Evaluation of classification results

Theory R functions **Examples**

The following examples are using the same dataset - Vltava data (please check the dataset description to get familiar with it).

Example 1: Project results of cluster analysis onto the diagram of unconstrained ordination

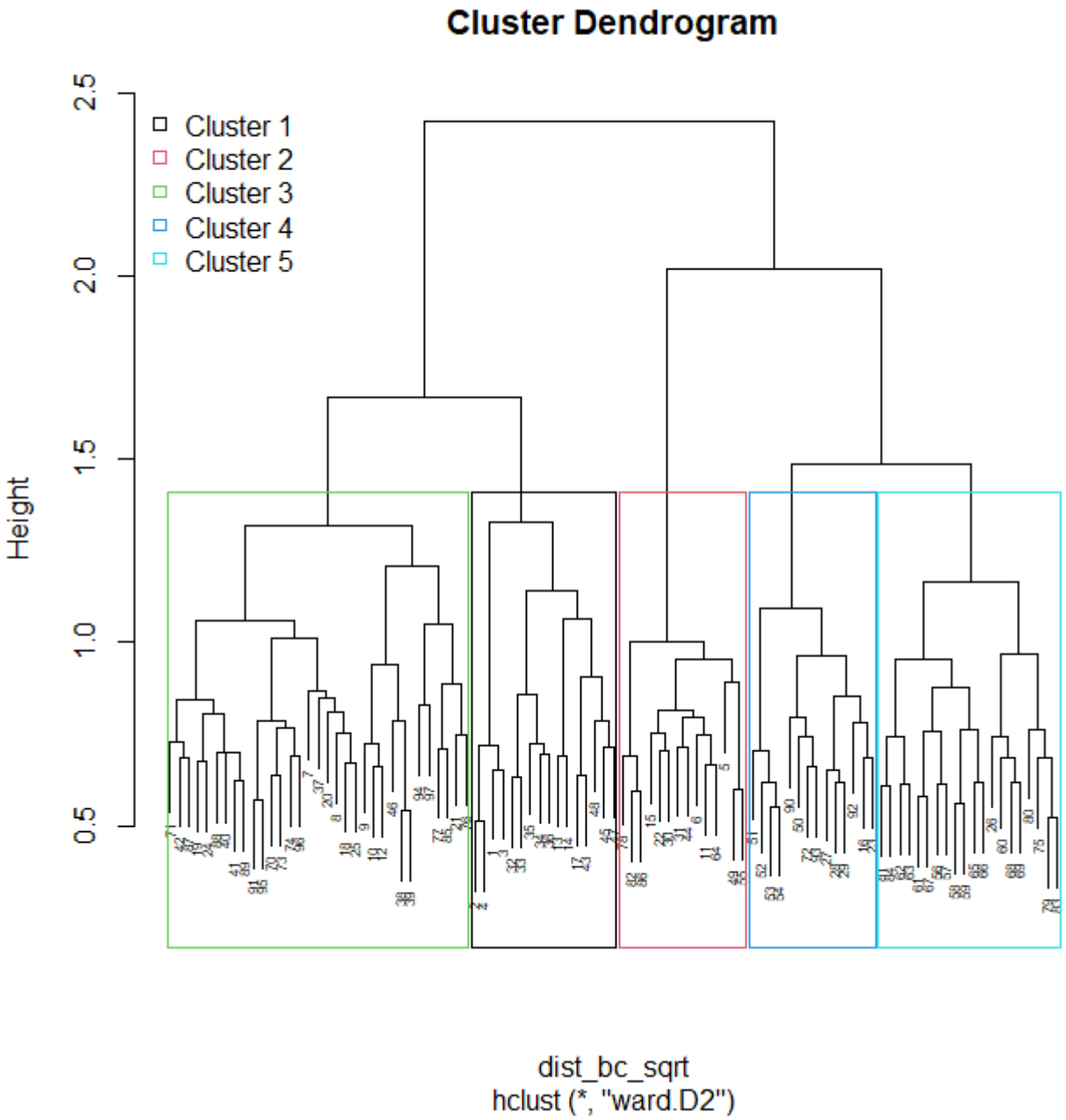
Combining cluster analysis with unconstrained ordination is a powerful tool, which will help us to understand 1) what is the relationship among individual groups/clusters, 2) whether the clusters are well separated from each other (ie the classification was successful in finding well-defined groups), and 3) possibly what is the relationship of individual clusters with the environment.

