BSG-MDS practical 1 Statistical Genetics

Eliya Tiram and Ximena Moure

07/11/2023, submission deadline 14/11/2023

```
data <- read.table("TSICHR22RAW.raw", header=TRUE)
genetic_data <- data[, 7:ncol(data)]
```

1. How many variants are there in this database? What percentage of the data is missing?

```
num_variants <- ncol(genetic_data)
percentage_missing <- mean(is.na(genetic_data)) * 100

cat("\nNum variants:",num_variants, "\n")
```

```
##
## Num variants: 20649
```

```
cat("\nPercentage missing:",percentage_missing, "\n")
```

```
##
## Percentage missing: 0.1986518
```

There are 20649 variants. The percentage of data missing is 0.20%.

2. Calculate the percentage of monomorphic variants. Exclude all monomorphics from the database for all posterior computations of the practical. How many variants do remain in your database?

```
is_monomorphic <- apply(genetic_data, 2, function(x) length(unique(na.omit(x))) == 1)
percentage_monomorphic <- sum(is_monomorphic) / length(is_monomorphic) * 100

# Removing monomorphic variants
polymorphic_data <- genetic_data[, !is_monomorphic]
remaining_variants <- ncol(polymorphic_data)

cat("\nPercentage monomorphic:",percentage_monomorphic, "%\n")
```

```
##
## Percentage monomorphic: 11.45818 %
```

```
cat("\nRemaining variants:",remaining_variants, "\n")
```

```
##
## Remaining variants: 18283
```

Percentage monomorphic: 11.45818 % Remaining variants: 18283

3. Report the genotype counts and the minor allele count of polymorphism rs8138488 C, and calculate the MAF of this variant.

```
rs8138488C_genotypes <- na.omit(polymorphic_data$rs8138488_C)

# Calculate the genotype counts for AA, AB, and BB
genotype_counts <- table(rs8138488C_genotypes)


aa_count <- genotype_counts["0"]
ab_count <- genotype_counts["1"]
bb_count <- genotype_counts["2"]

total_alleles <- 2 * (aa_count + ab_count + bb_count)
pA <- (2 * aa_count + ab_count) / total_alleles  # frequency of allele A
pB <- (2 * bb_count + ab_count) / total_alleles  # frequency of allele B

# Calculate MAF
maf <- min(pA, pB)

# Output the results
cat("Genotype counts for rs8138488 (AA AB BB):", genotype_counts, "\n")
```

```
## Genotype counts for rs8138488 (AA AB BB): 41 47 14
```

```
cat("Minor allele count for rs8138488_C:", maf * total_alleles, "\n")
```

```
## Minor allele count for rs8138488_C: 75
```

```
cat("MAF for rs8138488:", maf, "\n")
```

