

# Forensic genetics conference - Zakopane 2022

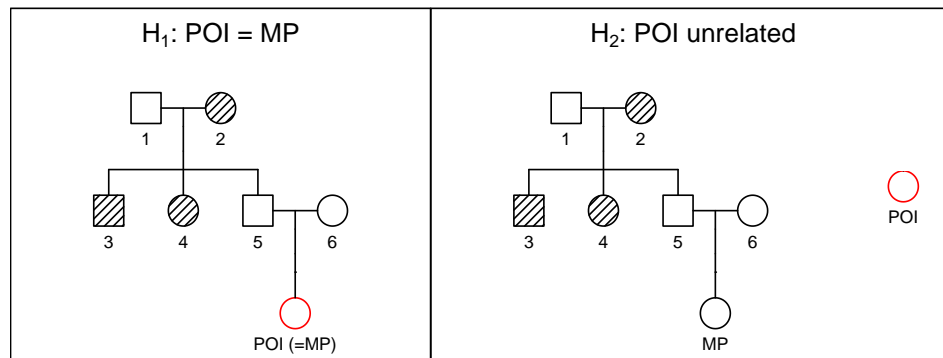
Magnus Dehli Vigeland

## Solutions for exercise set II

### Exercise II-1

a) No answer.

b) `missingPersonPlot(ref, missing = "MP")`



c) `ep = missingPersonEP(ref, missing = "MP")`  
`ep`

```
## Potential mismatches: 8 (D3S1358, D7S820, CSF1P0, PENTA_D, VWA, TPOX, D19S433, D2S1338)
## Expected mismatches: 1.679
## P(at least 1 mismatch): 0.863
```

The overall probability of exclusion is  $EP = 0.86$ , which is normally considered to be sufficient.

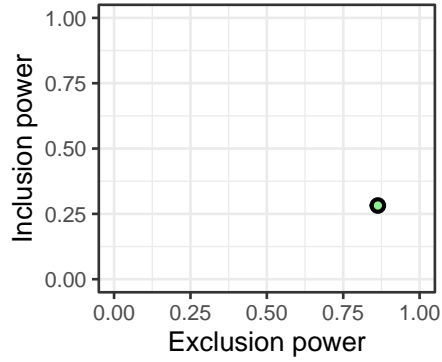
d) `ip = missingPersonIP(ref, missing = "MP", nsim = 1000, threshold = 10000, seed = 17)`  
`ip`

```
## Mean LR: 46998.29
## Mean log10(LR): 3.223
## Estimated power:
## P(LR >= 10000) = 0.282
```

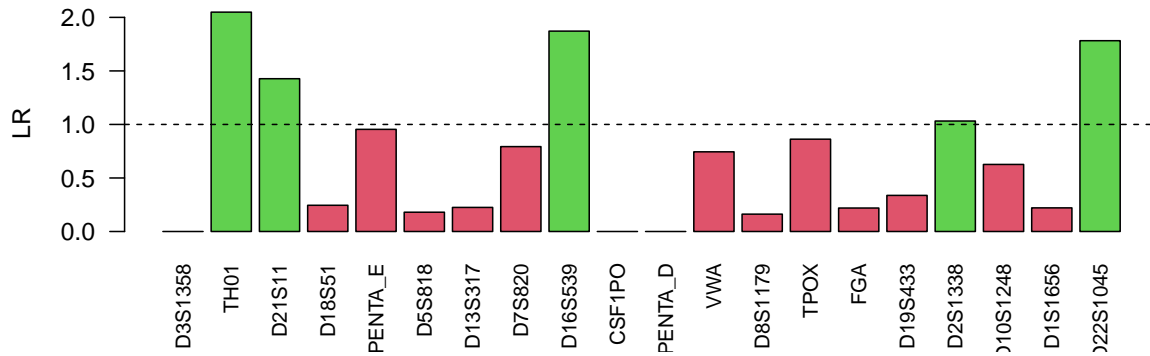
The output shows that only 28% of the simulations give an LR exceeding 10000. The answer depends to some extent on the seed, but with 1000 simulations the numbers will not change much.

If you wonder why the mean LR is much higher than  $10^{\text{mean log}(\text{LR})}$ , this is a simple consequence of the fact that the function  $\log(x)$  is concave. (Google *Jensen's inequality* for more details.)

e) `powerPlot(ep, ip)`

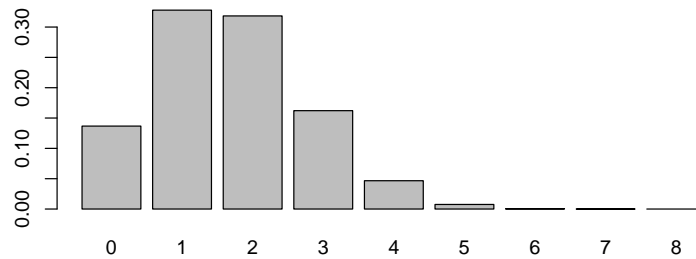


- f) The commands gives  $LR = 0$ , which excludes POI1 as the missing person (adding the usual caveats).  
g) There are 5 markers in favor ( $LR > 1$ , green bars) and 15 against ( $LR < 1$ , red bars).

