Numerical Ecology with R, second edition, 2018

This file provides corrections of small mistakes found in the printed edition (Errata), as well as improved explanations and updated, additional information (Addenda and updates).

Last update: 12 June 2024

Errata

Corrected words or other elements are in red.

P. 56, Table 3.1

Bottom right, last entry. One should read:

Binary variables:
Simple matching coefficient
`dist.binary(., method=2)`

Explanation: function **`dist.binary()`** accepts 0 and nonnegative values only (i.e., no standardized variables). All strictly positive values are converted to 1, and the coefficient `method = 2` computes the (square root of the one-complement of the) ratio between matching pairs (i.e. double zeros and double 1s) and the total number of variables.

P. 79, L. 1-2

The second sentence of the paragraph should read: *Colours highlight common clusters, whereas sites **connected by black lines** have different positions in the two trees.*

P. 133, last line of code

The line should read:

```
summary.MRT(spe.ch.mvpart.wrap)
```

Explanation: in this case, the usual **`summary()`** command displays the list of items contained in the result object of an **`MRT()`** run. In contrast, **`summary.MRT()`** has been tailored to display the most interesting results in an accessible format with supplementary results such as discriminant species and indicator values (IndVal).

P.134

The first paragraph should begin as follows:

The **`MRT()`** results are resented in a form close to canonical ordination results […]

The second paragraph should be amended as follows:

The **`summary.MRT()`** display of the result object provides information about…

The two last lines of the second paragraph should be amended as follows:

[…] for the right. The result object itself lists the complete results from which the **`summary.MRT()`** display is drawn.

See also the "Addenda" section below, for p. 134.

P. 173, Caption of Fig. 5.5

The caption should read: "Imputation of missing data in PCA. Procrustes rotation of the original PCA of environmental variables and the one performed on data where ~~three~~ missing values have been imputed. Scaling 1, only the sites are represented. Original sites: red; sites in imputed PCA: blue. Left: 3 missing values, or 1%; right: 32 missing values, or 10%." [delete "three"]

P. 308 bottom and p. 309 top

The description of the Holm correction is incorrect. Here is the correct procedure, as in Legendre & Legendre (2012, p.23) :

(1) Order the p-values from left to right, so that $p_1 \leq p_2 \leq \ldots \leq p_i \ldots \leq p_k$
(2) Compute adjusted $p_i' = (k - i + 1) \times p_i$; adjusted probabilities may be larger than 1
(3) Proceeding from left to right, if an adjusted p-value in the ordered series is smaller than the one occurring at its left, make the smallest equal to the largest one
(4) Replace the adjusted p (i.e., p') in the order of the initial vector
(5) If any p > 1, make it equal to 1
(6) Compare the adjusted p_i' to the unadjusted α significance level and make the statistical decision.

P. 345 line 2

Replace the value 1.0011188 by 1.011188 .

P. 398, R code

Add the red code line below:

```
fish.jac.neigh <- diag(fish.jac[-1, ]) # Jaccard Dj index
absc <- c(2:7, 9:30) # Abscissa
```

Explanation: the line is present in the script but has been inadvertently dropped from the book version.

Addenda and updates

The following entries are proposed to follow changes in recent versions of R and packages, to improve explanations or to add some recent pieces of information.

P. 27, text and code at bottom of page

Replace **`vegtrans()`** by **`abundtrans()`**.

Explanation: in package **`labdsv{}`** the name of function **`vegtrans()`** has been replaced by **`abundtrans()`**. The function is the same.

P. 42, second paragraph, second sentence

Complete the sentence as follows (addendum underscored):

The exercise consists in computing several dissimilarity matrices based on appropriate similarity coefficients: the Jaccard (S_7), Sørensen (S_8) and Ochiai (S_{14}) similarities.

P. 134, above the title of Sect. 4.12.4

Add the following text:

This is how the main results contained in the `spe.ch.mvpart.wrap` object we have just produced are provided. Users can extract the data most relevant to their scientific questions.

- `$pourct` : node × species matrix giving the contribution of each species to the R^2 of the analysis, expressed as a percentage for each node; the sum of each row (i.e., each node) is equal to 100.
- `$R2` : node × species matrix giving the contribution to the R^2 of each species, at each node; the sum of each row is equal to the contribution of each node to the global R^2; the sum of each column is equal to the total contribution of each species to the tree's explanation. The sum of this matrix is equal to the global R^2.
- `$MOYs` : branch × species matrix giving the average abundance of each species in the sites of the branch considered. In our example, there are 6 branches, 4 of which are terminal (leaves), not counting the tree root. Branches numbered 1 and 2 are those of the first node. 3 and 4 are the branches (and leaves) of the left-hand subdivision. 5 and 6 are those of the right-hand subdivision.
- `$RWhere` and `$LWhere` : line numbers of the sites carried by the right (R) and left (L) branches of each node.
- `$TABLE1` : please note that the last row of this table contains the column totals. For the rest:
    - columns 1 to 3: contribution of each species to the R^2 of the analysis for each explanatory variable at the thresholds used in the analysis. The sum of the totals of these three columns is the R2 of the analysis expressed as a percentage;
    - column 4: sum of previous columns; total contribution of each species to the explanation of the tree. The total of this column is the R^2 of the analysis expressed as a percentage; in our example, this value is 62.95%. The R^2 is therefore equal to 0.63, i.e. the 1-complement of the tree's RE, which is equal to 0.37 (Fig. 4.27);
    - column 5: contribution of each species to the total variance of the data, expressed as a percentage.

