Numerical classification

Hierarchical agglomerative classification

Theory R functions **Examples** Exercise 🔒

Example 1: Ward cluster algorithm applied on Barro Colorado Island data

In this example, we will use BCI dataset (data from tropical forest permanent plot) to conduct agglomerative cluster analysis, combining Ward's cluster algorithm with Bray-Curtis distance method. We will display resulting dendrogram with four distinguished vegetation clusters, project their spatial configuration and their differences in ordination diagram based on the same distance measure (NMDS using Bray-Curtis distances) and different distance measure (DCA based on chi-square distances).

First, let's load the data and log transform them (original data contain numbers of individuals):

```
library (vegan)
data (BCI)  # example using Baro Colorado data
BCI.log <- log1p (BCI)  # first, log transform species data, which contains
numbers of individuals
```

Function `vegdist` calculates distance matrix based on Bray-Curtis distances:

```
bc.dist <- vegdist (BCI.log, method = 'bray')
bc.dist
```

```
           1         2         3         4         5         6         7
8         9
2  0.2561624
3  0.2955959 0.2749839
4  0.3152203 0.2937687 0.2853559
5  0.3143658 0.3201234 0.3012306 0.2827513
6  0.3098983 0.3210060 0.3413611 0.3439200 0.3427188
7  0.3002090 0.3015008 0.3206233 0.3309095 0.3647759 0.2671652
8  0.3188611 0.2869892 0.2763554 0.2994629 0.3063330 0.3584762 0.3432045
9  0.3511634 0.3116517 0.2886386 0.3294022 0.3157618 0.3580870 0.3265907
0.2696480
10 0.3531552 0.3146364 0.2758954 0.2928692 0.3166512 0.3672627 0.3865676
0.2989141 0.2831257
...
```

Btw, if you type `bc.dist`, you actually use the function `print.dist`[1], which has several useful arguments (`?print.dist`). For example, if we want to print also diagonal values (zeros in this case, since we display distances, not similarites), w may use:

```
print (bc.dist, diag = TRUE)
```

```
          1         2         3         4         5         6         7
8         9
1  0.0000000
2  0.2561624 0.0000000
3  0.2955959 0.2749839 0.0000000
4  0.3152203 0.2937687 0.2853559 0.0000000
5  0.3143658 0.3201234 0.3012306 0.2827513 0.0000000
6  0.3098983 0.3210060 0.3413611 0.3439200 0.3427188 0.0000000
7  0.3002090 0.3015008 0.3206233 0.3309095 0.3647759 0.2671652 0.0000000
8  0.3188611 0.2869892 0.2763554 0.2994629 0.3063330 0.3584762 0.3432045
0.0000000
9  0.3511634 0.3116517 0.2886386 0.3294022 0.3157618 0.3580870 0.3265907
0.2696480 0.0000000
...
```

The Ward's algorithm is available in both the base function `hclust` and also the function `agnes` from the package `cluster`; the implementation, however, slightly differs. The function `hclust` contains two version of Ward's algorithm, and the default one (method = 'ward') is not the same as the one implemented in `agnes` with `method = 'ward'`. The `hclust` contains algorithm `ward.D` (default `ward`) and `ward.D2` (the latter being equivalent to Ward's algorithm in `agnes`). Check `?hclust` to see the difference between both algorithm, and optionally search for more details in Murtagh & Legendre (2014). The true Ward's algorithm is the one implemented in `agnes`, and in `hclust` as `method = 'ward.D2'`.

```
#install.packages ('cluster') # install if necessary
library (cluster)
clust <- agnes (bc.dist, method = 'ward') # calculate Ward's algorithm
plot (clust)  # this plots the results in interactive mode (needs to hit
enter to list between plots)
plot (clust, which = 2) # this plots only the dendrogram; for meaning of
argument "which", see the help file ?plot.agnes (since object "clust" is of
class "agnes")
groups <- cutree (clust, k = 4)  # which plots belong to which cluster?
Result is a vector of the same length as the number of samples in the
dataset.
```

To draw the spatil distribution of the plots classified into four clusters, we need spatial coordinates of each plot, which are in the dataset `BCI.env`:

```
BCI.env <- read.delim
('http://www.davidzeleny.net/anadat-r/lib/exe/fetch.php?media=data:bci.env.t
xt')
# BCI.env contains plot coordinates, UTM.NS (latitude) and UTM.EW
(longitude)
plot (UTM.NS ~ UTM.EW, data = BCI.env, pch = groups, cex = 3)
```

```
  # Plot ordination diagram with the results of classification
  NMDS <- metaMDS (bc.dist)
  ordiplot (NMDS, type = 'n')
  points (NMDS, pch = groups, col = groups)
  legend ('topright', pch = 1:4, col = 1:4, legend = 1:4, bty = 'n')
  # the same, this time with DCA
  DCA <- decorana (BCI.log)
  ordiplot (DCA, type = 'n', display = 'si')
  points (DCA, pch = groups, col = groups)
  legend ('topright', pch = 1:4, col = 1:4, legend = 1:4, bty = 'n')
  # Note that DCA is based on chi-square distances, while the NMDS above is based on Bray-Curtis distances (as is our cluster analysis). It is advisable to use the same distances among samples in both cluster analysis and ordination, in order to be able to see how well cluster analysis differentiates the groups. DCA is suitable for displaying results of TWINSPAN (which is based on DCA).
```

[1] When applied, the script contains only `print` function, which is a generic function choosing the specialized function by asking the class of the object on which it is applied; in case of `bc.dist`, the object is of the class `dist`, so the generic function `print` will choose function `print.dist` to proceed. Check `?print.dist` to see the helpfile for this function.

From:
https://www.davidzeleny.net/anadat-r/ - **Analysis of community ecology data in R**

Permanent link:
https://www.davidzeleny.net/anadat-r/doku.php/en:hier-agglom_examples?rev=1494263110

Last update: **2017/10/11 20:36**