assign1-sta-1039580

Jiayu Wang

Student ID: 1039580

Q1a

The first 5 phenotypes in study1 and study2 are shown in the output.

```
#Read data
setwd("C:/Users/iefad/Desktop/course2021/bioinfosta/assignment/data-assignmen
t1")
gt1 <- read.csv('genotypes1.csv')
gt2 <- read.csv('genotypes2.csv')
gt3 <- read.csv('genotypes3.csv')
pt1 <- read.csv('phenotype1.csv')
pt2 <- read.csv('phenotype2.csv')

#check the study sizes. As shown in the output below, the study sizes are con
sistent with what's reported.
dim(gt1) # sample size for genotypes1

## [1] 500 201

dim(gt2) # sample size for genotypes2

## [1] 1500  201

dim(gt3) # sample size for genotypes3

## [1] 100 201

dim(pt1) # sample size for phenotypes1

## [1] 500   2

dim(pt2) # sample size for phenotypes2

## [1] 1500    2

#Show the first 5 phenotypes in study1 and study2.
pt1[1:5,2]  # in study 1

## [1] 26.7 23.5 22.8 19.3 27.4

pt2[1:5,2]  # in study 2

## [1]  TRUE  TRUE  TRUE  TRUE FALSE
```

Q1b

```
recordpvalue <- data.frame(genotype = colnames(gt1)[2:201], pvalue = rep(0, 2
00))
for (i in 1:200) {
  model <-  lm(pt1$BMI ~ gt1[,i+1])
  recordpvalue[i,2] = anova(model)[1,5]
}
recordpvalue # record p-value
```

```
##     genotype       pvalue
## 1     snp001 2.788045e-01
## 2     snp002 4.213277e-02
## 3     snp003 1.106544e-02
## 4     snp004 4.598057e-01
## 5     snp005 4.773127e-01
## 6     snp006 6.269859e-01
## 7     snp007 1.180413e-01
## 8     snp008 5.823757e-02
## 9     snp009 9.247002e-01
## 10    snp010 2.928088e-01
## 11    snp011 3.567580e-01
## 12    snp012 6.542079e-02
## 13    snp013 6.278523e-02
## 14    snp014 6.534726e-02
## 15    snp015 1.663554e-01
## 16    snp016 9.426174e-02
## 17    snp017 6.009006e-01
## 18    snp018 5.801509e-01
## 19    snp019 3.733518e-01
## 20    snp020 7.043405e-04
## 21    snp021 5.358903e-01
## 22    snp022 9.937614e-02
## 23    snp023 1.358734e-01
## 24    snp024 8.037127e-01
## 25    snp025 6.565804e-01
## 26    snp026 3.640620e-01
## 27    snp027 1.415343e-01
## 28    snp028 2.404250e-01
## 29    snp029 5.957152e-01
## 30    snp030 2.979699e-01
## 31    snp031 4.335758e-01
## 32    snp032 7.584683e-03
## 33    snp033 1.447568e-02
## 34    snp034 7.685264e-01
## 35    snp035 1.509939e-01
## 36    snp036 1.308951e-01
## 37    snp037 9.744083e-01
## 38    snp038 1.403803e-04
## 39    snp039 5.448880e-01
## 40    snp040 6.025946e-01
```

```
## 41    snp041 3.586497e-04
## 42    snp042 8.194404e-02
## 43    snp043 7.576538e-01
## 44    snp044 6.649132e-03
## 45    snp045 7.188082e-01
## 46    snp046 1.912356e-01
## 47    snp047 1.332697e-03
## 48    snp048 5.376787e-01
## 49    snp049 7.584683e-03
## 50    snp050 6.045586e-01
## 51    snp051 5.682905e-05
## 52    snp052 2.280280e-07
## 53    snp053 9.539833e-06
## 54    snp054 5.173863e-04
## 55    snp055 3.443949e-01
## 56    snp056 9.714320e-05
## 57    snp057 3.016446e-01
## 58    snp058 1.234194e-01
## 59    snp059 1.572343e-02
## 60    snp060 4.493258e-02
## 61    snp061 4.763588e-01
## 62    snp062 9.275089e-01
## 63    snp063 2.776576e-01
## 64    snp064 2.432992e-02
## 65    snp065 7.536183e-02
## 66    snp066 9.828575e-02
## 67    snp067 2.002265e-02
## 68    snp068 2.655050e-01
## 69    snp069 7.664848e-01
## 70    snp070 8.204722e-02
## 71    snp071 1.745189e-01
## 72    snp072 7.124668e-03
## 73    snp073 2.270115e-01
## 74    snp074 2.923243e-01
## 75    snp075 4.030894e-01
## 76    snp076 8.539780e-03
## 77    snp077 9.883839e-02
## 78    snp078 9.651996e-03
## 79    snp079 1.150581e-02
## 80    snp080 7.428456e-02
## 81    snp081 5.460508e-01
## 82    snp082 6.463677e-01
## 83    snp083 5.671337e-01
## 84    snp084 3.178181e-01
## 85    snp085 2.068897e-02
## 86    snp086 6.739211e-02
## 87    snp087 4.497546e-02
## 88    snp088 3.433718e-01
## 89    snp089 7.141285e-03
## 90    snp090 7.349284e-01
```

```
## 91    snp091 3.103389e-01
## 92    snp092 3.789345e-01
## 93    snp093 1.064869e-01
## 94    snp094 3.256670e-01
## 95    snp095 1.155049e-02
## 96    snp096 7.824368e-01
## 97    snp097 1.959854e-01
## 98    snp098 2.469217e-03
## 99    snp099 4.673300e-01
## 100   snp100 2.940438e-01
## 101   snp101 4.141442e-02
## 102   snp102 1.077333e-02
## 103   snp103 1.067146e-03
## 104   snp104 5.380541e-01
## 105   snp105 3.280120e-02
## 106   snp106 8.048262e-03
## 107   snp107 4.088464e-03
## 108   snp108 6.298973e-01
## 109   snp109 7.401832e-01
## 110   snp110 5.864720e-01
## 111   snp111 7.884391e-01
## 112   snp112 4.293351e-01
## 113   snp113 2.933660e-01
## 114   snp114 4.493641e-01
## 115   snp115 1.748858e-02
## 116   snp116 7.759868e-02
## 117   snp117 5.656057e-01
## 118   snp118 5.424069e-01
## 119   snp119 9.489495e-01
## 120   snp120 2.194668e-01
## 121   snp121 2.656927e-02
## 122   snp122 1.897578e-01
## 123   snp123 2.615075e-01
## 124   snp124 3.306293e-01
## 125   snp125 1.117821e-01
## 126   snp126 6.405270e-01
## 127   snp127 3.880675e-01
## 128   snp128 7.469934e-01
## 129   snp129 2.056248e-01
## 130   snp130 1.174946e-01
## 131   snp131 9.904980e-01
## 132   snp132 7.994084e-01
## 133   snp133 5.923786e-02
## 134   snp134 4.849095e-01
## 135   snp135 9.297724e-01
## 136   snp136 3.202203e-01
## 137   snp137 1.129403e-02
## 138   snp138 9.275942e-01
## 139   snp139 6.976210e-01
## 140   snp140 6.773687e-03
```

