Practical 3

Statistical Genetics: Linkage disequilibrium and Haplotype estimation

Anna Putina Marine Mauzeau

2023-11-26

```
library(genetics)
library(HardyWeinberg)
library(haplo.stats)
library(dplyr)
library(graphics)
```

Linkage Disequilibrium

1. Load the `FOXP2.dat` file into the R environment. How many individuals and how many SNPs are there in the database? What percentage of the data is missing?

```
gen.data <- read.table('FOXP2.dat')

NA.counter <- sum(is.na.data.frame(gen.data))
NA.percentage <- NA.counter*100/prod(dim(gen.data)-1)
```

There are 104 individuals and 543 SNPs in the database.

The percentage of missing data in this database: 0 %.

2. Using the function LD from the genetics package, compute the LD statistic D for the SNPs *rs34684677* and *rs2894715* of the database. Is there significant association between the alleles of these two SNPs?

```
# Data processing
colnames(gen.data) <- as.vector(gen.data[1, ])
gen.data <- gen.data[-1,-1]

rs34684677 <- gen.data$rs34684677
rs2894715 <- gen.data$rs2894715

geno.rs34684677 <- genotype(rs34684677)
geno.rs2894715 <- genotype(rs2894715)

ld.results <- LD(geno.rs34684677, geno.rs2894715)
D <- ld.results$D
```

The coefficient of linkage disequilibrium D = -0.054937 could seem to be low (i.e. there is independance between the 2 genotypes), but it is actually high because |D'| is almost equal to 1 : there is linkage disequilibrium. The negative correlation coefficient suggests that the alleles at these loci are in repulsion. Thus, there seems to be a significant association between the alleles of these two SNPs.

3. Given your previous estimate of D for SNPs *rs34684677* and *rs2894715*, infer the haplotype frequencies. Which haplotype is the most common?

```
num <- 2*nrow(gen.data)
prop.G.rs2894715 <- sum(allele.count(geno.rs2894715, "G"))/num
prop.T.rs2894715 <- sum(allele.count(geno.rs2894715, "T"))/num

prop.G.rs34684677 <- sum(allele.count(geno.rs34684677, "G"))/num
prop.T.rs34684677 <- sum(allele.count(geno.rs34684677, "T"))/num

# Calculations
GG <- prop.G.rs34684677 * prop.G.rs2894715 - D
GT <- prop.G.rs34684677 * prop.T.rs2894715 + D
TG <- prop.T.rs34684677 * prop.G.rs2894715 + D
TT <- prop.T.rs34684677 * prop.T.rs2894715 - D

# Create a data frame
haplotype_table <- data.frame(
  Haplotype = c("GG", "GT", "TG", "TT"),
  Probability = c(GG, GT, TG, TT)
)

# Print the table
print(haplotype_table)
```

```
##   Haplotype  Probability
## 1        GG 3.364644e-01
## 2        GT 5.000741e-01
## 3        TG 7.406505e-05
## 4        TT 1.633875e-01
```

We can compare our results to the ones we get with the use of haplo.em function.

```
snp1 <- gsub("/", "", as.vector(rs34684677))
snp2 <- gsub("/", "", as.vector(rs2894715))

Geno <- cbind(substr(snp1,1,1),substr(snp1,2,2),
substr(snp2,1,1),substr(snp2,2,2))

snpnames <- c("rs34684677", "rs2894715")
HaploEM <- haplo.em(Geno, locus.label=snpnames)
print(HaploEM)
```

```
## ============================================================================
##                                  Haplotypes
## ============================================================================
##   rs34684677 rs2894715 hap.freq
## 1          G         G  0.33654
## 2          G         T  0.50000
## 3          T         G  0.00000
## 4          T         T  0.16346
## ============================================================================
##                                   Details
## ============================================================================
## lnlike =  -164.8458
```

```
## lr stat for no LD =  18.69923 , df =  0 , p-val =  NA
```

We get almost exactly the same results with both calculations.

The most common haplotype is G in *rs34684677* and T in *rs2894715* (GT).