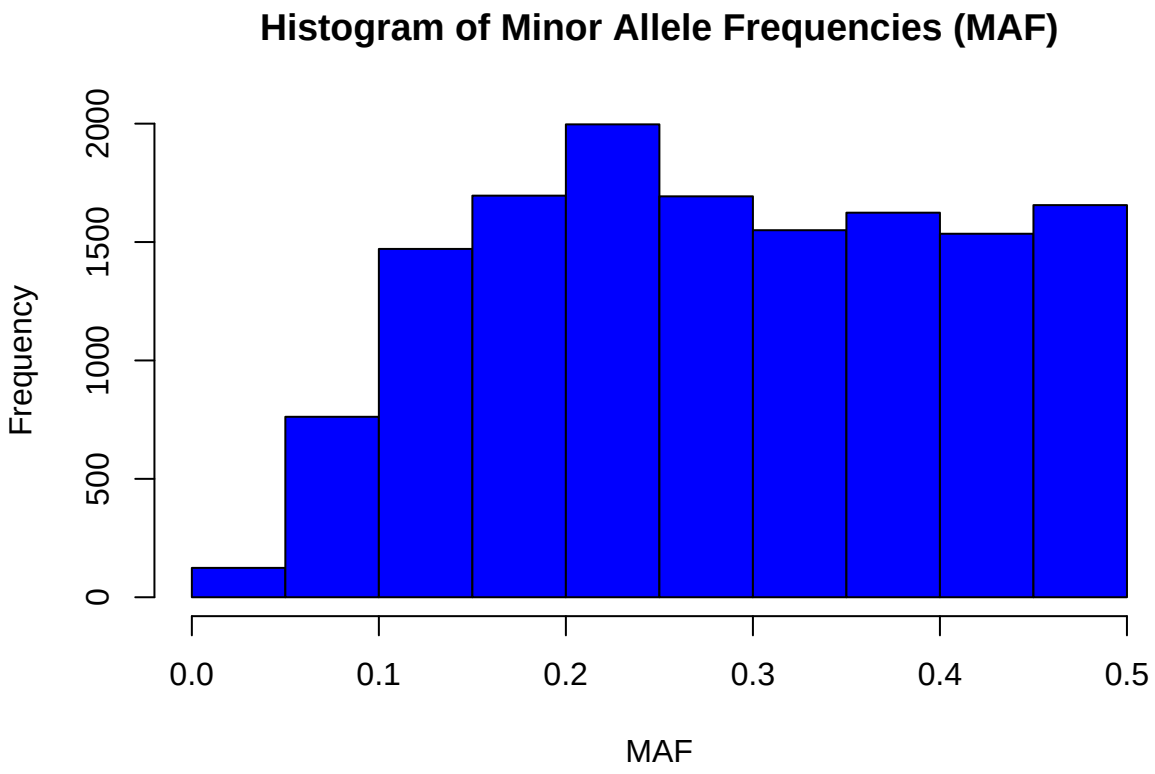
```
## MAF for rs8138488: 0.3676471
```

Genotype counts: - AA: 41 - AB: 47 - BB: 14

Minor allele count: 75 Maf: 0.3677

4.Compute the minor allele frequencies (MAF) for all markers, and make a histogram of it. Does the MAF follow a uniform distribution? What percentage of the markers have a MAF below 0.05? And below 0.01? Can you explain the observed pattern?

```
calculate_MAF <- function(genotype_column) {
  # Count the occurrences of each genotype
  genotype_counts <- table(genotype_column)

  pA <- (genotype_counts["0"] + 0.5 * genotype_counts["1"]) / sum(genotype_counts)
  pB <- (genotype_counts["2"] + 0.5 * genotype_counts["1"]) / sum(genotype_counts)

  # Calculate MAF
  MAF <- min(pA, pB)
  return(MAF)
}

# Apply the calculate_MAF function to each column (variant) in the data
MAF_values <- apply(polymorphic_data, 2, calculate_MAF)

# Omit NA values if there are any
MAF_values <- na.omit(MAF_values)

hist(MAF_values, main = "Histogram of Minor Allele Frequencies (MAF)",
     xlab = "MAF", ylab = "Frequency", col = "blue")
```




```
# Percentage of variants with MAF below 0.05 and below 0.01
percentage_below_5 <- sum(MAF_values < 0.05) / length(MAF_values) * 100
percentage_below_1 <- sum(MAF_values < 0.01) / length(MAF_values) * 100

# Output the percentages
cat("Percentage of variants with MAF below 0.05:", percentage_below_5, "%\n")
```

```
## Percentage of variants with MAF below 0.05: 0.8718458 %
```

```
cat("Percentage of variants with MAF below 0.01:", percentage_below_1, "%\n")
```

```
## Percentage of variants with MAF below 0.01: 0 %
```

The histogram displays a distribution that is non-uniform. There are relatively few values under 0.05, which can be attributed to the allele distribution across the variants — while alleles A and B are not equally distributed, they are not highly disproportionate either. This clustering effect in the data accounts for the non-uniform nature of the distribution.