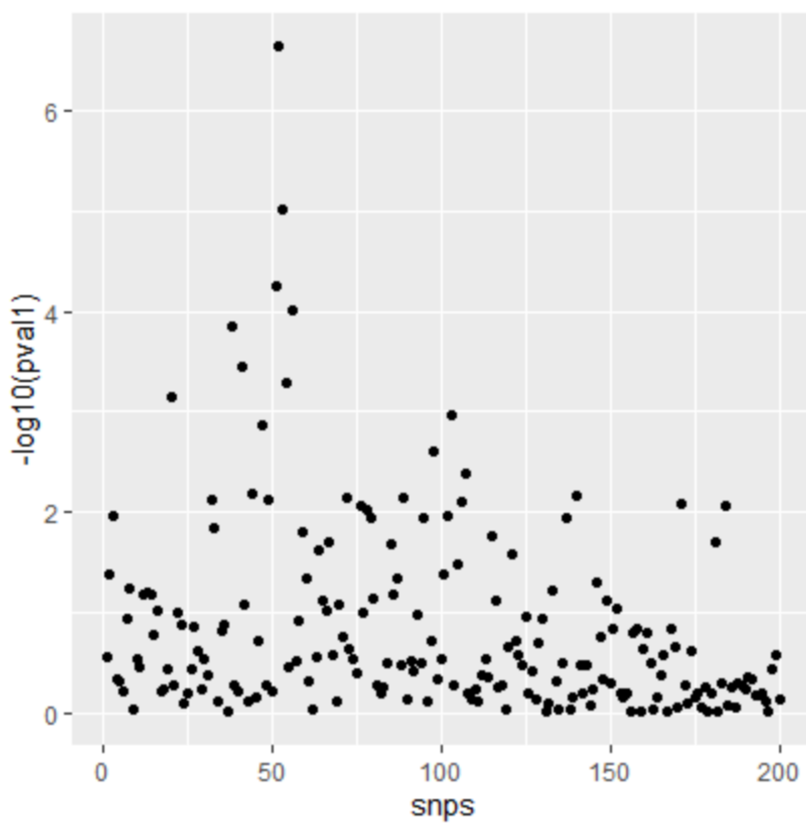
```
## 141   snp141 3.329425e-01
## 142   snp142 6.521029e-01
## 143   snp143 3.311811e-01
## 144   snp144 8.389249e-01
## 145   snp145 5.848875e-01
## 146   snp146 5.119488e-02
## 147   snp147 1.787775e-01
## 148   snp148 4.705875e-01
## 149   snp149 7.472226e-02
## 150   snp150 5.142107e-01
## 151   snp151 1.433640e-01
## 152   snp152 9.357016e-02
## 153   snp153 6.490185e-01
## 154   snp154 7.048606e-01
## 155   snp155 6.486447e-01
## 156   snp156 9.967293e-01
## 157   snp157 1.633182e-01
## 158   snp158 1.487430e-01
## 159   snp159 9.660547e-01
## 160   snp160 2.277225e-01
## 161   snp161 1.613559e-01
## 162   snp162 3.283292e-01
## 163   snp163 9.351326e-01
## 164   snp164 7.165016e-01
## 165   snp165 4.255025e-01
## 166   snp166 2.610420e-01
## 167   snp167 9.964303e-01
## 168   snp168 1.491455e-01
## 169   snp169 2.194073e-01
## 170   snp170 9.059113e-01
## 171   snp171 8.126962e-03
## 172   snp172 5.222187e-01
## 173   snp173 8.019461e-01
## 174   snp174 2.404460e-01
## 175   snp175 7.108361e-01
## 176   snp176 6.514491e-01
## 177   snp177 8.921099e-01
## 178   snp178 5.559946e-01
## 179   snp179 9.874595e-01
## 180   snp180 6.370010e-01
## 181   snp181 2.000435e-02
## 182   snp182 9.795191e-01
## 183   snp183 5.187020e-01
## 184   snp184 8.827462e-03
## 185   snp185 8.606555e-01
## 186   snp186 5.593104e-01
## 187   snp187 8.837700e-01
## 188   snp188 4.991781e-01
## 189   snp189 5.381265e-01
## 190   snp190 5.873973e-01
```

```
## 191    snp191 4.364517e-01
## 192    snp192 4.659470e-01
## 193    snp193 6.878316e-01
## 194    snp194 6.756790e-01
## 195    snp195 6.310356e-01
## 196    snp196 7.883439e-01
## 197    snp197 9.943127e-01
## 198    snp198 3.735410e-01
## 199    snp199 2.639042e-01
## 200    snp200 7.562790e-01
```

