Basics of Association Studies

Contents

1 A simple marker association test

1.1 Scenario

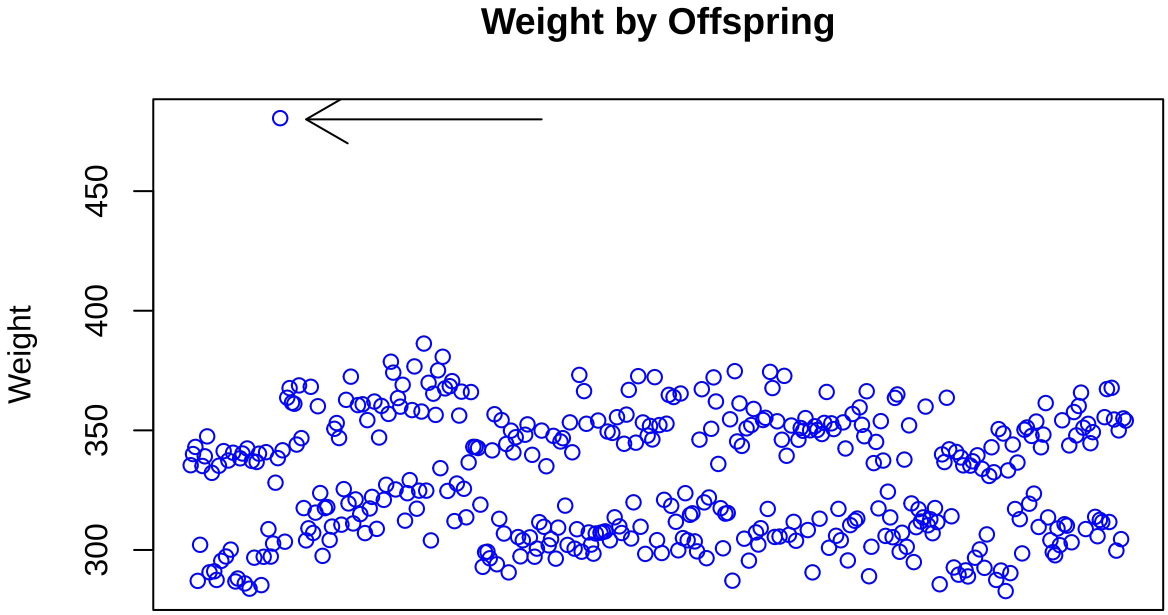
Here's our scenario: from previous mapping experiments a certain genomic region was identified as a potential QTL region, associated with weight in cattle. Even though its exact location is not well defined. Five microsatellite markers that we expect to be in full linkage with each of these genes were selected for the project. The researchers then set up a half sib experimental design with 10 heterozygote sires and each sire had 40 offspring with randomly selected females from a population with a similar genetic background. The sires and the offspring were genotyped for all five markers and phenotypic measures were recorded (the females were neither genotyped nor measured). We received this dataset and our task is to test for association between these markers and the phenotypes.

1.2 Data Cleaning

```r
# Load sires data
sires = read.table("Data/siredata.txt",
                   header = T,
                   sep = "\t",
                   skip = 3)

# Load offsprings data
prog = read.table("Data/progdata.txt",
                  header = T,
                  sep = "\t",
                  skip = 3)

# Change marker columns to factors
for (i in c(1, 3:12)) {
  sires[, i] <- as.factor(sires[, i])
}

for (i in c(2, 3, 5:14)) {
  prog[, i] <- as.factor(prog[, i])
}

# Remove the one offspring with the missing weight data
prog = prog[-which(prog$weight == "-"),]

# Drop the first record (it's a genotype error)
prog = prog[-1, ]

# Change weight variable to numeric
prog$weight = as.numeric(as.character(prog$weight))
```

```r
# Check for the outliers in the offsprings data
plot(
  prog$weight,
  main = "Weight by Offspring",
  xlab = "",
  ylab = "Weight",
  col = "blue",
  xaxt = "n"
)
```

```
arrows(150, 480, 50, 480, code = 2)
```



```
# Remove the outlier
prog = prog[-which(prog$weight > 400), ]
```

```
# Check data
head(sires, 5)
```

```
##      id weight m11 m12 m21 m22 m31 m32 m41 m42 m51 m52
## 1 sire1 334.14  M2  M1  M3  M2  M3  M4  M4  M2  M4  M2
## 2 sire2 364.81  M3  M2  M3  M2  M2  M3  M2  M4  M3  M1
## 3 sire3 383.95  M2  M4  M2  M4  M3  M2  M3  M4  M1  M4
## 4 sire4 349.88  M2  M1  M1  M2  M4  M3  M2  M1  M4  M3
## 5 sire5 357.87  M1  M3  M2  M1  M3  M1  M3  M4  M3  M2
```

```
head(prog, 5)
```

```
##    id  sire sex weight m11 m12 m21 m22 m31 m32 m41 m42 m51 m52
## 2 id2 sire1   M 335.43  M1  M4  M3  M1  M4  M5  M4  M3  M2  M2
## 3 id3 sire1   M 340.09  M2  M3  M2  M6  M3  M1  M4  M3  M4  M3
## 4 id4 sire1   M 343.08  M2  M3  M2  M1  M4  M6  M2  M3  M4  M5
## 5 id5 sire1   F 287.08  M1  M3  M2  M4  M4  M6  M4  M5  M2  M3
## 6 id6 sire1   F 302.17  M2  M2  M2  M5  M3  M5  M2  M4  M4  M1
```

Note: The first number refers to the marker and the second to the allele (i.e., m11 is marker one, allele one).

Now let's search for missing markers in the offsprings:

```
# select marker columns
index = grep("m", names(prog))

# Create an empty variable for the loop to use
missing = numeric()

# Loop
for (i in 1:length(index)) {
  missing = c(missing, which(prog[, index[i]] == "-"))
```

```r
}

missingU = unique(missing)

# Return the index of the rows with any missing marker
print(missingU)
```

```
## [1] 68
```

```r
# Return that row
print(prog[missingU, ])
```

```
##      id  sire sex weight m11 m12 m21 m22 m31 m32 m41 m42 m51 m52
## 70 id70 sire2   M 372.45   -   -   -   -   -   -   -   -   -   -
```

```r
# Remove the sample with missing markers
prog = prog[-missingU, ]

# Check marker levels
summary(prog$m11)
```

