Chapter 9 *Ordination in reduced space*

9.0 Projecting data sets in a few dimensions

Ordination

Ordination (from the Latin *ordinatio*, the action of setting in order) is the arrangement of units in some order. Gower (1984) points out that the term *ordination*, widely used in multivariate statistics, actually comes from ecology where it refers to the representation of objects (sites, stations, relevés, etc.) as points along one or several reference axes. In 1954, vegetation ecologist David Goodall was the first to apply factor analysis in community ecology. Goodall proposed the term "ordination" to designate this type of analysis, a term now widely used in community ecology textbooks and publications. Ordination consists in plotting object-points along an axis representing an ordered relationship, or forming a scatter diagram with two or more axes. The ordered relationships are usually quantitative, but it would suffice for them to be of the type "larger than", "equal to" or "smaller than" (semiquantitative relations) to serve as the basis for ordinations, as it is the case in nMDS (Section 9.4).

In ecology, several descriptors are usually observed for each object under study. In most instances, ecologists are interested in characterizing the main trends of variation of the objects with respect to all descriptors, not only a few of them. Looking at scatter plots of the objects with respect to all possible pairs of descriptors is a tedious approach, which generally does not shed much light on the problem at hand. In contrast, the multivariate approach consists in representing the scatter of objects in a multidimensional diagram, with as many axes as there are descriptors in the study. It is not possible to draw such a diagram on paper with more than two or eventually three dimensions, however, even though it is a perfectly valid mathematical construct. For the purpose of analysis, ecologists therefore project the multidimensional scatter diagram onto bivariate graphs whose axes are known to be of particular interest. The axes of these graphs are chosen to represent a large fraction of the variability of the multidimensional data matrix, in a space with reduced (i.e. lower) dimensionality relative to the original data set. Methods for *ordination in reduced space* also allow

Table 9.1 Domains of application of the ordination methods presented in this chapter.

Method	Distance preserved	Variables
Principal component analysis (PCA)	Euclidean distance	Quantitative data, linear relationships (beware of double-zeros)
Correspondence analysis (CA)	χ^2 distance	Non-negative, dimensionally homogeneous quantitative or binary data; species frequencies or presence/absence data
Principal coordinate analysis (PCoA), metric (multidimensional) scaling, classical scaling	Any distance measure	Quantitative, semiquantitative, qualitative, or mixed
Nonmetric multidimensional scaling (nMDS)	Any distance measure	Quantitative, semiquantitative, qualitative, or mixed

one to derive quantitative information on the quality of the projections, and study the relationships among descriptors as well as among objects.

Ordination in reduced space is often referred to as *factor* (or *inertia*) *analysis* since it is based on the extraction of the eigenvectors or *factors* of the association matrix. Factor analysis *sensu stricto* is mainly used in the social sciences; it aims at representing the covariance structure of the descriptors in terms of a hypothetical causal model. It is not discussed further in this book.

The domains of application of the techniques discussed in the present chapter are summarized in Table 9.1. Section 9.1 is devoted to principal component analysis (PCA), a powerful technique for ordination in reduced space which is, however, limited to quantitative descriptors. Because it preserves Euclidean distances, PCA results are sensitive to the presence of double-zeros. Section 9.2 discusses correspondence analysis (CA), an ordination method useful to analyse species presence/absence or abundance data. Sections 9.3 and 9.4 are devoted to principal coordinate analysis (metric scaling, PCoA) and nonmetric multidimensional scaling (nMDS), respectively. Both methods project, in reduced space, distance matrices among objects computed prior to ordination, based on user-chosen distance measures (Chapter 7); in some of these measures, the descriptors may be of any mathematical type. PCA, CA and PCoA are eigenvector-based methods, not nMDS. The presentation of various forms of canonical analysis, which are also eigenvector-based, is deferred to Chapter 11.

It often happens that the structure of the objects under study is not continuous. In such a case, an ordination in reduced space, or a scatter diagram produced using two important variables, may be sufficient to evidence the group structure of the objects. Ordination methods may thus sometimes be used to delineate clusters of objects (Fig. 8.1, Subsection 8.7.3). Ordinations can also be used as complements to cluster analyses. The reason is that clustering investigates pairwise distances among objects, looking for fine relationships, whereas ordination in reduced space considers the variability of the whole association matrix and thus brings out general gradients. Different methods for superimposing the results of clustering onto ordinations of the same objects are described in Section 10.1.