First, use Vltava data and calculate cluster analysis. Here we opt for Ward's algorithm combined with the Bray-Curtis distance metric. Note several things: 1) since the raw species composition data are on a percentage scale (cover of species estimated in the field using Braun-Blanquet scale, converted into percentage values), we first log transform them to reduce the importance of species with very high cover (using log (x+1) transformation to ensure that zero after log transformation still stays zero). 2) Since Ward's clustering algorithm requires metric distances, and Bray-Curtis distance is not metric, we need to square-root it to make it metric.

```
library (vegan)
vltava.spe <- read.delim
('https://raw.githubusercontent.com/zdealveindy/anadat-r/master/data/vltava-spe.txt
', row.names = 1)
spe_log <- log1p (vltava.spe)  # log-transform raw species composition data
dist_bc <- vegdist (spe_log, method = 'bray') # calculate Bray-Curtis distance
matrix
dist_bc_sqrt <- sqrt (dist_bc) # square-root BC distance matrix to make distances
metric
clus_ward <- hclust (dist_bc_sqrt, method = 'ward.D2')  # calculate Ward's
algorithm (using the correct ''method = 'ward.D2''')
clus_ward_cut <- cutree (clus_ward, k = 5) # "cut the tree" - to which groups
individual samples belong?
```

First, draw the cluster dendrogram and identify individual clusters as rectangles in the dendrogram (function `rect.hclust`). One important thing to remember: the order in which clusters are numbered in `cutree` function is not identical to the order of clusters in the dendrogram (i.e. the first rectangle on the left does not need to be cluster 1). See Example 2 in *Cluster analysis* to see details on how to solve this problem (this also explains why we need to create `clus_in_dendro` object below).

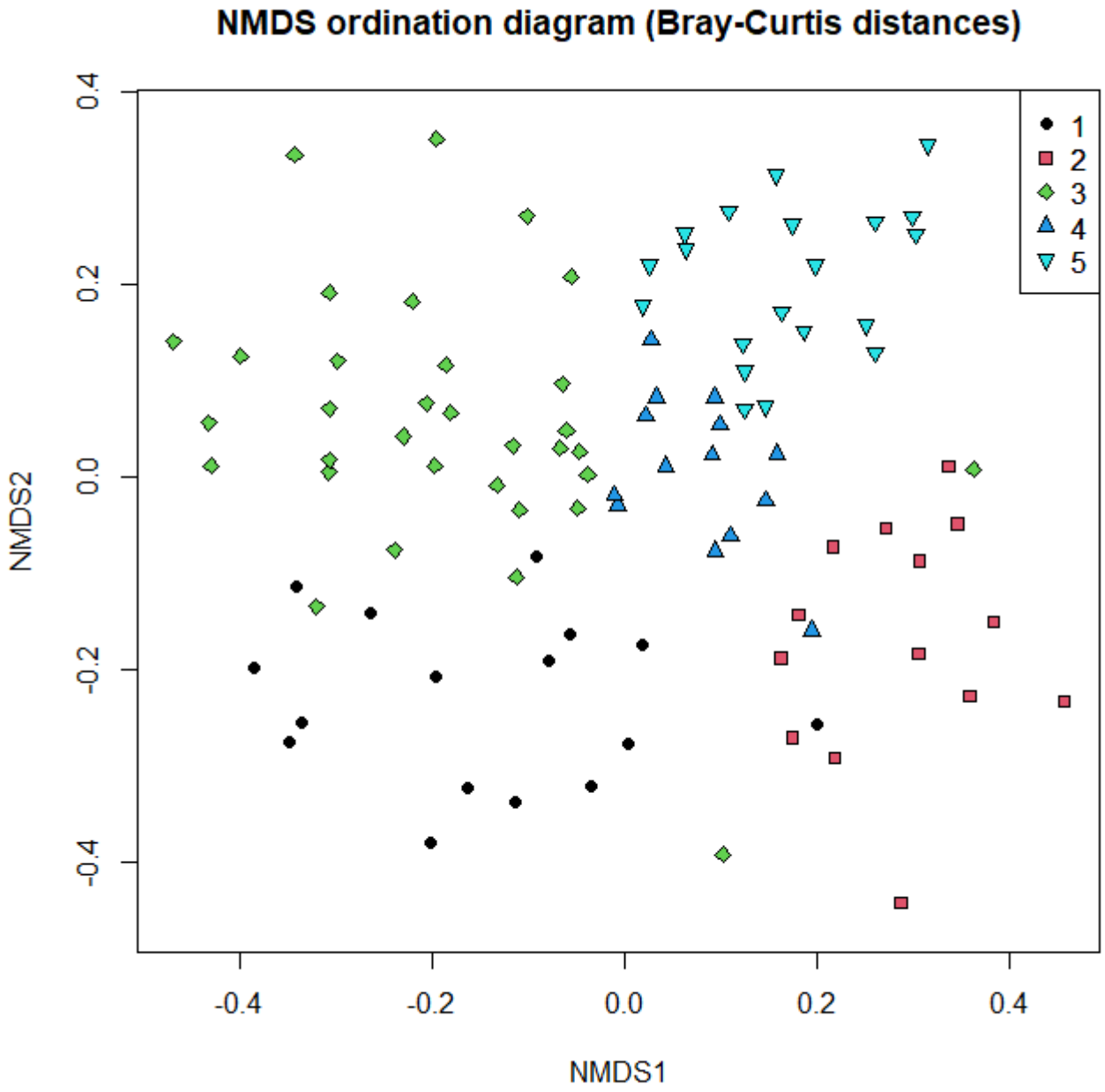
```
plot (clus_ward, cex = .5)  # argument cex reduced the size of the dendrogram leaf
labels to make them readable
clus_in_dendro <- unique (clus_ward_cut[clus_ward$order]) # make sure to know which
box is which cluster!
rect.hclust (clus_ward, k = 5, border = clus_in_dendro)
legend ('topleft', legend = paste ('Cluster', 1:5), col = 1:5, pch = 22, bty = 'n')
```



