CHL5224 Assignment 2

Over-fitting, Subtle Multiple Hypothesis Testing and Possible Remedies

Fangming Liao, 1001374997

November 27, 2018

In genetic association studies of a phenotype Y and genotype G of a SNP, you might want to consider different genetic models: additive, dominant, recessive, or genotypic. Then you might want to consider the minimum p-value of all the p-values obtained from these different models, just to be "safe". So, is this an "optimal" approach since we are maximizing the test statistic in a way. What is the power of this approach?

This is a rather difficult question to answer even using simulation studies, because to study power, we have to simulate the data under alternatives, which are composite: which genetic model to use (additive, dominant, recessive, or genotypic), what is the effect size etc., assuming type 1 error (T1E) of your test statistic was well controlled to start with.

So, let's consider a simpler problem: study T1E of the minimum p-value approach. And if you noticed a T1E problem, how would you correct for it?

Parameter setting:

- How many simulation replicates? Say n.rep=10,000
- What is the sample size? Say n=1000
- What kind of trait? Say $Y \sim N(170, 7^2)$
- What kind of SNP? Say the MAF of the SNP is 0.2 and obtain G under HWE

Here are the R code and outputs for simulation study using additive, dominant, recessive and genotypic models.

```r
# parameters set-up
n     <- 1000 # sample size
n.rep <- 10000 # simulation replicates
p     <- 0.2 # MAF of the SNP

# simulate genotype frequency under HWE
nAA <- n*p^2
nAa <- n*2*p*(1-p)
naa <- n*(1-p)^2

# obtain the actual genotype vector of 0, 1 and 2
# for additive, dominant, recessive and genotypic models
Gadditive  <- c(rep(0,naa), rep(1,nAa), rep(2,nAA))
Gdominant  <- c(rep(0,naa), rep(1,nAa+nAA))
Grecessive <- c(rep(0,naa+nAa), rep(1,nAA))
Ggenotypic <- cbind(c(rep(1,nAa), rep(0,nAA+naa)),
                    c(rep(0,nAa), rep(1,nAA), rep(0,naa)))
```

Additive Model:

$$Y = \alpha + \beta G + e, e \sim N(0, \sigma^2)$$

where G is a vector of n_{aa} 0's, n_{Aa} 1's, and n_{AA} 2's.

The null hypothesis for this model is: $H_0 : \beta = 0$.

Genotypic Model:

$$Y = \alpha + \beta_1 I_{G=Aa} + \beta_2 I_{G=AA} + e, e \sim N(0, \sigma^2)$$

where G is a coding matrix of n_{Aa} 1's, $n_{AA} + n_{aa}$ 0's in the first column, and n_{Aa} 0's, n_{AA} 1's, n_{aa}

0's in the second column.

The null hypothesis for this model is: $H_0 : \beta_1 = \beta_2 = 0$.

Recessive Model:

$$Y = \alpha + \beta G + e, e \sim N(0, \sigma^2)$$

where G is a vector of $n_{aa} + n_{Aa}$ 0's and n_{AA} 1's.

The null hypothesis for this model is: $H_0 : \beta = 0$.

Dominant Model:

$$Y = \alpha + \beta G + e, e \sim N(0, \sigma^2)$$

where G is a vector of $n_{AA} + n_{Aa}$ 1's and n_{aa} 0's.

The null hypothesis for this model is: $H_0 : \beta = 0$.

