

Practical 03 SG: LD and Haplotype estimation

Carlos Moyano & Kleber Reyes

2022-11-28

Linkage Disequilibrium

files: <https://www.cog-genomics.org/plink/1.9/formats#bim>

2. 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?

```
individuals <- nrow(genotypeData); individuals
```

```
## [1] 104
```

```
SNPs <- ncol(genotypeData); SNPs
```

```
## [1] 543
```

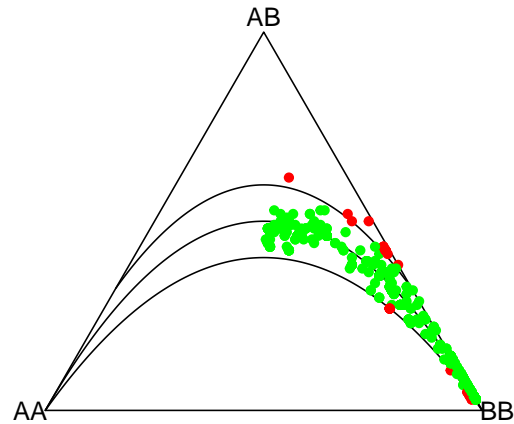
```
perc.mis <- 100*sum(is.na(genotypeData))/(SNPs*individuals); perc.mis
```

```
## [1] 0
```

We have 104 individuals and 543 SNPs (variants) in the dataset. No missing data.

3. Determine the genotype counts for each SNP, and depict all SNPs simultaneously in a ternary plot, and comment on your result. For how many variants do you reject Hardy-Weinberg equilibrium using an ordinary chi-square test without continuity correction? (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). ‘

```
alleles.SNP <- paste(variantsData[, "Allele.1"], variantsData[, "Allele.2"], sep="/")
genotype.counts <- MakeCounts(genotypeData, alleles.SNP, sep="/")[, 1:3]
HWTernaryPlot(genotype.counts)
```



It seems that the majority of the SNPs follow the HWE.

```
chisq.pvalues <- HWChisqStats(genotype.counts, pvalues = TRUE)
sum(chisq.pvalues < 0.05)
```

```
## [1] 33
```

There are 33 SNPs whose p-values are below the 0.05 threshold. This means that, statistically, these variants follow the HWE.

4. 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?

```
rs34684677 <- genotypeData[, "rs34684677"]
rs2894715 <- genotypeData[, "rs2894715"]
rs34684677.g <- genotype(rs34684677, sep="/")
rs2894715.g <- genotype(rs2894715, sep="/")
LD(rs34684677.g, rs2894715.g)
```

```
##
## Pairwise LD
## -----
##              D          D'          Corr
## Estimates: -0.05493703 0.9986536 -0.3144048
##
##              X^2      P-value      N
## LD Test: 20.56088 5.77645e-06 104
```

We can notice that the estimated D is negative (-0.055), then we are on the repulsion case, this entails that probability of heterozygotes is higher than the expected under independence, the standardized D or D' has value near to 1 (0.999). As D' is close to 1 we can say that these variables have a high LD. Since p-value is below 0.05 we can reject the H0 that states that D=0 (D'=0 and cor=0) and confirm that estimated values for D, D' and correlation are valid.

5. Also compute the LD statistic D for the SNPs rs34684677 and rs998302 of the database. Is there significant association between these two SNPs? Is there any reason why rs998302 could have stronger or weaker correlation than rs2894715?

```
# rs34684677 and rs998302
rs34684677 <- genotypeData[, "rs34684677"]
rs998302 <- genotypeData[, "rs998302"]
rs34684677.g <- genotype(rs34684677, sep="/")
rs998302.g <- genotype(rs998302, sep="/")
LD(rs34684677.g, rs998302.g)

##
## Pairwise LD
## -----
##           D           D'          Corr
## Estimates: 0.007208888 0.1792444 0.09112725
##
##           X^2    P-value    N
## LD Test: 1.727268 0.1887601 104
```

```
# rs2894715 and rs34684677
rs2894715 <- genotypeData[, "rs2894715"]
rs2894715.g <- genotype(rs2894715, sep="/")
LD(rs34684677.g, rs2894715.g)

##
## Pairwise LD
## -----
##           D           D'          Corr
## Estimates: -0.05493703 0.9986536 -0.3144048
##
##           X^2    P-value    N
## LD Test: 20.56088 5.77645e-06 104
```

In the case rs34684677 and rs998302, p-value is > 0.05 , therefore, we cannot reject the null hypothesis and we have to say that the D, D' and correlation estimates are not valid (not significantly different to 0).