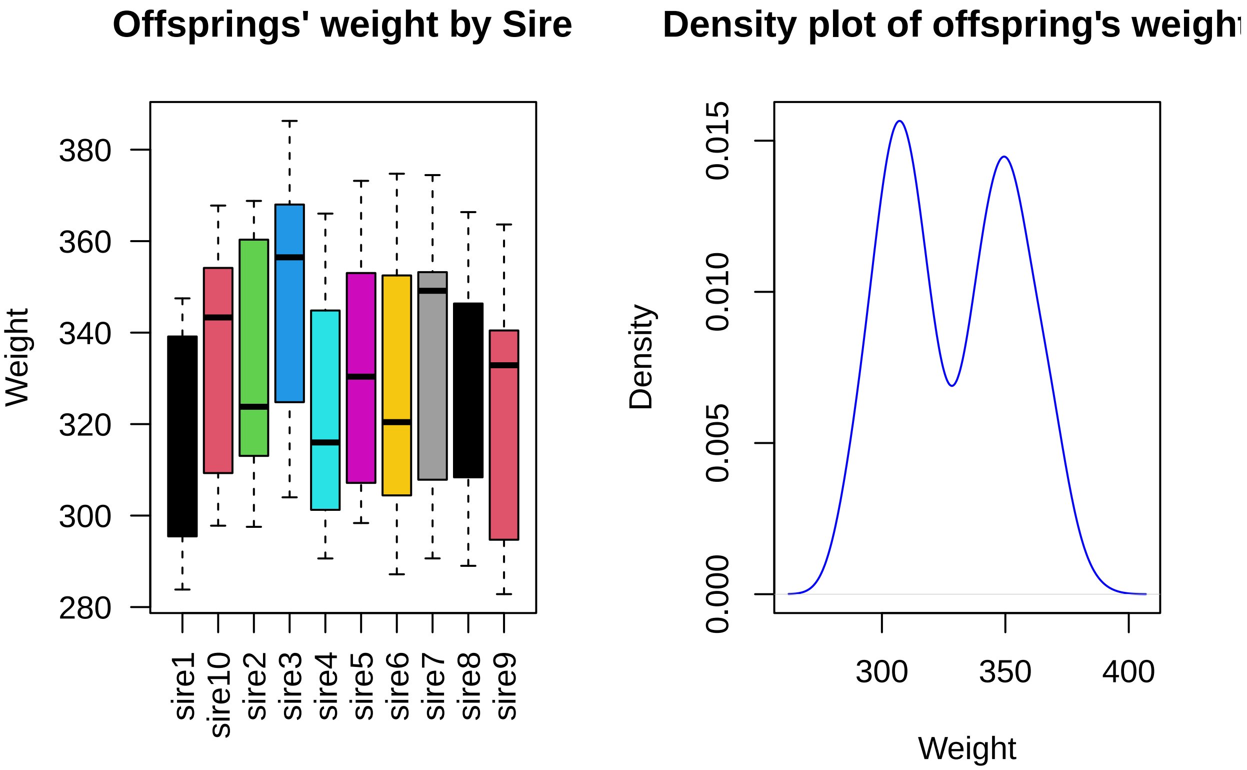
```
##   -  M1  M2  M3  M4  M5
##   0  97 139 101  59   0
```

```r
# Drop unnecessary level "-"
for (i in 5:14) {
  prog[, i] <- droplevels(prog[, i], "-")
}

# Check marker levels again
summary(prog$m11)
```

```
##  M1  M2  M3  M4
##  97 139 101  59
```

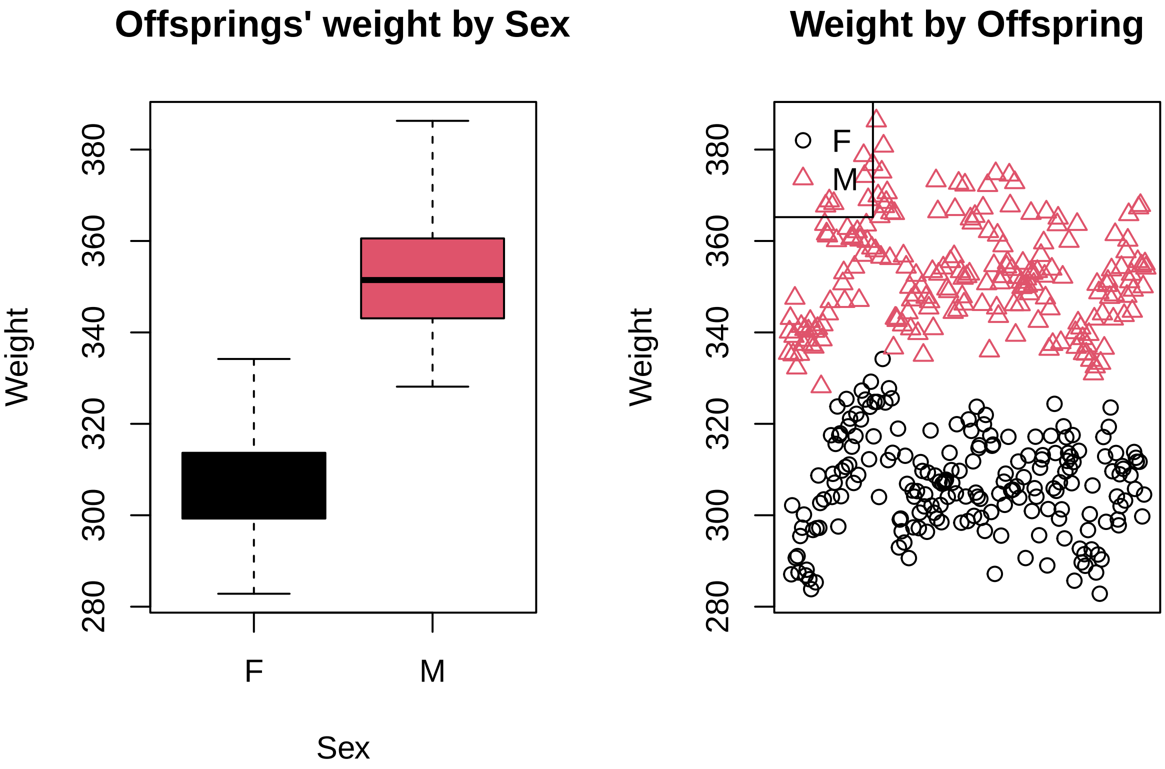
Now, let's check offsprings' weights by sex and the density plot of their weights.

```r
par(mfrow = c(1, 2))
# Plot sires weight vs. offsprings weight
boxplot(prog$weight ~ prog$sire,
        col = 1:length(levels(prog$sire)),
        main = "Offsprings' weight by Sire",
        ylab = "Weight",
        xlab = "",
        las = 2,
        cex.names=0.4,)

# Check the density plot for offsprings' weights
plot(density(prog$weight),
     col = "blue",
     xlab = "Weight",
     main = "Density plot of offspring's weights")
```



Now, plot offsprings' weight against their sex.

```r
par(mfrow = c(1, 2))
# Plot offsprings weight vs. offsprings sex
boxplot(prog$weight ~ prog$sex,
        col = 1:length(levels(prog$sex)),
        main = "Offsprings' weight by Sex",
        ylab = "Weight",
        xlab = "Sex")

# Plot the offsprings' weights vs their sex
plot(
  prog$weight,
  col = prog$sex,
  pch = as.numeric (prog$sex),
  main = "Weight by Offspring",
  xlab = "",
  xaxt = "n",
  ylab = "Weight"
)
legend("topleft",
       levels(prog$sex),
       col = 1:2,
       pch = 1:2)
```



Now, let's check the first marker:

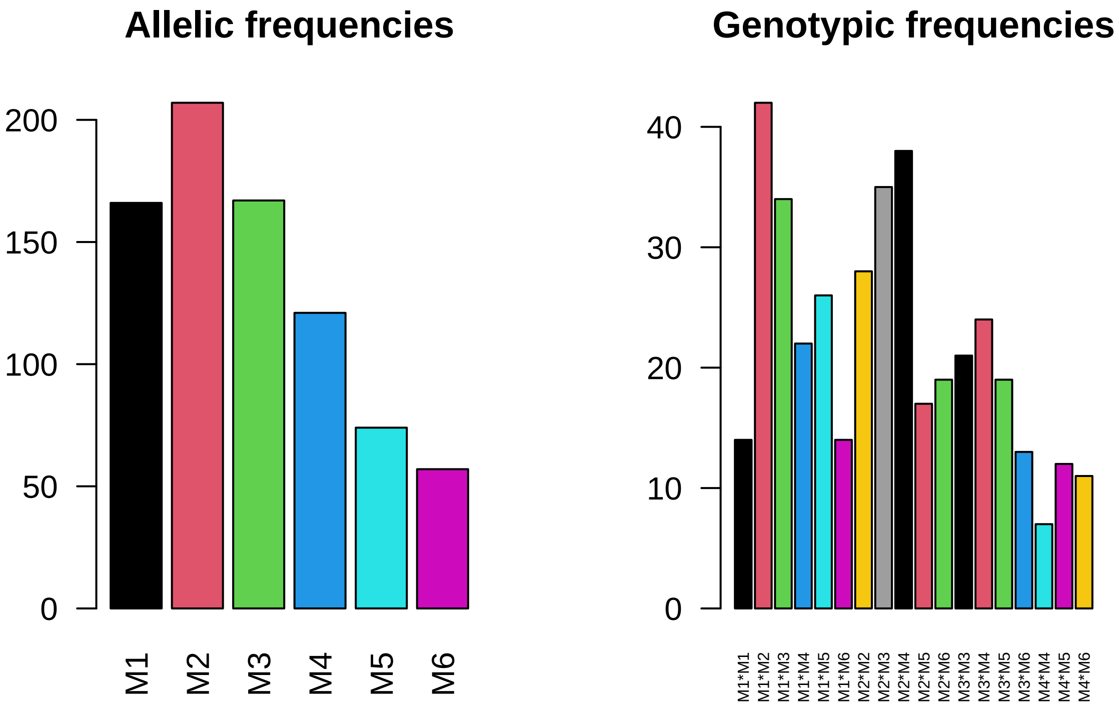
```
# Extract data for the first marker
marker_1 = data.frame(m11 = as.character(prog$m11), m12 = as.character(prog$m12))

# Sort the data
marker_1_sorted = character()
for (i in 1:length(marker_1[, 1])) {
  marker_1_sorted = rbind(marker_1_sorted, sort(as.character(marker_1[i, ])))
}

# Check allelic frequencies for the first marker
alleles = summary(factor(marker_1_sorted))
print(alleles)
```

```
##  M1  M2  M3  M4  M5  M6
## 166 207 167 121  74  57
```

```
# Check genotype frequencies for the first marker
genotypes = paste(marker_1_sorted[, 1], marker_1_sorted[, 2], sep = "*")
genotypes = summary(factor(genotypes))
print(genotypes)
```

```
## M1*M1 M1*M2 M1*M3 M1*M4 M1*M5 M1*M6 M2*M2 M2*M3 M2*M4 M2*M5 M2*M6 M3*M3 M3*M4
##    14    42    34    22    26    14    28    35    38    17    19    21    24
## M3*M5 M3*M6 M4*M4 M4*M5 M4*M6
##    19    13     7    12    11
```

```
# Plot the allelic and genotype frequencies for the first marker
par(mfrow = c(1, 2))
barplot(alleles,
        col = 1:11,
        las = 2,
        main = "Allelic frequencies")
barplot(
  genotypes,
  col = 1:11,
  las = 2,
  cex.names = 0.5,
  main = "Genotypic frequencies"
)
```



Now let's make a new data.frame with the same information but with a single column for the genotypes for each marker instead of two columns with alleles.

```
allgeno = NULL
indexms = grep("m", names(sires))
indexms = matrix(indexms, length(indexms) / 2, 2, byrow = T)
indexm = grep("m", names(prog))
indexm = matrix(indexm, length(indexm) / 2, 2, byrow = T)

for (i in 1:length(indexm[, 1])) {
  hold = data.frame(prog[indexm[i, ]])
  hold[, 1] = as.character(hold[, 1])
  hold[, 2] = as.character(hold[, 2])
  sorted = character()
  for (i in 1:length(hold[, 1])) {
    sorted = rbind(sorted, sort(as.character(hold[i, ])))
  }
  genotypes = paste(as.character(sorted[, 1]), as.character(sorted[, 2]), sep =
                      "_")
  allgeno = cbind(allgeno, genotypes)
}
colnames(allgeno) = c(
  "m1_both_allels",
  "m2_both_allels",
  "m3_both_allels",
  "m4_both_allels",
  "m5_both_allels"
)
markers = data.frame(prog[, 1:4], allgeno)
markers = markers[-1]
head(markers)
```

```
##    sire sex weight m1_both_allels m2_both_allels m3_both_allels m4_both_allels
## 2 sire1   M 335.43          M1_M4          M1_M3          M4_M5          M3_M4
## 3 sire1   M 340.09          M2_M3          M2_M6          M1_M3          M3_M4
## 4 sire1   M 343.08          M2_M3          M1_M2          M4_M6          M2_M3
## 5 sire1   F 287.08          M1_M3          M2_M4          M4_M6          M4_M5
## 6 sire1   F 302.17          M2_M2          M2_M5          M3_M5          M2_M4
## 7 sire1   M 335.11          M1_M2          M1_M2          M3_M3          M2_M4
##   m5_both_allels
## 2          M2_M2
## 3          M3_M4
## 4          M4_M5
## 5          M2_M3
## 6          M1_M4
## 7          M2_M5
```

```
# Remember the first form of data?
head(prog)
```

```
##    id  sire sex weight m11 m12 m21 m22 m31 m32 m41 m42 m51 m52
## 2 id2 sire1   M 335.43  M1  M4  M3  M1  M4  M5  M4  M3  M2  M2
## 3 id3 sire1   M 340.09  M2  M3  M2  M6  M3  M1  M4  M3  M4  M3
## 4 id4 sire1   M 343.08  M2  M3  M2  M1  M4  M6  M2  M3  M4  M5
## 5 id5 sire1   F 287.08  M1  M3  M2  M4  M4  M6  M4  M5  M2  M3
## 6 id6 sire1   F 302.17  M2  M2  M2  M5  M3  M5  M2  M4  M4  M1
## 7 id7 sire1   M 335.11  M2  M1  M2  M1  M3  M3  M2  M4  M2  M5
```

1.3 Analysis

First, let's check for the homogeneity of variances and normality of the weights.

```
# Fligner test for homogeneity of variances
fligner.test(weight ~ m1_both_allels, data = markers)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  weight by m1_both_allels
## Fligner-Killeen:med chi-squared = 7.6979, df = 17, p-value = 0.9726
```

```
# Shapiro test for normality
shapiro.test(markers$weight)
```

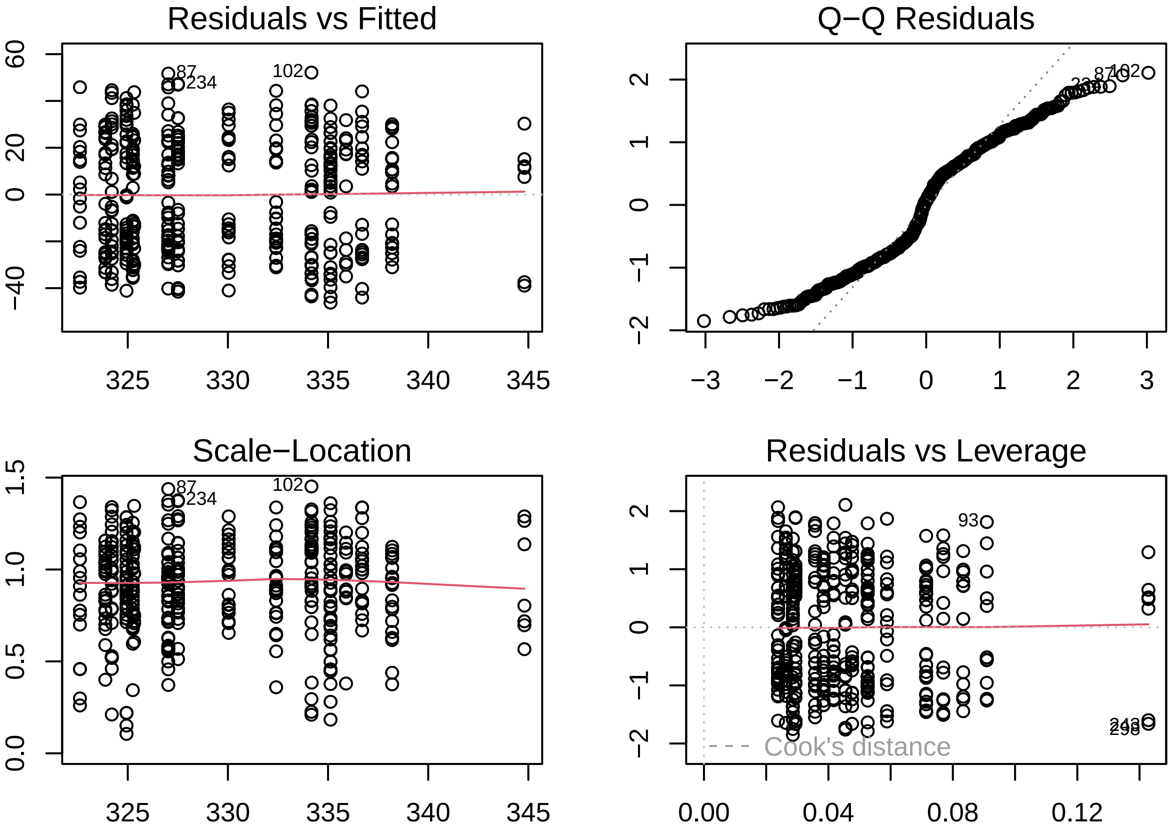
```
##
##  Shapiro-Wilk normality test
##
## data:  markers$weight
## W = 0.94831, p-value = 1.548e-10
```