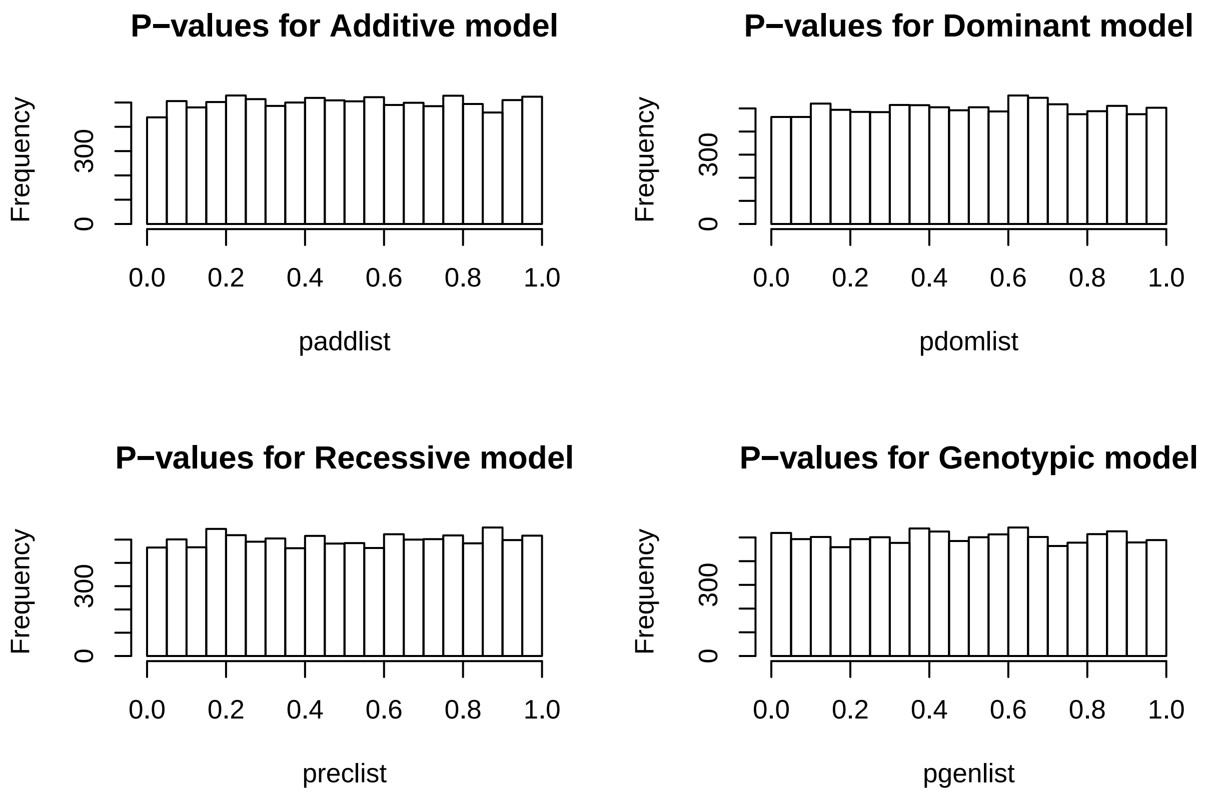
Then we fit each model for 10,000 times and get the p-values for each model, and the smallest one as the minimum.

```r
set.seed(1234)
# for each replicate, obtain the p values of all model and minimum p value among all
paddlist = pdomlist = preclist = pgenlist = pminlist = rep(0,n.rep)
for (i in 1:n.rep){
  Y <- rnorm(n,170,7) # traits
  # regression models for additive, dominant, recessive and genotypic models
  fit1 <- lm(Y ~ Gadditive)
  fit2 <- lm(Y ~ Gdominant)
  fit3 <- lm(Y ~ Grecessive)
  fit4 <- lm(Y ~ Ggenotypic)

  sm1 <- summary(fit1)
  sm2 <- summary(fit2)
  sm3 <- summary(fit3)
  sm4 <- summary(fit4)

  # p-values for each model
  paddlist[i] <- 1 - pf(sm1$fstatistic[1], sm1$fstatistic[2], sm1$fstatistic[3])
  pdomlist[i] <- 1 - pf(sm2$fstatistic[1], sm2$fstatistic[2], sm2$fstatistic[3])
  preclist[i] <- 1 - pf(sm3$fstatistic[1], sm3$fstatistic[2], sm3$fstatistic[3])
  pgenlist[i] <- 1 - pf(sm4$fstatistic[1], sm4$fstatistic[2], sm4$fstatistic[3])
  # minimum p-value
  pminlist[i] <- min(c(paddlist[i],pdomlist[i],preclist[i],pgenlist[i]))
}
```

```
# plot the histograms of p-values for each model
par(mfrow = c(2,2))
hist(paddlist, breaks=seq(0,1,by=0.05), main="P-values for Additive model")
hist(pdomlist, breaks=seq(0,1,by=0.05), main="P-values for Dominant model")
hist(preclist, breaks=seq(0,1,by=0.05), main="P-values for Recessive model")
hist(pgenlist, breaks=seq(0,1,by=0.05), main="P-values for Genotypic model")
```