In the case rs34684677 and rs2894715, p-value is $\ll 0.5$, meaning that we can reject the null hypothesis and take into account the estimated values of D, D' and correlation since they are significantly different to 0.

D' is very close to 1, so there's a high LD.

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

```
snp1 <- unname(sapply(rs34684677, function (x) str_replace_all(x, "/", "")))
snp2 <- unname(sapply(rs2894715, function (x) str_replace_all(x, "/", "")))
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,control=haplo.em.control(min.posterior=1e-4))
HaploEM$hap.prob
```

```
## [1] 0.3365385 0.5000000 0.1634615
```

From the EM algorithm output we can observe that the most common haplotype is GT, with probability 0.5.

7. Compute the LD statistics R2 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. Also compute an alternative estimate of R2 obtained by using the PLINK program.

For this purpose you should: * Download and install PLINK 1.90 from <https://www.cog-genomics.org/plink2/> * Take care to store the files FOXP2.bim, FOXP2.fam and FOXP2.bed in a directory where PLINK can find them. * Compute LD estimates with PLINK using `plink -bfile FOXP2 -r2 -matrix -out FOXP2.ld`. This creates a file with extension FOXP2.ld that contains a matrix with all R2 statistics. Read this file into the R environment. Make a scatter plot for R's LD estimates against PLINK's LD estimates. Are they identical or do they at least correlate? What's the difference between these two estimators? Which estimator would you prefer and why?

```
R_estimates <- LD(makeGenotypes(genotypeData))
R_estimates <- R_estimates$`R^2`
plinkstring01 <- "plink --bfile FOXP2 --r2 --matrix --out FOXP2"
system(plinkstring01)
PLINK_estimates <- read.table("FOXP2.ld", header = TRUE)
write.table(R_estimates, file="R_estimates_q_7.Rdata")
write.table(PLINK_estimates, file="PLINK_estimates_q_7.Rdata")
```

```
R_estimates_q_7 <- read.table("R_estimates_q_7.Rdata")
PLINK_estimates_q_7 <- read.table("PLINK_estimates_q_7.Rdata")
R_estimates_q_7 <- R_estimates_q_7[2:dim(R_estimates_q_7)[1],]
R_estimates_q_7 <- as.vector(t(replace(R_estimates_q_7, is.na(R_estimates_q_7), 0)))
PLINK_estimates_q_7 <- as.vector(t(replace(PLINK_estimates_q_7, is.na(PLINK_estimates_q_7), 0)))
plot(R_estimates_q_7, PLINK_estimates_q_7)
```

The scatter plot of both estimates (R vs. PLINK) shows that there's a correlation between them. However, we have noticed that there are cases where the PLINK estimate is 1 and the R estimate is 0, this is probably due to the fact that we have imputed the NAs as 0s, but still interesting because this means that in some cases R fails calculating the R2 and PLINK does not, so this later would be preferred.

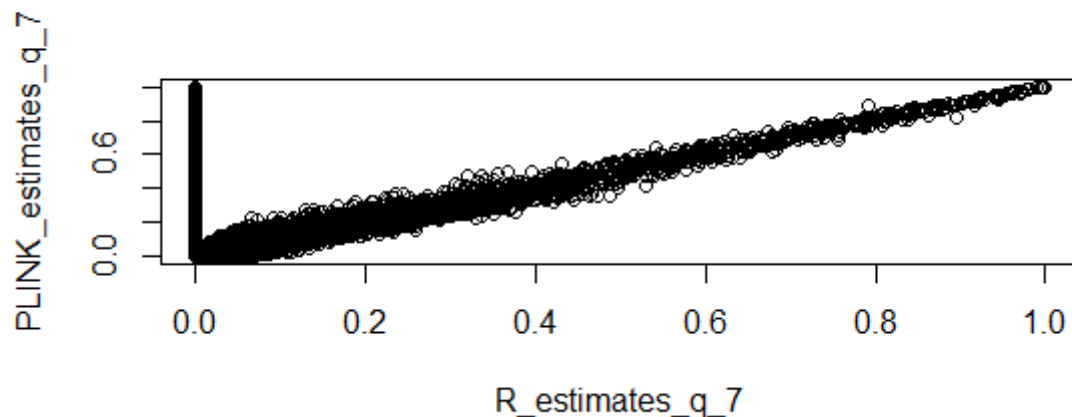


Figure 1: unchanged image

8. 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.

```
basepair_pos_df <- read.table("FOXP2.bim")
rownames(basepair_pos_df) <- basepair_pos_df$V2
basepair_pos_df <- basepair_pos_df %>% dplyr::select(V4)
basepair_dist_df <- as.matrix(dist(basepair_pos_df, method = "manhattan"))
diag(basepair_dist_df) <- NA # we have the elements of the diagonal in the matrix
basepair_dist <- as.vector(basepair_dist_df)
basepair_dist <- basepair_dist[!is.na(basepair_dist)]
plot(R_estimates_q_7, basepair_dist)
```

