Multivariate regression tree analysis (MRT) for the iAtlantic workshop

Pierre Legendre, Université de Montréal, May 2021

Proposed by marine ecologist Glenn De'ath in 2002, multivariate regression tree analysis (MRT) is an extension of *Classification and regression tree* analysis (CART) to multivariate response data. The method analyses a response data matrix as a function of a matrix of explanatory variables, like the asymmetric methods of canonical analysis (RDA and CCA). The method tries to identify discontinuities in the response data, e.g. community composition, and associate these discontinuities to specific values of the explanatory data, e.g. environmental.

1. The mathematics of MRT analysis

At each division step, as the tree develops, a single variable from **X** is chosen, the one that minimizes the sum of within-group sums-of-squares (SS_{within}). This is equivalent to finding, at each step, the division that maximizes the among-group sum-of-squares (SS_{among}). As in analysis of variance, $SS_{total} = SS_{within} + SS_{among}$, where SS_{total} is the total sum-of-squares of the response data matrix **Y**. SS_{total} is simply the sum of the total sums-of-squares of all response variables in **Y**. SS_{total} is a constant for any row permutation or division of matrix **Y**, so that minimizing SS_{within} is equivalent to maximizing SS_{among}.

Figure 1a shows a simple example, with a multivariate response data set **Y** on the left and a matrix of explanatory variables **X** on the right. There are three explanatory variables in **X**: $\mathbf{x}_1$ and $\mathbf{x}_2$ are quantitative in this example, whereas $\mathbf{x}_3$ is qualitative with three levels or states: A, B and C.

For the first split, the analysis will search for the best partition of **Y** in two groups, constrained by each variable **x** in turn.

• For variable $\mathbf{x}_1$, imagine that the rows of the two data sets, **Y** and **X**, are ordered following the increasing values of $\mathbf{x}_1$, as shown in the figure; the actual programming may differ from the description that follows. The function tries in turn all possible cut-points along variable $\mathbf{x}_1$. For each cut-point between successive (but non-identical) values of $\mathbf{x}_1$, imagine a line drawn across **Y** (dashed line in Figure 1a); it divides **Y** in two groups. $SS_{gr=1}$ is the within-group sums-of squares (also called the squared error) for the top group (gr=1) and $SS_{gr=2}$ is the within-group sums-of squares for the bottom group (gr=2). So the total within-group sum-of-squares (SS_{within}), or total error E^2, for that split of the objects is $E^2 = SS_{gr=1} + SS_{gr=2}$. As shown in the *Numerical ecology* book (Legendre & Legendre 2012, eqs. 8.5 and 8.6), the squared error can be computed either from a raw data file **Y** or from a distance matrix **D** derived from **Y** through an appropriate dissimilarity coefficient.

• The function tries in turn all possible cut-points along $\mathbf{x}_1$, making no cut between identical (tied) values, and it computes $E^2[\mathbf{x}_1]$. It notes the position of the cut where E^2 is minimum for variable $\mathbf{x}_1$ as well as the value of $E^2[\mathbf{x}_1]$ at that point.

• The process is repeated for variable $\mathbf{x}_2$: the rows of the two data matrices are reordered in such a way that the values of $\mathbf{x}_2$ are in increasing order, all possible cut points between non-identical values are tried in turn, and the cut that produces the smallest value of $E^2[\mathbf{x}_2]$ is noted.

• The third variable in Figure 1a is a qualitative variable, or anova factor. All possible combinations of factor levels are tried in turn. In this example, three solutions need to be studied: the group defined by state A versus the other objects, the group defined by state B, and finally the group

defined by state C. The combination that produces the smallest value of $E^2[\mathbf{x}_3]$ is noted. (In the example, the second split separates the rows with level B from those with levels A and C.)

• All values of $E^2[\mathbf{x}_j]$ (there are three variables, hence three values in this illustration) are compared: $E^2[\mathbf{x}_1]$, $E^2[\mathbf{x}_2]$, and $E^2[\mathbf{x}_3]$. The smallest of these values is used to draw the first split of the regression tree in Figure 1b (top split), which is the first partition of data set **Y** in two groups.