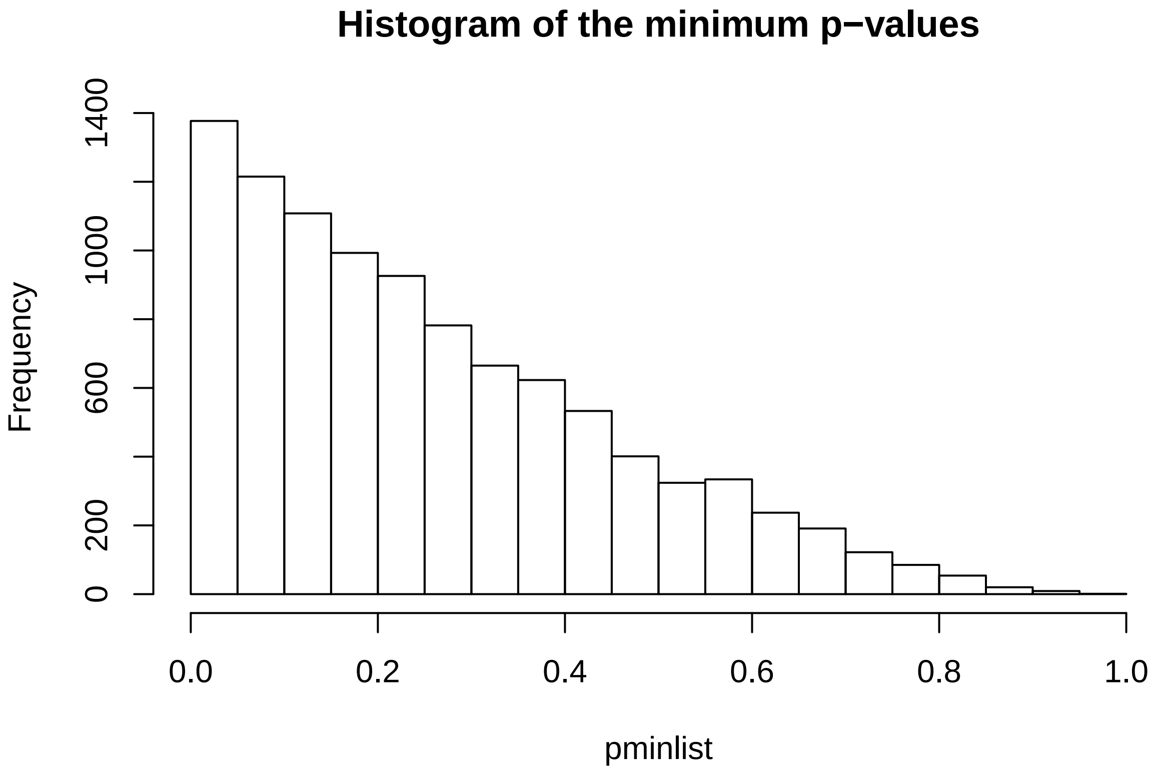
P−values for Additive model

P−values for Dominant model

P−values for Recessive model

P−values for Genotypic model

```
# histograms of the minimum p-values among models
par(mfrow = c(1,1))
hist(pminlist, breaks=seq(0,1,by=0.05), main="Histogram of the minimum p-values")
```



