# Solutions to exercises chapter 5

## 5.6.1 Expected LRs for diallelic SNPs with different allele frequency distributions

Different allele frequency distributions may yield different LR distributions. This information could be relevant when designing new DNA marker panels for kinship testing.

Study, by simulations, LR distributions for a full sibling vs unrelated case scenario for which the genotypes for one diallelic SNP is available. Compare three different allele frequency distributions; minor allele frequency, 0.1, 0.25 and 0.5. Assume the two potential siblings Sib-1 and Sib-2 will be genotyped. Let $H_1$: Sib-1 and Sib-2 are full siblings and $H_2$: Sib-1 and Sib-2 are unrelated. Mutation rates, population substructure, and other complicating factors can be disregarded.

(a) Manually define databases for each minor allele frequency given above.
(b) Define the pedigrees and generate data using 10,000 simulations. Set the seed to 1234. Compute the LR for each simulation and true hypothesis. Summarize the results as the median LR and the probability that the LR will exceed 1.
(c) Repeat (a)-(b) for each minor allele frequency.
(d) Open the frequency database containing data for 100 SNPs and repeat (b).
(e) Repeat (d) for each minor allele frequency.

**Solution:** We start with some theoretical results. Let p be the minor allele frequency (MAF) (0.1, 0.25 and 0.5) and q = 1-p the major. It can be shown that

- The smallest possible LR = 0.25 occurs if Sib-1 and Sib-2 are homozygous for different alleles.
- The largest LR occurs if they are both homozygous for the rare allele in which case LR = 0.25+0.5/p+0.25/p2. These maximum values are 30.25, 6.25 and 2.25 for 0.1, 0.25 and 0.5. Note that in general, a large number of simulations is needed to secure that the extreme LR values will occur in the simulations. In this case 10,000 simulations should suffice.
- It can be shown that the mean LR is 21/16 = 1.3125 if H1 is true and 1 if H2 is true, regardless of allele frequencies.

We use Familias to do the simulations. We repeat the following steps three times (one time for each allele frequency distribution): Add one marker with two alleles and their corresponding allele frequencies in the "Edit database" window. The mutation rate is set to 0. Add four individuals (mother, father and two children) in the "Persons" window, and create two pedigrees (full-sibs and unrelated) in the "Pedigrees" window. Perform the simulations via the "Simulate" button. In this solution we set the seed to 1234, use 10,000 simulations and set that the two children would have DNA data.

The following results were obtained from the simulations (LR defined here as Pr(data|full sibs)/Pr(data|unrelated)):

| **Parameter** | **Minor allele frequency=0.1** | | **Minor allele frequency=0.25** | | **Minor allele frequency=0.5** | |
|---|---|---|---|---|---|---|
| True | Full sibs | Unrelated | Full sibs | Unrelated | Full sibs | Unrelated |
| Mean LR | 1.317 | 0.997 | 1.304 | 1.005 | 1.306 | 0.997 |

| | | | | | | |
|---|---|---|---|---|---|---|
| Median LR | 1.114 | 1.114 | 1.3611 | 1.25 | 1.25 | 0.75 |
| Max LR | 30.25 | 30.25 | 6.25 | 6.25 | 2.25 | 2.25 |
| Min LR | 0.25 | 0.25 | 0.25 | 0.25 | 0.25 | 0.25 |
| Pr(LR>1\|full sibs) | 0.8357 | - | 0.7315 | - | 0.5914 | |
| Pr(LR>1\|unrelated) | - | 0.6921 | - | 0.5041 | | 0.3766 |

From the results we can see that using a diallelic marker with minor allele frequency (MAF) 0.1 will give the largest maximum LR, but the median LR (when full siblings is the true relationship) is less than the medians for the other MAFs. MAF = 0.5 generated the largest difference between median LRs for true full sibs compared with true unrelated individuals. We also see that while MAF = 0.1 gives the highest exceedance probability for true full sibs (LR larger than 1), it also generates to highest level of false positives (LR > 1 for true unrelated).

### 5.6.2 Comparison between expected LRs for a microhaplotype marker panel and a SNP marker panel

The purpose of this exercise is to study, by simulations, LR distributions for a full sibling versus unrelated case scenario for which data are available for 40 microhaplotype loci (frequency distribution as in Fig. 5.7 above). Comparisons are made to a marker panel comprising 40 diallelic SNPs (with a 0.4/0.6 frequency distribution per marker). Let $H_1$: Sib-1 and Sib-2 are full siblings and $H_2$: Sib-1 and Sib-2 are unrelated. Frequency data are given in the online supplementary files..