We want to test the markers with a continuous trait (weight) for association. A good way to go about it is with a linear model for analysis of variance (ANOVA).

```
# Fit the model for marker 1
model3 = lm(weight ~ m1_both_allels, data = markers)
summary(model3)
```

```
##
## Call:
## lm(formula = weight ~ m1_both_allels, data = markers)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.185 -22.118   1.607  21.368  52.109
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            3.253e+02  6.763e+00  48.095   <2e-16 ***
## m1_both_allelsM1_M2  1.771e+00  7.809e+00   0.227   0.8207
## m1_both_allelsM1_M3  2.256e+00  8.035e+00   0.281   0.7790
## m1_both_allelsM1_M4  8.919e+00  8.651e+00   1.031   0.3032
## m1_both_allelsM1_M5 -1.372e+00  8.388e+00  -0.164   0.8702
## m1_both_allelsM1_M6  7.143e-04  9.564e+00   0.000   0.9999
## m1_both_allelsM2_M2 -1.049e+00  8.283e+00  -0.127   0.8993
## m1_both_allelsM2_M3  9.872e+00  8.002e+00   1.234   0.2181
## m1_both_allelsM2_M4 -3.144e-01  7.911e+00  -0.040   0.9683
## m1_both_allelsM2_M5 -2.640e+00  9.132e+00  -0.289   0.7727
## m1_both_allelsM2_M6  1.145e+01  8.913e+00   1.284   0.1998
## m1_both_allelsM3_M3  4.787e+00  8.731e+00   0.548   0.5838
## m1_both_allelsM3_M4  7.156e+00  8.510e+00   0.841   0.4009
## m1_both_allelsM3_M5  1.295e+01  8.913e+00   1.453   0.1472
## m1_both_allelsM3_M6  8.932e+00  9.746e+00   0.916   0.3600
## m1_both_allelsM4_M4  1.955e+01  1.171e+01   1.669   0.0959 .
## m1_both_allelsM4_M5  1.065e+01  9.954e+00   1.069   0.2856
## m1_both_allelsM4_M6  6.623e-02  1.020e+01   0.006   0.9948
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.3 on 378 degrees of freedom
## Multiple R-squared:  0.0432, Adjusted R-squared:  0.0001707
## F-statistic: 1.004 on 17 and 378 DF,  p-value: 0.4529
```

```
# Check quality control plots for the model
par(mfrow = c(2, 2), pin = c(2.5, 1.5))
plot(model3)
```



```r
# ANOVA test
anova(model3)
```

```
## Analysis of Variance Table
##
## Response: weight
##                 Df Sum Sq Mean Sq F value Pr(>F)
## m1_both_alleles  17  10928  642.83   1.004 0.4529
## Residuals       378 242028  640.29
```

The marker is not significant (does not explain much of the variability in weight). We also know that sex seems to have a pretty big effect on our data, so let's include sex in the model.

```r
model2 = lm(weight ~ sex + m1_both_alleles, data = markers)
summary(model2)
```

```
##
## Call:
## lm(formula = weight ~ sex + m1_both_alleles, data = markers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.1174  -7.2400  -0.7689   7.1190  31.3916
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           2.992e+02  2.900e+00 103.168  < 2e-16 ***
## sexM                  4.558e+01  1.081e+00  42.143  < 2e-16 ***
## m1_both_allelsM1_M2 6.111e+00  3.274e+00   1.867 0.062693 .
## m1_both_allelsM1_M3 5.512e+00  3.368e+00   1.637 0.102537
## m1_both_allelsM1_M4 1.010e+01  3.625e+00   2.787 0.005590 **
## m1_both_allelsM1_M5 1.884e+00  3.516e+00   0.536 0.592370
## m1_both_allelsM1_M6 7.143e-04  4.007e+00   0.000 0.999858
## m1_both_allelsM2_M2 7.090e+00  3.476e+00   2.040 0.042079 *
## m1_both_allelsM2_M3 5.965e+00  3.354e+00   1.779 0.076118 .
## m1_both_allelsM2_M4 7.739e+00  3.320e+00   2.331 0.020292 *
## m1_both_allelsM2_M5 4.637e+00  3.830e+00   1.211 0.226794
## m1_both_allelsM2_M6 1.350e+01  3.735e+00   3.615 0.000341 ***
## m1_both_allelsM3_M3 9.128e+00  3.660e+00   2.494 0.013051 *
## m1_both_allelsM3_M4 1.421e+01  3.569e+00   3.981 8.24e-05 ***
## m1_both_allelsM3_M5 1.260e+01  3.734e+00   3.375 0.000815 ***
## m1_both_allelsM3_M6 1.043e+01  4.084e+00   2.555 0.011010 *
## m1_both_allelsM4_M4 1.304e+01  4.910e+00   2.655 0.008258 **
## m1_both_allelsM4_M5 1.010e+01  4.171e+00   2.422 0.015898 *
## m1_both_allelsM4_M6 9.537e+00  4.278e+00   2.229 0.026375 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.6 on 377 degrees of freedom
## Multiple R-squared:  0.8325, Adjusted R-squared:  0.8245
## F-statistic: 104.1 on 18 and 377 DF,  p-value: < 2.2e-16
```

```
anova(model2)
```

```
## Analysis of Variance Table
##
## Response: weight
##                 Df Sum Sq Mean Sq   F value    Pr(>F)
## sex              1 204821  204821 1822.0785 < 2.2e-16 ***
## m1_both_allels  17   5756     339    3.0122 6.124e-05 ***
## Residuals      377  42379     112
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Now let's consider the effect of sire in our model.