Now, display the same clusters in the unconstrained ordination diagram. We should follow the rule that the distance measure used in ordination should be the same as the distance measure used in cluster analysis, i.e. Bray-Curtis in our example. The only two unconstrained ordination methods which can implement Bray-Curtis distance are non-metric multidimensional scaling (NMDS) and principal coordinate analysis (PCoA) - we choose the first one here. In the case of NMDS, it does not matter whether we use the original Bray-Curtis distances (object `dist_bc`) or their square rooted version used in Ward's cluster algorithm (`dist_bc_sqrt`), since NMDS, as the first step, converts distances into ranks, used for the calculation, and square-rooting is a monotonic transformation, which preserves the rank of the values:

```
NMDS <- metaMDS (dist_bc)
ordiplot (NMDS, type = 'n', main = 'NMDS ordination diagram (Bray-Curtis
distances)')
points (NMDS, pch = clus_ward_cut+20, bg = clus_ward_cut)
legend ('topright', legend = 1:5, pch = (1:5)+20, pt.bg = 1:5)
```

(adding 20 to the cluster number 1 to 5 allowed using symbols 21 to 25 for displaying the samples in the ordination diagram).



From both the dendrogram and ordination diagram we see that cluster 4 and 5 are rather similar in species composition. In the dendrogram, cluster 4 and 5 merges at a rather low height of 1.5 (note that height here does not represent the Bray-Curtis distance, which is in the range between 0 and 1, but the total error sum of squares, which is the criterion association with Ward's clustering algorithm). In the ordination diagram, their samples are rather close to each other and partially overlapping. In contrast, cluster 2 (the red one) is rather different from other clusters - it branches rather high in the dendrogram, and is quite far from other groups on the ordination diagram.

Also, from the ordination, it seems that the classification results in reasonably distinct clusters, even though there are some samples “ejected” to the other groups, e.g. the lonely sample belonging to cluster 3 (green) in the far right part of the ordination diagram, among samples of cluster 2 (red).

Example 2: Environmental differences between clusters

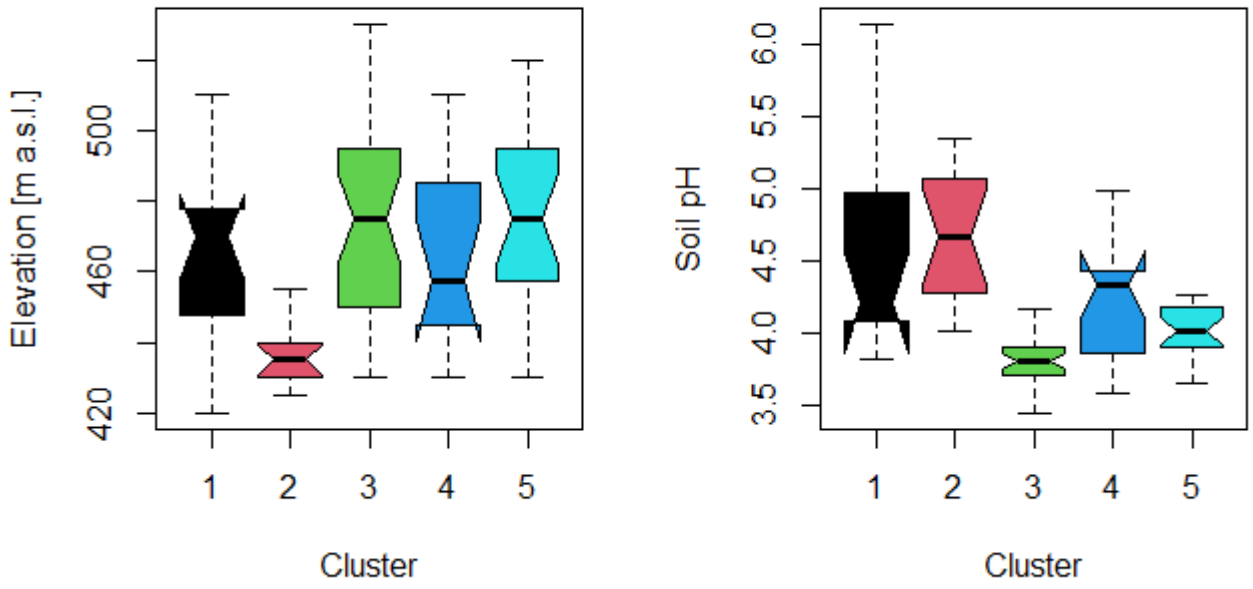
If we have environmental information about individual samples, it would be useful to quantify and describe environmental differences among individual clusters. In the end, we are ecologists, and this is what we are usually interested in.