• Each branch of the tree is then analysed separately; a *branch* is a group formed by a split. The search for a meaningful split is first done for the left branch of the tree. All explanatory variables in **X** are tried in turn and the variable that produces the split with the smallest value of $E^2[\mathbf{x}_j]$ is used for the next split on the left-hand side of the tree. Similarly, the search is carried out for the objects in the right-hand branch of the tree and the variable of **X** that produces the split with the smallest value of $E^2[\mathbf{x}_j]$ is used for that split. Any variable may be used for several splits. Figure 1b shows a tree produced for a data set **Y** with 3 species; the data in **Y** and **X** shown in Appendix 1.

The process could go on until the tree is fully resolved and individual objects form the terminal *leaves* of the tree. Users, however, are usually not interested in the fully resolved tree, but instead in a tree that presents informative partitions. That shorter tree is found by *pruning* the tree, an operation that consists in removing the smallest branches.

The optimal size of the tree is decided by a resampling analysis called cross-validation. How the cross-validation is used to determine the size of the final tree is described in section 2.

MRT belongs to the family of Euclidean methods because it is based on sums of squared deviations from the means, like anova and *K*-means partitioning. The appropriateness of MRT analysis for the analysis of species data matrices containing many zeros may be highly enhanced by transforming the species abundances with transformations like the chord, Hellinger and log-chord transformations. Data transformation could greatly improve the interpretability and usefulness of the trees as explanatory models of community response data.

2. A full example of MRT analysis: the spider data

De'ath (2002) reanalysed the hunting spider data of Aart & Smeenk-Enserink (1975), using the spider and environmental data transformed and recoded by ter Braak (1986, Table 3); ter Braak had used these data to illustrate canonical correspondence analysis in his seminal paper. The recoded data are available in a data file of package {mvpart} (De'ath, 2011): 28 sites, 12 species and 6 environmental variables (water, sand, moss, light reflection, twigs, and herbs, transformed into classes from 0 to 9).

Following De'ath (2002), the species data were transformed by dividing each abundance value by its column mean, then by the row mean recomputed on the resulting file. The size of the tree was selected after cross-validation: the minimum value of the cross-validation error (*CV Error* = 0.483) was used to decide on the size of the tree (4 groups, Figure 2). The *R*-square of that tree (1 – relative error) was 0.788. The first split separated a group of 8 sites that harboured more twigs (≥ 8) than the other sites; that group had higher abundances of species 2 and 7 than the other sites. The second split isolated a group of 6 sites found on dryer ground (water < 2.5); it had higher abundances of the last two species. The last split separated two groups ($n = 6$ and 8 respectively) according to soil humidity (water < 5.5 versus ≥ 5.5); the group of 6 sites is dominated by species 9, while its sister group, containing 8 sites, is the only one to show substantial abundances of species 1, 4, 5 and 6. An identical partition of the spider data sites into four groups was obtained by applying MRT to the chi-square transformed spider data.

3. MRT as a form of space- or time-constrained clustering

The multivariate regression tree method can be used as a form of spatially-constrained or time-constrained cluster analysis. This idea was first proposed by Borcard et al. (2011, section 4.11.5) and Legendre & Legendre (2012, section 12.6.5), and illustrated by the analysis of the fish community composition along a river. The river course was considered to be a curved spatial transect. The constraining variable may be the site numbers or, in the script in Appendix 3, the variable "dfs" (distance from the source) in data frame "env".

A constrained clustering solution is obtained by analysing a multivariate response matrix **Y** using a single quantitative or rank-ordered variable **x** representing the sampling sequence through space or time. The difference with Figure 1 is that matrix **X** only contains a single explanatory variable **x**, which serves as the constraint in the cluster analysis. **Y** may contain community composition data transformed in an appropriate way. For a weekly time series over a year, for example, the constraining variable **x** may be a vector containing the sampling dates counted from January 1st, or the week order numbers from 1 to 52; the results will be identical since MRT segments **Y** at cutting points along the explanatory, or constraining, variable **x**. The observations do not have to be equispaced along **x**. For spatial transects, the constraining variable may be the ordered sequence of site numbers along the transect or some order variable ordering the sites following their geographical order, as it is the case with variable 'dfs' (distance from the source of the river) in the example developed in the next section.