```
model1 = lm(weight ~ sex + sire + m1_both_allels, data = markers)
summary(model1)
```

```
##
## Call:
## lm(formula = weight ~ sex + sire + m1_both_allels, data = markers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.5650  -5.7110  -0.0889   5.2261  21.5778
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         293.8628     2.6104 112.574  < 2e-16 ***
## sexM                 45.5573     0.8566  53.186  < 2e-16 ***
## siresire10           14.5459     1.9541   7.444 7.00e-13 ***
## siresire2            18.2750     2.0120   9.083  < 2e-16 ***
## siresire3            27.9691     2.0110  13.908  < 2e-16 ***
## siresire4             8.5190     1.9165   4.445 1.16e-05 ***
## siresire5            14.2383     1.9915   7.150 4.72e-12 ***
## siresire6            15.8241     2.0043   7.895 3.37e-14 ***
## siresire7            14.2136     2.0673   6.875 2.66e-11 ***
## siresire8            14.1905     2.1265   6.673 9.22e-11 ***
## siresire9             3.6338     1.9234   1.889   0.0596 .
## m1_both_allelsM1_M2   1.0713     2.6239   0.408   0.6833
## m1_both_allelsM1_M3  -0.7715     2.7549  -0.280   0.7796
## m1_both_allelsM1_M4   0.9751     3.0089   0.324   0.7461
## m1_both_allelsM1_M5  -1.6311     2.8036  -0.582   0.5611
## m1_both_allelsM1_M6  -1.6844     3.1542  -0.534   0.5937
## m1_both_allelsM2_M2  -0.4491     2.8385  -0.158   0.8744
## m1_both_allelsM2_M3  -2.0030     2.7399  -0.731   0.4652
## m1_both_allelsM2_M4  -1.4636     2.7087  -0.540   0.5893
## m1_both_allelsM2_M5  -1.9519     3.0634  -0.637   0.5244
## m1_both_allelsM2_M6   2.4693     3.0634   0.806   0.4207
## m1_both_allelsM3_M3  -0.8174     3.0775  -0.266   0.7907
## m1_both_allelsM3_M4   2.1514     2.9849   0.721   0.4715
## m1_both_allelsM3_M5   2.2666     3.1662   0.716   0.4745
## m1_both_allelsM3_M6   0.4037     3.3847   0.119   0.9051
## m1_both_allelsM4_M4  -3.6682     4.0730  -0.901   0.3684
## m1_both_allelsM4_M5   1.2526     3.6037   0.348   0.7283
## m1_both_allelsM4_M6  -3.0625     3.6157  -0.847   0.3975
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.303 on 368 degrees of freedom
## Multiple R-squared:  0.8997, Adjusted R-squared:  0.8923
## F-statistic: 122.3 on 27 and 368 DF,  p-value: < 2.2e-16
```

```
anova(model1)
```

```
## Analysis of Variance Table
##
## Response: weight
##                  Df Sum Sq Mean Sq   F value Pr(>F)
## sex               1 204821  204821 2971.0104 <2e-16 ***
## sire              9  21807    2423   35.1467 <2e-16 ***
## m1_both_allels   17    958      56    0.8175 0.6729
## Residuals       368  25370      69
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

To formally test the difference in models (essentially proportion of variation explained by a term) do an anova on them.

```
model1 = lm(weight ~ sex + sire + m1_both_allels, data = markers)
model2 = lm(weight ~ sex + m1_both_allels, data = markers)
model3 = lm(weight ~ m1_both_allels, data = markers)
null_model = lm(weight ~ 1, data = markers)
```

- **ANOVA of model 1 (sex + sire + M1) vs model 2 (sex + M1)**

```
anova(model1, model2)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ sex + sire + m1_both_allels
## Model 2: weight ~ sex + m1_both_allels
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    368 25370
## 2    377 42379 -9    -17009 27.413 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
b = "================================================"
```

- **ANOVA of model 2 (sex + M1) vs model 3 (M1)**

```
anova(model2, model3)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ sex + m1_both_allels
## Model 2: weight ~ m1_both_allels
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    377  42379
## 2    378 242028 -1   -199649 1776.1 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
b = "================================================"
```

- **ANOVA of model 3 (M1) vs null model**

```
anova(model3, null_model)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ m1_both_allels
## Model 2: weight ~ 1
##   Res.Df    RSS  Df Sum of Sq     F Pr(>F)
## 1    378 242028
## 2    395 252956 -17    -10928 1.004 0.4529
```

```
b = "================================================"
```

This confirms it: our marker is not providing any information at all.

- **ANOVA of model 2 (sex + M1) vs null model**

In contrast look at sex:

```
anova(model2, null_model)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ sex + m1_both_allels
## Model 2: weight ~ 1
##   Res.Df    RSS  Df Sum of Sq      F    Pr(>F)
## 1    377  42379
## 2    395 252956 -18   -210577 104.07 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
b = "================================================"
```

2 Single SNP analysis

2.1 Scenario

A dataset containing epidemiological information and 51 SNPs from a case-control study on asthma. We are interested in finding those SNPs associated with the asthma status.

2.2 Load data

```r
# Import packages
library(SNPassoc)

# Load data
asthma <- read.table("Data/asthma.txt", header = TRUE)

# Check data's first 20 headers (the rest are SNPs as well)
colnames(asthma)[1:20]
```

```
##  [1] "country"     "gender"      "age"         "bmi"         "smoke"
##  [6] "casecontrol" "rs4490198"   "rs4849332"   "rs1367179"   "rs11123242"
## [11] "rs13014858"  "rs1430094"   "rs1430093"   "rs746710"    "rs1430090"
## [16] "rs6737251"   "rs11685217"  "rs1430097"   "rs10496465"  "rs3756688"
```

```r
# Check some samples
t(asthma[1:3, 1:7])
```

```
##             1          2          3
## country     "Germany"  "Germany"  "Germany"
## gender      "Males"    "Males"    "Males"
## age         "42.80630" "50.22861" "46.68857"
## bmi         "20.14797" "24.69136" "27.73230"
## smoke       "2"        "0"        "1"
## casecontrol "0"        "0"        "0"
## rs4490198   "GG"       "GG"       "GG"
```

```r
# Check some SNPs
asthma[1:5, 7:9]
```

```
##   rs4490198 rs4849332 rs1367179
## 1        GG        TT        GC
## 2        GG        GT        GC
## 3        GG        TT        GC
## 4        AG        GT        GG
## 5        AG        GG        GG
```

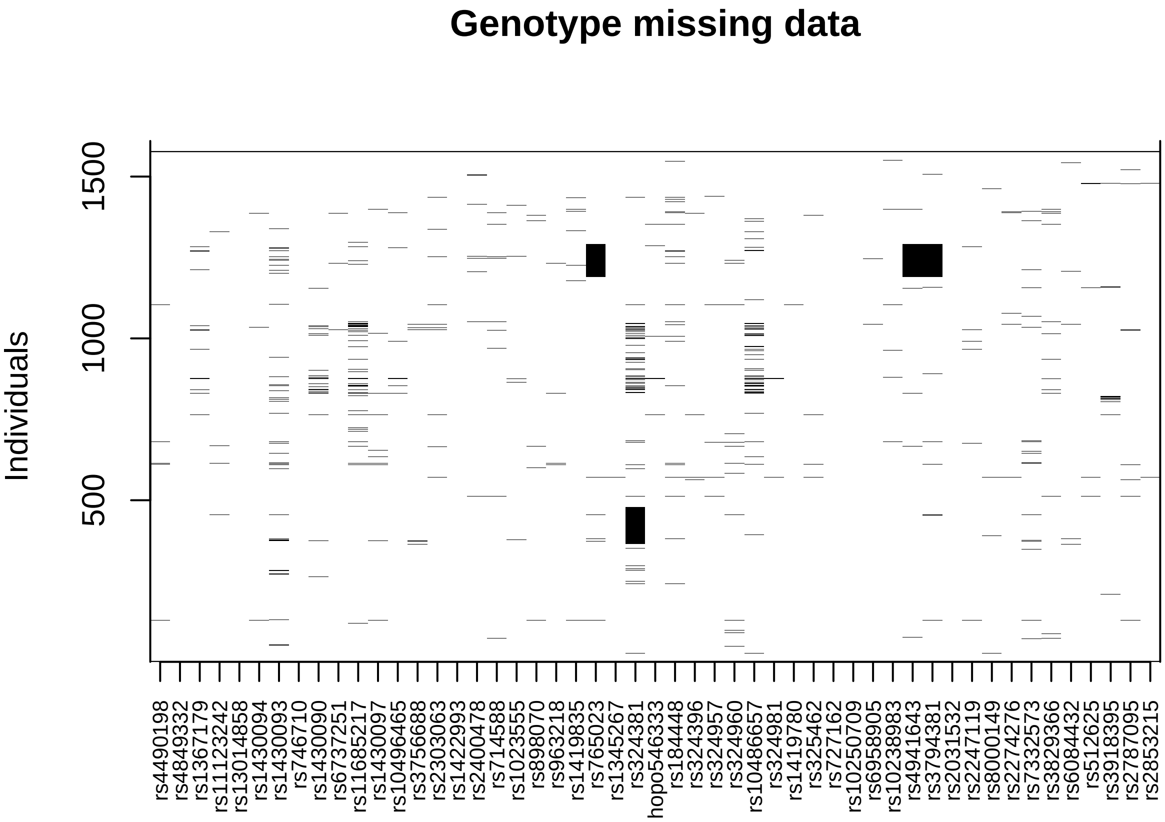
```r
# Indicate which columns of the dataset contain the SNP data
asthma.s <- setupSNP(data = asthma,
                     colSNPs = 7:ncol(asthma),
                     sep = "")

# Check SNP data now
asthma.s[1:5, 7:9]
```

```
##   rs4490198 rs4849332 rs1367179
## 1       G/G       T/T       G/C
## 2       G/G       G/T       G/C
## 3       G/G       T/T       G/C
## 4       A/G       G/T       G/G
## 5       A/G       G/G       G/G
```

```
# Check the genotype and allele frequencies for a SNP
summary(asthma.s$rs1422993)
```

```
## Genotypes:
##     frequency percentage
## G/G       903  57.224335
## G/T       570  36.121673
## T/T       105   6.653992
##
## Alleles:
##   frequency percentage
## G      2376   75.28517
## T       780   24.71483
##
## HWE (p value): 0.250093
```

```
# Plot missing genotypes across all SNPs
plotMissing(asthma.s, print.labels.SNPs = TRUE)
```