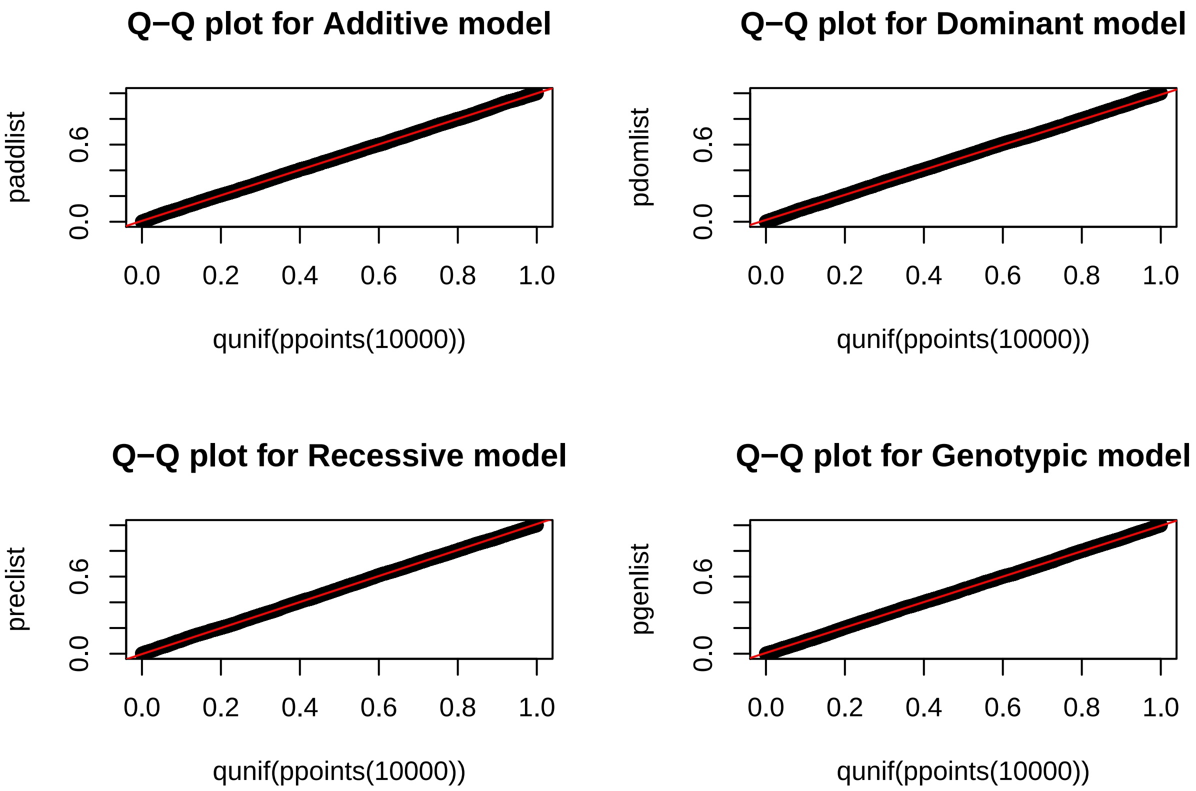
From the histogram we notice that the p-values for each individual test through all 10,000 replicates are uniformly distributed between 0 and 1. However, the minimum p-values among all 4 tests of each replicate is highly left skewed. By setting the bar size of our histogram to 0.05, it is easy to see the proportion of minumum p-values that are less than 0.05, which are in the first bar.

The results indicated that the type 1 error rate is high for our minimum p-values approach.