MRT is a least-squares algorithm. In the present application, it segments matrix **Y** in such a way that the sum of the multivariate within-group sums of squares is minimum, with the constraint that the sampling dates within each group be adjacent along the sampling sequence. MRT can be used to segment spatial series, e.g. transect data as shown in the following ecological application, as well as time series.

4. An example of space-constrained clustering by MRT: the Doubs River fish data

The example describes the space-constrained clustering of 29 sites along the course of Doubs River, in eastern France, computed for the fish community composition data.

The R code in Appendix 3 produces a space-constrained clustering of the Doubs River fish community data by MRT. The fish abundance data were Hellinger-transformed for this example. The regression tree is shown in Figure 3. — Pre-treatment of the fish data by the chord transformation also produced 5 groups (shown in the script in Appendix 3), which only differed in minor details from the 5 groups obtained from Hellinger-transformed data and shown in Figure 3.

A schematic map of the sites along the Doubs River, showing the partition into the 5 spatially-constrained groups, is presented in Figure 4. The "pick" graph showed that the partition into 9 groups had the minimum *CV Error*, but the most parsimonious partition within the confidence interval of the minimum *CV Error* was in 2 groups. Five groups, located mid-way between these two solutions, produced a map that could be interpreted in terms of differences in species composition among the groups.

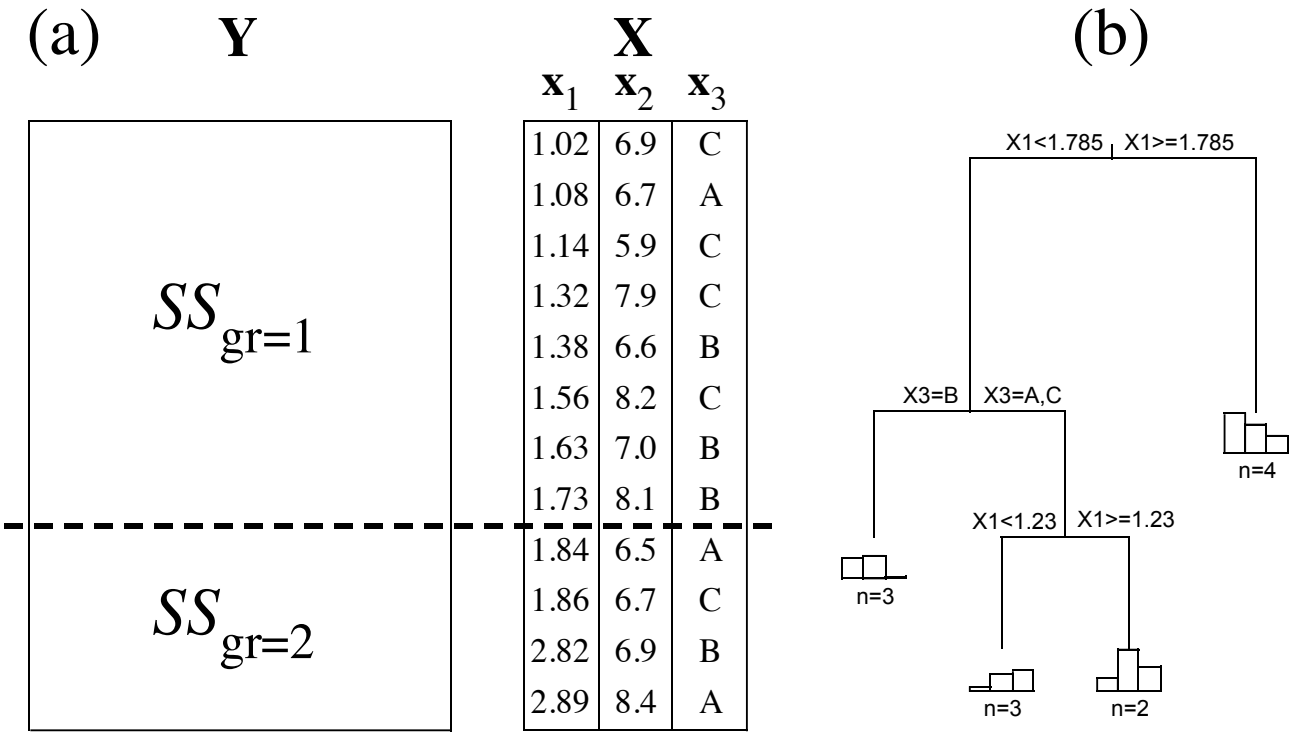

Figure 1 – Schematic description of MRT analysis.