2.3 H-W equilibrium

Genotype calling error can be detected by a H-W equilibrium test. Note that H-W must be checked only in controls. One is interested in keeping those SNPs that do not reject the null hypothesis (p > 0.05).

```
# H-W test
hwe <- tableHWE(asthma.s, casecontrol)
```

```
# Check some results
hwe[1:4,]
```

```
##            all groups          0         1
## rs4490198   0.1741325 0.09795485 0.9115690
## rs4849332   0.5220596 0.63029064 0.6458515
## rs1367179   0.7381531 1.00000000 0.4919761
## rs11123242  0.9328981 0.92328571 0.5990026
```

```
# Keeping SNPs that pass the H-W test
snps.ok <- rownames(hwe)[hwe[, 2] >= 0.05]
pos <- which(colnames(asthma) %in% snps.ok, useNames = FALSE)
asthma.s <- setupSNP(data = asthma,
                     colSNPs = pos,
                     sep = "")
```

2.4 SNP association analysis

Now, let's check the association between disease status (**casecontrol**) and a single SNP.

Note: AIC (Akaike information criteria): it can be used to decide which is the best model of inheritance (the lower the better the model is).

```
# Test for group ~ SNP
association(formula = casecontrol ~ rs1422993, data = asthma.s)
```

```
##
## SNP: rs1422993  adjusted by:
##                 0    %   1    %   OR lower upper  p-value  AIC
## Codominant
## G/G           730 59.0 173 50.9 1.00             0.017768 1642
## G/T           425 34.3 145 42.6 1.44  1.12  1.85
## T/T            83  6.7  22  6.5 1.12  0.68  1.84
## Dominant
## G/G           730 59.0 173 50.9 1.00             0.007826 1642
## G/T-T/T       508 41.0 167 49.1 1.39  1.09  1.77
## Recessive
## G/G-G/T      1155 93.3 318 93.5 1.00             0.877863 1649
## T/T            83  6.7  22  6.5 0.96  0.59  1.57
## Overdominant
## G/G-T/T       813 65.7 195 57.4 1.00             0.005026 1641
## G/T           425 34.3 145 42.6 1.42  1.11  1.82
## log-Additive
## 0,1,2        1238 78.5 340 21.5 1.22  1.01  1.47 0.040151 1644
```

```
# Incorporate covariates in the model
association(formula = casecontrol ~ rs1422993 + country + smoke, data = asthma.s)
```

```
##
## SNP: rs1422993  adjusted by: country smoke
##                 0    %   1    %   OR lower upper p-value  AIC
## Codominant
## G/G           728 59.1 173 51.0 1.00             0.06957 1407
## G/T           423 34.3 144 42.5 1.38  1.05  1.82
## T/T            81  6.6  22  6.5 1.07  0.62  1.85
## Dominant
## G/G           728 59.1 173 51.0 1.00             0.03380 1406
## G/T-T/T       504 40.9 166 49.0 1.33  1.02  1.73
## Recessive
## G/G-G/T      1151 93.4 317 93.5 1.00             0.80821 1411
## T/T            81  6.6  22  6.5 0.94  0.55  1.60
## Overdominant
## G/G-T/T       809 65.7 195 57.5 1.00             0.02163 1406
## G/T           423 34.3 144 42.5 1.37  1.05  1.79
## log-Additive
## 0,1,2        1232 78.4 339 21.6 1.19  0.96  1.46 0.10926 1408
```

```r
# Stratify for gender
association(formula = casecontrol ~ rs1422993 + survival::strata(gender),
            data = asthma.s)
```

```
##
## SNP: rs1422993  adjusted by: survival::strata(gender)
##                  0    %   1    %   OR lower upper  p-value  AIC
## Codominant
## G/G           730 59.0 173 50.9 1.00             0.022940 1634
## G/T           425 34.3 145 42.6 1.42  1.11  1.83
## T/T            83  6.7  22  6.5 1.09  0.66  1.80
## Dominant
## G/G           730 59.0 173 50.9 1.00             0.011144 1633
## G/T-T/T       508 41.0 167 49.1 1.37  1.07  1.74
## Recessive
## G/G-G/T      1155 93.3 318 93.5 1.00             0.805330 1640
## T/T            83  6.7  22  6.5 0.94  0.58  1.53
## Overdominant
## G/G-T/T       813 65.7 195 57.4 1.00             0.006378 1632
## G/T           425 34.3 145 42.6 1.41  1.10  1.80
## log-Additive
## 0,1,2        1238 78.5 340 21.5 1.21  1.00  1.46 0.055231 1636
```

```r
# Train the model only on a subset of individuals
association(
  formula = casecontrol ~ rs1422993,
  data = asthma.s,
  subset = country == "Spain"
)
```