Reduced space

Ecologists generally use ordination methods to study the relative positions of objects in reduced space. An important aspect to consider is the representativeness of the representation in reduced space, which usually has $d = 2$ or 3 dimensions. To what extent does the reduced space preserve the distance relationships among objects? To answer this, one can compute the distances between all pairs of objects, both in the multidimensional space of the original p descriptors and in the reduced d-dimensional space. The resulting values are plotted in a scatter diagram such as Fig. 9.1. When the projection in reduced space accounts for a high fraction of the variance, the distances between projections of the objects in reduced space are quite similar to the original distances in multidimensional space (case a). When the projection is less efficient, the distances between objects in reduced space are much smaller than in the original space. Two situations may then occur. When the objects are at *proportionally* similar distances in the two spaces (case b), the projection is still useful even if it accounts for a small fraction of the variance. When, however, the relative object positions are not the same in the two spaces (case c), the projection is useless. Ecologists often disregard the interpretation of ordinations when the reduced space does not account for a high fraction of the variance. This is not entirely justified, since a projection in reduced space may be informative even if that space only accounts for a small fraction of the variance (case b).

Shepard diagram

The scatter diagram of Fig. 9.1, which is often referred to as a Shepard diagram (Shepard, 1962; diagrams in Shepard's paper had their axes transposed relative to Fig. 9.1), may be used to estimate the representativeness of ordinations obtained using any reduced-space ordination method. In principal component analysis (Section 9.1), the distances among objects, in both the multidimensional space of original descriptors and the reduced space, are calculated using Euclidean distances (D_1, eq. 7.32). The **F** matrix of principal components (eq. 9.4 below) gives the coordinates of the objects in the reduced space. In principal coordinate analysis (Section 9.3) and nonmetric multidimensional scaling (Section 9.4), Euclidean distances among the objects in reduced space are compared to distances D_{hi} found in matrix **D** used as the basis for computing the ordination. In correspondence analysis (Section 9.2), it is the χ^2 distance (D_{16}, eq. 7.55) among objects that is used on the abscissa of the Shepard diagram. Shepard-like diagrams can also be constructed for cluster analysis (Fig. 8.24).

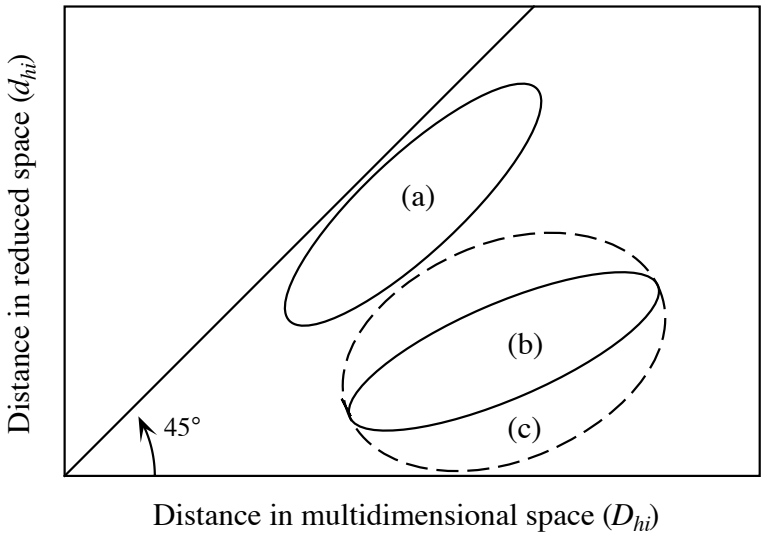


Figure 9.1 Shepard diagram. Three situations encountered when comparing distances among objects, in the *p*-dimensional space of the *p* original descriptors (abscissa) *versus* the *d*-dimensional reduced space (ordinate). The figure only shows the contours of the scatters of points. (a) The projection in reduced space accounts for a high fraction of the variance; the relative positions of objects in the *d*-dimensional reduced space are similar to those in the *p*-dimensional space. (b) The projection accounts for a small fraction of the variance, but the relative positions of the objects are similar in the two spaces. (c) Same as (b), but the relative positions of the objects differ in the two spaces. Adapted from Rohlf (1972). Compare to Fig. 8.24.

The following sections discuss the ordination methods most useful to ecologists. The sections are written to be easily understood by ecologists, so that they may not entirely fulfil the expectations of statisticians. Many programs are available to carry out ordination analysis; several of them are described by Michael Palmer*. R functions are listed in Section 9.5. For detailed discussions on the theory or computing methods, one can refer to ter Braak (1987c) and Legendre & Birks (2012). Important references about correspondence analysis are Benzécri and coll. (1973), Hill (1974), Greenacre (1983), and ter Braak (1987c). Gower (1984, 1987) reviewed the ordination methods described in this chapter, plus a number of other techniques developed by psychometricians. Some of these are progressively finding their way into numerical ecology. They include methods of metric scaling other than principal coordinate analysis, multidimensional unfolding, orthogonal Procrustes analysis (Subsection 11.5.2) and its generalized form, scaling methods for several distance matrices, and a method for ordination of non-symmetric matrices.

* Web page: http://ordination.okstate.edu/.

Ordination vocabulary* Box 9.1

Major axis. Axis in the direction of maximum variance of a scatter of points.