(a) Data: **Y** is the response data set. There are three explanatory variables in **X**; $\mathbf{x}_1$ and $\mathbf{x}_2$ are quantitative in this example, and $\mathbf{x}_3$ is qualitative, with three factor levels (or qualitative states). The dashed horizontal line indicates a cut-point along the values of $\mathbf{x}_1$. The line is extended across **Y**, which is thus divided in two groups. The data used for this analysis are shown in Appendix 1.

(b) Multivariate regression tree computed by function mvpart() of the {mvpart} package; there were three "species" in **Y**; these variables are shown in Appendix 1. Variable $\mathbf{x}_1$ controls the first split (the split occurs at the position of the dashed line in panel a). Variable $\mathbf{x}_3$ controls the second split; the objects with level B are in the left-hand group, those with levels A and C are in the right-hand group. Variable $\mathbf{x}_1$ was used again for the third split. The number of objects in each group is shown underneath each *leaf* (terminal group) of the tree, together with a histogram showing the relative abundances of the three species in that group in matrix **Y**. Figure from Legendre & Legendre (2012, Fig. 8.22), with permission from the authors.

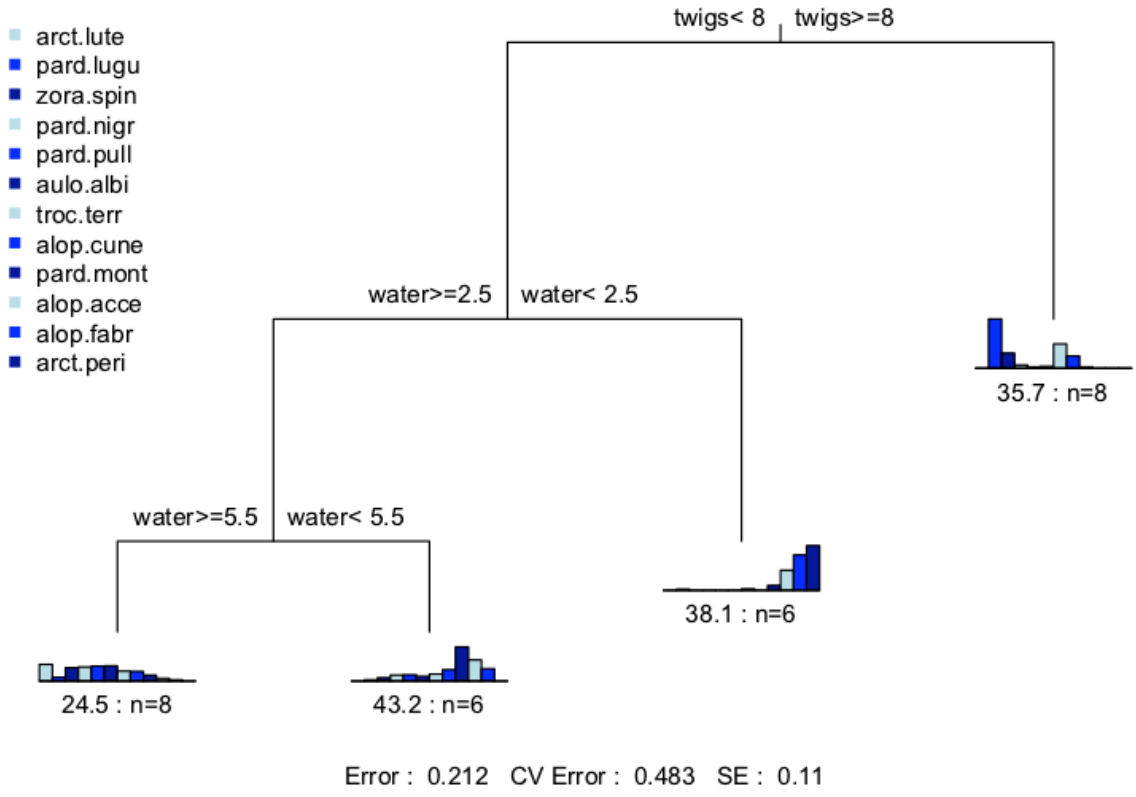

Figure 2 – Multivariate regression tree for the hunting spider data analysed by De'ath (2002). The tree shown has $k = 4$ *leaves* (or clusters). The relative abundances of the 12 species are shown in histograms positioned at the tips of the branches, with the species in the same order as in the **Y** input file; the species names are shown in the upper-left portion of the plot as they appear in the **Y** data file. Under each histogram, *n* is the number of sites in the leaf (group); the value before *n* is the sum of squared errors for the group, i.e. the SS_{gr} statistic.