Q1c

From the plot we can conclude that most p-values are larger than 0.01, which indicates a less significant association between the corresponding SNPs and BMI. The peak y value is higher than 6, which stands for less than e-6 p-value and the strongest association among these SNPs.

```
library("ggplot2")
snps <- 1:200
pval1 <- recordpvalue$pvalue
ggplot(recordpvalue, aes(snps, -log10(pval1))) + geom_point()
```



Q1d

As shown in output, snp052 has the smallest p-value at 2.28028e-07.

```
recordpvalue[order(recordpvalue$pvalue), ][1,]
```

```
##    genotype      pvalue
## 52   snp052 2.28028e-07
```

Q2a

```
colnames(pt1)[2] = 'overweight'
pt1[,2] = as.logical(pt1[,2] > 25) # treat phenotypes from study1 the same as
 study2
pt = rbind(pt1,pt2)  #creating a single data frame for the phenotype
gt = rbind(gt1,gt2)  ##creating a single data frame for the genotype
```

Q2b

```
combinepvalue <- data.frame(genotype = colnames(gt)[2:201], pvalue = rep(0, 2
00))
for (i in 1:200) {
  model <-  lm(as.numeric(pt$overweight) ~ as.numeric(gt[,i+1]))
  combinepvalue[i,2] = summary(model)$coefficients[2,4]
}
combinepvalue
```

```
##     genotype       pvalue
## 1     snp001 7.347895e-01
## 2     snp002 2.754467e-02
## 3     snp003 7.606861e-01
## 4     snp004 6.202933e-01
## 5     snp005 6.447473e-02
## 6     snp006 3.606788e-01
## 7     snp007 1.539003e-01
## 8     snp008 1.039244e-01
## 9     snp009 3.465682e-01
## 10    snp010 3.137228e-01
## 11    snp011 5.913890e-01
## 12    snp012 2.078579e-03
## 13    snp013 4.423170e-01
## 14    snp014 1.616898e-01
## 15    snp015 4.813904e-01
## 16    snp016 4.405304e-01
## 17    snp017 1.845063e-01
## 18    snp018 1.504788e-01
## 19    snp019 7.077208e-01
## 20    snp020 1.199570e-01
## 21    snp021 3.729892e-01
## 22    snp022 1.608155e-01
## 23    snp023 3.175448e-03
```

```
## 24    snp024 3.381555e-01
## 25    snp025 1.552742e-01
## 26    snp026 7.442203e-01
## 27    snp027 6.283863e-03
## 28    snp028 1.737209e-01
## 29    snp029 3.273679e-01
## 30    snp030 6.200770e-01
## 31    snp031 9.924207e-01
## 32    snp032 2.323039e-01
## 33    snp033 2.830157e-04
## 34    snp034 6.873381e-01
## 35    snp035 8.799028e-01
## 36    snp036 4.630821e-01
## 37    snp037 9.218493e-01
## 38    snp038 2.394928e-07
## 39    snp039 7.933904e-01
## 40    snp040 3.574962e-01
## 41    snp041 8.027248e-07
## 42    snp042 5.996413e-01
## 43    snp043 9.694138e-01
## 44    snp044 1.513415e-04
## 45    snp045 3.001582e-01
## 46    snp046 2.954667e-04
## 47    snp047 8.563084e-04
## 48    snp048 4.215389e-01
## 49    snp049 4.629563e-01
## 50    snp050 8.343850e-01
## 51    snp051 1.090388e-13
## 52    snp052 2.853124e-14
## 53    snp053 5.688590e-11
## 54    snp054 7.754202e-10
## 55    snp055 5.104635e-03
## 56    snp056 1.073751e-10
## 57    snp057 1.996013e-01
## 58    snp058 7.938272e-01
## 59    snp059 4.923808e-03
## 60    snp060 8.173204e-03
## 61    snp061 6.127101e-01
## 62    snp062 9.823904e-01
## 63    snp063 1.379992e-03
## 64    snp064 1.357014e-02
## 65    snp065 8.083341e-01
## 66    snp066 9.884708e-01
## 67    snp067 1.309211e-01
## 68    snp068 1.035369e-01
## 69    snp069 1.239875e-02
## 70    snp070 8.021752e-01
## 71    snp071 4.359181e-01
## 72    snp072 2.401995e-02
## 73    snp073 2.131951e-01
```

```
## 74    snp074 6.901060e-02
## 75    snp075 3.732020e-02
## 76    snp076 9.018782e-03
## 77    snp077 3.497538e-02
## 78    snp078 1.248096e-02
## 79    snp079 1.925664e-04
## 80    snp080 4.062044e-02
## 81    snp081 5.903523e-01
## 82    snp082 7.499138e-01
## 83    snp083 3.689904e-01
## 84    snp084 7.171029e-02
## 85    snp085 1.328008e-01
## 86    snp086 7.349165e-05
## 87    snp087 8.088520e-02
## 88    snp088 3.147495e-01
## 89    snp089 7.030465e-05
## 90    snp090 7.889398e-01
## 91    snp091 1.619470e-01
## 92    snp092 4.602575e-01
## 93    snp093 1.779591e-01
## 94    snp094 6.758913e-01
## 95    snp095 3.509545e-02
## 96    snp096 4.391856e-01
## 97    snp097 4.529272e-02
## 98    snp098 6.269115e-05
## 99    snp099 7.065321e-01
## 100   snp100 5.788417e-01
## 101   snp101 4.175910e-04
## 102   snp102 4.991869e-04
## 103   snp103 1.053401e-04
## 104   snp104 8.646844e-01
## 105   snp105 1.986769e-10
## 106   snp106 5.010754e-10
## 107   snp107 7.394690e-11
## 108   snp108 4.495587e-01
## 109   snp109 6.431232e-01
## 110   snp110 3.228068e-02
## 111   snp111 6.020478e-01
## 112   snp112 5.954196e-01
## 113   snp113 9.355466e-01
## 114   snp114 4.851323e-01
## 115   snp115 2.397480e-02
## 116   snp116 1.382865e-01
## 117   snp117 7.083962e-01
## 118   snp118 7.101746e-01
## 119   snp119 2.717641e-01
## 120   snp120 1.256549e-01
## 121   snp121 1.388144e-02
## 122   snp122 1.794045e-01
## 123   snp123 8.610190e-01
```