4. Determine the genotype counts for each SNP. For how many variants do you reject Hardy-Weinberg equilibrium using an ordinary chi-square test without continuity correction? Is this what you would expect by chance? (hint: you can read the .bim in R in order to determine the alleles of each SNP, and use function MakeCounts from the HardyWeinberg package to create a matrix of genotype counts).

```
FOXP2.bim <- read.table("FOXP2.bim")

alleles <- paste(FOXP2.bim$V5, FOXP2.bim$V6, sep = "/")
genotype.counts <- MakeCounts(gen.data, alleles, sep="/")

chi_sqared <- HWChisqStats(genotype.counts)
pval <- HWChisqStats(genotype.counts, pvalues = T)

nb.rejected <- sum(pval < 0.05)
```

We reject Hardy-Weinberg equilibrium for 33 variants.

To determine whether the number of rejections is what we would expect by chance, we consider the significance level we used for the chi-square tests.

We expect approximately 5% of tests to result in a rejection by chance alone.

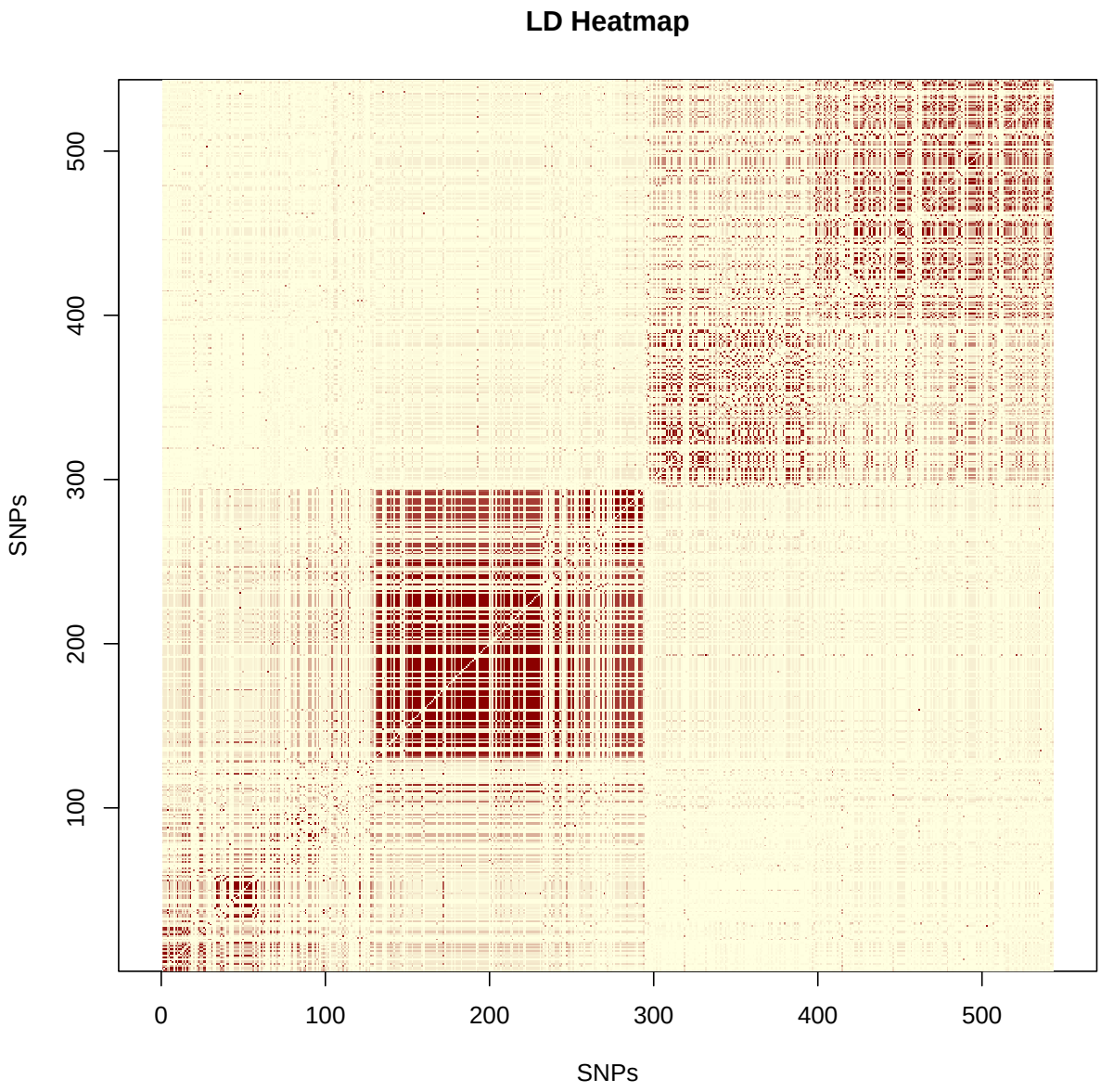
In our case:

Expected number of rejections $= 543 \times 0.05 = 27.15$.

This means, on average, we might expect around 27 rejections by chance alone. Since we observed 33 rejections, it is slightly higher than the expected value by chance.

5. Compute the LD for all the marker pairs in this data base, using the LD function of the packages genetics. Be prepared that this make take a few minutes. Extract the R2 statistics and make an LD heatmap (hint: you can use the command `image`) using the R2 statistic.

```
gen.data.genotype <- makeGenotypes(gen.data)

ld_results <- LD(gen.data.genotype)

r2_matrix <- ld_results$"R^2"
r2_matrix[is.na(r2_matrix)] <- 0
# Create a square symmetric matrix
square_r2_matrix <- r2_matrix + t(r2_matrix)

# Define a custom color palette from light yellow to dark red
my_color_palette <- colorRampPalette(c("lightyellow", "darkred"))
# Create an LD heatmap using the image function
image(1:ncol(square_r2_matrix), 1:nrow(square_r2_matrix), square_r2_matrix,
      main = "LD Heatmap", xlab = "SNPs", ylab = "SNPs",
      col = my_color_palette(100), zlim = c(0, 1), asp = 1)
```

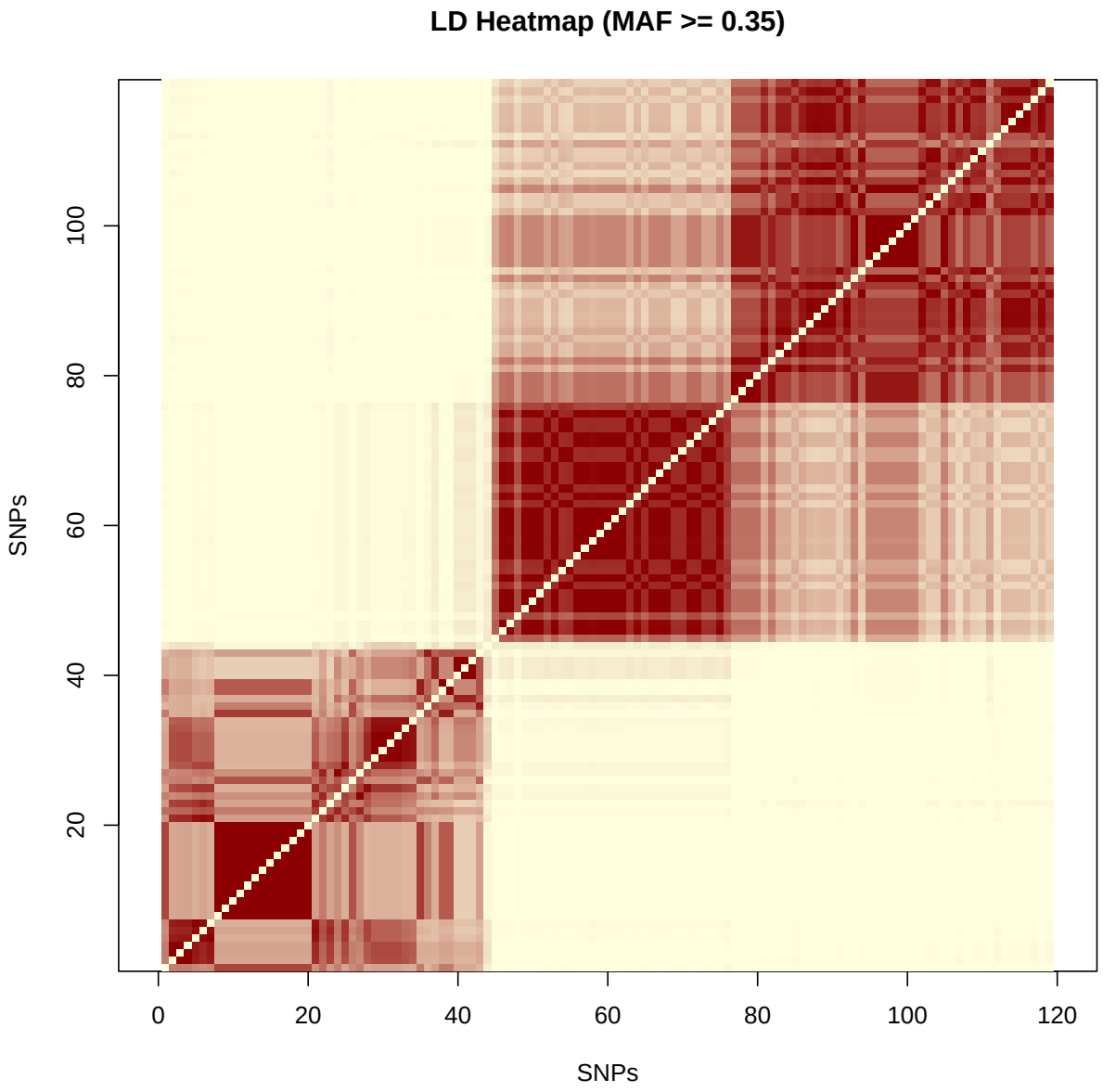