Statistics shown underneath the plot

• *Error* is the total relative error of the tree with the chosen size (this figure shows the tree with $k = 3$ leaves); the relative error statistics are represented by green dots in the "pick" graph (not shown here). The *R*-square of the tree model is (1 – *Error*).

• *CV Error* is the cross-validation error of the selected tree. *SE* is the standard error of the cross-validation statistic for the selected tree. Run the analysis from the Examples section of the function documentation file and examine the first graph obtained when requesting xv="pick". The minimum value of the cross-validation error (the red dot in that graph) indicates the best-fitting tree model. One may also choose a good-fitting model that is more parsimonious than the one with the smallest *CV Error*, i.e. a tree with fewer splits and leaves. For that purpose, authors are recommending to select a tree whose *CV Error* is within one standard error (*SE*, vertical blue lines) of the smallest *CV Error* value (horizontal red line). The smallest tree meeting this condition is indicated by an orange dot in the "pick" graph.

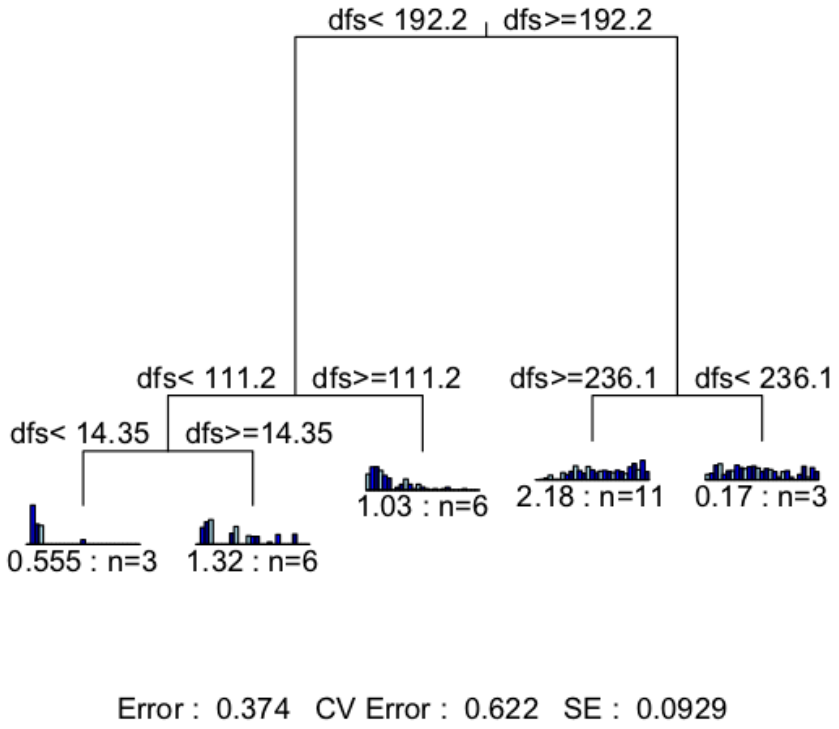

Figure 3 – Space-constrained clustering by MRT of the Doubs River fish community data, 29 sites. The constraining variable is 'dfs', the distance of the sites from the river source.