(a) Create a project in Familias with the microhaplotype marker data. Define the pedigrees, set the seed to 1234 and generate data using 10,000 simulations, conditioned both on $H_1$ and $H_2$. Summarize the results as the median LR and the probability that the LR will exceed 1. *Create distribution plots and exceedance plots, i.e., plot estimates of P(LR > x|Hi), i=1, 2, for a range of x-values.

(b) Repeat (a), with the SNP marker data.

(c) Compare the results from (a) and (b).

**Solution:** We used Familias for the simulations and we start by creating all necessary input files. The allele frequency file for the microhaplotype panel is already given, and we create an analogous file for the SNP markers, such as:

```
SNP-1
A       0.4
B       0.6

SNP-2
A       0.4
B       0.6
```

and so on until we have 40 SNPs.

We then perform the simulations as in the previous exercise. We need to do this twice, one time with the microhaplotype panel marker data and once with the 40 SNPs panel marker data. We define four individuals, set up the two hypotheses (full siblings and unrelated) and run the simulations as in 5.7.1. We set the seed to 1234 and use 10,000 simulations.

The following LR distributions were obtained

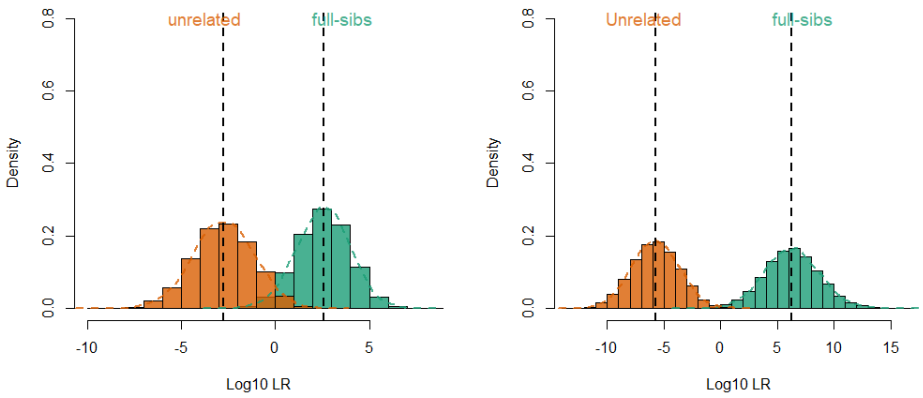



Figure. LR distributions for the 40 SNPs panel (left) and the 40 microhaplotypes panel (right).

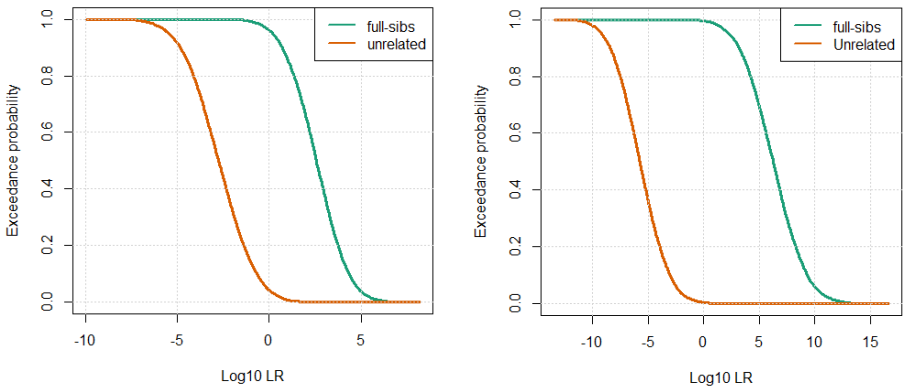



Figure. Exceedance plots for the 40 SNPs panel (left) and the 40 microhaplotypes panel (right).

As expected, the microhaplotype panel is more informative when it comes to solving this full sibling case scenario. The two different marker panels comprise the same number of markers, but the microhaplotype panel generated larger LRs due to the higher number of alleles per marker. The median LR for the 40 SNPs panel was 374 for true full siblings and 0.0018 for true unrelated, and the median LR for the 40 microhaplotypes panel was 1,844,415 for true full siblings and 1.7E-06 for true unrelated.

### 5.6.3 *Relationship inference from a large number of SNP markers

In this exercise, we have SNP data for two individuals and we would like to estimate the degree of relationship between the two individuals. More specifically, we have genotype data for 23,742 SNPs all located on chromosome 1. In this exercise, we will focus on the segment approach, P(IBD ¼ 0) and kinship coefficient. Data are given in the online supplementary files.