Let's import environmental data `vltava.env`, and use a subset of four variables to draw differences among clusters.

```
vltava.env <- read.delim
('https://raw.githubusercontent.com/zdealveindy/anadat-r/master/data/vltava-env.txt
')
```

Let's select just two of them (elevation and soil pH) and draw boxplots illustrating the differences:

```
par (mfrow = c(1,2))
boxplot (vltava.env$ELEVATION ~ clus_ward_cut, notch = TRUE, col = 1:5, xlab =
'Cluster', ylab = 'Elevation [m a.s.l.]', outline = FALSE)
boxplot (vltava.env$pH ~ clus_ward_cut, notch = TRUE, col = 1:5, xlab = 'Cluster',
ylab = 'Soil pH', outline = FALSE)
```



Colours of boxplots respect the colours of clusters in Example 1; non-overlapping notches indicate that env. variables among given clusters are likely to be significantly different (check the help file `?boxplot` to see details what `notch = TRUE` means). We also deliberately decided to suppress displaying outliers in the boxplots (`outline = FALSE`).

Let's take elevation and test the differences between clusters. First, we should do the analysis of variance (ANOVA), which will reveal whether the mean of at least one of the groups is significantly different from the overall mean. If ANOVA is significant, we can go a step further and test pair-wise differences between clusters, using e.g. Tukey HSD comparison.

```
clus_ward_cut_fact <- as.factor (clus_ward_cut)
ANOVA_elev <- aov (vltava.env$ELEVATION ~ clus_ward_cut_fact)
summary (ANOVA_elev) # F = 8.832, P < 0.001
```

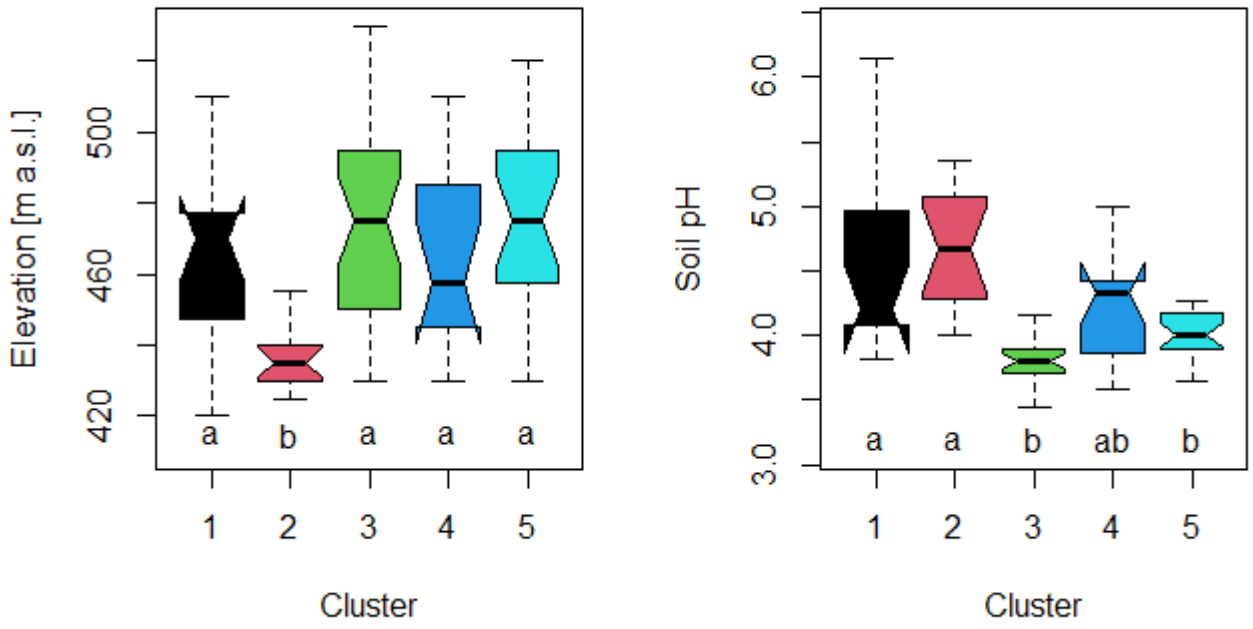
Yes - as you see ANOVA is significant, so we can happily resolve pairwise comparison. R contains the function `Tukey.HSD`, which can do that, but here we use a bit more advanced function `HSD.test` from the package `agricolae` (install it if you haven't):

```
library (agricolae)
hsd_elev <- HSD.test(ANOVA_elev, trt = "clus_ward_cut_fact", group = T, console =
T)
hsd_elev
```

```
$statistics
   MSerror Df     Mean       CV
  548.1152 92 466.1649 5.022227

$parameters
   test             name.t ntr StudentizedRange alpha
  Tukey clus_ward_cut_fact   5         3.935217  0.05

$means
  vltava.env$ELEVATION      std  r Min Max    Q25   Q50    Q75
1             464.0625 24.91444 16 420 510 451.25 470.0 476.25
2             434.1429 11.58153 14 405 455 430.00 435.0 438.75
3             474.4545 25.68449 33 430 530 450.00 475.0 495.00
4             464.6429 24.05637 14 430 510 446.25 457.5 482.50
5             477.6500 23.77366 20 430 520 458.75 475.0 495.00

$comparison
NULL

$groups
  vltava.env$ELEVATION groups
5             477.6500      a
3             474.4545      a
4             464.6429      a
1             464.0625      a
2             434.1429      b
```