6. Make another heatmap obtained by filtering out all variants with a MAF below 0.35, and redoing the computations to obtain the R2 statistics in R. Can you explain any differences observed between the two heatmaps?

```
genotype.counts.frame <- as.data.frame(genotype.counts)

# Function to calculate Minor Allele Frequency (MAF) from genotype counts
calculate_maf <- function(counts) {
  total_alleles <- sum(counts)

  count.A <- counts[1] + counts[2]/2
  count.B <- counts[3] + counts[2]/2
```

```
  MAF <- min(count.A, count.B) / total_alleles

  return (MAF)
}

# Calculate MAF for each SNP
maf_values <- apply(genotype.counts.frame, 1, calculate_maf)
```

```
maf_threshold <- 0.35

filtered_gen.data.genotype <- gen.data.genotype[, maf_values > maf_threshold]

filtered_r2_matrix <- square_r2_matrix[maf_values > maf_threshold, maf_values > maf_threshold]

# Create an LD heatmap using the image function
image(1:ncol(filtered_r2_matrix), 1:nrow(filtered_r2_matrix), filtered_r2_matrix,
      main = "LD Heatmap (MAF >= 0.35)", xlab = "SNPs", ylab = "SNPs",
      col = my_color_palette(100), zlim = c(0, 1), asp = 1)
```