First principal axis (of the concentration ellipsoid in a multinormal distribution; Fig. 4.9). Line passing through the greatest dimension of the ellipsoid; major axis of the ellipsoid.

Principal components. New variates (*variates* = random variables) specified by the axes of a rigid rotation of the original system of coordinates, and corresponding to the successive directions of maximum variance of the scatter of points. The principal components give the positions of the objects in the new system of coordinates.

Principal-component axes (also called *principal axes* or *component axes*). System of axes resulting from the rotation described above.

*Adapted from Morrison (1990, pp. 87 and 323-325).

9.1 Principal component analysis (PCA)

In this book, principal component analysis* is defined as the eigenanalysis of the dispersion matrix $\mathbf{S} = (n-1)^{-1}\mathbf{Y}_c'\mathbf{Y}_c$, where $\mathbf{Y}_c$ is matrix $\mathbf{Y}$ column-centred. In other books, it may be defined as the eigenanalysis of $\mathbf{Y}_c'\mathbf{Y}_c$ without division by $(n-1)$. How to compute the principal axes (Box 9.1) of $\mathbf{S}$ was explained in Section 4.4. In a nutshell, in a multinormal distribution, the first principal axis is the line that goes through the greatest dimension of the concentration ellipsoid describing the distribution. The following principal axes (orthogonal to one another and successively shorter) go through the following greatest dimensions of the p-dimensional ellipsoid. A maximum of p principal axes can be derived from a data matrix containing p variables (Fig. 4.9). The principal axes of a dispersion matrix $\mathbf{S}$ are found by solving

$$(\mathbf{S} - \lambda_k\mathbf{I})\,\mathbf{u}_k = \mathbf{0} \qquad \textbf{(9.1)}$$

(eq. 4.23) whose characteristic equation

$$\left|\mathbf{S} - \lambda_k\mathbf{I}\right| = 0 \qquad \textbf{(9.2)}$$

* Because the pronunciation of "principal" and "principle" is similar in English, the erroneous name "*Principle* component analysis" is sometimes found in the literature, even recent.

Eigenvalue
Eigenvector

is used to compute the *eigenvalues* λ_k. The eigenvectors $\mathbf{u}_k$ associated with the eigenvalues λ_k are found by putting the different values λ_k in turn into eq. 9.1. These eigenvectors are the *principal axes* of dispersion matrix **S** (Section 4.4). The eigenvectors are normalized (i.e. scaled to unit length, Section 2.4) before computing the *principal components,* which give the coordinates of the objects on the successive principal axes. Principal component analysis (PCA) was originally described by Pearson (1901) although it is more often attributed to Hotelling (1933) who proposed it independently. The method and several of its implications for data analysis are presented in the seminal paper of Rao (1964). PCA possesses the following properties, which make it a powerful tool for the analysis of ecological data:

Principal components

1) Since any dispersion matrix **S** is symmetric, its principal axes $\mathbf{u}_k$ are *orthogonal* to one another. In other words, they correspond to *linearly independent directions* in the concentration ellipsoid of the distribution of objects (Section 2.9).

2) The eigenvalues λ_k of a dispersion matrix **S** are all positive or null because **S** is positive semidefinite (Section 4.1, Table 2.2). PCA does not produce negative eigenvalues. The eigenvalues represent the amounts of *variance* of the data along the successive principal axes (Section 4.4).

3) Because of the first two properties, principal component analysis can often *summarize, in a few dimensions, most of the variability* of a dispersion matrix of a large number of descriptors. It also provides a measure of the amount of variance explained by these few independent principal axes.

The present section shows how to compute the relationships among objects and among descriptors, as well as the relationships between the principal axes and the original descriptors. A simple numerical example is developed, involving five objects and two quantitative descriptors:

$$\mathbf{Y} = \begin{bmatrix} 2 & 1 \\ 3 & 4 \\ 5 & 0 \\ 7 & 6 \\ 9 & 2 \end{bmatrix} \quad \text{After centring on the column means, } \mathbf{Y}_c = [y - \bar{y}] = \begin{bmatrix} -3.2 & -1.6 \\ -2.2 & 1.4 \\ -0.2 & -2.6 \\ 1.8 & 3.4 \\ 3.8 & -0.6 \end{bmatrix}$$

where $\mathbf{Y}_c$ is the matrix of column-centred data. In practice, principal component analysis is never used for two descriptors only; in such a case, the objects can simply be represented in a two-dimensional scatter diagram (Fig. 9.2a). A two-dimensional example is used here for simplicity, in order to show that the main result of principal component analysis is to rotate the axes, using the centroid of the objects as pivot.