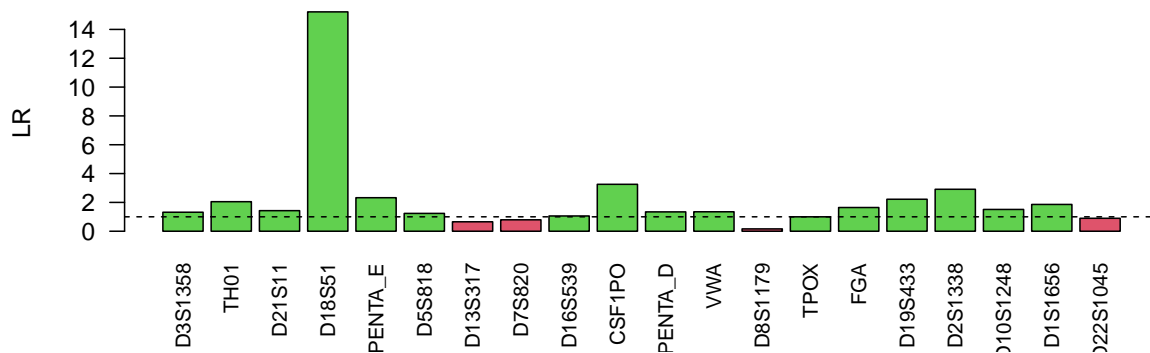


- h) 3 exclusions. This is within the normal range for this reference data, as indicated by this plot:

```
barplot(ep$distribMismatch)
```



- i) The suggested code gives  $LR = 2369.091$  for POI2, with the marker-wise LRs as shown below.

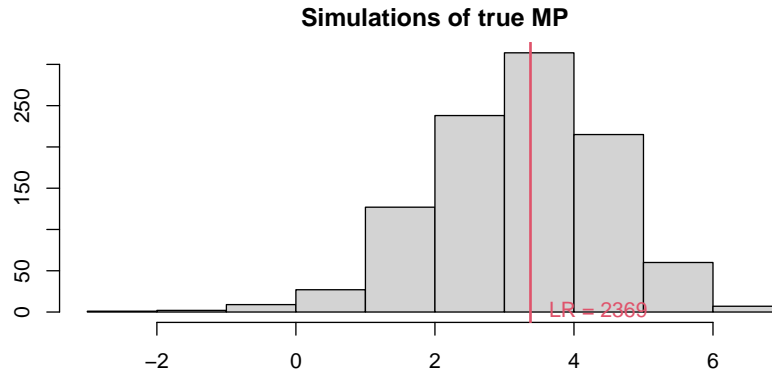


The marker D18S51 gives the largest LR:

```
test2$LRperMarker["D18S51"]
```

```
## D18S51
## 15.2168
```

- j) The LR for POI2 is quite as expected for the true POI, as indicated by the distribution plot:



We conclude that POI2 is quite likely to be the missing person, but that some further evidence is needed to declare a positive match.

## Exercise II-2

- b) The plot was produced with the following command:

```
refs = typedMembers(am)
plot(am, marker = 1, hatched = refs, labs = c(refs, missing),
     col = list(red = missing, blue = refs))
```

- c) The total *a priori* number of solutions is:

```
ncomb(5,5,3,3)
```

```
## [1] 52564
```

The numbers in the function call are, respectively:

- the number of female victims
- the number of female missing persons
- the number of male victims
- the number of male missing persons

- d) MP6 has inbreeding coefficient  $f = 1/16$ , and her parents are first cousins. Code:

```
inbreeding(am, "MP6")
verbalise(am, parents(am, "MP6"))
```

- e) The pairwise LR matrix is found as follows:

```
prw = pairwiseLR(pm, am, missing)
prw$LRmatrix
```

```
##           MP1           MP2           MP3           MP4           MP5           MP6
## V1 479971259           0 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
## V2           0 67760107189 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
## V3           0           0 6.409841e+14 0.000000e+00 0.000000e+00 0.000000e+00
## V4           0           0 0.000000e+00 1.803600e+12 1.803600e+12 0.000000e+00
```

```
## V5      0      0 0.000000e+00 1.030067e+11 1.030067e+11 0.000000e+00
## V6      0      0 0.000000e+00 0.000000e+00 0.000000e+00 8.817392e+12
## V7      0      0 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
## V8      0      0 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
##      MP7      MP8
## V1      0 0.0000000
## V2      0 0.5512209
## V3      0 0.0000000
## V4      0 0.0000000
## V5      0 0.0000000
## V6      0 0.0000000
## V7 16946051 295.8389523
## V8      0 0.2684890
```