In the resulting plot we can observe that SNPs which are very distant from each other are generally not highly correlated, except in some very rare cases that could be considered outliers.

9. Make an LD heatmap of the markers in this database, using the R2 statistic with the LD function. 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?

```
rgb.palette <- colorRampPalette(rev(c("blue", "orange", "red")), space = "rgb")
LDheatmap(makeGenotypes(genotypeData), LDmeasure="r", color=rgb.palette(18)) # r stands for R2
maf_mask <- maf(genotype.counts) < 0.35
LDheatmap(makeGenotypes(genotypeData[,maf_mask]), LDmeasure="r", color=rgb.palette(18)) # r stands for R2
```

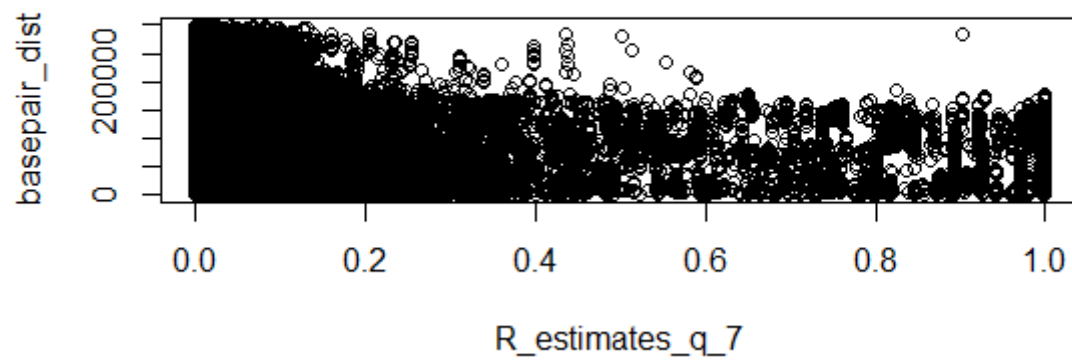


Figure 2: unchanged image

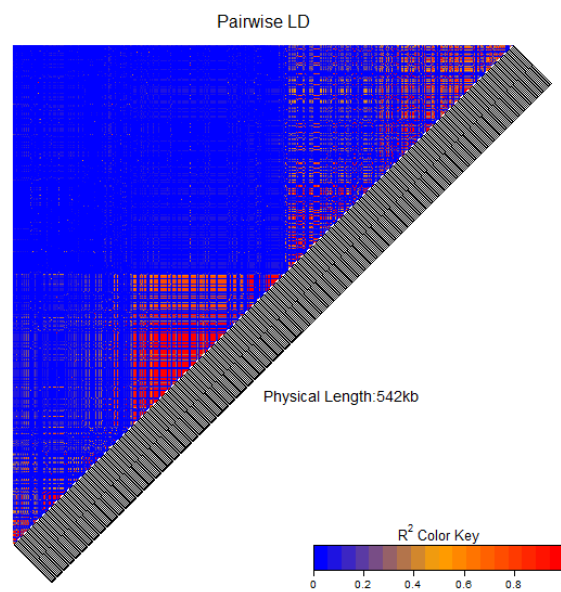


Figure 3: half-size image

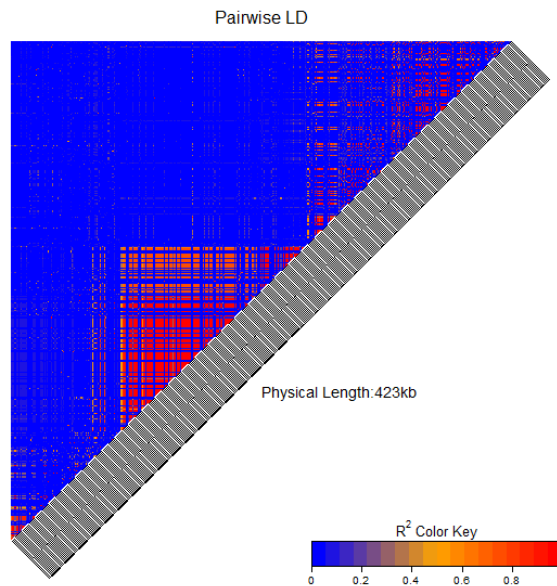


Figure 4: half-size image

10. Can you distinguish blocks of correlated markers in the area of the FOXP2 gene? How many blocks do you think that at least seem to exist?