Important on the output of the function is the last table, which shows differences among clusters, expressed as letters (clusters with the same letter are not significantly different). We can use these letters and display them onto the boxplot (a small added space below the boxes by extending a bit `ylim` range in `boxplot` function will help to squeeze them there):

```
par (mfrow = c(1,2))
boxplot (vltava.env$ELEVATION ~ clus_ward_cut, notch = TRUE, col = 1:5, xlab =
'Cluster', ylab = 'Elevation [m a.s.l.]', outline = FALSE, ylim = c(410, 530))
group_no_elev <- as.numeric (rownames (hsd_elev$groups))
letter_elev <- hsd_elev$groups$groups
text (x = group_no_elev, y = 415, labels = letter_elev)

ANOVA_pH <- aov (vltava.env$pH ~ clus_ward_cut_fact)
summary (ANOVA_pH) # F = 10.92, P < 0.001
hsd_pH <- HSD.test(ANOVA_pH, trt = "clus_ward_cut_fact", group = T, console = T)

boxplot (vltava.env$pH ~ clus_ward_cut, notch = TRUE, col = 1:5, xlab = 'Cluster',
ylab = 'Soil pH', outline = FALSE, ylim = c(3.1, 6.4))
group_no_pH <- as.numeric (rownames (hsd$groups))
letter_pH <- hsd$groups$groups
text (x = group_no_pH, y = 3.2, letter_pH)
```



We can see that for elevation, plots in cluster 2 are significantly lower than other clusters. When considering soil pH, plots in cluster 3 and 5 are significantly more acid than those in cluster 1 and 2, while cluster 4 stands in between.

Example 3: Fidelity of species to individual clusters

Two common ways how to identify which species are diagnostic to which cluster is the IndVal approach and Φ (phi) coefficient of fidelity. Here we will practice how to calculate Φ, how to test it, how to print results and interpret them. We use the same data as in the exercises above (Vltava data, classified by Ward's clustering algorithm applied on square-rooted Bray-Curtis distances).

We will need library `indicspecies` written by Miquel de Cáceres to calculate fidelity (`install.packages('indicspecies')` if you haven't):

```
library (indicspecies)
```

First, we need to transform the species composition data into presences absence (Φ coefficient is calculated on presence-absence data), and then apply the function `multipatt` (abbreviation for *Multi-level pattern analysis*) to calculate fidelity. We set the argument `fun = 'r.g'`, where `r` indicates correlation coefficient (Φ coefficient is closely related to Pearson's r correlation) and `g` stands for standardizing clusters to the same size ("group size").

```
vltava.spe.pa <- decostand (vltava.spe, method = 'pa')
phi <- multipatt (vltava.spe.pa, cluster = clus_ward_cut, fun = 'r.g')
```