f) The joint solutions are found as follows:

```
res = jointDVI(pm, am, missing)
head(res)
```

```
##      V1 V2 V3 V4 V5 V6 V7 V8      loglik      LR      posterior
## 1 MP1 MP2 MP3 MP4 MP5 MP6 MP7 * -737.0038 1.374125e+90 6.904732e-01
## 2 MP1 MP2 MP3 MP4 MP5 MP6 MP7 MP8 -737.8061 6.159953e+89 3.095266e-01
## 3 MP1 MP2 MP3 MP4 MP5 MP6 * * -752.3418 2.997856e+83 1.506369e-07
## 4 MP1 MP2 MP3 MP4 MP5 MP6 * MP8 -753.3430 1.101543e+83 5.535057e-08
## 5 MP1 MP2 MP3 * MP5 MP6 MP7 MP8 -773.8441 1.375599e+74 6.912137e-17
## 6 MP1 MP2 MP3 * MP5 MP6 MP7 * -774.8047 5.263521e+73 2.644825e-17
```

*Comment:* The top two joint solutions, which are much more likely than the rest, agree that  $V_i = MP_i$  for  $i = 1, \dots, 7$ , while disagreeing whether  $V_8$  is  $MP_8$  or has no match. The uncertainty about  $V_8$  cannot be resolved with the given data. While the pairwise LR's could not separate the sisters  $V_4$  and  $V_5$ , the joint solution uniquely identifies them directly.

### Exercise II-3

a) 24 markers, as reported by `summary()`:

```
x = readPed("data/reconstruct-fathers.ped")
x = setFreqDatabase(x, "data/reconstruct-fathers.freq")
summary(x)
```

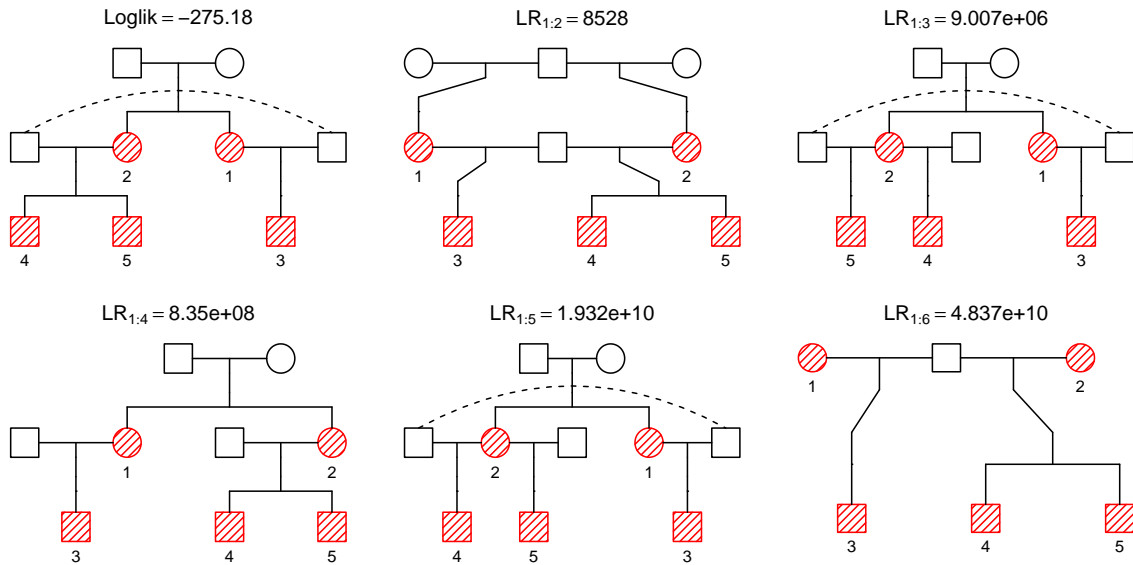
```
## List of 5 singletons.
## Labels: 1 (female), 2 (female), 3 (male), 4 (male), 5 (male).
## 24 attached markers.
```

b) As explained in the documentation `?reconstruct`, the meanings of the optional arguments are:

- `connected = TRUE`: only consider connected pedigrees
- `knownPO`: pairs known to be parent-offspring (in some order)
- `noChildren`: individuals without children of their own
- `linearInb = FALSE`: disallow mating between direct descendants

c) The 6 most likely pedigrees:

```
plot(res, top = 6)
```



- d) All three children have the same father! Indeed, this is supported by both the two first pedigrees above, which are overwhelmingly more likely than the others
- e) Without the restrictions, the program needs to compute the likelihood of 161852 pedigrees. This takes a long time! It is also completely unnecessary, since most of these pedigrees are highly unrealistic for humans.