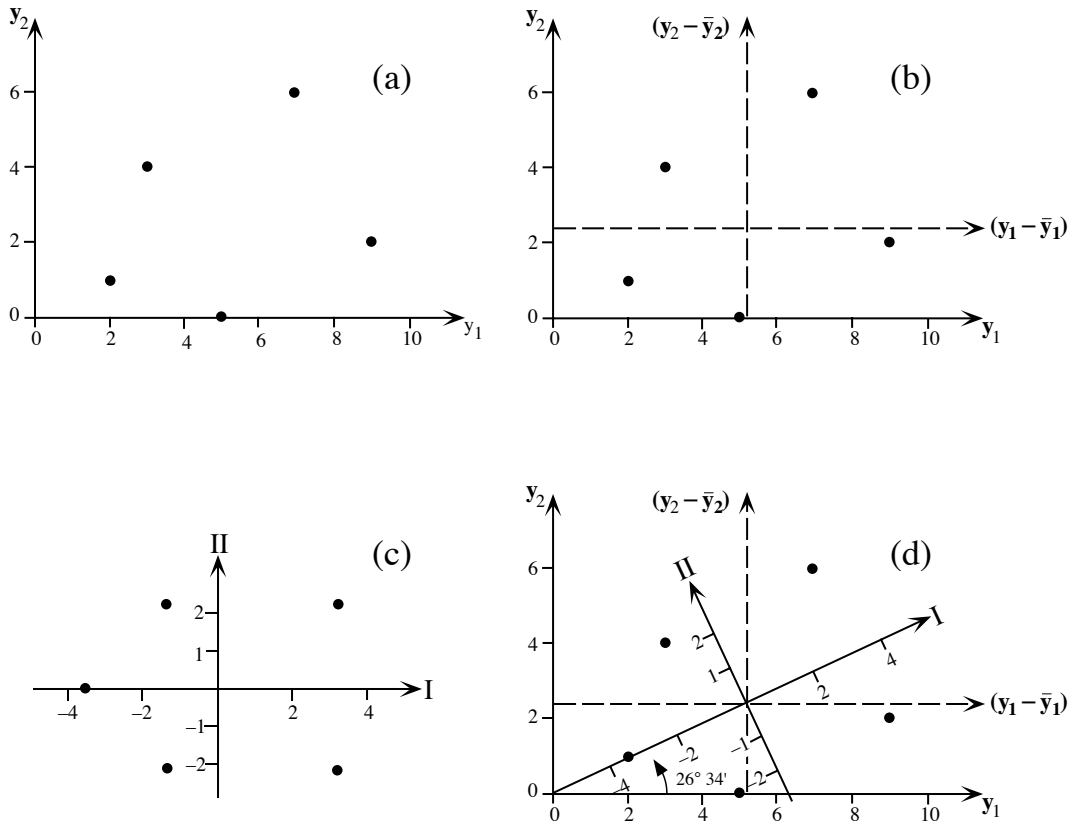


Figure 9.2 Numerical example of principal component analysis. (a) Five objects are plotted with respect to descriptors $\mathbf{y}_1$ and $\mathbf{y}_2$. (b) After centring the data, the objects are now plotted with respect to $(\mathbf{y}_1 - \bar{\mathbf{y}}_1)$ and $(\mathbf{y}_2 - \bar{\mathbf{y}}_2)$, represented by dashed axes. (c) The objects are plotted with reference to principal axes I and II, which are centred with respect to the scatter of points. (d) The two systems of axes (b and c) can be superimposed after a rotation of 26°34'.

1 — Computing the eigenvectors of a dispersion matrix

The dispersion matrix (eq. 4.6) of the descriptors in the above example is:

$$\mathbf{S} = \frac{1}{n-1}\,\mathbf{Y}_c'\,\mathbf{Y}_c = \begin{bmatrix} 8.2 & 1.6 \\ 1.6 & 5.8 \end{bmatrix}$$

The corresponding characteristic equation (eq. 2.23) is:

$$\left|\mathbf{S} - \lambda_k \mathbf{I}\right| = \left| \begin{bmatrix} 8.2 & 1.6 \\ 1.6 & 5.8 \end{bmatrix} - \begin{bmatrix} \lambda_k & 0 \\ 0 & \lambda_k \end{bmatrix} \right| = 0$$

It has two eigenvalues, $\lambda_1 = 9$ and $\lambda_2 = 5$. The total variance (sum of diagonal values) in the matrix of eigenvalues is the same as in **S**, but it is partitioned in a different way: the sum of the variances in **S**, (8.2 + 5.8 = 14), is equal to the sum of the eigenvalues, (9 + 5 = 14). $\lambda_1 = 9$ accounts for 64.3% of the variance and λ_2 makes up for the difference (35.7%). There are as many eigenvalues as there are descriptors. The successive eigenvalues account for progressively smaller fractions of the variance. Introducing, in turn, the λ_k's in matrix equation 9.1:

$$(\mathbf{S} - \lambda_k \mathbf{I})\, \mathbf{u}_k = \mathbf{0}$$

provides the eigenvectors associated with the eigenvalues. Once these vectors have been normalized (i.e. scaled to unit length, $\mathbf{u}'\mathbf{u} = 1$) they become the *columns* of matrix **U**:

$$\mathbf{U} = \begin{bmatrix} 0.8944 & -0.4472 \\ 0.4472 & 0.8944 \end{bmatrix}$$

If a different sign had been arbitrarily assigned to one of the terms of matrix **U** during calculation of the eigenvectors, a mirror image would have been produced for Figs. 9.2c. That image would have been as good at representing the data as Fig. 9.2c.