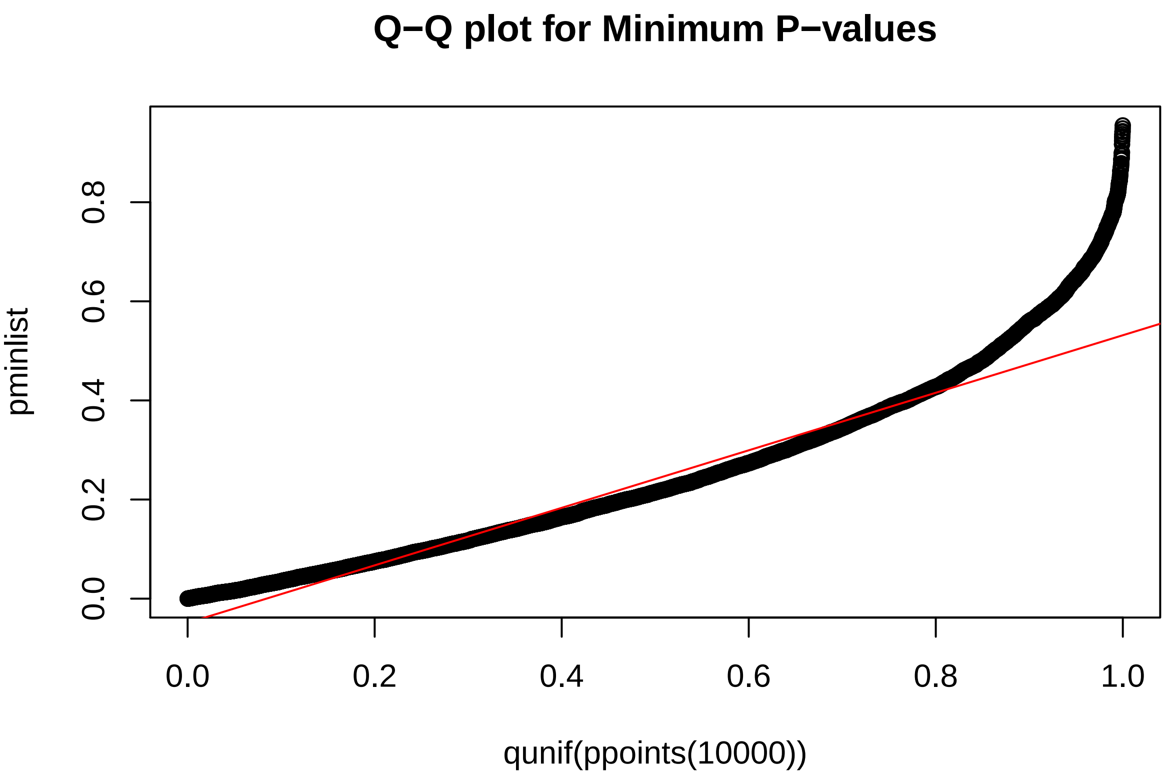
Additionally, we plotted the Quantile-Quantile plot for each model test and the minimum p-values, comparing to uniform distribution as following:

```r
# plot the qqplot of p-values for each model comparing to uniform dist'n
par(mfrow = c(2,2))
qqplot(qunif(ppoints(10000)), paddlist, main="Q-Q plot for Additive model")
qqline(paddlist, distribution = function(p) qunif(p), col = 2)
qqplot(qunif(ppoints(10000)), pdomlist, main="Q-Q plot for Dominant model")
qqline(pdomlist, distribution = function(p) qunif(p), col = 2)
qqplot(qunif(ppoints(10000)), preclist, main="Q-Q plot for Recessive model")
qqline(preclist, distribution = function(p) qunif(p), col = 2)
```

```
qqplot(qunif(ppoints(10000)), pgenlist, main="Q-Q plot for Genotypic model")
qqline(pgenlist, distribution = function(p) qunif(p), col = 2)
```



```
# qqplot of the minimum p-values among models
par(mfrow = c(1,1))
qqplot(qunif(ppoints(10000)), pminlist, main="Q-Q plot for Minimum P-values")
qqline(pminlist, distribution = function(p) qunif(p), col = 2)
```



Seeing from the Q-Q-plot, we found that although the quantile of ordered p-values of individual test model are pretty close to the uniform distribution's quantile, the ditribution of minimum p-values is not close to uniform.