Percentage of variants with MAF below 0.05: 0.8718458 % Percentage of variants with MAF below 0.01: 0 %

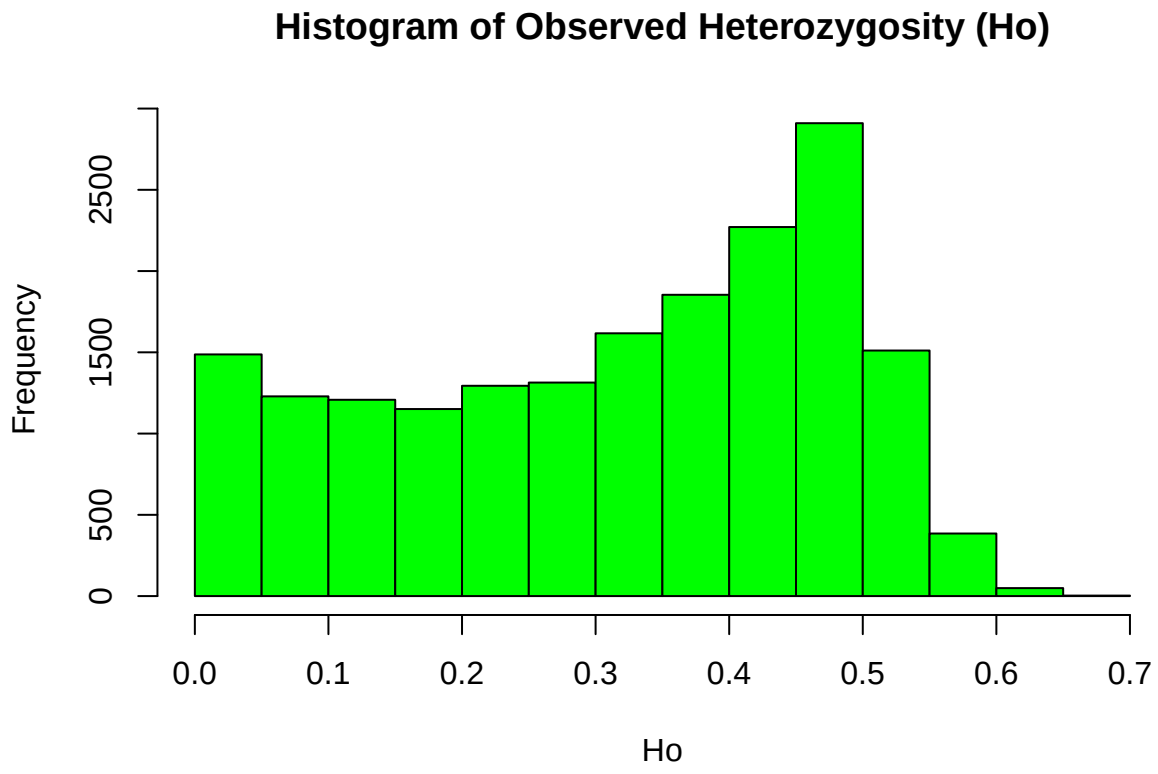
The histogram demonstrates that most variants have a MAF between 0.2 and 0.5, indicating that a substantial portion of alleles are not extremely rare in the population. Very few variants have a MAF below 0.05 (approximately 0.87%), and none have a MAF below 0.01 (0%).

The absence of variants with a Minor Allele Frequency (MAF) below 0.01 could be for different reasons: - relatively homogeneous population genetically. - if the sample size of the population studied is too small,

it might not be large enough to capture the rarest alleles. Rare variants, by definition, appear in a very small proportion of the population, so a larger sample size is needed to detect them. - the technology or methodology used to detect genetic variants can significantly influence which alleles are observed.

5. Calculate the observed heterozygosity H0, and make a histogram of it. What is, theoretically, the range of variation of this statistic?

```
# Calculate observed heterozygosity (Ho) for each marker, which is the frequency of the AB genotype
calculate_ho <- function(genotype) {
  ab_count <- sum(genotype == 1,  na.rm = TRUE)  # Count the number of AB genotypes
  total_genotypes <- sum(!is.na(genotype))     # Count the total number of non-missing genotypes
  ho <- ab_count / total_genotypes               # Calculate Ho as the proportion of AB genotypes
  return(ho)
}

ho_all <- apply(polymorphic_data, 2, calculate_ho)

ho_all <- na.omit(ho_all)
hist(ho_all, main = "Histogram of Observed Heterozygosity (Ho)",
     xlab = "Ho", ylab = "Frequency", col = "green")
```