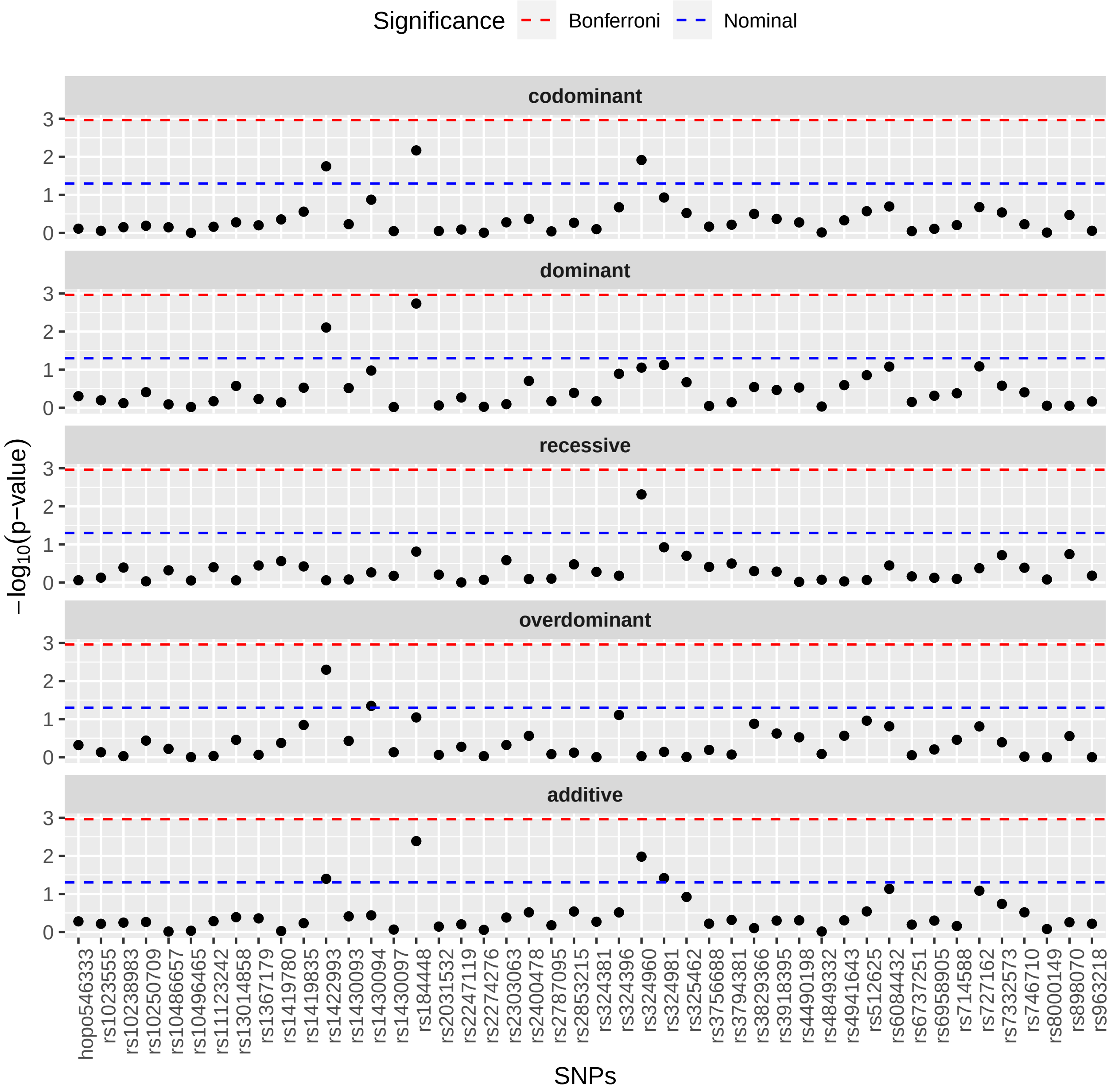
```
##
## SNP: rs1422993  adjusted by:
##                 0    %  1    %   OR lower upper p-value   AIC
## Codominant
## G/G          179 54.6 22 44.9 1.00              0.3550 295.2
## G/T          125 38.1 24 49.0 1.56  0.84  2.91
## T/T           24  7.3  3  6.1 1.02  0.28  3.66
## Dominant
## G/G          179 54.6 22 44.9 1.00              0.2059 293.7
## G/T-T/T      149 45.4 27 55.1 1.47  0.81  2.70
## Recessive
## G/G-G/T      304 92.7 46 93.9 1.00              0.7576 295.2
## T/T           24  7.3  3  6.1 0.83  0.24  2.85
## Overdominant
## G/G-T/T      203 61.9 25 51.0 1.00              0.1502 293.2
## G/T          125 38.1 24 49.0 1.56  0.85  2.85
## log-Additive
## 0,1,2        328 87.0 49 13.0 1.23  0.77  1.96  0.3816 294.5
```

Check the association between disease status (**casecontrol**) and all SNPs.

```r
# Massive univariate testing (MUT)
ans <- WGassociation(formula = casecontrol, data = asthma.s)
ans[1:3, ]
```

```
##           comments codominant dominant recessive overdominant log-additive
## rs4490198        -    0.52765  0.29503   0.96400      0.29998      0.49506
## rs4849332        -    0.96912  0.92986   0.84806      0.82327      0.97049
## rs1367179        -    0.62775  0.59205   0.35786      0.86419      0.43994
```

```r
# Plot p-values from the MUT
plot(ans)
```



We can also fit max-statistics model.

```
# Calculate p-value for a certain SNP under the max-statistics model
maxstat(asthma.s$casecontrol, asthma.s$rs1422993)
```

```
##      dominant recessive log-additive MAX-statistic Pr(>z)
## [1,]    7.073     0.024        4.291         7.073 0.0144 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

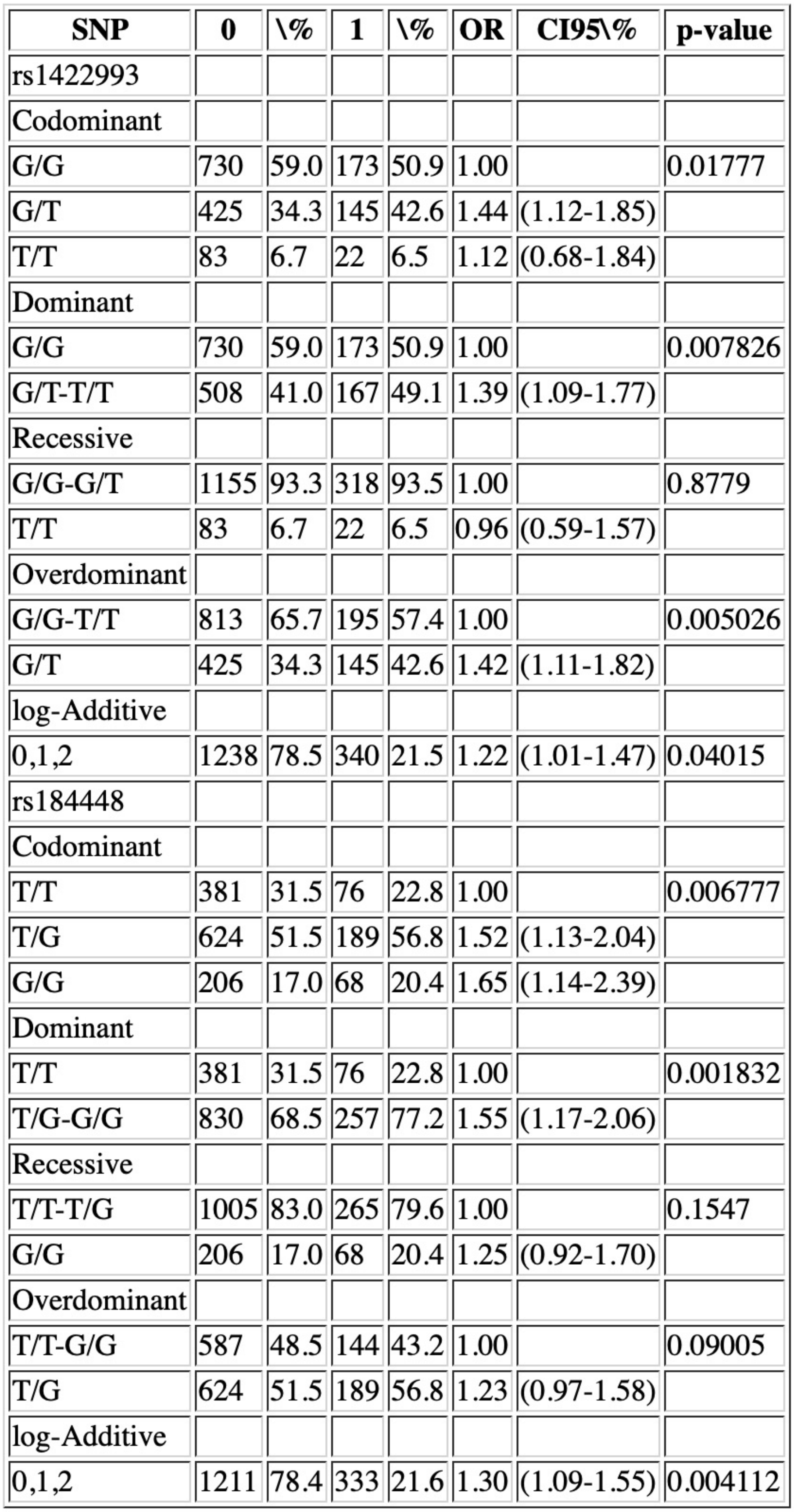
```
# Max-statistics for all SNPs
ans.max <- maxstat(asthma.s, casecontrol)

# Check output
ans.max[1:5, 1:2]
```

```
##                  rs4490198   rs4849332
## dominant      1.096514182 0.007748582
## recessive     0.002036721 0.036707719
## log-additive  0.466455726 0.001368585
## MAX-statistic 1.096514182 0.036707719
## Pr(>z)        0.501708984 0.976293499
```

Finally, let's create a table output for the results.