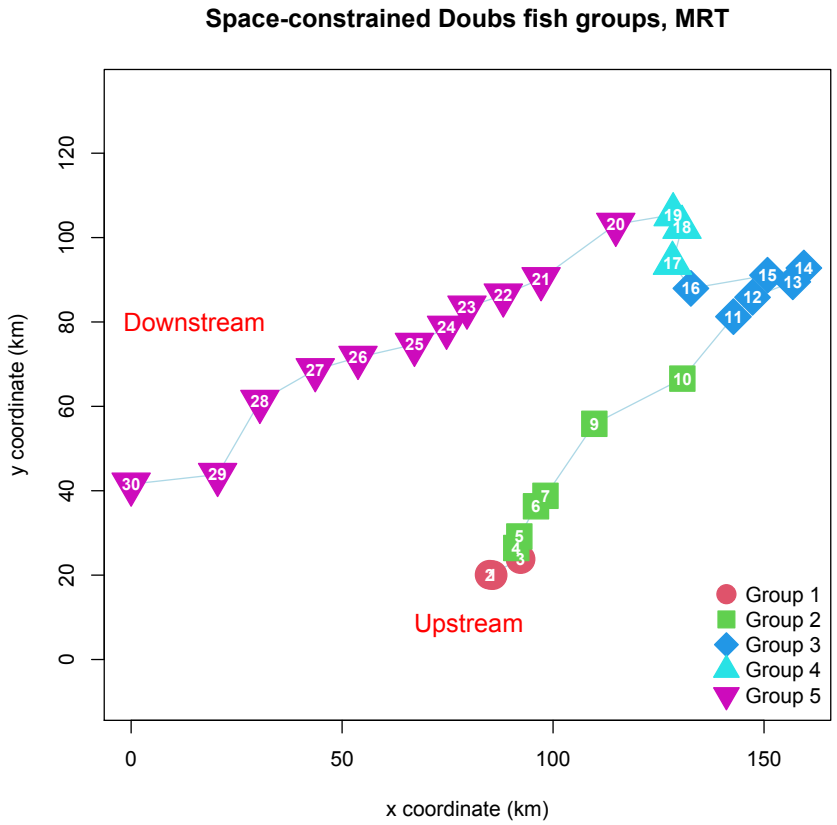

Figure 4 – Schematic map of the sites along Doubs River. The symbols and colours represent the five spatially-constrained groups of sites. Sites 1 and 2 are on top of each other in the plot.

References

Aart, P. J. M. (van der) & N. Smeenk-Enserink. 1975. Correlations between distributions of hunting spiders (Lycosidae, Ctenidae) and environmental characteristics in a dune area. *Netherlands Journal of Zoology* 25: 1-45.

Borcard, D., F. Gillet & P. Legendre. 2011. *Numerical ecology with R.* Use R! series, Springer Science, New York. xi + 306 pp.

Borcard, D., F. Gillet & P. Legendre. 2018. Numerical ecology with R, 2nd edition. Use R! series, Springer International Publishing AG. xv + 435 pp.

De'ath, G. 2002. Multivariate regression trees: a new technique for modeling species-environment relationships. *Ecology* 83: 1105-1117.

De'ath, G. 2012. mvpart: Multivariate partitioning. R package version 1.6-0. http://cran.rproject.org/web/packages/mvpart/.

Legendre, P. & L. Legendre. 2012. *Numerical ecology, 3rd English edition.* Elsevier Science BV, Amsterdam. xvi + 990 pp.

ter Braak, C. J. F. 1986. Canonical correspondence analysis: a new eigenvector technique for multivariate direct gradient analysis. *Ecology* 67: 1167-1179.

Appendix 1

Data files used to compute the multivariate regression tree in Figure 1. In the matrices that follow, the data in **Y** and **X** are ordered by the values of variable $\mathbf{x}_1$, as in Figure 1a.

Matrix **Y**

```
              Y1         Y2         Y3
 [1,] 0.1910978 1.5642915 1.3534104
 [2,] 0.5743539 1.4040023 0.7107540
 [3,] 0.0000000 0.0000000 1.6401854
 [4,] 0.7523856 2.2506030 1.1379542
 [5,] 1.0987933 1.3578862 0.2126343
 [6,] 0.7835680 2.9186479 1.8296707
 [7,] 1.8139399 1.5591685 0.0000000
 [8,] 0.9436658 1.5667416 0.2651598
 [9,] 2.0875316 1.4574247 1.1136134
[10,] 1.7829821 2.0335457 1.7756639
[11,] 3.0812930 0.6086476 1.1863433
[12,] 2.8692957 3.0200150 0.3664296
```

Matrix **X**

```
          X1   X2  X3
Site1   1.02  6.9   C
Site2   1.08  6.7   A
Site3   1.14  5.9   C
Site4   1.32  7.9   C
Site5   1.38  6.6   B
Site6   1.56  8.2   C
Site7   1.63  7.0   B
Site8   1.73  8.1   B
Site9   1.84  6.5   A
Site10  1.86  6.7   C
Site11  2.82  6.9   B
Site12  2.89  8.4   A
```

Analysis that produced the tree in Figure 1 –

```
mvpart(data.matrix(Y) ~ ., data=mat.X, xv="pick", xvmult=100)
```

The tree with size 4 (i.e. with 4 clusters) was selected.

