Kinship statistics and pedigree analysis

Magnus Dehli Vigeland and Thore Egeland

Exercise set IV. Missing person identification and DVI

To get started, open RStudio and load the packages used in these exercises:

```r
library(pedsuite)
library(dvir) # not a core package; must be loaded separately
```

If needed, download the datasets with the following command:

```r
download.file("https://magnusdv.github.io/pedinr/datasets/data.zip", dest = "data.zip")
unzip("data.zip")
```

Exercise IV-1   (MPI case study)

In this exercise we will work through a case study in missing person identification. Genotypes and allele frequencies are given in the files `mpi-example.ped` and `mpi-example.freq`, included in the `data` folder.

a) Load the data in R with the code below, and inspect it using `summary()` and `plotPedList()`.

```r
mpi = readPed("data/mpi-example.ped")
mpi = setFreqDatabase(mpi, "data/mpi-example.freq")
```

b) Use the following code to plot the hypotheses. How many reference individuals have been typed, and what are their relationships to the missing person?

```r
ref = mpi$Reference
missingPersonPlot(ref, missing = "MP")
```

c) We want to test if POI1 is the missing person. Find the LR and give a conclusion after running

```r
poi1 = mpi$POI1
mpiTest1 = missingPersonLR(ref, missing = "MP", poi = poi1)
mpiTest1
```

d) Use the code below to plot the LR for each marker. Which markers have LR = 0?

```r
lr1 = mpiTest1$LRperMarker
cols = ifelse(lr1 > 1, 8, 2)
barplot(lr1, col = cols, ylab = "LR", las = 2, cex.names = 0.8)
abline(h = 1, lty = 2)
```

e) Find the LR for POI2. Which marker gives the largest LR?

f) Explain why marker D18S51 gives a high LR for POI2, but not for POI1.
   Hint: Look at `plotPedList(mpi, marker = "D18S51")` and `afreq(mpi, marker = "D18S51")`.

g) Find the *exclusion power* by running the code below. Interpret the output.

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

h) Find the *inclusion power* by running the code below. Interpret the output.

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

i) Present the two power analyses in a *power plot* and comment on the output.

```
powerPlot(ep, ip)
```

Exercise IV-2 (DVI analysis, grave data)

We will analyse the DVI dataset `grave` included in the **dvir** package. This is a complex case and some commands may take a few minutes to run. Here is an overview of the data:
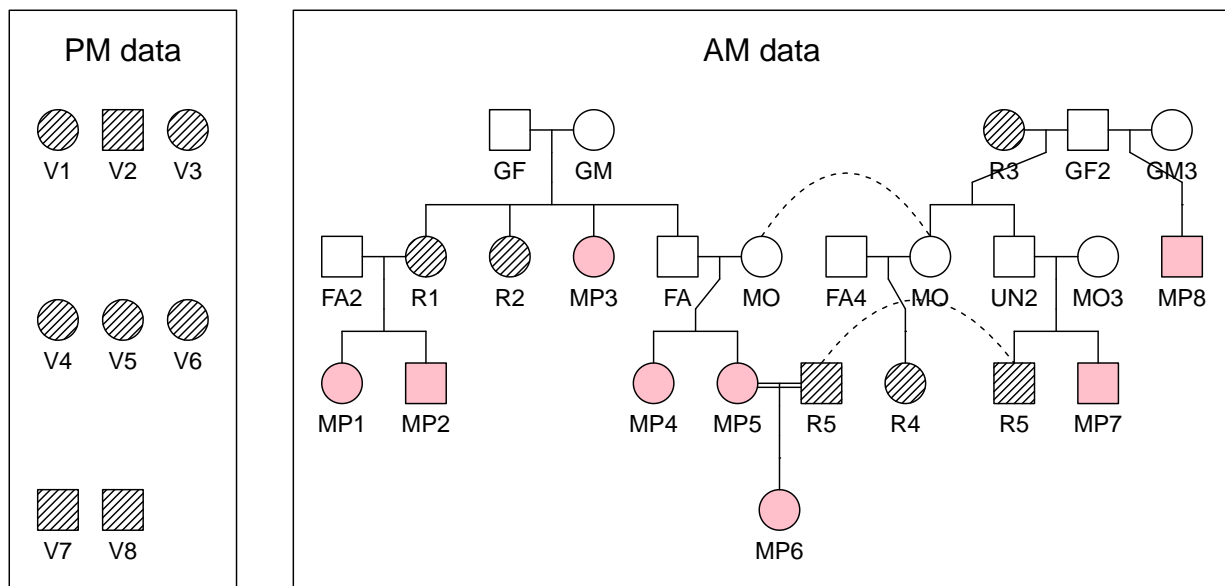
a) Familiarise yourself with the dataset by printing the `grave` object to the console, and also its main components `pm`, `am` and `missing`.

```
grave
grave$pm
grave$am
grave$missing
```