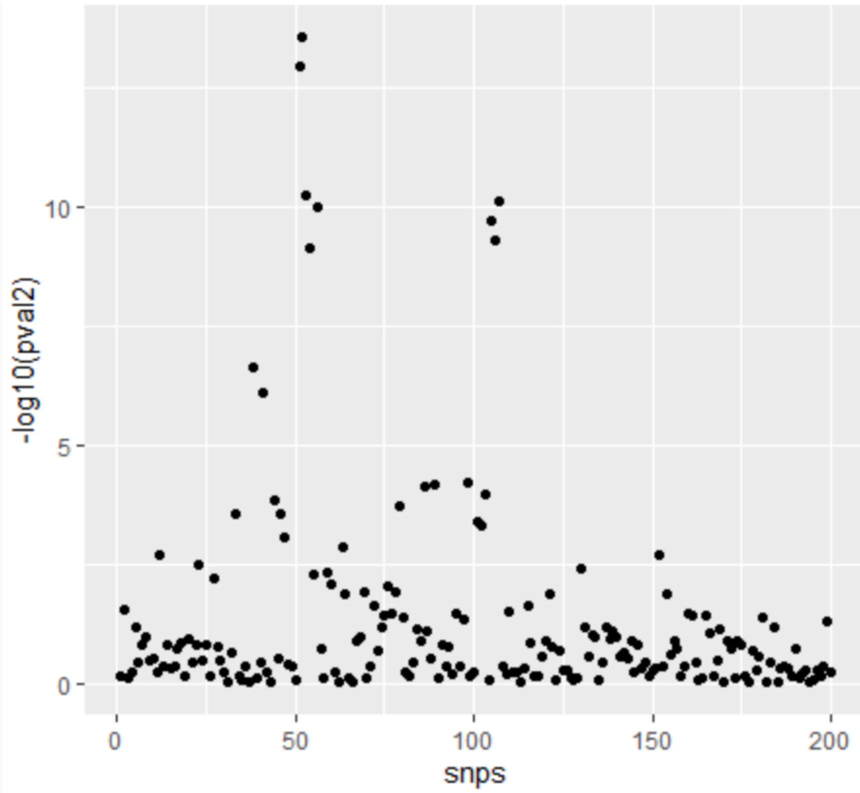
```
## 124   snp124 2.071795e-01
## 125   snp125 5.346964e-01
## 126   snp126 5.247947e-01
## 127   snp127 7.852816e-01
## 128   snp128 8.616783e-01
## 129   snp129 8.205817e-01
## 130   snp130 4.080885e-03
## 131   snp131 6.930556e-02
## 132   snp132 2.771219e-01
## 133   snp133 1.024566e-01
## 134   snp134 1.127692e-01
## 135   snp135 8.977938e-01
## 136   snp136 3.855438e-01
## 137   snp137 6.498843e-02
## 138   snp138 1.176825e-01
## 139   snp139 8.496159e-02
## 140   snp140 1.099622e-01
## 141   snp141 2.678742e-01
## 142   snp142 2.237624e-01
## 143   snp143 3.058052e-01
## 144   snp144 1.361713e-01
## 145   snp145 6.114179e-01
## 146   snp146 1.601984e-01
## 147   snp147 4.910927e-01
## 148   snp148 3.647457e-01
## 149   snp149 7.311475e-01
## 150   snp150 5.327648e-01
## 151   snp151 4.865474e-01
## 152   snp152 2.003915e-03
## 153   snp153 4.387514e-01
## 154   snp154 1.369534e-02
## 155   snp155 2.422154e-01
## 156   snp156 1.296499e-01
## 157   snp157 1.827772e-01
## 158   snp158 6.965121e-01
## 159   snp159 4.464146e-01
## 160   snp160 3.487072e-02
## 161   snp161 3.844111e-02
## 162   snp162 3.543502e-01
## 163   snp163 9.039203e-01
## 164   snp164 7.620743e-01
## 165   snp165 3.973554e-02
## 166   snp166 9.036475e-02
## 167   snp167 7.380562e-01
## 168   snp168 3.257838e-01
## 169   snp169 7.338117e-02
## 170   snp170 9.812604e-01
## 171   snp171 1.303396e-01
## 172   snp172 1.935454e-01
## 173   snp173 7.861820e-01
```

```
## 174   snp174 1.309960e-01
## 175   snp175 1.598008e-01
## 176   snp176 7.310219e-01
## 177   snp177 9.764182e-01
## 178   snp178 2.161579e-01
## 179   snp179 5.274178e-01
## 180   snp180 2.807450e-01
## 181   snp181 4.359697e-02
## 182   snp182 9.086116e-01
## 183   snp183 3.615387e-01
## 184   snp184 6.723081e-02
## 185   snp185 9.737579e-01
## 186   snp186 4.785079e-01
## 187   snp187 4.483336e-01
## 188   snp188 4.816559e-01
## 189   snp189 7.138295e-01
## 190   snp190 1.957596e-01
## 191   snp191 8.033078e-01
## 192   snp192 6.672748e-01
## 193   snp193 5.409684e-01
## 194   snp194 9.344393e-01
## 195   snp195 8.568820e-01
## 196   snp196 5.433710e-01
## 197   snp197 7.324146e-01
## 198   snp198 4.587318e-01
## 199   snp199 5.304996e-02
## 200   snp200 6.074201e-01
```