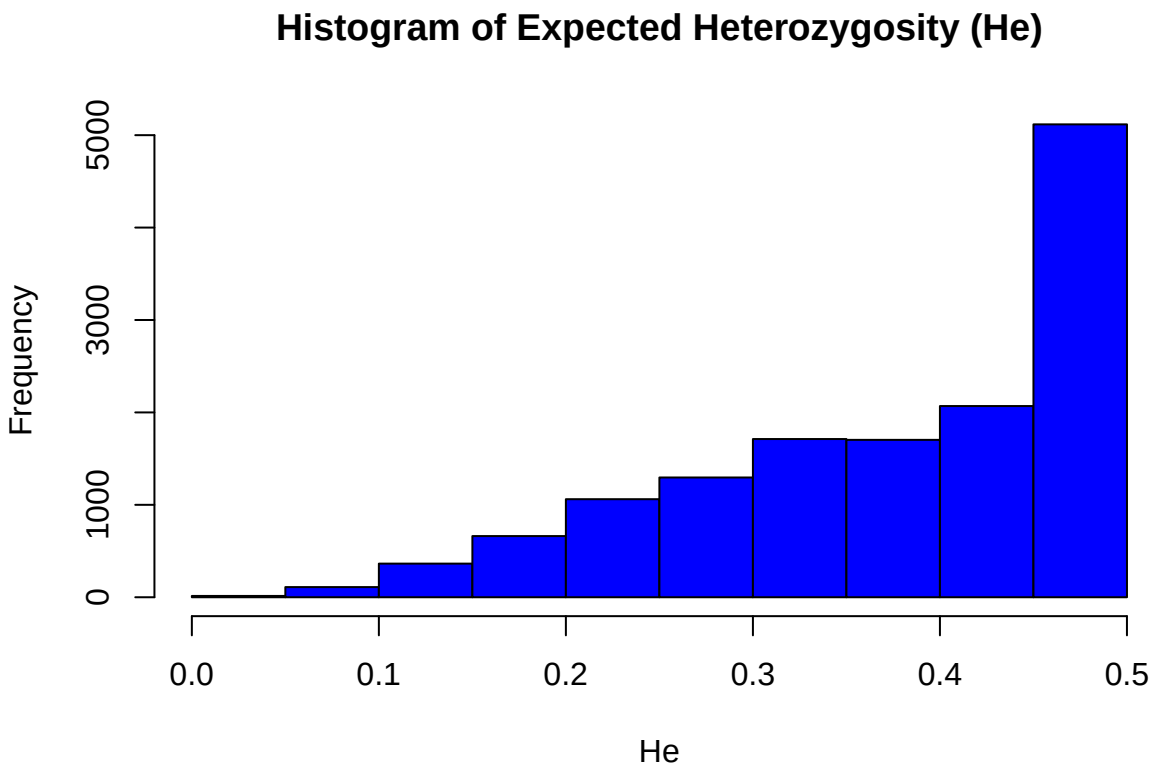

```
summary_ho <- summary(ho_all)
print(summary_ho)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.1765  0.3431  0.3145  0.4510  0.6765
```

The theoretical range of variation for Ho is from 0 to 1, where 0 means no heterozygosity and 1 means complete heterozygosity.