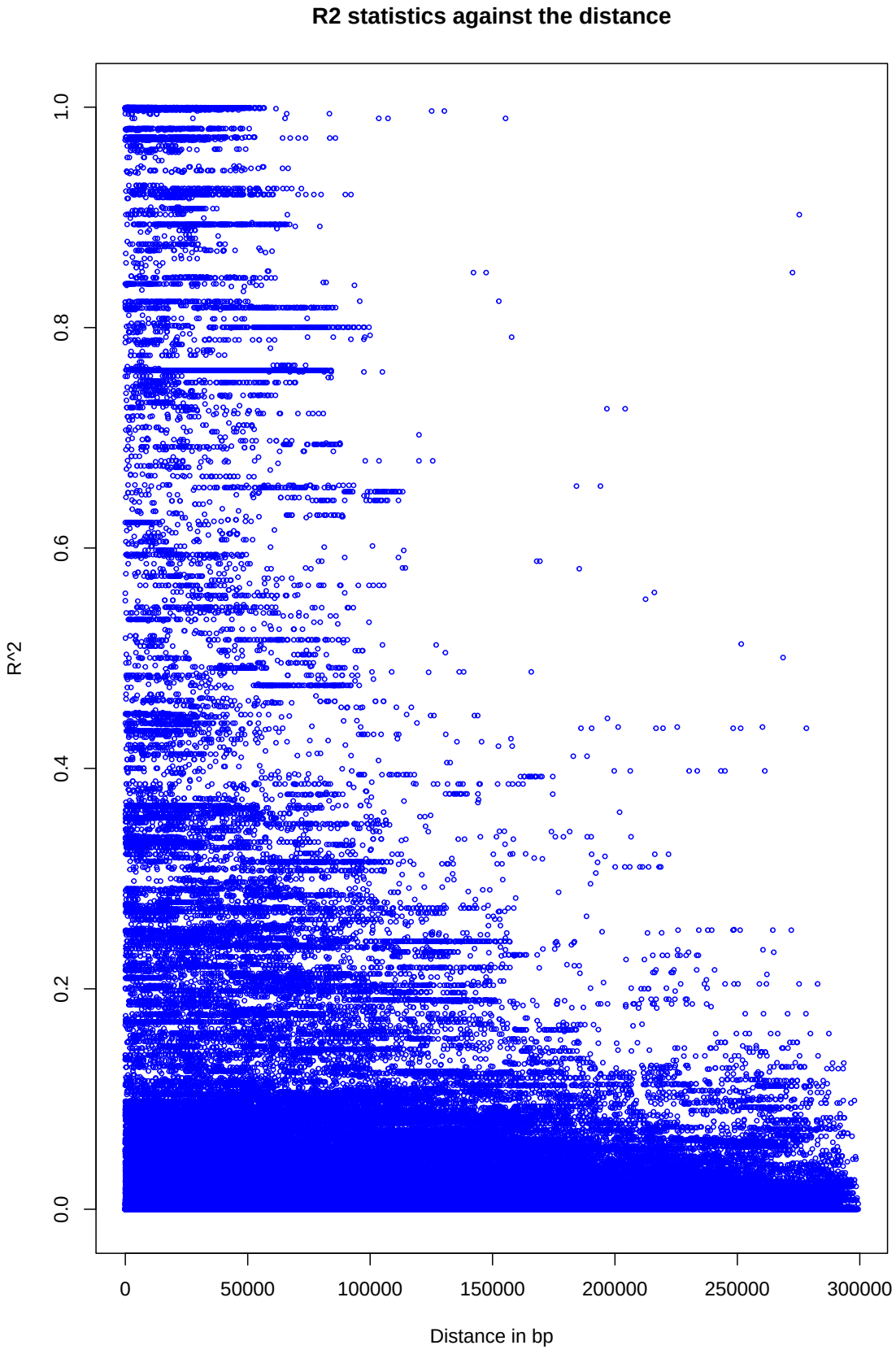
The second heatmap has the much more distinct dark red blocks. This suggests a higher correlation between SNPs in the second heatmap (after removing the SNPs with low MAFs).

7. Compute a distance matrix with the distance in base pairs between all possible pairs of SNPs, using the basepair position of each SNP given in the .bim file. Make a plot of R's R2 statistics against the distance (expressed as the number of basepairs) between the markers. Comment on your results.

```
position.frame <- FOXP2.bim[, c("V2", "V4")]
colnames(position.frame) <- c("name", "position")

# Create a matrix of positions
positions_matrix <- as.matrix(position.frame[["position"]])
```

```r
# Calculate pairwise distances
pairwise_distances <- dist(positions_matrix, method = "manhattan", diag = TRUE)

# Convert the result to a square matrix
pairwise_distances_matrix <- as.matrix(pairwise_distances)

# Set row and column names
rownames(pairwise_distances_matrix) <- position.frame$name
colnames(pairwise_distances_matrix) <- position.frame$name
```

```r
plot(pairwise_distances_matrix, r2_matrix,
     main = "R2 statistics against the distance",
     xlab = "Distance in bp", ylab = "R^2",
     pch = 1, col = "blue", cex = 0.5)
```


R2 statistics against the distance
R^2
0.0
0.2
0.4
0.6
0.8
1.0
0
50000
100000
150000
200000
250000
300000
Distance in bp

From this plot, we can infer that as the distance between markers increases, the correlation between them decreases — a finding consistent with common sense.