(a) What is the estimated length of shared segments for a parent-child duo? Assume the total length of the markers included in the exercise is 249 cM.
(b) What is the expected P(IBD=0) and the kinship coefficient for a first cousin relationship?
(c) Estimate the total length of shared segments. Use a threshold of 7 cM and 100 SNPs to include a segment in the accumulation toward a total length.

(d) Estimate P(IBD=0) and the kinship coefficient.

**Solution:** We have made a script in R which can be found at http://familias.name/BookKEP/PossibleSolutionUsingR.R

(a) The estimated length for a parent/child duo is 249 cM, since a parent and a child will share one allele IBD for all markers.
(b) The expected P(IBD=0) for a first cousin relationship is 0.75, and the expected kinship coefficient is 0.0625.
(c) The core of the segment approach is to find sections of DNA for which the two individuals share at least one allele, and the genetic distance between the starting point and endpoint of such section is measured. In other words, a shared segment will be "broken" if the two individuals are homozygous for different alleles (given that we only have di-allelic SNPs). Another thing to think about is the centromere region. There are normally not any markers within this region, which means that if not accounted for the individuals might get a very large shared segment if alleles are shared for the SNPs surrounding the centromere region. In this case we see in the marker file ("chr1.map") that SNP rs11249395 is the last marker before the centromere, and rs10907360 is the first SNP after the centromere. The length of the centromere is in this case 23.162743 cM, which means that if a segment includes the centromere, 23.162743 cM should be subtracted from its length.

Segments are usually measured in cM, and in this case we assume that 1 Mb = 1cM.

In practice, one would like to exclude very small segments and also segments only comprising a low number of SNPs. This is due to the fact that such segments may not represent a shared historical genealogy, and would most probably be an adventitious match.

If we run our script, based on the assumptions above with a minimum segment threshold of 7 cM and minimum number of SNPs threshold of 100 we get the following segments:

[1] Segments, min_cM = 7 , min_SNP = 100

[,1] [,2]

segment_temp 20.80722 1824

segment_temp 29.99400 2953

segment_temp 17.75963 2120

Since we don't have any reference data for expected shared segment length for different relationships, given the marker data included in this exercise, we cannot really estimate the degree of relationship. BUT we assume that chromosome 1 is around 280 cM minus the centromere, which means around 257 cM. We can then roughly estimate the proportion of shared segments, in total for various relationships. We refer to table 5-4 and get the following estimates:

Table 1. Rough estimates of expected total shared segment length for chr 1.

| **Degree of relationship** | **Total shared segment length (mean in cM)** | **Proportion of total shared segment length** | **Expected total shared segment length for chr 1 (cM)** |
|---|---|---|---|
| Parent/child | 3485 | 1 | 257 |
| Full siblings | 2629 | 0.754 | 193.9 |
| Grandparent/grandchild | 1766 | 0.507 | 130.2 |
| 1st cousins | 874 | 0.251 | 64.5 |
| 2nd cousins | 233 | 0.067 | 17.2 |

If we sum the length of the shared segments in our case, we end up with 68.6 cM, which is very close to the expected total shared segment length for first cousins (64.5 cM)!

(d) In this exercise we will use the definitions of Pr(IBD = 0) and kinship coefficient that we introduced in the book. There might, however, be other slightly different ways to estimate these parameters in practice.

We have implemented our solution in the R script that we referred to above. We get the following results:

[1] Pr=0:
[1] 0.7267708
[1] Rel coeff:
[1] 0.06236505

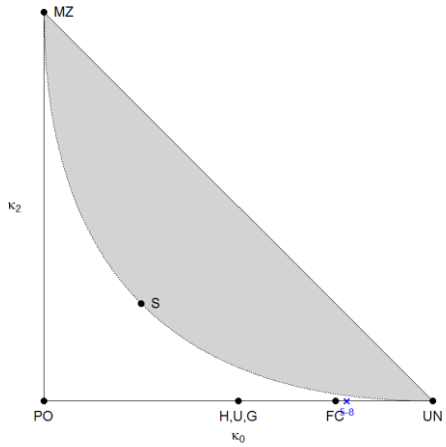
How to interpret these value? We once again refer to Table 5-4:

| **Relationship** | **Kinship coefficient** (Expected average proportion of shared alleles IBD) | **Pr(IBD = 0)** | **Segment length** (Total, in cM) |
|---|---|---|---|
| Parent/child | 0.5 | 0 | 3485 |
| Siblings | 0.25 | 0.250 | 2629 |
| First cousins | 0.0625 | 0.750 | 874 |
| Second cousins | 0.0156 | 0.938 | 233 |
| Third cousins | 0.00390 | 0.984 | 74 |

From these reference values we see that our case is close to the expected values for a first cousin relationship. Pr(IBD=0) was estimated to 0.727 which is very close to the expected 0.75 for a first cousin relationship. The kinship coefficient for a first cousin relationship is expected to be around 0.0625, which also is very close to the 0.0624 obtained in our case!