6. Compute for each marker its expected heterozygosity (He), where the expected het- erozygosity for a bi-allelic marker is defined as $1 - \sum_{i=1}^{k} p_i^2$,where p_i^2 is the frequency of the ith allele. Make a histogram of the expected heterozygosity. What is, theoretically, the range of variation of this statistic? What is the average of He for this database?

```r
# Calculate expected heterozygosity (He) for each marker
calculate_He <- function(genotype_column) {

  clean_genotype_column <- na.omit(genotype_column)

  # If all values are NA, return NA for He
  if (length(clean_genotype_column) == 0) {
    return(NA)
  }

  # Count the occurrences of each genotype
  genotype_counts <- table(clean_genotype_column)

  # Calculate allele frequencies
  # AA counts as two A alleles, AB as one A and one B, BB as two B alleles
  num_A <- 2 * genotype_counts["0"] + genotype_counts["1"]
  num_B <- 2 * genotype_counts["2"] + genotype_counts["1"]
  total_alleles <- num_A + num_B

  # Calculate allele frequencies
  pA <- num_A / total_alleles
  pB <- num_B / total_alleles

  He <- 1 - (pA^2 + pB^2)

  return(He)
}


He_values <- apply(polymorphic_data, 2, calculate_He)

# Remove NA values from the result
He_values <- na.omit(He_values)

# Create a histogram of He values
hist(He_values, main = "Histogram of Expected Heterozygosity (He)",
     xlab = "He", ylab = "Frequency", col = "blue")
```




```
# Calculate the average He for the dataset
average_He <- mean(He_values)

# Output the average He
cat("Average expected heterozygosity (He) for this database:", average_He, "\n")
```

```
## Average expected heterozygosity (He) for this database: 0.3770766
```

Theoretically it can vary from 0 to 0.5. A value of 0 indicates a lack of genetic diversity, meaning all individuals in the population possess the same allele. Conversely, a value of 0.5 denotes maximum genetic diversity, where there is an even distribution of the two alleles within the population.

Average expected heterozygosity (He) for this database: 0.3770766

STR dataset

```
library(HardyWeinberg)
```

```
## Loading required package: mice
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##     filter

## The following objects are masked from 'package:base':
##
##     cbind, rbind

## Loading required package: Rsolnp

## Loading required package: nnet
```

```
data("NistSTRs")
```

1. How many individuals and how many STRs contains the database?

```
num_individuals <- nrow(NistSTRs)
num_STRs <- ncol(NistSTRs) / 2  # Since each STR is represented by two columns (alleles)

num_individuals
```

```
## [1] 361
```

```
num_STRs
```

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

```
# Print the results
cat("Number of individuals:", num_individuals, "\n")
```

```
## Number of individuals: 361
```

```
cat("Number of STRs:", num_STRs, "\n")
```

```
## Number of STRs: 29
```

Number of individuals: 361 Number of STRs: 29

2. Write a function that determines the number of alleles for a STR. Determine the number of alleles for each STR in the database. Compute basic descriptive statistics of the number of alleles (mean, standard deviation, median, minimum, maximum)

```r
# Custom function to determine the number of alleles for a given STR
num_alleles_per_STR <- function(str_index) {
  alleles_1 <- NistSTRs[, (2 * str_index - 1)]
  alleles_2 <- NistSTRs[, (2 * str_index)]

  # Combine alleles from both columns
  all_alleles <- c(alleles_1, alleles_2)

  # Count the unique alleles
  num_unique_alleles <- length(unique(all_alleles))

  return(num_unique_alleles)
}

# Determine the number of alleles for each STR in the database
alleles_per_STR <- sapply(1:(ncol(NistSTRs) / 2), num_alleles_per_STR)

# Descriptive statistics
mean_alleles <- mean(alleles_per_STR)
sd_alleles <- sd(alleles_per_STR)
median_alleles <- median(alleles_per_STR)
min_alleles <- min(alleles_per_STR)
max_alleles <- max(alleles_per_STR)

# Return descriptive statistics
list(mean = mean_alleles, sd = sd_alleles, median = median_alleles, min = min_alleles, max = max_allele
```

```
## $mean
## [1] 11.86207
##
## $sd
## [1] 6.226236
##
## $median
## [1] 10
##
## $min
## [1] 6
##
## $max
## [1] 39
```

```r
alleles_per_STR
```

