic models: dominant, recessive, additive, and the genotypic model were built to take into account the possible genomic feature of the SNP.

By regressing Y to G (genotype obtained by SNP under Hardy Weinberg Equilibrium), under the assumption of four different models, we got 10,000 sets of p-values. The minimum p-value within each set was extracted to form another vector. Normally in single hypothesis testing, a p-value of 0.05 is used to reject the null hypothesis if the likelihood of the observed data under the null is 5%. However, in multiple hypothesis testings, the likelihood of incorrectly rejecting a null hypothesis, namely, making a type I error, increases. In cases when we conduct a family of tests on a set of data, the probability of making at least one type I error is called the family-wise error rate (FWE).

Various methodologies have been invented to control for false discovery rates brought by the family wise errors. In general, the correction methods can be classified into two categories: a single step or a step-wise procedure.

Methods

Single step procedure

The Bonferroni correction is an example of a single step procedure to correct for type I error. It is done by dividing the α by the number of tests conducted [2]. The new α level is then applied to each individual test to examine statistical significance. However, the single-step Bonferroni method is considered a overly conservative method which does not have enough ability to detect true differences when the tests are not mutually dependent [3].

Step-wise procedure

The Benjamini–Hochberg procedure adjusts for multiple comparisons by producing a graduated series of significance threshold using the formula: $\frac{\text{Rank of the p-value}}{\text{Total number of tests}} \times$ a specificed false discovery rate (α) [4]. The method involves ordering the original p-values by ascending order while the formula above is used to calculate a significance threshold for each rank of p-value. The first threshold gives the same threshold as the Bonferroni method, and the last threshold would be same as the unadjusted value of α [4].

Discussion

Correcting for type 1 error is a critical issue in multiple hypothesis testings; however, overemphasizing the prevention of type I errors could be at the cost of type II errors, which means a decreased ability in rejecting a false null hypothesis. It is also important to note that these adjustment methods all assume mutual independence between tests. However, this assumption is often violated in real-world studies. Based on

the results of the Pearson correlation test in our example, it is also evident that our four models were not mutually independent at all.

To conclude, in different scenarios when we decide the usage of different techniques to adjust for type I error, it is essential to consider the cost of type I error compared to type II error so that we could avoid overly suppressing α by sacrificing statistical power. The step-wise Benjamini–Hochberg procedure which gives an adjusted p-value that is not as stringent as the Bonferroni method is a more appropriate way to address the problem of type I error.

The histogram below presents the non-uniformly distributed minimum p-values of the four genetic models. Since the distribution is not uniform and the four tests are not mutually independent, applying a p-value of 0.05 would be too liberal while simply using the Bonferroni correction would provide a too conservative p-value. The blue line shows the p value of 0.05. The green line is the significance threshold provided by Bonferroni correction. The red line is the 5% quantile of the minimum p-value, which should be the optimal siginicance threshold if the ideal false discovery rate is set at 5%.