Observing the plots from question 9, it seems that there exist at least 2 blocks of correlated markers.

11. Simulate independent SNPs under the assumption of Hardy-Weinberg equilibrium, using R's sample instruction [...]. Simulate as many SNPs as you have in your database, and take care to match each SNP in your database with a simulated SNP that has the same sample size and allele frequency. Make an LD heatmap of the simulated SNPs, using R2 as your statistic. Compare the results with the LD heatmap of the FOXP2 region. What do you observe? State your conclusions.

```
n <- dim(genotypeData)[1]
m <- dim(genotypeData)[2]
M <- matrix(0, n, m)
for (i in 1:m) {
  p <- unname(af(genotype.counts[i,]))
  q <- 1 - p
  M[,i] <- sample(c("AA", "AB", "BB"), n, replace=TRUE, prob=c(p*p, 2*p*q, q*q))
}
M <- as.data.frame(M)
M <- M %>% mutate(across(everything(), function(x) genotype(x, sep="")))
```

```
rgb.palette <- colorRampPalette(rev(c("blue", "orange", "red")), space = "rgb")
LDheatmap(makeGenotypes(M), LDmeasure="r", color=rgb.palette(18))
```

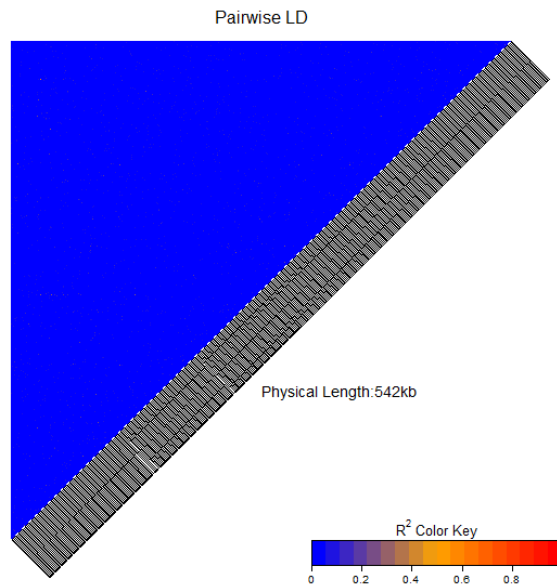


Figure 5: half-size image

In the heatmap of the simulated SNPs under Hardy-Weinberg equilibrium, it is clearly observable that there are no identifiable correlation blocks of SNPs as in the case of the heatmap of the FOXP2 region's SNPs. This present correlation for some SNPs of the FOXP2 region suggests that the underlying cause is that there are not independent from each other.

Haplotype estimation

1. Load data

```
haploData <- read.table("APOE/APOE.dat", header = TRUE)
individuals <- haploData[,1]
haploData <- haploData[, 2: ncol(haploData)]
```

2. How many individuals and how many SNPs are there in the database? What percentage of the data is missing?

```
individuals <- nrow(haploData); individuals
```



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

```
SNPs <- ncol(haploData); SNPs
```

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

```
perc.mis <- 100*sum(is.na(haploData))/(SNPs*individuals); perc.mis
```

```
## [1] 0
```

We have 107 individuals and 162 SNPs (variants) in the dataset. No missing data.

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

Assuming all SNPs are bi-allelic, there are 2^m possible haplotypes, so 2^{162} being m the number of SNPs ($5.8460065e+48$).