The object `phi` created by function `multipatt` is a list, and the component `$sign` contains the table with species indicating to which cluster they are diagnostic. We can filter out the species which are significant (P<0.01 here) and print them:

```
re <- phi$sign[phi$sign$p.value<=0.01,]
```

```
re
```

```
          s.1 s.2 s.3 s.4 s.5 index      stat p.value
Abiealb23   0   0   0   1   1    15 0.6642737   0.005
Acerpla23   1   1   0   0   0     6 0.4154034   0.005
Alnuglu23   0   1   0   0   0     2 0.8164966   0.005
Fraxexc23   1   1   0   0   0     6 0.3918460   0.005
Pinusyl23   1   0   1   1   0    19 0.4382082   0.005
Prunpad23   0   1   0   0   0     2 0.6740816   0.005
...
```

The table contains species abbreviation, and then an indication to which group (s.1 to s.5) is the species diagnostic. The column `index` indicates which combination of groups the species is diagnostic to (1 means cluster 1, 2 means cluster 2, 6 means cluster 1 & 2, and so on). The column `stat` is for Φ coefficient (in this case), and `p.value` for significance.

It may be useful to actually sort the species in the table (otherwise sorted as they appear in the raw data) according to which cluster they are diagnostic to. We want to first list species diagnostic to cluster 1, then cluster 2, and so on, and the cluster 1 and 2, 2 and 3, and so on. Below is using `dplyr` function `arrange`, which does exactly this. Note that the sorting is a bit complex: we want to make sure that we first display species diagnostic to only a single cluster (that is why we need to first calculate variable `no_diag_clust`, indicating what number of clusters the species is diagnostic to; then we sort species in descending order according to their diagnostic values for clusters s.1 to s.5, and finally, also in descending order, according to their Φ value (column `stat`):

```
library (tidyverse)
no_diag_sp <- rowSums (select (re, s.1, s.2, s.3, s.4, s.5))
re %>% arrange (no_diag_sp, desc (s.1), desc (s.2), desc (s.3), desc (s.4), desc
(s.5), desc (stat))
```

```
          s.1 s.2 s.3 s.4 s.5 index      stat p.value
Vinchir1    1   0   0   0   0     1 0.6804190   0.005
Euphcyp1    1   0   0   0   0     1 0.6666667   0.005
Hylomax1    1   0   0   0   0     1 0.5693680   0.005
Campper1    1   0   0   0   0     1 0.5479992   0.005
Camprot1    1   0   0   0   0     1 0.5244745   0.005
Genitin1    1   0   0   0   0     1 0.5163978   0.005
Bracpin1    1   0   0   0   0     1 0.5092775   0.005
Polyodo1    1   0   0   0   0     1 0.5092466   0.005
Lychvis1    1   0   0   0   0     1 0.4958468   0.005
Silenut1    1   0   0   0   0     1 0.4958468   0.005
Cytinig1    1   0   0   0   0     1 0.4611528   0.005
Achitan1    1   0   0   0   0     1 0.4588315   0.005
Diancar1    1   0   0   0   0     1 0.4588315   0.005
Pimpsax1    1   0   0   0   0     1 0.4588315   0.005
Trifalp1    1   0   0   0   0     1 0.4588315   0.005
Hypeper1    1   0   0   0   0     1 0.4526211   0.010
Poa.nem1    1   0   0   0   0     1 0.4495051   0.005
Digigra1    1   0   0   0   0     1 0.4246831   0.005
Anthram1    1   0   0   0   0     1 0.4215364   0.005
Geniger1    1   0   0   0   0     1 0.4215364   0.010
Festpal1    1   0   0   0   0     1 0.4010513   0.005
Liguvul23   1   0   0   0   0     1 0.3947710   0.010
Allisen1    1   0   0   0   0     1 0.3947710   0.010
Bromben1    1   0   0   0   0     1 0.3947710   0.005
Festovi1    1   0   0   0   0     1 0.3525932   0.010
```