========

Appendix 2

This Appendix contains practical notes on the use of function mvpart.R from package {mvpart} (De'ath 2012).

```
library(mvpart)
?mvpart                    # Documentation file
```

Many of the arguments are the same as in the rpart function of package {rpart}.

Required data files

Y = response data; class: data.matrix

X = explanatory data; class: data.frame

Description of some graphical parameters used by mvpart – See also: ?par

Usage – An example

```
res = mvpart(data.matrix(Y) ~ pH+Lat+Lon, data=X, margin=0.08, xv="pick", xvmult=100)
```

- xv="p" is short for xv="pick", which means: pick a number of groups by clicking
- xvmult: number of cross-validation steps

If you have chosen xv="pick" or xv="p", a first graph is produced –

- green line and dots: relative error for different values of k; this line has no minimum.
- blue line: cross validation error; choose the minimum value (indicated by red dot), or a smaller value within the confidence interval of the minimum (e.g. orange dot), shown by the horizontal red line in the graph.

Click on a point in this graph to indicate your choice of the number of groups (k). The second graph appears; it contains the multivariate regression tree.

Additional functions to help interpret the MRT results

```
summary.rpart(res)
```

Produce a PCA plot of the data showing the groups, number as selected. Each group is surrounded by a convex hull and the group centroids are linked by the tree structure.

```
?rpart.pca
rpart.pca(res, interact=FALSE, wgt.ave=FALSE)
```

If interact=TRUE, the plot can be viewed from different angles by left-clicking around the plot.

Classification tree graph

At the bottom of the tree –

• Error: relative error of the tree. R-square = 1 - Error

• CV error: cross-validation error

• SE: standard error

Label on each split:

• the X variable used for that split and the values in the left and right branches

Underneath each leaf of the tree

• the value is the error sum-of-squares of the group ('leaf' of the tree)

• n = number of objects in the group ('leaf')

• histogram of the abundances of the p species in matrix Y

Output object of function 'mvpart'

Type the following to obtain details on the mvpart output file:

?rpart.object

• object_mvpart$y contains the response matrix

• object_mvpart$call contains the call to the mvpart function

• object_mvpart$cptable contains a tree structure summary. CP = complexity parameter

• object_mvpart$frame presents the tree in data.frame form, one row for each node of the tree. This element shows the number of objects in each cluster ('leaf').

• object_mvpart$where contains a list of the groups to which individual objects belong in the classification.

========

Appendix 3

This Appendix contains R code to compute a spatially-constrained analysis by MRT of the fish community data along the Doubs River, using function mvpart.R of package {mvpart}[1]. The freshwater fish community data from the Doubs River in eastern France are used throughout the Borcard et al. (2011, 2018) books[2].

```
# Load the necessary packages
library(vegan)
library(mvpart)

# Load the following function. It will be used to renumber the clusters sequentially
renumber.cl <- function(gr) {
      aa <- 1
      gr2 <- rep(1,length(gr))
      for (i in 2:length(gr)) {
      if (gr[i]!=gr[i-1]) aa <- aa+1
      gr2[i] <- aa
      }
      gr2
      }

# Read the Doubs.RData file, obtained in the material downloaded from the NEwR Web site.

# Site #8, where no fish had been caught, must be removed from all data files to prevent that site from terminating a spatial segment of the sites
spe <- spe[-8,]                    # Species data
env <- env[-8,]                    # Environmental data
spa <- spa[-8,]                    # Spatial data: geographic coordinates (for plotting maps)

# Transform the species data with the Hellinger and chord transformations
spe.hel <- decostand(spe, "hellinger")      # Hellinger transformation
spe.norm <- decostand(spe, "normalize")     # chord transformation
```

[1] If package mvpart is not already installed on your computer, install it manually following the instructions in Appendix 4.