4. Estimate haplotype frequencies using the haplo.stats package (set the minimum posterior probability to 0.001). How many haplotypes do you find? List the estimated probabilities in decreasing order. Which haplotype number is the most common?

```
snp_mat <- haploData %>%
  mutate(across(everything(),
    function(x) unname(sapply(x, function(x) str_replace_all(x, "/", ""))))))
index <- 1
Geno <- cbind(substr(snp_mat[,index],1,1),substr(snp_mat[,index],2,2))
snpnames <- c(names(haploData)[index])
for (snp_i in 1:(SNPs - 1)) {
  index <- index + 1
  Geno <- cbind(Geno, substr(snp_mat[,index],1,1),substr(snp_mat[,index],2,2))
  snpnames <- c(snpnames, names(haploData)[index])
}
HaploEM <- haplo.em(Geno,locus.label=snpnames,control=haplo.em.control(min.posterior=1e-4))
sort(HaploEM$hap.prob, decreasing=TRUE)
```

```
## [1] 0.3995355708 0.1308411215 0.0744771058 0.0683865911 0.0501820026
## [6] 0.0467289720 0.0358445308 0.0351607954 0.0225438117 0.0204953083
## [11] 0.0186915888 0.0161107523 0.0086864717 0.0073492154 0.0046728972
## [16] 0.0046728972 0.0046728972 0.0046728972 0.0046728972 0.0046728972
## [21] 0.0046728972 0.0046728972 0.0046728972 0.0046728972 0.0040261016
## [26] 0.0034178360 0.0033026973 0.0028691777 0.0021412538 0.0016594492
## [31] 0.0008206743
```

```
which.max(HaploEM$hap.prob)
```

```
## [1] 27
```

Haplotype number 27 of the 31 is the most common.

5. Is the haplotypic constitution of any of the individuals in the database ambiguous or uncertain? For how many? What is the most likely haplotypic constitution of individual NA20763? (identify the constitution by the corresponding haplotype numbers).

```
#variantsColnames <- c("FID", "IID", "FatherID", "MotherID", "Sex", "Phenotype")
#individualsData <- read.table("APOE/APOE.fam", header = FALSE, col.names = variantsColnames)
```

```
NA20763.ID <- which(individuals == "NA20763")
```

```
HaploEM$hap1code[NA20763.ID]
```

```
## numeric(0)
```

```
HaploEM$hap2code[NA20763.ID]
```

```
## numeric(0)
```

```
snp_mat.NA20763.ID <- haploData[NA20763.ID,] %>%
  mutate(across(everything(),
    function(x) unname(sapply(x, function(x) str_replace_all(x, "/", ""))))))
```

The possible haplotypic constitution of NA20763 are haplotypes 8 and 23. ## 6. Suppose we would delete polymorphism rs374311741 from the database prior to haplotype estimation. Would this affect the results obtained? Justify your answer.

```
rs374311741 <- haploData[, "rs374311741"]; rs374311741
```

```
## [1] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
## [13] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
## [25] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
## [37] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
## [49] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
## [61] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
## [73] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
## [85] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
## [97] "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C" "C/C"
```

```
haploData.2 <- data.frame(haploData)
```

```
haploData.2[, "rs374311741"] <- NULL
```

```
SNPs.2 <- ncol(haploData.2);
```

```
snp_mat.2 <- haploData.2 %>%
  mutate(across(everything(),
    function(x) unname(sapply(x, function(x) str_replace_all(x, "/", ""))))))
```

```
index <- 1
```

```
Geno.2 <- cbind(substr(snp_mat.2[, index], 1, 1), substr(snp_mat.2[, index], 2, 2))
```

```
snpnames.2 <- c(names(haploData.2)[index])
```

```

for (snp_i in 1:(SNPs.2 - 1)) {
  index <- index + 1
  Geno.2 <- cbind(Geno.2, substr(snp_mat.2[,index],1,1),substr(snp_mat.2[,index],2,2))
  snpnames.2 <- c(snpnames.2, names(haploData.2)[index])
}
HaploEM.2 <- haplo.em(Geno.2,locus.label=snpnames.2,control=haplo.em.control(min.posterior=1e-4))
sort(HaploEM.2$hap.prob, decreasing=TRUE)

```

```

## [1] 0.3995008681 0.1308411215 0.0744798784 0.0684167298 0.0501814042
## [6] 0.0467289720 0.0358600998 0.0351613684 0.0225548462 0.0204956148
## [11] 0.0186915888 0.0161182357 0.0086858516 0.0073507432 0.0046728972
## [16] 0.0046728972 0.0046728972 0.0046728972 0.0046728972 0.0046728972
## [21] 0.0046728972 0.0046728972 0.0046728972 0.0046728972 0.0040258352
## [26] 0.0034034855 0.0033020560 0.0028688712 0.0021365001 0.0016573180
## [31] 0.0008096397

```

```

which.max(HaploEM.2$hap.prob)

```

```

## [1] 27

```

If we have a look to the variant rs374311741, we can realize that is an homozygote (only “C/C”). The result will not have a huge impact since a homozygote is considered a unambiguous haplotype and the variability will not be affected. If we recalculate the Haplotypes we will find that values remain without important changes.