Phalaru1	0	1	0	0	0	2	0.9107143	0.005
Lycoeur1	0	1	0	0	0	2	0.9085257	0.005
Stacsyl1	0	1	0	0	0	2	0.8635755	0.005
Festgig1	0	1	0	0	0	2	0.8214286	0.005
Alnuglu23	0	1	0	0	0	2	0.8164966	0.005
Calysep1	0	1	0	0	0	2	0.8014485	0.005
Myospal1	0	1	0	0	0	2	0.7959350	0.005
Stelnem1	0	1	0	0	0	2	0.7777560	0.005
Sympoff1	0	1	0	0	0	2	0.7068371	0.005
Ranurep1	0	1	0	0	0	2	0.6931919	0.005
Aegopod1	0	1	0	0	0	2	0.6770620	0.005
Prunpad23	0	1	0	0	0	2	0.6740816	0.005
Impagla1	0	1	0	0	0	2	0.6530294	0.005
Valeexc1	0	1	0	0	0	2	0.6123724	0.005
Carebri1	0	1	0	0	0	2	0.6089894	0.005
Lysivul1	0	1	0	0	0	2	0.6000616	0.005
Rumeaqu1	0	1	0	0	0	2	0.5566219	0.005
Salifra23	0	1	0	0	0	2	0.5547002	0.005
Bidefro1	0	1	0	0	0	2	0.5547002	0.005
Galiapa1	0	1	0	0	0	2	0.5547002	0.005
Mentx.v1	0	1	0	0	0	2	0.5547002	0.005
Cirsole1	0	1	0	0	0	2	0.5532670	0.005
Angesyl1	0	1	0	0	0	2	0.5476190	0.005
Soladul1	0	1	0	0	0	2	0.5173224	0.005
Artevul1	0	1	0	0	0	2	0.4923660	0.005
Chaehir1	0	1	0	0	0	2	0.4923660	0.005
Myosaqu1	0	1	0	0	0	2	0.4923660	0.005
Humulup1	0	1	0	0	0	2	0.4611148	0.005
Lythsal1	0	1	0	0	0	2	0.4572230	0.010
Urtidio1	0	1	0	0	0	2	0.4340518	0.005
Aconvar1	0	1	0	0	0	2	0.4232074	0.005
Cusceur1	0	1	0	0	0	2	0.4232074	0.010
Poa.pal1	0	1	0	0	0	2	0.4232074	0.005
Lamimac1	0	1	0	0	0	2	0.4220196	0.010
Glycmax1	0	1	0	0	0	2	0.3894008	0.010
Euoneur23	0	1	0	0	0	2	0.3474972	0.010
Vaccmyr1	0	0	1	0	0	3	0.6011300	0.005
Avenfle1	0	0	1	0	0	3	0.4593426	0.005
Betupen23	0	0	1	0	0	3	0.4327338	0.010
Callvul1	0	0	1	0	0	3	0.3791005	0.010
Galespc1	0	0	0	1	0	4	0.3804001	0.005
Lunared1	0	0	0	0	1	5	0.7097582	0.005
Milieff1	0	0	0	0	1	5	0.6437549	0.005
Mercper1	0	0	0	0	1	5	0.5897678	0.005
Prenpur1	0	0	0	0	1	5	0.5897678	0.005
Cardimp1	0	0	0	0	1	5	0.5230789	0.005
Seneova1	0	0	0	0	1	5	0.5158033	0.005
Acerpla23	1	1	0	0	0	6	0.4154034	0.005
Fraxexc23	1	1	0	0	0	6	0.3918460	0.005
Fragmos1	1	1	0	0	0	6	0.3616508	0.010
Hiersab1	1	0	1	0	0	7	0.5596310	0.005
Hierlac1	1	0	1	0	0	7	0.4450991	0.005
Querpet23	1	0	1	0	0	7	0.4330667	0.005
Tilicor23	1	0	0	1	0	8	0.5338429	0.005
Clinvul1	1	0	0	1	0	8	0.4476660	0.005
Pulmobs1	1	0	0	1	0	8	0.3709077	0.005

```
Scronod1    0   1   0   1   0    11 0.4225365   0.005
Fagusyl23   0   0   1   0   1    14 0.3873290   0.010
Abiealb23   0   0   0   1   1    15 0.6642737   0.005
Dryofil1    0   0   0   1   1    15 0.4911746   0.005
Mycemur1    0   0   0   1   1    15 0.4170078   0.005
Actaspi1    0   0   0   1   1    15 0.4035034   0.005
Dryocar1    0   0   0   1   1    15 0.3994412   0.010
Moehtri1    0   0   0   1   1    15 0.3829024   0.005
Stelhol1    1   1   0   1   0    17 0.4742312   0.005
Pinusyl23   1   0   1   1   0    19 0.4382082   0.005
Luzuluz1    1   0   1   1   0    19 0.3806619   0.005
Impanol1    0   1   0   1   1    24 0.5359159   0.005
Oxalace1    0   1   0   1   1    24 0.5351096   0.010
Asareur1    0   1   0   1   1    24 0.5098884   0.005
Dryodil1    0   1   0   1   1    24 0.3991077   0.010
Galemon1    0   1   0   1   1    24 0.3883148   0.010
```

This is very similar to the way how vegetation ecologists sort synoptic tables of species when they describe vegetation types distinguished in their study.