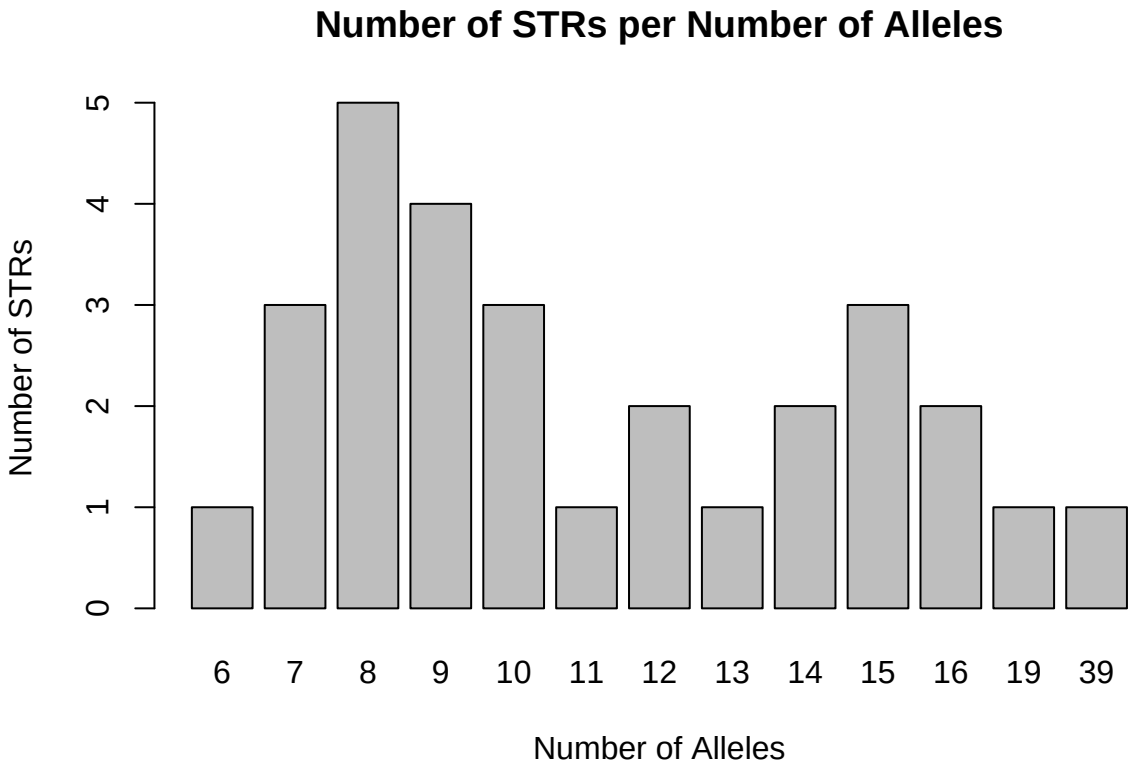
```
##  [1]  7  9 16  8  7 15 15 15 16  8 12 11  9  9 14  9 10 12  6  7 14  8 10 13 19
## [26] 39  8  8 10
```

Mean:11.86 Standard deviation: 6.226236 Median: 10 Minimum: 6 Maximum: 39

Alleles per STR: 7 9 16 8 7 15 15 15 16 8 12 11 9 9 14 9 10 12 6 7 14 8 10 13 19 39 8 8 10

3. Make a table with the number of STRs for a given number of alleles and present a barplot of the number STRs in each category. What is the most common number of alleles for an STR?

```
allele_counts <- table(alleles_per_STR)
barplot(allele_counts, main = "Number of STRs per Number of Alleles", xlab = "Number of Alleles", ylab =
```