7. 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?

```

maf.list <- vector(mode="numeric", length=SNPs)

for (i in 1:SNPs) {
  variant.g <- genotype(haploData[, i],sep="/")
  variant.g.summary <- summary(variant.g)

  # MAF
  maf.list[i] <- min(variant.g.summary$allele.freq[, "Proportion"], na.rm = T)
}

maf.ind <- which(maf.list < 0.1)

haploData.2 <- haploData[,-maf.ind]

```

```

SNPs.2 <- ncol(haploData.2);
snp_mat.2 <- haploData.2 %>%
  mutate(across(everything(),
    function(x) unname(sapply(x, function (x) str_replace_all(x, "/", ""))))))
index <- 1

```

```

Geno.2 <- cbind(substr(snp_mat.2[,index],1,1),substr(snp_mat.2[,index],2,2))
snpnames.2 <- c(names(haploData.2)[index])
for (snp_i in 1:(SNPs.2 - 1)) {
  index <- index + 1
  Geno.2 <- cbind(Geno.2, substr(snp_mat.2[,index],1,1),substr(snp_mat.2[,index],2,2))
  snpnames.2 <- c(snpnames.2, names(haploData.2)[index])
}
HaploEM.2 <- haplo.em(Geno.2,locus.label=snpnames.2,control=haplo.em.control(min.posterior=1e-4))
sort(HaploEM.2$hap.prob, decreasing=TRUE)

```

```

## [1] 0.620635586 0.130841121 0.113009274 0.074766355 0.031850539 0.018691589
## [7] 0.005532638 0.004672897

```

```

which.max(HaploEM.2$hap.prob)

```

```

## [1] 8

```

The number of haplotypes decrease significantly. Since haplotypes variations are added by the MAF (that introduce variations to the existing haplotypes).

8. We could consider the newly created haplotypes in our last run of haplo.em as the alleles of a new superlocus. Which is, under the assumption of Hardy-Weinberg equilibrium, the most likely genotype at this new locus? What is the probability of this genotype? Which genotype is the second most likely, and what is its probability?

```

haplo <- HaploEM.2$haplotype
haplo.freq <- HaploEM.2$hap.prob

hPair <- get.hapPair(haplo, haplo.freq, base.index=1)

gen.prob.sorted <- sort(hPair$p.g, index.return=TRUE, decreasing = TRUE)$ix

first <- gen.prob.sorted[1]
second <- gen.prob.sorted[2]

hPair$p.g[first]

```

```

## [1] 0.3851885

```

```

hPair$x.haplo[first,]

```

```

## [1] 0 0 0 0 0 0 2

```

```

hPair$p.g[second]

```

```

## [1] 0.1624093

```

```
hPair$x.haplo[second,]
```

```
## [1] 0 0 0 0 0 0 1
```

Under HWE the most likely genotype is the one with a pair of haplotype 8 with a probability of 0.3851883. The second most likely genotype has just 1 haplotype 8, its probability is 0.1624093.

The haplotype 8 is made up of the following alleles:

```
as.character(haplo[8,])
```

```
## [1] "C" "C" "G" "C" "C" "G" "C" "T" "G" "C" "C" "G" "G" "C" "T" "C" "A" "T"
## [19] "T" "C" "T" "T" "A" "T" "C" "A" "C" "C" "T" "A" "G" "C" "C" "C" "A" "C"
## [37] "C" "C" "C" "G" "T" "G" "T" "T" "T" "G" "C" "C" "C" "T" "G" "C" "A" "C"
## [55] "A" "G" "G" "G" "G" "C" "A" "G" "C" "G" "C" "T" "T" "C" "G" "G" "C" "A"
## [73] "T" "G" "A" "C" "C" "C" "G" "G" "T" "A" "T" "T" "G" "A" "A" "T" "T" "A"
## [91] "G" "T" "T" "G" "C" "G" "A" "C" "G" "G" "G" "G" "G" "T" "T" "C" "T" "T"
## [109] "C" "C" "G" "A" "A" "C" "A" "T" "T" "C" "C" "C" "C" "G" "T" "C" "A" "A"
## [127] "C" "C" "C" "C" "A" "C" "A" "G" "G" "C" "G" "G" "T" "A" "G"
```