To adjust the minimum p-vlaues, I first considered the Bonferroni correction; however, the Bonferroni correction is conservative when the tests are dependent. So we need check the independence of our tests on different models by checking the correlation of p-values.

```
set.seed(1234)
# check the correlation of p-values between tests
allpval <- cbind(paddlist,pdomlist,preclist,pgenlist)
cor(allpval)
```

```
##           paddlist   pdomlist     preclist     pgenlist
## paddlist 1.0000000 0.80725103  0.210667004  0.144403499
## pdomlist 0.8072510 1.00000000  0.020376543  0.159025526
## preclist 0.2106670 0.02037654  1.000000000 -0.009522452
## pgenlist 0.1444035 0.15902553 -0.009522452  1.000000000
```

From the covariance matrix, we found that the correlation between p-values of test for additive model and donimant model is 0.80725, which indicates a high correlation, and for between additive model and recessive model, additive model and genotypic model, dominant model and genotypic model can't be ignored as they are greater than 0.1.

```
# see how much fall into <0.05
quantile(pminlist,0.05)
```

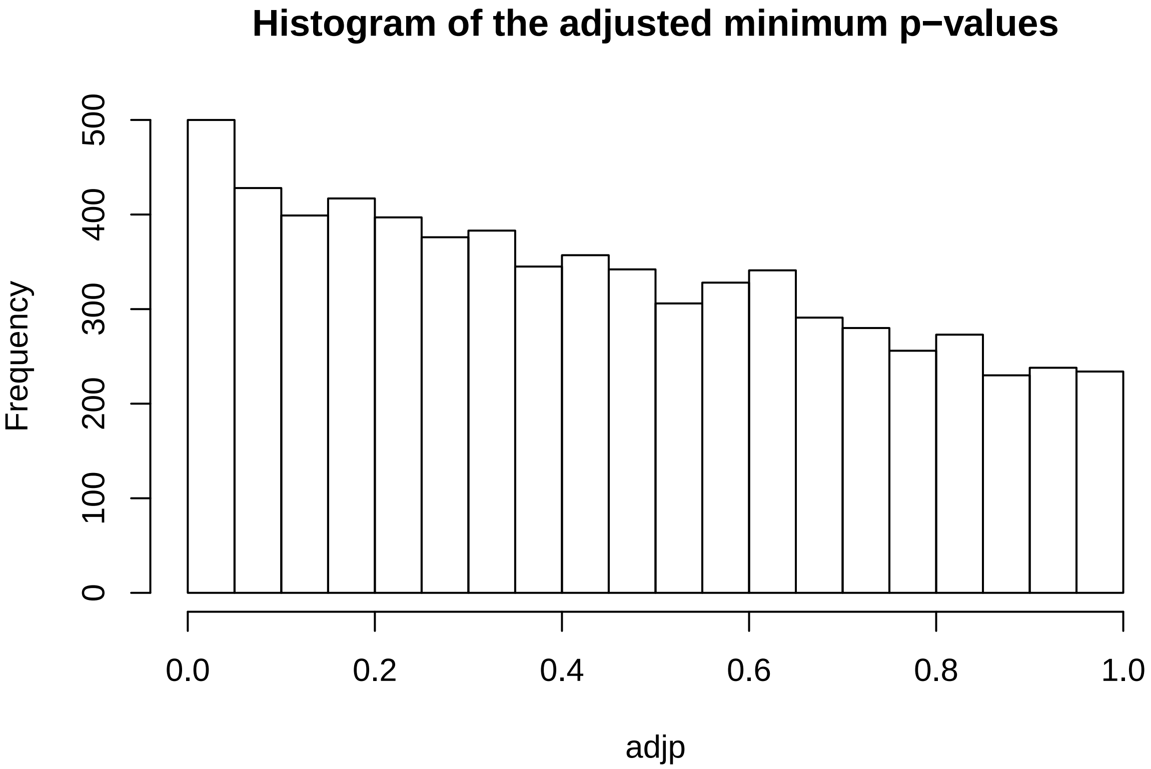
```
##         5%
## 0.01615397
```

By computing the number of minimum p-values that fall into the region smaller than 0.05, we simply find the 0.05 quantile of ordered minimum p-values, which gives us a result of 0.01615. This results indicates that we should look at a significant level of $\alpha = 0.01615$ instead of 0.05.

```
# get the scale for Bonferroni correction (between 1 and 4)
(alpha <- 0.05/quantile(pminlist,0.05))
```