Q2c

From the plot we can conclude that most p-values are still larger than 0.01, which means weaker association with the overweight state. However, compared with the plot before combining data from study1 and study2, peak y values are higher and less y values are below 2, which indicates an overall stronger association with the overweight state(trait) here in this plot.

```
pval2 <- combinepvalue$pvalue
ggplot(combinepvalue, aes(x=snps, y=-log10(pval2))) + geom_point()
```



Q2d

As shown in output, snp052 has the smallest pvalue at 2.853124e-14

```
sortedpvalue <- combinepvalue[order(combinepvalue$pvalue), ]
sortedpvalue[1,]
```

```
##    genotype       pvalue
## 52   snp052 2.853124e-14
```

Q2e(i)

The number is 16.

```
m = length(combinepvalue$pvalue)
sum(combinepvalue$pvalue < 0.05/m)
```

```
## [1] 16
```

Q2e(ii)

The number is 29.

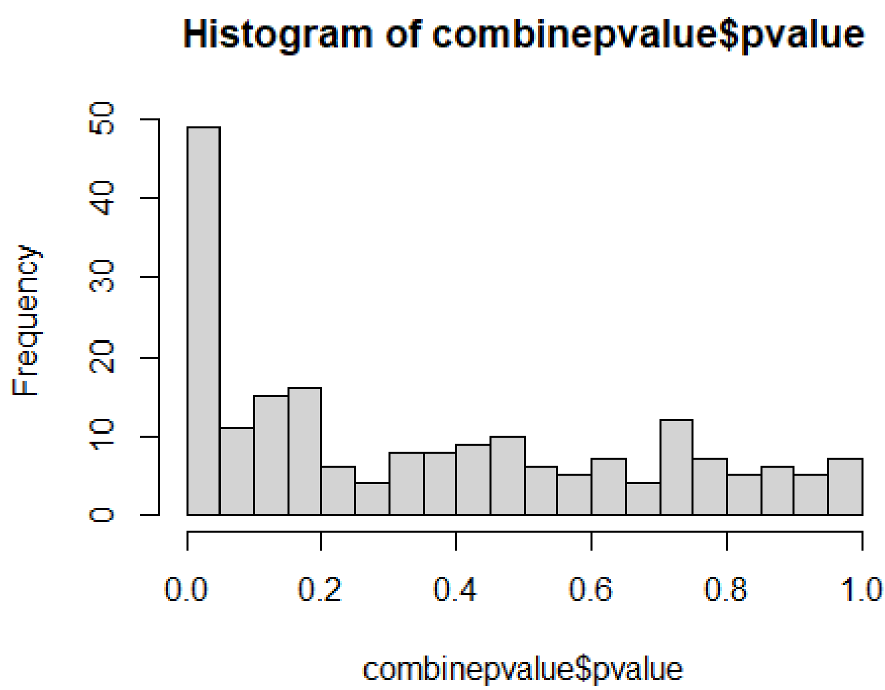
```
max(which(sort(combinepvalue$pvalue) <= 0.05*seq(1, m)/m))
```

```
## [1] 29
```