Codes:

```
## Set parameters
n= 1000
mu= 170
sd= 7

## Simulate 1000 samples for Y normal distribution
y= rnorm(n, mu, sd)

## Simulate 1000 samples for X multinomial distribution
x= rmultinom(1000, 1, prob=c(0.04,0.64,0.32)) #p^2, q^2, 2pq

## Four types of models
# Dominant model
i<-1
x.dom=1
for (i in 1:1000){
 if(x[1,i]==1 | x[3,i]==1){
   x.dom[i] <-1
 }
else {x.dom[i]<-0}
}

# Recessive model
i<-1
x.rec=1
for (i in 1:1000){
  if(x[1,i]==1){
    x.rec[i] <-1
  }  else {x.rec[i]<-0}
}

# Additive model #Has Gradient
i<-1
x.add=1
for (i in 1:1000){
  if(x[1,i]==1){
```

```
    x.add[i]<-2
  } else if (x[3,i]==1){
    x.add[i]<- 1}
    else {x.add[i]<-0}
}

# Genotypic model
i<-1
x.gen<-1
for (i in 1:1000){
  if(x[1,i]==1){
    x.gen[i] <-2}
    else if (x[3,i]==1){
    x.gen[i]<-1}
    else {x.gen[i]<-3}
}

x.gen<-as.character(x.gen)

data<- data.frame(y, x.dom, x.rec, x.add, x.gen)

m1<- lm(y~x.dom, data= data)
m2<- lm(y~x.rec, data= data)
m3<- lm(y~x.add, data= data)
m4<- lm(y~x.gen, data= data)

p1<- summary(m1)$coefficient[2,4]
p2<- summary(m2)$coefficient[2,4]
p3<- summary(m3)$coefficient[2,4]
p4<- summary(m4)$coefficient[2,4]
p.dat<-c(p1, p2, p3, p4)
p5<-min(p.dat[1], p.dat[2], p.dat[3], p.dat[4])

y.dist= rnorm(1000, 170, 7)
n.rep=10000
p.dat.rep=matrix(0,nrow=n.rep, ncol=5)
y.dist= matrix(0,nrow=n.rep, ncol=1000)

i.rep <- 1
n.rep <- 10000
p.dat.rep <- matrix(0,nrow=n.rep, ncol=5)
y.dist <- matrix(0,nrow=n.rep, ncol=1000)

for (i.rep in 1:n.rep){
  p.dat <- p.dat.rep[i.rep,]
  y.dist[i.rep,] <- rnorm (1000, mean=170, sd=7)
  p.dat[1] <-summary(lm(y.dist[i.rep,]~x.dom))$coefficients[2, 4]
  p.dat[2] <-summary(lm(y.dist[i.rep,]~x.rec))$coefficients[2, 4]
  p.dat[3] <-summary(lm(y.dist[i.rep,]~x.add))$coefficients[2, 4]
  p.dat[4] <-summary(m4 <- lm(y.dist[i.rep,]~x.gen))$coefficients[2, 4]
  p.dat[5] <- min(p.dat[1], p.dat[2], p.dat[3], p.dat[4])
  p.dat.rep[i.rep,]<-p.dat
}
```

```r
## Plot histogram
hist(p.dat.rep)
# p value= 0.05
abline(v=0.05,col="blue")
# Bonferroni correction, p= 0.0125
abline(v=0.0125,col="green")
# 5% data point, p= 0.02
abline(v=quantile(p.dat.rep[,5],0.05),col="red")
```

```r
# Pearson correlarion
cor.test(p.dat.rep[,1], p.dat.rep[,2], method= "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  p.dat.rep[, 1] and p.dat.rep[, 2]
## t = 3.1647, df = 9998, p-value = 0.001557
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.01204151 0.05120244
## sample estimates:
##        cor
## 0.03163412
```

```r
cor.test(p.dat.rep[,1], p.dat.rep[,3], method= "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  p.dat.rep[, 1] and p.dat.rep[, 3]
## t = 135.3, df = 9998, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7971869 0.8110369
## sample estimates:
##       cor
## 0.8042211
```

```r
cor.test(p.dat.rep[,1], p.dat.rep[,4], method= "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  p.dat.rep[, 1] and p.dat.rep[, 4]
## t = -1.6378, df = 9998, p-value = 0.1015
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.035966311  0.003223318
## sample estimates:
##         cor
## -0.01637779
```

```r
cor.test(p.dat.rep[,2], p.dat.rep[,3], method= "pearson")
```

```
##
##  Pearson's product-moment correlation
```