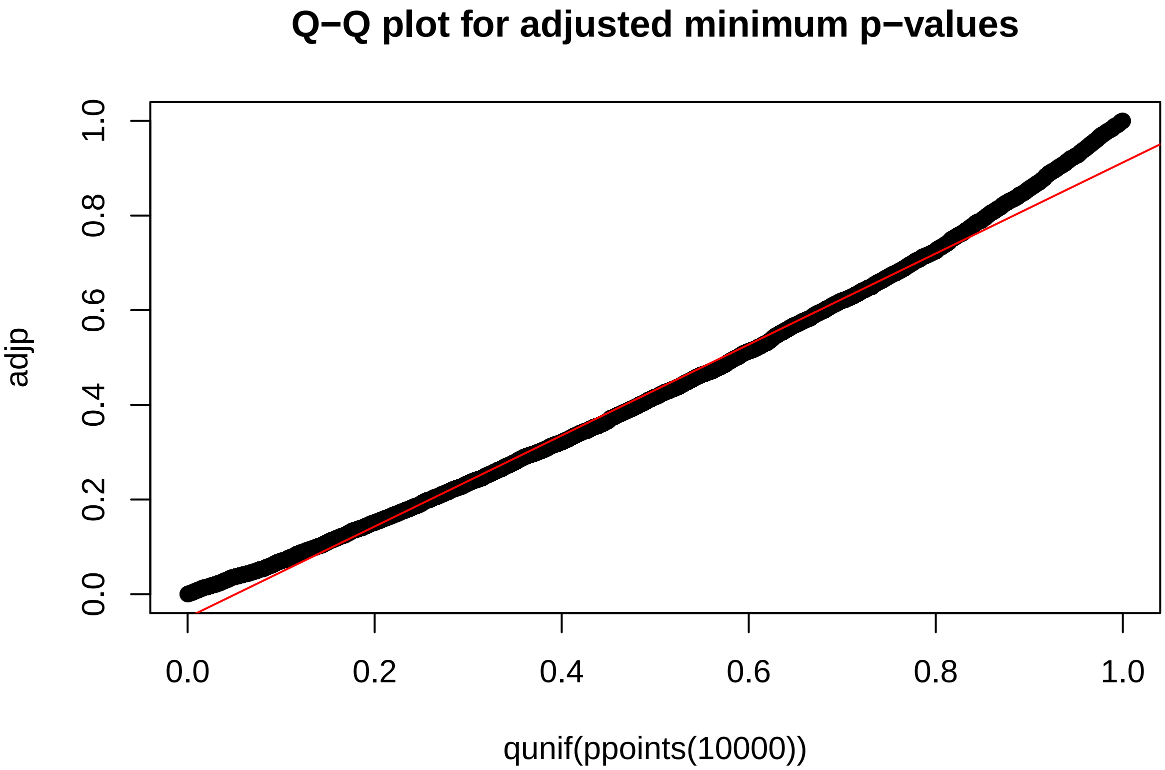
```
##       5%
## 3.095214
```

```
# adjusted minimum p-values
adjp<-pminlist*alpha
# histogram of adjusted minimum p-values
# remove p-values that are greater than 1
adjp <- adjp[adjp<=1]
hist(adjp,breaks=seq(0,1,by=0.05),
     main="Histogram of the adjusted minimum p-values")
```



Alternatively, we apply the Bonferroni correction using a coefficient of $\frac{0.05}{0.01615} = 3.095$ instead of the numnber of tests, 4, and only keep the ones that are smaller than or equal to 1. In this way, the adjusted minimum p-values have a distribution that looks much more uniform than the original one.

```
qqplot(qunif(ppoints(10000)), adjp,
       main="Q-Q plot for adjusted minimum p-values")
qqline(adjp, distribution = function(p) qunif(p), col = 2)
```



Moreover, the Quantile-Quantile plot for adjusted minimum p-values comparing to an uniform distribution also supports our conclusion that they are uniformly distributed.

Since we have only have one SNP to test in our study with 4 models, it is just a case of subtle multiple hypothesis testing, other p-value adjustment method such as Hochberg's and Hommel's methods are not suitable.