The R package IBDestimate (explained in Exercise 5.6 below) can also be used to solve this exercise. The input files need to be adjusted, and slightly different results are obtained:

```
con <- url("http://familias.name/BookKEP/ex63TE.RData")

load(con)

close(con) # Finished loading data

library(forrel) # install from cran once

(est = ibdEstimate(ex63TE))

Estimating 'kappa' coefficients
Initial search value: (0.333, 0.333, 0.333)
Pairs of individuals: 1
  5 vs. 8: estimate = (0.779, 0.221, 0), iterations = 13
Total time: 2.51 secs


id1 id2     N    kappa0    kappa1      kappa2

1   5   8 23742 0.779 0.221 0

Shows estimate in IBD triangle:

showInTriangle(est, labels = TRUE)
```




### 5.6.4 Biogeographical ancestry prediction

Assume that we want to predict from which population Mr X originates and that we have DNA data for a single SNP marker, for which this unknown individual is homozygous for allele T. We have three different populations P1, P2 and P3. With the following allele frequencies P1 (T:0.1, C:0.9), P2 (T:0.9, C:0.1) and P3, (T:0.5, C:0.5) . Use a Bayesian approach and calculate the posterior probabilities for the hypotheses that Mr. X belongs to P1, P2 and P3. Assume HWE and a flat prior.

**Solution:** Bayes theorem gives, using R:

```
pP1 = 0.1^2; pP2 = 0.9^2; pP3 = 0.5^2; tot = pP1 + pP2 +pP3;

pP1/tot # Probability of belonging to P1

pP2/tot # Probability of belonging to P2

pP3/tot # Probability of belonging to P3
```

This gives that the probabilities are respectively 0.009, 0.757, 0.234 and we see that Mr X as expected most likely comes from the population for which T is most frequent, i.e., P2. The posterior probability that X comes from this population is 0.757.

### 5.6.5 Finding LR distributions for marker panels with length based allele frequencies and marker panels with DNA sequence based allele frequencies

In Example 5.1, we compared the difference in LRs reflecting if sequence variation in STR alleles is accounted for. We considered two single markers. In this exercise, we will repeat this approach by comparing expected LRs for the 27 STR markers included in the ForenSeq marker panel. We have access to length-based allele frequencies and to sequence-based allele frequencies for a Swedish population. We would like to study how informative this panel is for a first cousin versus unrelated case. Allele frequencies are given in the online supplementary files.

(a) For each database, create a project in Familias and define persons and pedigrees.
(b) Perform simulations and estimate the median LR, mean LR, and 1%-99% intervals. Estimate exceedance probabilities (LR > 10, LR > 100, LR > 1000) for the two different allele frequency distributions. Set the seed to 1234 in the simulation window in Familias and perform 10,000 simulations for each allele database and true hypothesis. (In this exercise we ignore possible impact of linkage).
(c) *Plot LR distributions and exceedance probability distributions.

**Solution:** In this exercise we once again use Familias to perform the simulations. We do this in a similar fashion as we have done in exercises 5.7.1 and 5.7.2.

The results are given below. Note, however, that numbers may differ slightly depending on how the pedigrees are defined.

| | TRUE | median | mean | 1 % | 99 % | LR10 | LR100 | LR1000 |
|---|---|---|---|---|---|---|---|---|
| exc seq | cousins | 3.377 | 33 | 0.091 | 498.6 | 26.62 % | 4.38 % | 0.54 % |
| incl seq | cousins | 4.68 | 165.3 | 0.084 | 1920 | 36.41 % | 9.78 % | 1.59 % |
| exc seq | unrelated | 0.3128 | 1.059 | 0.0174 | 10.65 | 1.10 % | 0.40 % | 0.00 % |
| incl seq | unrelated | 0.2397 | 1.037 | 0.011 | 13.42 | 1.51 % | 0.04 % | 0.00 % |

The central part of the distributions is best measured by the median and does not differ much. Obviously, there will be more large LRs when the sequence database is used, but not terribly so according to the above simulations.