Q2e(iii)

The number is 21, the resulting FDR estimate is 0.007407407 as shown in output.

```r
hist(combinepvalue$pvalue,n=20)
```




```r
nsig = sum(combinepvalue$pvalue < 0.001)
nullfrac = mean(combinepvalue$pvalue>0.1)/0.9
falsep = nullfrac*m*0.001
FDR = falsep/nsig
print(list(nsig,FDR))
```

```
## [[1]]
## [1] 21
##
## [[2]]
## [1] 0.007407407
```

Q2f

I would report the number of significantly associated SNPs as 29 by using BH method. BH and Storey methods are better than Bonferroni since Bonferroni is too strict and can miss valuable true discovery. If choosing between BH and Storey, here I think the FDR of Storey method, 0.007407407, might be too strict. So I will report the 29 SNPs as significantly associated and the possibility of error (here I use False Discovery Rate) would be less than 0.05.

Q3a

The SNPs with 8 smallest p-values and their p-values are shown as output.

```
combinepvalue[order(combinepvalue$pvalue),][1:8, ]
```

```
##     genotype       pvalue
## 52    snp052 2.853124e-14
## 51    snp051 1.090388e-13
## 53    snp053 5.688590e-11
## 107   snp107 7.394690e-11
## 56    snp056 1.073751e-10
## 105   snp105 1.986769e-10
## 106   snp106 5.010754e-10
## 54    snp054 7.754202e-10
```

Q3b

The correlation matrix is shown in output.

The linkage disequilibrium between some pairs of SNPs are relatively high with correlation higher than 0.5.

```
small8 <- combinepvalue[order(combinepvalue$pvalue),][1:8, ]
cormatrix <- matrix(ncol = 8, nrow = 8)
colnames(cormatrix) <- small8$genotype
rownames(cormatrix) <- small8$genotype

index <- as.numeric(row.names(small8))

for (i in 1:8) {
  for (j in 1:8) {
    cormatrix[i,j] <-
cor(as.numeric(gt[,index[i]+1]),as.numeric(gt[,index[j]+1]))
  }
}
cormatrix
```

```
##           snp052    snp051    snp053    snp107    snp056    snp105    snp106
## snp052 1.0000000 0.8547424 0.8965812 0.1462807 0.7937844 0.1312771 0.1634972
## snp051 0.8547424 1.0000000 0.7808815 0.1395784 0.9012220 0.1253443 0.1528309
## snp053 0.8965812 0.7808815 1.0000000 0.1334606 0.7382154 0.1253515 0.1503977
## snp107 0.1462807 0.1395784 0.1334606 1.0000000 0.1600531 0.9400909 0.9379026
## snp056 0.7937844 0.9012220 0.7382154 0.1600531 1.0000000 0.1419182 0.1704282
## snp105 0.1312771 0.1253443 0.1253515 0.9400909 0.1419182 1.0000000 0.9046226
## snp106 0.1634972 0.1528309 0.1503977 0.9379026 0.1704282 0.9046226 1.0000000
## snp054 0.7777999 0.8943596 0.8066411 0.1300315 0.8526201 0.1188114 0.1458209
```

```
##           snp054
## snp052 0.7777999
## snp051 0.8943596
## snp053 0.8066411
## snp107 0.1300315
## snp056 0.8526201
## snp105 0.1188114
## snp106 0.1458209
## snp054 1.0000000
```

```
sum(cormatrix>0.5) #check the correlation > 0.5 which might be helpful to con
clude
```

```
## [1] 34
```

Q3c

Tag SNPs are shown in the output.

```
i <- 1
j <- 1
while (i <= 2){
  while (j <= nrow(cormatrix)){
    if (cormatrix[i,j] > 0.5 & (i != j)){
      cormatrix <- cormatrix[-j,-j]
    }
    else {
      j <- j+1
    }
  }
  j <- 1
  i <- i+1
}
cormatrix
```

```
##           snp052    snp107
## snp052 1.0000000 0.1462807
## snp107 0.1462807 1.0000000
```

Q4a

The parameter estimates and standard errors are shown in the output.

```
overweightstate <- pt$overweight
has_052 <- as.numeric(gt[,53])
has_107 <- as.numeric(gt[,108])

fit1 = glm(overweightstate ~ has_052 + has_107,family="binomial")
coef(summary(fit1))[2:3,1:2]
```

```
##           Estimate Std. Error
## has_052 0.5029238 0.07573746
## has_107 0.4025903 0.07375765
```

Q4b

The OR for SNP052 and SNP107 are 1.653549 and 1.495694 respectively.