Ortho-gonality

It is easy to check the orthogonality of the two eigenvectors: their cross-product $\mathbf{u}'_1\mathbf{u}_2 = (0.8944 \times (-0.4472)) + (0.4472 \times 0.8944) = 0$. Moreover, Section 4.4 has shown that the elements of **U** are direction cosines of the angles between the original descriptors and the principal axes. Using this property, one finds that the system of principal axes specifies a rotation of (arc cos 0.8944) = 26°34' of the system of reference defined by the original descriptors. Hence, Figure 9.2 shows that principal component analysis has performed a rotation of the system of axes (descriptors) without changing the positions of the objects with respect to one another.

2 — Computing and representing the principal components

Loading

Principal component

The elements of the eigenvectors are also weights, or *loadings* of the original descriptors, in the linear combination of descriptors from which the principal components are computed. The *principal components* give the positions of the objects with respect to the new system of principal axes. Thus the position of an object $\mathbf{x}_i$ on the first principal axis is given by the following function, or linear combination:

$$f_{i1} = (y_{i1} - \bar{y}_1)\, u_{11} + \ldots + (y_{ip} - \bar{y}_p)\, u_{p1} = [y - \bar{y}]_i \mathbf{u}_1 \qquad \textbf{(9.3)}$$

The values $(y_{ij} - \bar{y}_j)$ are the coordinates of object $\mathbf{x}_i$ on the various centred descriptors j and the values u_{j1} are the loadings of the descriptors on the first eigenvector. The positions of all objects with respect to the system of principal axes is given by matrix **F** of the transformed variables. It is also called the *matrix of principal components*:

Matrix of principal components

$$\mathbf{F} = \mathbf{Y}_c \mathbf{U} \qquad \textbf{(9.4)}$$

where **U** is the matrix of eigenvectors and $\mathbf{Y}_c$ is the matrix of centred observations. The system of principal axes is centred with respect to the scatter of point-objects. This would not be the case if **U** had been multiplied by **Y** instead of the centred matrix $\mathbf{Y}_c$, as in some special forms of principal component analysis (*non-centred PCA*). For the numerical example, the principal components are computed as follows:

$$\mathbf{F} = \begin{bmatrix} -3.2 & -1.6 \\ -2.2 & 1.4 \\ -0.2 & -2.6 \\ 1.8 & 3.4 \\ 3.8 & -0.6 \end{bmatrix} \begin{bmatrix} 0.8944 & -0.4472 \\ 0.4472 & 0.8944 \end{bmatrix} = \begin{bmatrix} -3.578 & 0 \\ -1.342 & 2.236 \\ -1.342 & -2.236 \\ 3.130 & 2.236 \\ 3.130 & -2.236 \end{bmatrix}$$

The variance of the two columns of **F** are $\lambda_1 = 9$ and $\lambda_2 = 5$ respectively. Since the two columns of the matrix of component scores are the coordinates of the five objects with respect to the principal axes, they can be used to plot the objects with respect to principal axes I and II (Fig. 9.2c). It is easy to verify (Fig. 9.2d) that, in this two-descriptor example, the objects are positioned by the principal components in the same way as in the original system of descriptor-axes. Principal component analysis has simply rotated the axes by 26° 34' in such a way that the new axes correspond to the two main components of variability. When there are more than two descriptors, as it is usually the case in ecology, principal component analysis still only performs a rotation of the system of descriptor-axes, but now in multidimensional space. In that case, principal components I and II define the plane allowing the representation of the largest amount of variance. The objects are projected on that plane in such a way as to preserve, as much as possible, the relative Euclidean distances they have in the multidimensional space of the original descriptors.

Euclidean distance

The relative positions of the objects in the rotated p-dimensional space of principal components are the same as in the p-dimensional space of the original descriptors (Fig. 9.2d). This means that *the Euclidean distances among objects* (D_1, eq. 7.32) *have been preserved through the rotation of axes*. This important property of principal component analysis is noted in Table 9.1. The quality of the representation in a reduced Euclidean space with m dimensions only ($m \leq p$) may be assessed using the following ratio:

R^2-like ratio

$$\left(\sum_{k=1}^{m} \lambda_k \right) \Big/ \left(\sum_{k=1}^{p} \lambda_k \right) \qquad \textbf{(9.5)}$$

This ratio is the equivalent of a coefficient of determination (R^2, eq. 10.20) in regression analysis. The denominator of eq. 9.5 is actually equal to the trace of matrix **S** (sum of the diagonal elements). Thus, with the current numerical example, a representation of the objects, along the first principal component only, would account for a proportion 9/(9+5) = 0.643 of the total variance in the data matrix. This value is identical to that given in Subsection 9.1.1 for the fraction of the variance of **Y** that is accounted for by λ_1.