### 5.7.6 Estimation of IBD coefficients

In this exercise, we will estimate IBD coefficients using R.

a) Create and plot a first cousin pedigree. *Hint*: See documentation of the function cousinPed in the R library pedtools:

```
install.packages("pedtools")
```

```
library(pedtools)

x = cousinPed(1)

plot(x)
```

b) Find the kappa coefficients between the first cousins. *Hint*: See the documentation of the function kappaIBD in the R library ribd.

```
install.packages("ribd")

library(ribd)

kappaIBD(x, leaves(x))

[1] 0.75 0.25 0.00
```

c) Simulate 100 equifrequent SNP markers. *Hint*: See documentation of the function markerSim in R library forrel

```
install.packages("forrel")

library(forrel)

x = markerSim(x, N = 100, alleles = 1:2, seed = 12345, verbose = FALSE)
```

An example how “x” may look like:

```
> x
 id fid mid sex <1> <2> <3> <4> <5>
  1   *   *   1 1/2 1/1 2/2 1/2 1/2
  2   *   *   2 1/2 1/2 2/2 2/2 1/1
  3   1   2   1 2/2 1/1 2/2 1/2 1/1
  4   *   *   2 1/2 2/2 1/2 1/1 1/2
  5   1   2   1 1/1 1/2 2/2 1/2 1/1
  6   *   *   2 1/1 2/2 1/1 1/2 2/2
  7   3   4   1 1/2 1/2 2/2 1/2 1/1
  8   5   6   1 1/1 2/2 1/2 1/2 1/2
Only 5 (out of 100) markers are shown.
```

d) Estimate the IBD coefficients. Pedigree information is not used. *Hint*: See documentation of the appropriate function in R library forrel

```
ibdEstimate (x, ids = leaves(x))

Estimating 'kappa' coefficients
Initial search value: (0.333, 0.333, 0.333)
Pairs of individuals: 1
  7 vs. 8: estimate = (0.539, 0.461, 0), iterations = 8
Total time: 0.017 secs
  id1 id2   N      k0      k1 k2
1   7   8 100 0.53852 0.46148  0
```

This compares well with the theoretical values 0.75, 0.25 and 0. However, we can't expect to get these theoretical values, even with an extremely large number of markers. The kappa values (0.75, 0.25, 0) are the theoretical or average values for first cousins. Some first cousins may be closer to unrelated, some closer to half-sibs.

e) Try different seeds and reestimate.

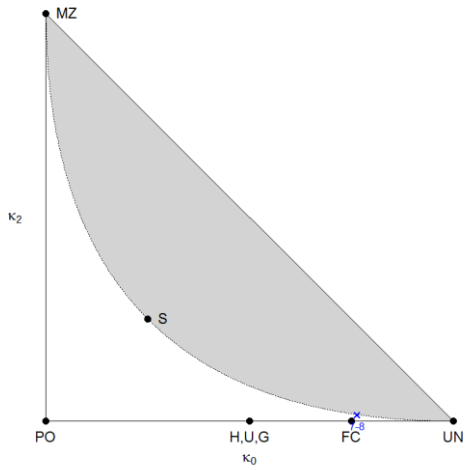
```
x = markerSim(x, N = 100, alleles = 1:2,seed = 17, verbose = FALSE)

ibdEstimate (x, ids = leaves(x))

Estimating 'kappa' coefficients
Initial search value: (0.333, 0.333, 0.333)
Pairs of individuals: 1
  7 vs. 8: estimate = (0.908, 0.032, 0.06), iterations = 36
Total time: 0.007 secs
  id1 id2   N      k0      k1      k2
1   7   8 100 0.90849 0.03154 0.05997
```

f) Increasing the number of markers (say to N = 10,000) will yield kappa values that fit with the theoretical values 0.75, 0.25, and 0. Verify this.

```
x = markerSim(x, N = 10000, alleles = 1:2, seed=113,verbose = FALSE)

ibdEstimate (x, ids = leaves(x))

Estimating 'kappa' coefficients
Initial search value: (0.333, 0.333, 0.333)
Pairs of individuals: 1
  7 vs. 8: estimate = (0.764, 0.221, 0.015), iterations = 31
Total time: 0.147 secs
  id1 id2     N      k0      k1     k2
1   7   8 10000 0.76364 0.22116 0.0152
```

g) Show the result in the IBD triangle. *Hint*: Use the function showInTriangle.

```
x = markerSim(x, N = 10000, alleles = 1:2, seed=113,verbose = FALSE)

est=ibdEstimate (x, ids = leaves(x))

showInTriangle(est, labels = TRUE)
```


MZ
κ2
S
PO
H,U,G
FC
UN
κ0