Finally, we can also use the function `summary` to display the output in yet another structured way (using the same significance threshold $P < 0.01$ as in the table above):

```
summary (phi, alpha = 0.01)
```

```
Multilevel pattern analysis
 ---------------------------

 Association function: r.g
 Significance level (alpha): 0.01

 Total number of species: 274
 Selected number of species: 97
 Number of species associated to 1 group: 72
 Number of species associated to 2 groups: 17
 Number of species associated to 3 groups: 8
 Number of species associated to 4 groups: 0

 List of species associated to each combination:

 Group 1  #sps.  25
           stat p.value
Vinchir1  0.680   0.005 **
Euphcyp1  0.667   0.005 **
Hylomax1  0.569   0.005 **
Campper1  0.548   0.005 **
Camprot1  0.524   0.005 **
Genitin1  0.516   0.005 **
Bracpin1  0.509   0.005 **
Polyodo1  0.509   0.005 **
Lychvis1  0.496   0.005 **
Silenut1  0.496   0.005 **
Cytinig1  0.461   0.005 **
Achitan1  0.459   0.005 **
Diancar1  0.459   0.005 **
Pimpsax1  0.459   0.005 **
```

```
Trifalp1  0.459   0.005 **
Hypeper1  0.453   0.010 **
Poa.nem1  0.450   0.005 **
Digigra1  0.425   0.005 **
Anthram1  0.422   0.005 **
Geniger1  0.422   0.010 **
Festpal1  0.401   0.005 **
Liguvul23 0.395   0.010 **
Allisen1  0.395   0.010 **
Bromben1  0.395   0.005 **
Festovi1  0.353   0.010 **

 Group 2  #sps.  36
           stat p.value
Phalaru1  0.911   0.005 **
Lycoeur1  0.909   0.005 **
Stacsyl1  0.864   0.005 **
Festgig1  0.821   0.005 **
Alnuglu23 0.816   0.005 **
Calysep1  0.801   0.005 **
Myospal1  0.796   0.005 **
Stelnem1  0.778   0.005 **
Sympoff1  0.707   0.005 **
Ranurep1  0.693   0.005 **
Aegopod1  0.677   0.005 **
Prunpad23 0.674   0.005 **
Impagla1  0.653   0.005 **
Valeexc1  0.612   0.005 **
Carebri1  0.609   0.005 **
Lysivul1  0.600   0.005 **
Rumeaqu1  0.557   0.005 **
Salifra23 0.555   0.005 **
Bidefro1  0.555   0.005 **
Galiapa1  0.555   0.005 **
Mentx.v1  0.555   0.005 **
Cirsole1  0.553   0.005 **
Angesyl1  0.548   0.005 **
Soladul1  0.517   0.005 **
Artevul1  0.492   0.005 **
Chaehir1  0.492   0.005 **
Myosaqu1  0.492   0.005 **
Humulup1  0.461   0.005 **
Lythsal1  0.457   0.010 **
Urtidio1  0.434   0.005 **
Aconvar1  0.423   0.005 **
Cusceur1  0.423   0.010 **
Poa.pal1  0.423   0.005 **
Lamimac1  0.422   0.010 **
Glycmax1  0.389   0.010 **
Euoneur23 0.347   0.010 **

 Group 3  #sps.  4
           stat p.value
Vaccmyr1  0.601   0.005 **
Avenfle1  0.459   0.005 **
Betupen23 0.433   0.010 **
```

```
Callvul1  0.379   0.010 **

 Group 4  #sps.  1
         stat p.value
Galespc1 0.38   0.005 **

 Group 5  #sps.  6
          stat p.value
Lunared1 0.710   0.005 **
Milieff1 0.644   0.005 **
Mercper1 0.590   0.005 **
Prenpur1 0.590   0.005 **
Cardimp1 0.523   0.005 **
Seneova1 0.516   0.005 **

 Group 1+2  #sps.  3
           stat p.value
Acerpla23 0.415   0.005 **
Fraxexc23 0.392   0.005 **
Fragmos1  0.362   0.010 **

 Group 1+3  #sps.  3
           stat p.value
Hiersab1  0.560   0.005 **
Hierlac1  0.445   0.005 **
Querpet23 0.433   0.005 **

 Group 1+4  #sps.  3
           stat p.value
Tilicor23 0.534   0.005 **
Clinvul1  0.448   0.005 **
Pulmobs1  0.371   0.005 **

 Group 2+4  #sps.  1
          stat p.value
Scronod1 0.423   0.005 **

 Group 3+5  #sps.  1
           stat p.value
Fagusyl23 0.387    0.01 **
...
...
...
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

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

Permanent link:
https://anadat-r.davidzeleny.net/doku.php/en:class-eval_examples

Last update: **2023/05/04 13:34**