Haplotype estimation

```
APOE.dat <- read.table('APOE.dat')
APOE.bim <- read.table('APOE.bim')
```

1. How many individuals and how many SNPs are there in the database? What percent-age of the data is missing?

```
colnames(APOE.dat) <- as.vector(APOE.dat[1, ])
APOE.dat <- APOE.dat[-1,-1]
dim(APOE.dat)
```

```
## [1] 107 162
```

```
NA.counter <- sum(is.na.data.frame(APOE.dat))
NA.percentage <- NA.counter*100/prod(dim(APOE.dat))
```

There are 107 individuals and 162 SNPs in this database.

The percentage of missing data in this database: 0 %..

2. Assuming that all SNPs are bi-allelic, how many haplotypes can theoretically be found for this data set?

Theoretically, as all SNPs are bi-allelic, for each SNP there are 2 different choices : 2^{162} haplotypes.

3. Estimate haplotype frequencies using the haplo.em function that you will find in the haplo.stats package. How many haplotypes do you find? List the estimated probabilities in decreasing order. Which haplotype number is the most common?

```
col.vectors <- lapply(APOE.dat, identity)

Geno <- c()
for (col.vec in col.vectors){
  snp <- gsub("/", "", col.vec)
  Geno <- cbind(Geno, substr(snp,1,1), substr(snp,2,2))
}

HaploEM <- haplo.em(Geno, locus.label=colnames(APOE.dat))
```

There are 34 different haplotypes.

```
# list of the estimated probabilities in decreasing order
hap.prob.dec <- sort(HaploEM$hap.prob, decreasing = T)
```

Probabilities of haplotypes in decreasing order: 0.4027626, 0.1308411, 0.0716681, 0.0681801, 0.0502102, 0.0454821, 0.0357199, 0.0349952, 0.0225032, 0.0190724, 0.0186916, 0.0157937, 0.0087047, 0.0072285, 0.0046729, 0.0046729, 0.0046729, 0.0046729, 0.0046729, 0.0046729, 0.0046729, 0.0046729, 0.0046729, 0.0040448, 0.0034961, 0.0033865, 0.0030156, 0.0022284, 0.0018612, 0.0012766, 0.0012469, 8.6128028×10^{-4}, 5.2443151×10^{-7}.

The 29th haplotype is the most common (with a probability of 0.4027626).

4. Remove all genetic variants that have a minor allele frequency below 0.10 from the database, and re-run haplo.em. How does this affect the number of haplotypes? Comment on your results.

```
# find all genetic variants that have a minor allele frequency below 0.10
i <- 1
ind.to.remove <- c()
for (col.vec in col.vectors){
  geno <- genotype(col.vec)
  prop1 <- sum(allele.count(geno, APOE.bim[i, 5]))/324
  prop2 <- sum(allele.count(geno, APOE.bim[i, 6]))/324
  props <- c(prop1, prop2)
  if (min(props) < 0.1){
    ind.to.remove <- c(ind.to.remove, i)
  }
  i <- i+1
}

# remove them from the dataset
APOE.dat <- APOE.dat[,-ind.to.remove]

# rerun haplo.em
col.vectors <- lapply(APOE.dat, identity)
Geno <- c()
for (col.vec in col.vectors){
  snp <- gsub("/", "", col.vec)
  Geno <- cbind(Geno, substr(snp,1,1), substr(snp,2,2))
}
HaploEM <- haplo.em(Geno, locus.label=colnames(APOE.dat))
```

There are 8 different haplotypes.

```
# list of the estimated probabilities in decreasing order
hap.prob.dec <- sort(HaploEM$hap.prob, decreasing = T)
```

Probabilities of haplotypes in decreasing order: 0.6208947, 0.1629508, 0.1127502, 0.0747664, 0.0186916, 0.0052735, 0.0046729, $5.4303735 \times 10^{-10}$.

The 8th haplotype is the most common (with a probability of 0.6208947).

Filtering out genetic variants with a minor allele frequency below 0.10 resulted in a notable reduction in haplotype diversity, decreasing the number of haplotypes from 34 to 8. This simplification suggests a focus on more common variants and a potential loss of information related to rare genetic variation. The impact could be considered in terms of the goals of the analysis, with a trade-off between interpretability and retaining nuanced genetic structures.