When the observations have been made along a temporal or spatial axis, or on a geographic surface (i.e. a map giving the coordinates of the sampling sites), one may plot the principal component values along the sampling axis, or on the geographic map. Figure 9.15 shows an example of such a map for the first ordination axis of a detrended correspondence analysis. The same approach can be used with the results of a principal component analysis, or any other ordination method.

3 — Contributions of the descriptors

Principal component analysis provides the information needed to understand the role of the original descriptors in the formation of the principal components. It may also be used to show the relationships among the original descriptors. The role of the descriptors in principal component analysis is now examined under various aspects.

1. *The matrix of eigenvectors **U**.* — In Subsection 9.1.1, the relationships among the *normalized eigenvectors*, which are the columns of matrix **U**, were studied using an expression of the form **U**'**U**. For the numerical example:

$$\mathbf{U'U} = \begin{bmatrix} 0.8944 & 0.4472 \\ -0.4472 & 0.8944 \end{bmatrix} \begin{bmatrix} 0.8944 & -0.4472 \\ 0.4472 & 0.8944 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \mathbf{I}$$

The diagonal terms of **U**'**U** result from the scalar product of the eigenvectors with themselves. These values are the (length)^2 of the eigenvectors, here equal to unity because the eigenvectors were scaled to 1. The nondiagonal terms, resulting from the multiplication of two different eigenvectors, are equal to zero because the eigenvectors are orthogonal. This result would be the same for any matrix **U** of normalized eigenvectors computed from a symmetric matrix. Matrix **U** is a square *orthonormal matrix* (Section 4.4); several properties of such matrices are described in Section 2.8.

Orthonormal matrix

In the same way, the relationships among *descriptors*, which correspond to the rows of matrix **U**, can be studied through the product **U**'**U**. The diagonal and nondiagonal terms of **U**'**U** have the same meaning as in **U**'**U**, except that they now concern the relationships among descriptors. This is PCA *scaling 1*, which is explained in more details in Subsection 9.1.4. The relationships among the rows of a square orthonormal matrix are the same as among the columns (Section 2.8, property 7), so that:

Scaling 1

$$\mathbf{U'U} = \mathbf{I} \qquad \textbf{(9.6)}$$

The descriptors are therefore of unit lengths in the multidimensional space and they lie at 90° of one another (orthogonality).

Principal component analysis is simply a rotation, in the multidimensional space, of the original system of axes (Figs. 9.2 and 9.3a, for a two-dimensional space). It therefore follows that, after the analysis (rotation), the original descriptor-axes are still at 90° of one another. Furthermore, normalizing the eigenvectors simultaneously normalizes the descriptor-axes (the lengths of the row and column vectors are given outside the matrix):

$$\mathbf{U} = \begin{bmatrix} u_{11} & \cdots & u_{1p} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ u_{p1} & \cdots & u_{pp} \end{bmatrix} \begin{matrix} \sqrt{\Sigma u_{1k}^2} = 1 \\ \cdot \\ \cdot \\ \cdot \\ \sqrt{\Sigma u_{pk}^2} = 1 \end{matrix} \qquad \textbf{(9.7)}$$

$$\sqrt{\Sigma u_{j1}^2} = 1 \quad \cdots \quad \sqrt{\Sigma u_{jp}^2} = 1$$

There is a second approach to the study of the relationships among descriptors. It consists in scaling the eigenvectors in such a way that the cosines of the angles between descriptor-axes be proportional to their *covariances*. In this approach, the angles between descriptor-axes are between 0° (maximum positive covariance) and 180° (maximum negative covariance); an angle of 90° indicates a null covariance (orthogonality). This result is achieved by scaling each eigenvector k to a length equal to its standard deviation $\sqrt{\lambda_k}$ *. This is PCA *scaling 2*, explained in Subsection 9.1.4. With this scaling for the eigenvectors, the Euclidean distances among objects are not preserved.