Odds ratio = Odds(x1+1)/Odds(x1) = exp(β0 + β1(x1+1) + β2x2)/exp(β0 + β1x1 + β2x2) = exp(β1). Here ORs for SNP052 and SNP107 are greater than 1, which means the associations between overwight state and SNP052, and between overwight state and SNP107. And since OR for SNP052 is higher than OR for SNP107, association between overweight state and SNP052 is stronger than that for SNP107.

```
beta052 <- as.numeric(coef(fit1)[2])
beta107 <- as.numeric(coef(fit1)[3])
```

```
OR_052 <- exp(beta052)
OR_107 <- exp(beta107)
```

```
OR_052
```

```
## [1] 1.653549

OR_107

## [1] 1.495694
```

Q4c

C.I for SNP052 is (1.425441, 1.918160). C.I for SNP052 is (1.294375, 1.728325).

```
#Calculate SE
se_052 <- coef(summary(fit1))[2,2]
se_107 <- coef(summary(fit1))[3,2]
se_052

## [1] 0.07573746

se_107

## [1] 0.07375765

#Calculate C.I.
C.I_052 <- exp(beta052 + c(-1,1) * qnorm(0.975) * se_052)
C.I_107 <- exp(beta107 + c(-1,1) * qnorm(0.975) * se_107)

C.I_052
```

```
## [1] 1.425441 1.918160
```

```
C.I_107
```

```
## [1] 1.294375 1.728325
```

Q5a

The risk is as shown in the output.

```
new <- data.frame(has_052 = as.numeric(gt3[,53]), has_107 = as.numeric(gt3[,1
08]))
prerisk <- predict.glm(fit1,new,type = "response")
prerisk
```

```
##         1         2         3         4         5         6         7         8
## 0.5803120 0.2526774 0.2526774 0.2526774 0.2526774 0.4306492 0.2526774 0.3585968
##         9        10        11        12        13        14        15        16
## 0.3585968 0.2526774 0.4554015 0.3358612 0.2526774 0.2526774 0.2526774 0.2526774
##        17        18        19        20        21        22        23        24
## 0.2526774 0.3358612 0.4306492 0.2526774 0.3358612 0.4803761 0.2526774 0.5556981
##        25        26        27        28        29        30        31        32
## 0.4554015 0.4306492 0.3585968 0.3358612 0.2526774 0.5556981 0.2526774 0.2526774
##        33        34        35        36        37        38        39        40
## 0.4554015 0.2526774 0.3585968 0.2526774 0.4306492 0.2526774 0.4554015 0.3358612
##        41        42        43        44        45        46        47        48
## 0.5556981 0.2526774 0.2526774 0.2526774 0.2526774 0.3358612 0.3358612 0.2526774
##        49        50        51        52        53        54        55        56
## 0.2526774 0.2526774 0.3585968 0.4554015 0.4803761 0.3585968 0.2526774 0.3585968
##        57        58        59        60        61        62        63        64
## 0.2526774 0.3358612 0.3358612 0.2526774 0.3585968 0.4554015 0.2526774 0.2526774
##        65        66        67        68        69        70        71        72
## 0.3585968 0.3358612 0.4306492 0.3358612 0.3358612 0.4306492 0.2526774 0.4554015
##        73        74        75        76        77        78        79        80
## 0.2526774 0.2526774 0.4306492 0.2526774 0.2526774 0.2526774 0.4554015 0.2526774
##        81        82        83        84        85        86        87        88
## 0.3358612 0.4554015 0.4554015 0.3358612 0.3358612 0.2526774 0.2526774 0.4554015
##        89        90        91        92        93        94        95        96
## 0.2526774 0.4803761 0.3358612 0.3585968 0.5556981 0.3358612 0.3585968 0.3358612
##        97        98        99       100
## 0.5556981 0.2526774 0.2526774 0.5556981
```

Q5b

The risk is as shown in the output.