[2] To obtain the Doubs River fish community data, go to the Web page http://adn.biol.umontreal.ca/~numericalecology/numecolR/ of the book *Numerical ecology with R, 2nd edition* (2018). In the white square, click on the link "Complete material, updated for R 4.0.4" to download the scripts of R code, the two data sets used in the book and some R functions.

(a) MRT of the Hellinger-transformed abundance data <== Used in figs. MRT.3 and MRT.4.
The spatially constraining variable is 'dsf', distance of the sites from the river source.

```
dev.new(title="MRT with sequential constraint, Hellinger")
spe.hel.seq <- mvpart(as.matrix(spe.hel) ~ dfs, data=env, xv="pick",
    margin=0.08, xvmult=100)
```

"pick" graph: click on the desired number of groups, e.g. 5 groups
Examine the sites composing the 5 groups, which form the leaves of the regression tree

```
(gr5.hel <- spe.hel.seq$where)
 [1] 4 4 4 5 5 5 5 5 5 6 6 6 6 6 6 9 9 9 8 8 8 8 8 8 8 8 8 8
```

Renumber the 5 groups with function "renumber.cl()", so that the group numbers in Figure 4 will start with 1 at the head of the river

```
 ( gr5.hel.ren = renumber.cl(gr5.hel) )
 [1] 1 1 1 2 2 2 2 2 2 3 3 3 3 3 3 4 4 4 5 5 5 5 5 5 5 5 5 5
```

End Hellinger

(b) MRT of the chord-transformed abundance data;
Readers can produce the MRT tree, as in the Hellinger case.

```
dev.new(title="MRT with sequential constraint, chord")
spe.ch.seq <- mvpart(as.matrix(spe.norm) ~ dfs, data=env, xv="pick",
    margin=0.08, xvmult=100)
```

"pick" graph: click on the desired number of groups, e.g. 5 groups

```
( gr5.ch <- spe.ch.seq$where )
 [1] 5 5 5 6 6 6 6 6 6 7 7 7 7 8 8 8 9 9 9 9 9 9 9 9 9 9 9 9
```

Renumber the 5 groups with function renumber.cl(

```
( gr5.ch.ren = renumber.cl(gr5.ch) )
 [1] 1 1 1 2 2 2 2 2 2 3 3 3 3 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5
```

End chord

Plot the clusters on a schematic map of the Doubs river using the following script.
The coordinates of the sites are found in file "spa". The map is shown in Figure 4

The groups from Hellinger-transformed data are identified as variable gr2, used in the script

```
gr2 = gr5.hel.ren

dev.new(title="Map of MRT groups on Doubs river")
plot(spa, type="n", main="Space-constrained Doubs fish groups, MRT",
   xlab="x coordinate (km)", ylab="y coordinate (km)", asp=1)
lines(spa, col="light blue")
text(80, 8, "Upstream", cex=1.2, col="red")
text(15, 80, "Downstream", cex=1.2, col="red")
k <- length(levels(factor(gr2)))
for (i in 1:k) {
   points(spa[gr2==i,1], spa[gr2==i,2], pch=i+20, cex=3, col=i+1, bg=i+1)
   }
text(spa, row.names(spa), cex=0.8, col="white", font=2)
legend("bottomright", paste("Group",1:k), pch=(1:k)+20, col=2:(k+1),
   pt.bg=2:(k+1), pt.cex=2, bty="n")
```

Appendix 4

Installing package **mvpart**

If package mvpart is not already installed on your computer, install it manually from
the GitHub repository by typing the following commands in the R console.
Unfortunately, package mvpart is no longer available from CRAN.

On Windows machines, Rtools (4.0 and above) must be installed **first**. In a Web browser, go to:
https://cran.r-project.org/bin/windows/Rtools/

Following that, copy or type the following commands:

```
install.packages("devtools")
library(devtools)
install_github("cran/mvpart", force = TRUE)
```

If the "install_github" command returns an error about the *namespace* file (this may happen
due to your computer platform and System version), copy or type the following commands:

```
assignInNamespace("version_info",
                  c(devtools:::version_info,
                  list("4.0" = list(version_min = "3.3.0",
                  version_max = "99.99.99",
                  path = "bin"))),
                  "devtools")
install_github("cran/mvpart", force = TRUE)
```

========