Scaling 2

Using the diagonal matrix $\mathbf{\Lambda}$ of eigenvalues (eq. 2.20), the new matrix of eigenvectors, called $\mathbf{U}_{sc2}$ (i.e. $\mathbf{U}$ for scaling 2), can be directly computed by means of the expression $\mathbf{U\Lambda}^{1/2}$. For the numerical example:

$$\mathbf{U}_{sc2} = \mathbf{U\Lambda}^{1/2} = \begin{bmatrix} 0.8944 & -0.4472 \\ 0.4472 & 0.8944 \end{bmatrix} \begin{bmatrix} \sqrt{9} & 0 \\ 0 & \sqrt{5} \end{bmatrix} = \begin{bmatrix} 2.6833 & -1.0000 \\ 1.3416 & 2.0000 \end{bmatrix} \qquad \textbf{(9.8)}$$

In scaling 2, the relationships among descriptors are the same as in the dispersion matrix $\mathbf{S}$ (on which the analysis is based), since

$$(\mathbf{U\Lambda}^{1/2})\,(\mathbf{U\Lambda}^{1/2})' = \mathbf{U\Lambda U'} = \mathbf{U\Lambda U}^{-1} = \mathbf{S} \qquad \textbf{(9.9)}$$

* In some computer packages, PCA only scales the eigenvectors to length $\sqrt{\lambda}$ and only provides a plot of the descriptor-axes; no plot of the objects in reduced space is available.

Equation $\mathbf{U\Lambda U}^{-1} = \mathbf{S}$ is derived directly from the general equation of eigenvectors $\mathbf{SU} = \mathbf{U\Lambda}$ (eq. 2.27). In other words, the new matrix $\mathbf{U\Lambda}^{1/2}$ is of the following form (the lengths of the row and column vectors are given outside the matrix):

$$\mathbf{U}_{sc2} = \mathbf{U\Lambda}^{1/2} = \begin{bmatrix} u_{11}\sqrt{\lambda_1} & \cdots & u_{1p}\sqrt{\lambda_p} & \sqrt{\Sigma\,(u_{1k}\sqrt{\lambda_k})^2} = s_1 \\ \vdots & & \vdots & \vdots \\ u_{p1}\sqrt{\lambda_1} & \cdots & u_{pp}\sqrt{\lambda_p} & \sqrt{\Sigma\,(u_{pk}\sqrt{\lambda_k})^2} = s_p \\ \sqrt{\Sigma\,(u_{j1}\sqrt{\lambda_1})^2} = \sqrt{\lambda_1} & \cdots & \sqrt{\Sigma\,(u_{jp}\sqrt{\lambda_p})^2} = \sqrt{\lambda_p} & \end{bmatrix} \quad \textbf{(9.10)}$$

This equation shows that, when the eigenvectors are scaled to the lengths of their respective standard deviations $\sqrt{\lambda_k}$, the lengths of the descriptor-axes are $\sqrt{s_j^2} = s_j$ (i.e. their standard deviations) in multidimensional space. The product of two descriptor-axes, which corresponds to their angle in the multidimensional space, is therefore equal to their *covariance* s_{jl}.

2. *Projection of descriptors in reduced space, scaling 1: matrix* **U**. — When matrix **U** is used to project the descriptor-axes in a PCA plot, the descriptor-axes are of unit lengths and at right angles in multidimensional space (Fig. 9.3a). The angles between descriptor-axes and principal axes are projections of the *rotation angles* corresponding to the elements of matrix **U** (Fig. 4.10). For the numerical example, the angles between descriptors and principal axes are computed as in Section 4.4 using matrix **U**:

$$\mathbf{U} = \begin{bmatrix} 0.8944 & -0.4472 \\ 0.4472 & 0.8944 \end{bmatrix} \xrightarrow{\text{arc cos}} \begin{bmatrix} 26°34' & 116°34' \\ 63°26' & 26°34' \end{bmatrix}$$

The values of angles in the inset of Fig. 9.3a are thus: $\beta = 26°34'$, $\gamma = 63°26'$, $\delta = 26°34'$. The correlations between descriptors j and principal axes k are the same as in scaling 2 (below) because the two scalings only differ by the stretching of the axes. In scaling 1, the correlations among descriptors are equal to 0 because descriptors are orthogonal (i.e. at right angles) in this representation.

Projection u_{jk} of a descriptor-axis j on a principal axis k is *proportional* to *the covariance* of that descriptor with the principal axis. The proportionality factor is different for each principal axis, so that it is not possible to compare the projection of a descriptor on one axis to its projection on another axis. It is correct, however, to compare the projections of different descriptor-axes on the same principal axis. It can be shown that an isogonal projection (with respectively equal angles) of p orthogonal axes of unit lengths gives a length $\sqrt{d/p}$ to each axis in d-dimensional space. In Fig. 9.4, the equilibrium projection of each of the three orthogonal unit axes, in two-

