Tutorial: Visualizing high-dimensional data and phylogenetic tree reconstruction

Jelle Zuidema, ILLC, UvA, 27-8-2014

In this tutorial we will look at the techniques for reconstructing phylogenetic trees, used in evolutionary genetics as well as historical linguistics. We will work in R, using the packages “ape” and “phangorn”, so start by starting up R and installing those packages:

```
install.packages("ape")
install.packages("phangorn")
library(ape)
library(phangorn)
```

The first example we will look at is a dataset with genetic data (RNA samples) from many different species. It comes with the phangorn package and is made available by typing:

```
data(Laurasiatherian)
```

(data originally comes from http://www.allanwilsoncentre.ac.nz/ . Have a look there to see if you can find out more than I know).

With the command `str(Laurasiatherian)` you can get a summary of the data. It is quite a lot, so let's first pick subset of species to run our analysis on. With

```
mysubset <- subset(Laurasiatherian, subset=c(25,26,27))
```

you can pick out your subset. Try different numbers than 25, 26, 27, to pick out three species you are interested in: two closely related, and one more distantly related. (To include all, use: `mysubset <- subset(Laurasiatherian, subset=1:47)`).

Phylogenetic analysis starts with comparing (genetic) features of each species or individual and building a distance matrix. You can do that with the command:

```
dm <- dist.ml(mysubset)
```

Have a look at that matrix (by typing dm) and make sure you understand what it represents and why some numbers are small and some numbers are large.

You can visualize the information contained in the distance matrix with a technique called multidimensional scaling, which tries to find points in a (by default) 2-dimensional space where the distances between points are as similar as possible to the distances given in the matrix. The R-command is `cmdscale` (for *classical* multidimensional scaling):

```
loc = cmdscale(dm)
x <- loc[, 1]
y <- loc[, 2]
plot(x, y, type="n", xlab="", ylab="", asp=1, axes=FALSE)
text(x, y, rownames(loc), cex=0.6)
```

To reconstruct the likely evolutionary history of the sample (i.e., phylogenetic tree reconstruction) we work with the original distance matrix. We use the matrix in a so-called "agglomerative clustering" procedure: step 1 is to create a separate cluster for each species (a cluster with N=1). Step 2 is to search for the two clusters that are most similar, and we merge them to form a bigger cluster. We then continue with step 2 over and over again until we are left with a single, massive cluster.

Perform these steps with pen and paper for the three species you selected. Now repeat the whole analysis with a distance metric you create with 5 species.

For bigger datasets hierarchical clustering by hand becomes very time consuming. Fortunately, the whole procedure is automatized with the function NJ. By running the following commands you can build a phylogenetic tree for the entire dataset.

```
dm2 <- dist.ml(Laurasiatherian)
tree <- NJ(dm2)
plot(tree)
```

Part 2: Language

Have a look at the website below to get an impression about a dataset for cognates in indoeuropean languages: http://language.cs.auckland.ac.nz/what-we-did/

Go to `ielex.mpi.nl` and download the cognate dataset ("nexus file"). In the file menu of R, change the working directory to the download folder and read in the data into R by typing:
`ielex <- read.nexus.data("IELex_Bouckaert2012.nex")`

Have a look at the summary of this datafile by typing `str(ielex)` and `ielex$Dutch` and `ielex$German`. This massive dataset gives you for each word from the "basic vocabulary" in each language, whether or not there is a cognate in the focal language.

To run the automatic phylogenetic tree analysis, we first need to remove the question marks (missing data) from the datafile:

```
for (i in names(ielex)) {
ielex[[i]][ielex[[i]]=="?"]<-"0" }
```

and convert the data to the format needed by the packages we use:

```
mydata <- phyDat(ielex,type="USER",levels=c("0","1"))
```

Now run multidimensional scaling and the phylogenetic analysis on a subset of the data, and describe the analysis that comes out. Choose an interesting subset.

```
dm <- dist.hamming(subset(mydata,subset=2*(1:51)))
loc = cmdscale(dm)
plot(loc[,1],loc[,2],type="n",xlab="",ylab="",asp=1,axes=FALSE)
text(loc[,1],loc[,2], rownames(loc), cex = 0.6)
```

And then:

```
tree <- NJ(dm)
plot(tree,use.edge.length=FALSE,cex=.7)
```

Finally, phylogenetic tree analysis is quite sensitive to the exact method used for hierarchical clustering and for measuring distances between feature vectors. Try to find alternatives for dist.ml and NJ() in the phangorn package, and report whether the analysis you found earlier is robust to changes.