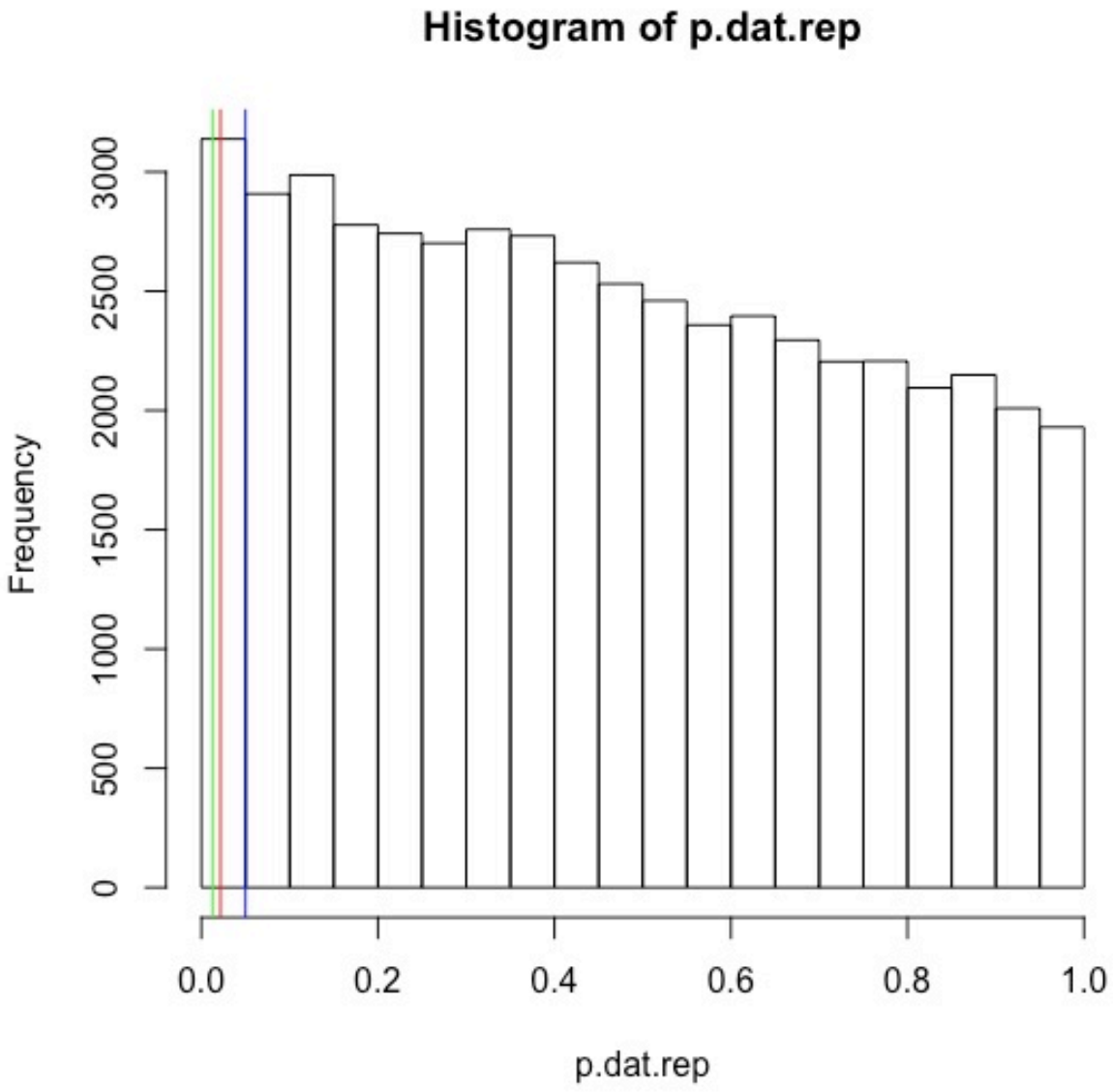
```
##
## data:  p.dat.rep[, 2] and p.dat.rep[, 3]
## t = 22.335, df = 9998, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1992497 0.2365876
## sample estimates:
##       cor
## 0.2179984
```

```
cor.test(p.dat.rep[,2], p.dat.rep[,4], method= "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  p.dat.rep[, 2] and p.dat.rep[, 4]
## t = 167.94, df = 9998, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8540162 0.8642785
## sample estimates:
##       cor
## 0.8592338
```

```
cor.test(p.dat.rep[,3], p.dat.rep[,4], method= "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  p.dat.rep[, 3] and p.dat.rep[, 4]
## t = 5.1831, df = 9998, p-value = 2.224e-07
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.03219964 0.07129477
## sample estimates:
##       cor
## 0.05176704
```



References

[1] Jelle J Goeman and Aldo Solari. Multiple hypothesis testing in genomics. *Statistics in medicine*, 33(11):1946–1978, 2014.

[2] Carlo Bonferroni. Teoria statistica delle classi e calcolo delle probabilita. *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze*, 8:3–62, 1936.

[3] Priya Ranganathan, CS Pramesh, and Marc Buyse. Common pitfalls in statistical analysis: The perils of multiple testing. *Perspectives in clinical research*, 7(2):106, 2016.

[4] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.