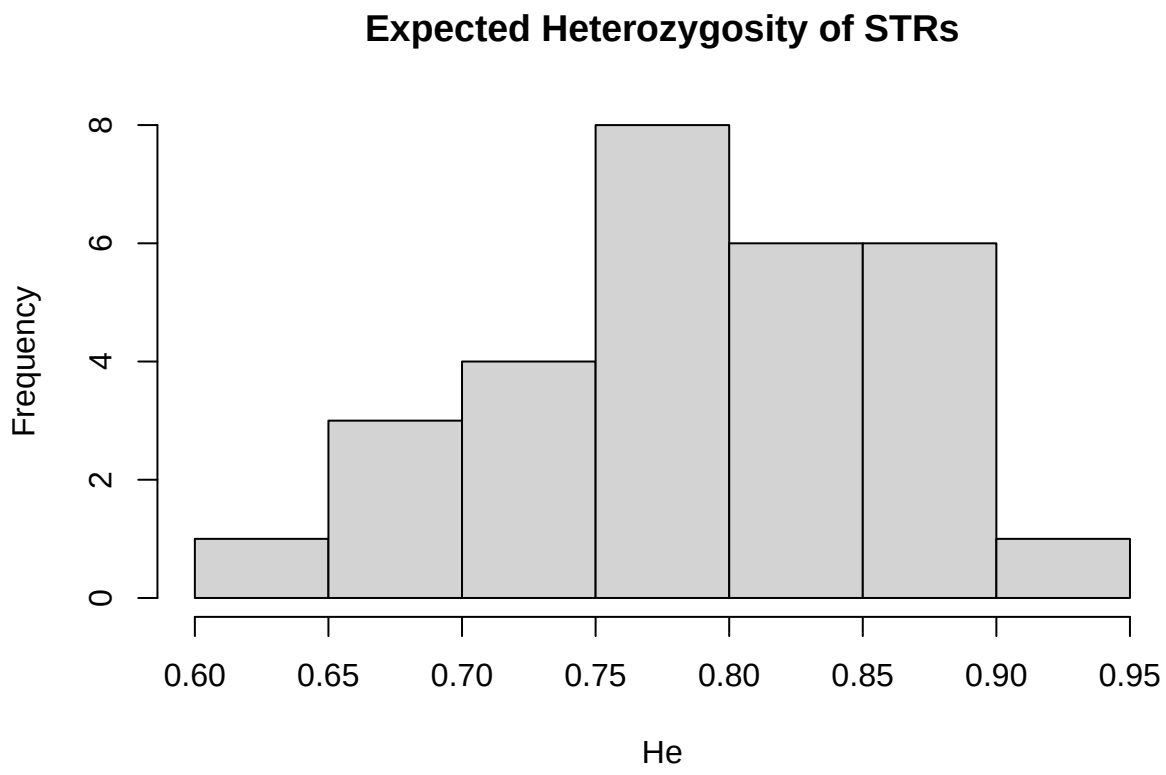

```
most_common_alleles <- as.integer(names(allele_counts)[which.max(allele_counts)])
```

The most common is 8.

4. Compute the expected heterozygosity for each STR. Make a histogram of the expected heterozygosity over all STRS. Compute the average expected heterozygosity over all STRs.

```
calculate_He <- function(STR_data) {
  alleles <- unlist(STR_data)  # Combine both columns into one vector
  allele_frequencies <- table(alleles) / length(alleles)  # Calculate frequencies
  He <- 1 - sum(allele_frequencies^2)
  return(He)
}
```

```r
expected_heterozygosity <- sapply(seq(1, ncol(NistSTRs), by = 2), function(i) {
  calculate_He(NistSTRs[, c(i, i+1)])
})

hist(expected_heterozygosity, main="Expected Heterozygosity of STRs", xlab="He")
```




```r
average_He <- mean(expected_heterozygosity)

average_He
```