P. 141 under caption of Fig. 4.34

A new clustering method with spatial or temporal contiguity constraint, based on the general agglomerative clustering algorithm of Lance and Williams (1967), is described in a paper by Guénard and Legendre (2022). The method is implemented in function **`constr.hclust()`** of the **`adespatial`** package in **R**. As graphical output, the function produces a series of maps

corresponding to the different clustering levels; it can also produce a dendrogram. The method is summarized in a teaching document by Legendre (2021).

New references

Guénard, G. et P. Legendre. Hierarchical clustering with contiguity constraint in R. Journal of Statistical Software 103(7): 1-26 (2022)

Legendre, P. Space-constrained hierarchical clustering for the iAtlantic workshop. Teaching document, iAtlantic Ocean Time Series Workshop / Université de Montréal. 9 pp. Available on http://numericalecology.com/documents_enseignement/Space-constrained_hierarchical_clustering.pdf (2021)

P. 146, text

4.15.2 Noise clustering using the `vegclust()` function

Original paragraph in the published book:

A recent package called **`vegclust`,** developed by Miquel De Cáceres, provides a large [……] in the "Noise" cluster.

Replace by the following, more detailed paragraph:

A recent package called **`vegclust`**, developed by Miquel De Cáceres, provides a large range of options to perform non-hierarchical or hierarchical fuzzy clustering of community data under different models (De Cáceres et al. 2010). An interesting one is called "noise clustering" (Davé and Krishnapuram 1997). This method is an attempt to make fuzzy clustering more robust to outliers. Outliers are defined as follows: once "true cluster" centroids have been defined, capture into a fictitious "Noise" cluster the objects that lie farther than a distance δ from the "true cluster" centroids (De Cáceres et al. 2010). The choice of the value of δ is critical: too small a δ value results in an overly large number of outliers, i.e., a large membership in the "Noise" cluster. Note also that if a cluster has a larger intragroup dispersion than the others, this increases the likelihood that some of its legitimate members be considered as outliers.

P. 169, first block of text (below the **R** code)

Original text:

`envfit()` also proposes permutation tests to assess the significance of the R^2 of each explanatory variable regressed on the two axes of the biplot. But this is not, by far, the best way to test the effect of explanatory variables on a table of response variables. We will explore this topic in Chap. 6.

Replace by the expanded text below:

`envfit()` also proposes permutation tests to assess the significance of the R^2 of each explanatory variable regressed on the two axes of the biplot. The R^2 statistics (noted `r2` in the **`envfit()`** output) are produced for quantitative explanatory variables and for factors. They measure the fit of the data to the explanatory variables. With the default option `choices = c(1, 2)`, only the first two axes of the ordination are considered and the R^2 measures the fit of the data ordinated in two dimensions to each explanatory variable. If the calculation is made to involve all dimensions of a PCA ordination (this can be obtained by changing the values in argument `choices`), the R^2 statistic measures the fit of the full-dimensional data to the explanatory variables. If the ordination was produced by PCoA (Sect. 5.5) or NMDS (Sect. 5.6) of a dissimilarity matrix, the fit is between the response variables and the data transformed by the dissimilarity index used in the ordination. If the ordination is a PCA and the envfit analysis involves all PCA

axes, the R^2 is identical to that produced by **`adonis2()`** (Chap. 6). Note, however that function **`envfit()`** has not been designed to replace this other function, which was designed for multivariate analysis of variance by RDA (Sect. 6.3.2.9); its role is to draw explanatory variables onto simple ordination plots.

P. 234, last text paragraph

Original text:

The three RDAs can be tested as usual, and fractions [a] and [c] can be computed and tested by means of partial RDA. Fraction [b], however, is not an adjusted component of variance and cannot be estimated and tested by regression methods. It has zero degree of freedom. […]

Expanded text:

The three RDAs can be tested as usual, and fractions [a] and [c] can be computed and tested by means of partial RDA. Fraction [b], however, is not an adjusted component of variance and cannot be estimated and tested by regression methods. It has zero degree of freedom. However, an elegant workaround has been devised by Bauman et al. (2018) in the special case of the shared space-environment relationship (i.e., the [b] fraction of two explanatory matrices, one of them modeling spatial structures using methods such as MEM variables [Sect. 7.4]), by means of special permutation procedures based on the spatial layout of the sampling units (torus translations and, in the case of irregular sampling, Moran spectral randomization, Wagner and Dray 2015).

Additional references:

Bauman, D., Vleminckx, J., Hardy, O. J., Drouet, T.: Testing and interpreting the shared space-environment fraction in variation partitioning analyses of ecological data. Oikos **128**, 274-285 (2018)

Wagner, H.H., Dray, S.: Generating spatially constrained null models for irregularly spaced data using Moran spectral randomization methods. Methods in Ecology and Evolution **6**, 1169-1178 (2015)

P. 234, R code

Depending on the version of package {spdep}, due to a reversal in the recording of the two dimensions of the grid (despite the absence of change in the names of the arguments), the line of code involving function **`cell2nb`** must be adapted:

```
if(packageVersion("spdep") < 0.8 {
nb <- cell2nb(4, 10, "queen")
} else {
nb <- cell2nb(nrow = 10, ncol = 4, type = "queen")
# or, equivalent
# nb <- cell2nb(4, 10, "queen", legacy = TRUE)
}
```

If the code above return error messages, use the following:

```
nb <- cell2nb(nrow = 10, ncol = 4, type = "queen")
```