Finally, use the function `plotDVI()` to reproduce the plot shown above.

b) Find the total number of *a priori* possible solutions, using the `ncomb()` function as follows. Explain the input to the function.

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

c) Describe the relationship between the parents of MP6. Find the inbreeding coefficient of MP6.

d) Use `pairwiseLR()` to compute the matrix of pairwise likelihood ratios. Explain what the entries of this matrix are. What is the LR for the pairing V1 = MP1?

```
prw = pairwiseLR(grave)
prw$LRmatrix
```

e) Explain the output from

```
excl = findExcluded(grave)
excl$exclusionMatrix
```

f) Explain the output from

```
und = findUndisputed(grave)
und$summary
```

g) Use `jointDVI()` to find the optimal (joint) solution, and inspect the top five alternatives. Comment on your findings.

h) Use the function `plotSolution()` to visualise the best joint solution.

Exercise IV-3   (DVI analysis continued, grave data)

This exercise continues on the previous. We will now use the function `dviSolve()`. This wraps several other functions into a complete pipeline for solving a DVI case.

a) Run the comment below, and comment whether there are any

- problems with the data,
- nonidentifiable missing persons,
- exclusions.

```
gr = dviSolve(grave)
```

b) Study the output `gr$AM` and explain why a joint analysis is needed to find MP4 = V4 and MP5 = V5.

c) Assume we cannot trust the information on sex. Re-run the analysis as follows:

```
gr = dviSolve(grave, ignoreSex = T)
```

What conclusion can now be drawn about the missing persons `MP1` and `MP2`?

Diviana

The remaining exercises are based on the Diviana app, which is available online here: https://magnusdv.shinyapps.io/diviana/.

If you want to run Diviana with real (i.e., sensitive) data, we recommend to run Diviana locally from RStudio. To set this up, restart R and execute the following:

```
install.packages("pak")
pak::pak(c("magnusdv/dvir", "magnusdv/diviana"), ask = F)
```

After installing the packages, you may launch Diviana:

```
diviana::launchApp()
```

Exercise IV-4   (Example: `planecrash`)

a) Load the `planecrash` example data and click *Overview* (in the `AM DATA` tab) to view a plot of the dataset. How many victims, missing individuals, and reference families are there?
b) Open the *Analysis* tab and click *Solve*. Study the resulting plot and the `AM` and `PM` output tables. Explain the solution.
c) Study the output log. Was joint analysis used in this case? Why not?
d) Reduce the LR threshold to 1000, and re-analyse the data. Explain the changes.
e) Go to *Relatedness* and check the estimated PM-PM relationships. One pair stands out. Who are they, and what is their most likely relationship?

Exercise IV-5   (Example: `planecrash` - cont.)

Reset all and load the `planecrash` dataset again. In each point below, do the described changes, save the pedigree and re-analyse the dataset. Explain the conclusions for the family in question.

a) Modify family F1 so that the mother (individual 1) is missing instead of the daughter.
b) In family F2, add a full sister to the missing person M2, named "M2-2", who is also missing.
c) Add a new family (F6) to the dataset. This should be a family with two missing sisters, named A and B. The family has no typed reference individuals.