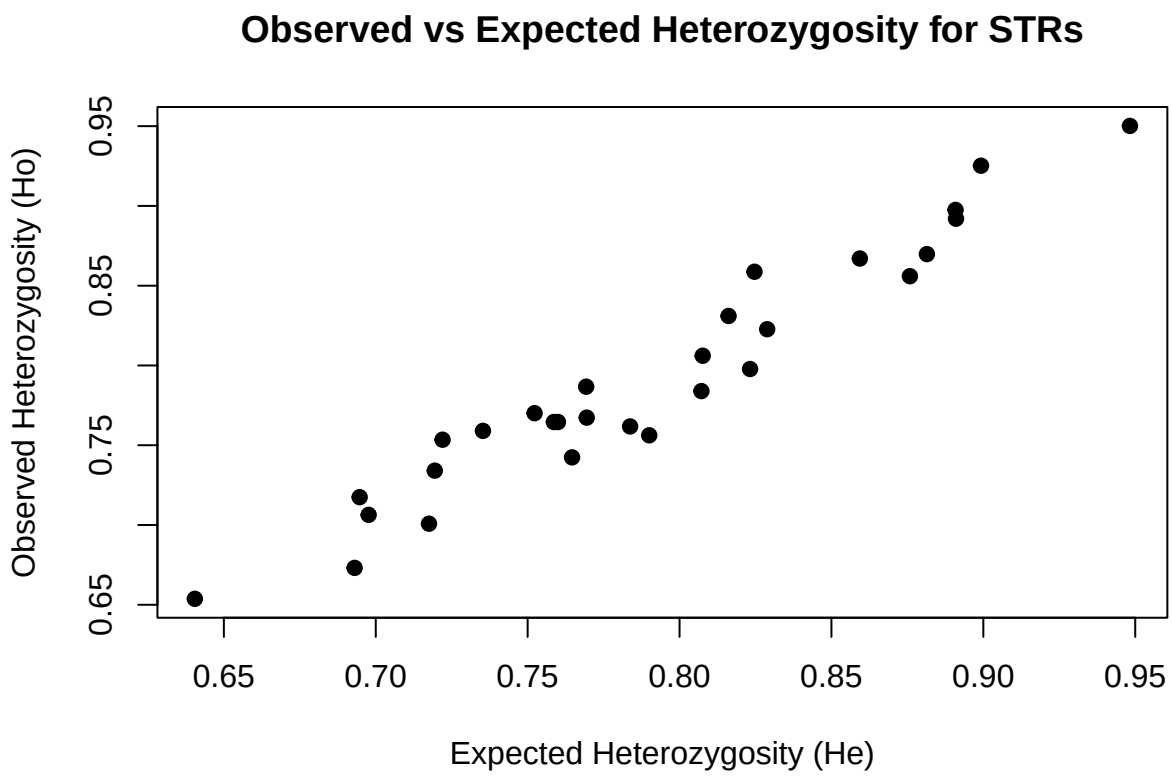
```
## [1] 0.7904043
```

The average is: 0.7904043

5. Calculate also the observed heterozygosity for each STR. Plot observed against ex- pected heterozy- gosity, using all STRs. What do you observe?

```r
calculate_Ho <- function(STR_data) {
  heterozygotes <- sum(STR_data[,1] != STR_data[,2])
  Ho <- heterozygotes / nrow(STR_data)
  return(Ho)
```

```
}

observed_heterozygosity <- sapply(seq(1, ncol(NistSTRs), by = 2), function(i) {
  calculate_Ho(NistSTRs[, c(i, i+1)])
})

plot(expected_heterozygosity, observed_heterozygosity,
     xlab = "Expected Heterozygosity (He)",
     ylab = "Observed Heterozygosity (Ho)",
     main = "Observed vs Expected Heterozygosity for STRs",
     pch = 19)
```




```
#abline(0, 1)
```

Observations: - There don't seem to be any extreme outliers where the observed heterozygosity is drastically different from the expected heterozygosity.

- Some points lie below the line y=x, indicating that the observed heterozygosity is slightly less than expected. This could be due to various factors, including selection, genetic drift, inbreeding, or a small population size.
- The range of heterozygosity values (both Ho and He) from approximately 0.65 to 0.95 indicates good genetic diversity within the STRs analyzed.

6. Compare, overall, the results you obtained for the SNP database with those you obtained for the STR database. What differences do you observe between these two types of genetic markers?

The SNP markers have a smaller number of alleles relative to STR markers because SNPs are bi-allelic by nature (this inherently limits their allelic diversity). STRs typically exhibit mutations that are due to replication slippage, which refers to errors made during DNA replication that can lead to the insertion or deletion of repeats. These types of mutations can introduce new alleles into the population more frequently than single nucleotide changes. Therefore, this leads to generally lower levels of both observed and expected heterozygosity in SNPs, given their narrower allelic diversity.