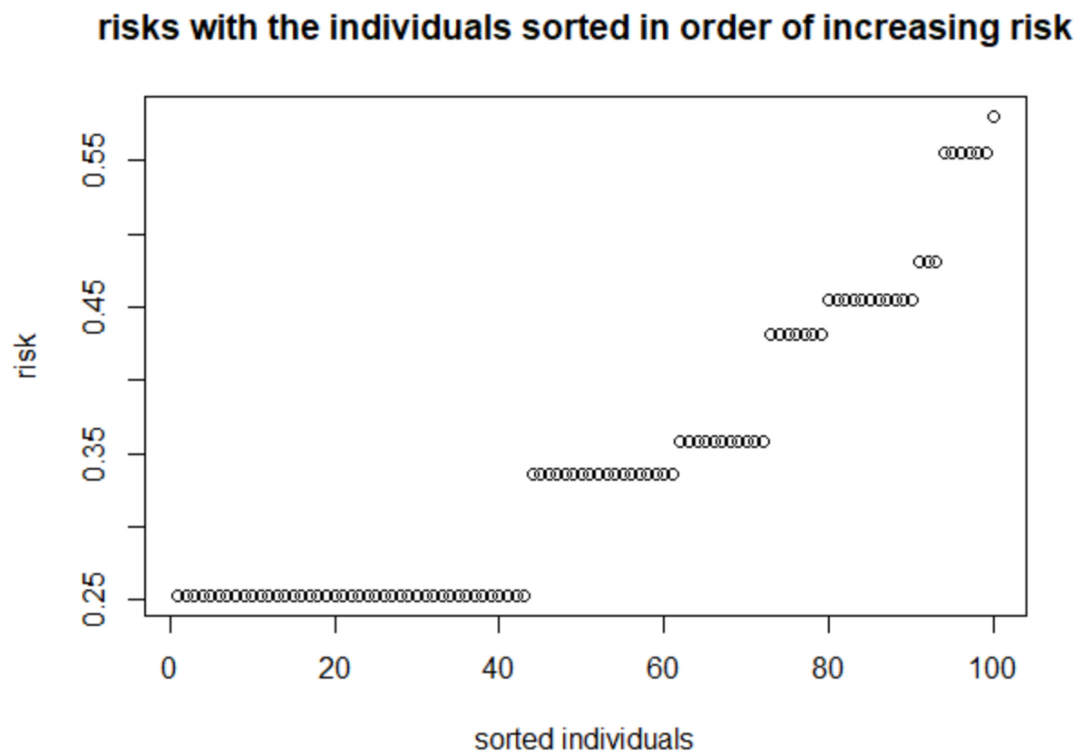
```
premodel <- data.frame(individuals = gt3$X, risk = prerisk)
premodel[order(premodel$risk), ]
```

```
##     individuals      risk
## 2     indiv2002 0.2526774
## 3     indiv2003 0.2526774
## 4     indiv2004 0.2526774
## 5     indiv2005 0.2526774
```

```
## 7     indiv2007 0.2526774
## 10    indiv2010 0.2526774
## 13    indiv2013 0.2526774
## 14    indiv2014 0.2526774
## 15    indiv2015 0.2526774
## 16    indiv2016 0.2526774
## 17    indiv2017 0.2526774
## 20    indiv2020 0.2526774
## 23    indiv2023 0.2526774
## 29    indiv2029 0.2526774
## 31    indiv2031 0.2526774
## 32    indiv2032 0.2526774
## 34    indiv2034 0.2526774
## 36    indiv2036 0.2526774
## 38    indiv2038 0.2526774
## 42    indiv2042 0.2526774
## 43    indiv2043 0.2526774
## 44    indiv2044 0.2526774
## 45    indiv2045 0.2526774
## 48    indiv2048 0.2526774
## 49    indiv2049 0.2526774
## 50    indiv2050 0.2526774
## 55    indiv2055 0.2526774
## 57    indiv2057 0.2526774
## 60    indiv2060 0.2526774
## 63    indiv2063 0.2526774
## 64    indiv2064 0.2526774
## 71    indiv2071 0.2526774
## 73    indiv2073 0.2526774
## 74    indiv2074 0.2526774
## 76    indiv2076 0.2526774
## 77    indiv2077 0.2526774
## 78    indiv2078 0.2526774
## 80    indiv2080 0.2526774
## 86    indiv2086 0.2526774
## 87    indiv2087 0.2526774
## 89    indiv2089 0.2526774
## 98    indiv2098 0.2526774
## 99    indiv2099 0.2526774
## 12    indiv2012 0.3358612
## 18    indiv2018 0.3358612
## 21    indiv2021 0.3358612
## 28    indiv2028 0.3358612
## 40    indiv2040 0.3358612
## 46    indiv2046 0.3358612
## 47    indiv2047 0.3358612
## 58    indiv2058 0.3358612
## 59    indiv2059 0.3358612
## 66    indiv2066 0.3358612
## 68    indiv2068 0.3358612
```

```
## 69    indiv2069 0.3358612
## 81    indiv2081 0.3358612
## 84    indiv2084 0.3358612
## 85    indiv2085 0.3358612
## 91    indiv2091 0.3358612
## 94    indiv2094 0.3358612
## 96    indiv2096 0.3358612
## 8     indiv2008 0.3585968
## 9     indiv2009 0.3585968
## 27    indiv2027 0.3585968
## 35    indiv2035 0.3585968
## 51    indiv2051 0.3585968
## 54    indiv2054 0.3585968
## 56    indiv2056 0.3585968
## 61    indiv2061 0.3585968
## 65    indiv2065 0.3585968
## 92    indiv2092 0.3585968
## 95    indiv2095 0.3585968
## 6     indiv2006 0.4306492
## 19    indiv2019 0.4306492
## 26    indiv2026 0.4306492
## 37    indiv2037 0.4306492
## 67    indiv2067 0.4306492
## 70    indiv2070 0.4306492
## 75    indiv2075 0.4306492
## 11    indiv2011 0.4554015
## 25    indiv2025 0.4554015
## 33    indiv2033 0.4554015
## 39    indiv2039 0.4554015
## 52    indiv2052 0.4554015
## 62    indiv2062 0.4554015
## 72    indiv2072 0.4554015
## 79    indiv2079 0.4554015
## 82    indiv2082 0.4554015
## 83    indiv2083 0.4554015
## 88    indiv2088 0.4554015
## 22    indiv2022 0.4803761
## 53    indiv2053 0.4803761
## 90    indiv2090 0.4803761
## 24    indiv2024 0.5556981
## 30    indiv2030 0.5556981
## 41    indiv2041 0.5556981
## 93    indiv2093 0.5556981
## 97    indiv2097 0.5556981
## 100   indiv2100 0.5556981
## 1     indiv2001 0.5803120
```

```r
plot(premodel[order(premodel$risk), ]$risk, xlab = "sorted individuals", ylab
 = "risk", main = "risks with the individuals sorted in order of increasing r
isk")
```



Q5c

The risk of indiv2001 is 0.580312.

```
premodel[premodel$individuals=='indiv2001',]

##   individuals     risk
## 1   indiv2001 0.580312
```

Q5d

The OR is 4.089562.

```
maxrisk <- max(prerisk)
minrisk <- min(prerisk)

ODmax <- maxrisk/(1 - maxrisk)
ODmin <- minrisk/(1 - minrisk)
OR <- ODmax/ODmin
OR

## [1] 4.089562
```