```
# Create a table output for two specific SNPs
library(xtable)
invisible(capture.output(out <-
                           getNiceTable(ans[c("rs1422993", "rs184448")])))
nlines <- attr(out, "nlines")
hlines <- c(-1,-1, 0, cumsum(nlines + 1), nrow(out), nrow(out))
print(
  xtable(out, caption = "", label = 'tab-2SNPs'),
  tabular.enviroment = "longtable",
  type = "html",
  file = "Output/tableSNPs.html",
  floating = FALSE,
  include.rownames = FALSE,
  hline.after = hlines,
  sanitize.text.function = identity
)
```

The output table looks like this:



SNP	0	\%	1	\%	OR	CI95\%	p-value
rs1422993							
Codominant							
G/G	730	59.0	173	50.9	1.00		0.01777
G/T	425	34.3	145	42.6	1.44	(1.12-1.85)	
T/T	83	6.7	22	6.5	1.12	(0.68-1.84)	
Dominant							
G/G	730	59.0	173	50.9	1.00		0.007826
G/T-T/T	508	41.0	167	49.1	1.39	(1.09-1.77)	
Recessive							
G/G-G/T	1155	93.3	318	93.5	1.00		0.8779
T/T	83	6.7	22	6.5	0.96	(0.59-1.57)	
Overdominant							
G/G-T/T	813	65.7	195	57.4	1.00		0.005026
G/T	425	34.3	145	42.6	1.42	(1.11-1.82)	
log-Additive							
0,1,2	1238	78.5	340	21.5	1.22	(1.01-1.47)	0.04015
rs184448							
Codominant							
T/T	381	31.5	76	22.8	1.00		0.006777
T/G	624	51.5	189	56.8	1.52	(1.13-2.04)	
G/G	206	17.0	68	20.4	1.65	(1.14-2.39)	
Dominant							
T/T	381	31.5	76	22.8	1.00		0.001832
T/G-G/G	830	68.5	257	77.2	1.55	(1.17-2.06)	
Recessive							
T/T-T/G	1005	83.0	265	79.6	1.00		0.1547
G/G	206	17.0	68	20.4	1.25	(0.92-1.70)	
Overdominant							
T/T-G/G	587	48.5	144	43.2	1.00		0.09005
T/G	624	51.5	189	56.8	1.23	(0.97-1.58)	
log-Additive							
0,1,2	1211	78.4	333	21.6	1.30	(1.09-1.55)	0.004112

2.5 Gene x environment and gene x gene interactions

```
# G x E by smoking as the factor
association(formula = casecontrol ~ dominant(rs1422993) * factor(smoke),
            data = asthma.s)
```

```
##
##       SNP: dominant(rs1422993  adjusted by:
##  Interaction
## ---------------------
##           0      OR lower upper   1      OR lower upper   0  1    2 lower upper
## G/G     273 86 1.00    NA    NA 213 44 0.66  0.44  0.98 182 34 0.59  0.38  0.92
## G/T-T/T 210 77 1.16  0.82  1.66 144 51 1.12  0.75  1.68 115 24 0.66  0.40  1.09
##          0 1    3 lower upper  0 1    4 lower upper
## G/G     40 2 0.16  0.04  0.67 20 7 1.11  0.45  2.72
## G/T-T/T 19 7 1.17  0.48  2.88 16 7 1.39  0.55  3.49
##
## p interaction: 0.13712
##
##  factor(smoke) within dominant(rs1422993
## ---------------------
## G/G
##     0  1   OR lower upper
## 0 273 86 1.00    NA    NA
## 1 213 44 0.66  0.44  0.98
## 2 182 34 0.59  0.38  0.92
## 3  40  2 0.16  0.04  0.67
## 4  20  7 1.11  0.45  2.72
##
## G/T-T/T
##     0  1   OR lower upper
## 0 210 77 1.00    NA    NA
## 1 144 51 0.97  0.64  1.46
## 2 115 24 0.57  0.34  0.95
## 3  19  7 1.00  0.41  2.48
## 4  16  7 1.19  0.47  3.01
##
## p trend: 0.13712
##
##  dominant(rs1422993 within factor(smoke)
## ---------------------
## 0
##           0  1   OR lower upper
## G/G     273 86 1.00    NA    NA
## G/T-T/T 210 77 1.16  0.82  1.66
##
## 1
##           0  1   OR lower upper
## G/G     213 44 1.00    NA    NA
## G/T-T/T 144 51 1.71  1.09   2.7
##
## 2
##           0  1   OR lower upper
## G/G     182 34 1.00    NA    NA
## G/T-T/T 115 24 1.12  0.63  1.98
##
## 3
##          0 1   OR lower upper
## G/G     40 2 1.00    NA    NA
## G/T-T/T 19 7 7.37   1.4 38.89
##
## 4
##          0 1   OR lower upper
## G/G     20 7 1.00    NA    NA
## G/T-T/T 16 7 1.25  0.36  4.31
##
## p trend: 0.278
```

```
# G x G by rs184448 as the factor
association(
  formula = casecontrol ~ rs1422993 * factor(rs184448),
  data = asthma.s,
  model.interaction = "dominant"
)
```

```
##
##       SNP: rs1422993  adjusted by:
##  Interaction
## ---------------------
##         T/T      OR lower upper T/G      OR lower upper   0  1  G/G lower upper
## G/G     227 43 1.00    NA    NA 359 96 1.41  0.95  2.10 128 30 1.24  0.74  2.07
## G/T-T/T 154 33 1.13  0.69  1.86 265 93 1.85  1.24  2.77  78 38 2.57  1.55  4.27
##
## p interaction: 0.24499
##
##  factor(rs184448) within rs1422993
## ---------------------
## G/G
##       0  1   OR lower upper
## T/T 227 43 1.00    NA    NA
## T/G 359 96 1.41  0.95  2.10
## G/G 128 30 1.24  0.74  2.07
##
## G/T-T/T
##       0  1   OR lower upper
## T/T 154 33 1.00    NA    NA
## T/G 265 93 1.64  1.05  2.55
## G/G  78 38 2.27  1.32  3.90
##
## p trend: 0.24499
##
##  rs1422993 within factor(rs184448)
## ---------------------
## T/T
##           0  1   OR lower upper
## G/G     227 43 1.00    NA    NA
## G/T-T/T 154 33 1.13  0.69  1.86
##
## T/G
##           0  1   OR lower upper
## G/G     359 96 1.00    NA    NA
## G/T-T/T 265 93 1.31  0.95  1.82
##
## G/G
##           0  1   OR lower upper
## G/G     128 30 1.00    NA    NA
## G/T-T/T  78 38 2.08  1.19  3.62
##
## p trend: 0.12743
```