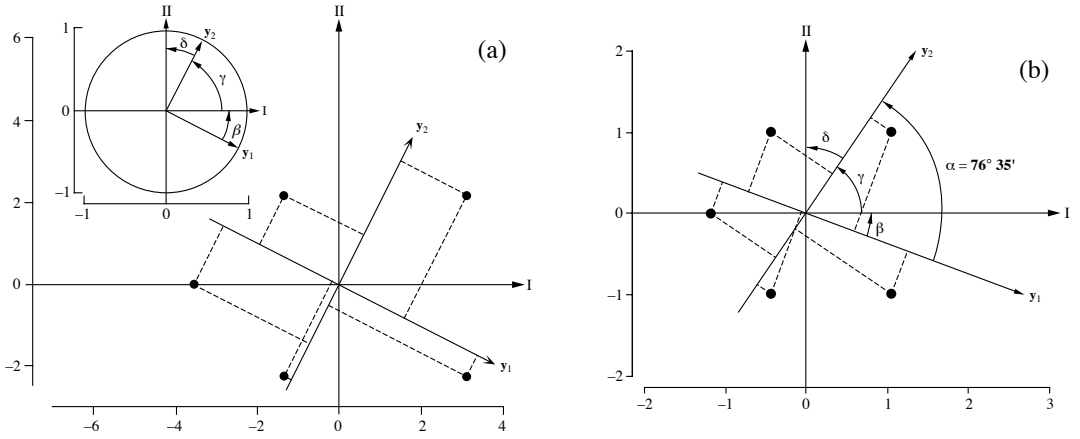


Figure 9.3 Numerical example from Fig. 9.2. Distance and correlation biplots are discussed in Subsection 9.1.4. (a) Distance biplot (scaling 1). The eigenvectors are scaled to lengths 1. Inset: descriptors (matrix **U**). Main graph: descriptors (matrix **U**; arrows) and objects (matrix **F**; dots). The interpretation of the object-descriptor relationships is not based on their proximity, but on orthogonal projections (dashed lines) of the objects on the descriptor-axes or their extensions. The lengths of the arrows were multiplied by 4 for clarity of the diagram. (b) Correlation biplot (scaling 2). Descriptors (matrix $\mathbf{U}\mathbf{\Lambda}^{1/2}$; arrows) with a covariance angle of 76°35'. Objects (matrix **G**; dots). Projecting the objects orthogonally on a descriptor (dashed lines) reconstructs the values of the objects along that descriptors, to within a multiplicative constant.

dimensional space, has a length of $\sqrt{2/3}$. This is due to the fact that an isogonal projection results in an equal association of all descriptor-axes with the principal axes.

Equilibrium circle

An *equilibrium circle of descriptors,* with radius $\sqrt{d/p}$, may be drawn as reference to assess the contribution of each descriptor to the formation of the reduced space (Fig. 9.4). The circle is also drawn in the inset of Fig. 9.3a; its radius is $\sqrt{2/2} = 1$ because, in the numerical example, both the reduced space and the total space are two-dimensional. If one was only interested in the equilibrium contribution of descriptors to the first principal axis, the one-dimensional "circle" would then have a "radius" of $\sqrt{1/2} = 0.7071$. For the example, the projection of the first descriptor on the first principal axis is equal to 0.8944 (examine matrix **U** and Fig. 9.3a), so that this descriptor contributes in an important way to the formation of axis I. This is not the case for the second descriptor, whose projection on the first axis is only 0.4472.

3. *Projection of descriptors in reduced space, scaling 2: matrix* $\mathbf{U}_{sc2}$. — Ecologists using principal component analysis are not interested in the whole multidimensional space but only in a simplified *projection* of the objects in a *reduced space* (generally a two-dimensional plane). The elements $u_{jk}\sqrt{\lambda_k}$ of the eigenvectors scaled to $\sqrt{\lambda_k}$ are the coordinates of the projections of descriptors j on the different principal axes k.

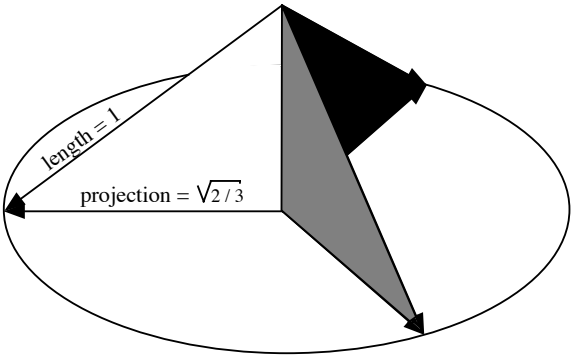


Figure 9.4 Equilibrium projection, in a plane, of three orthogonal vectors with unit lengths, and equilibrium circle of the three descriptors.

They are scaled in such a way that the projections of descriptor-axes can be drawn in the reduced space formed by the principal axes (Fig. 9.3b). As in scaling 1, the descriptors are represented by arrows since they are *axes*. In a reduced-dimension plane, projections of descriptor-axes are shorter than or equal to their lengths in the multidimensional space. In the case of Fig. 9.3b, the lengths are the same in the projection plane as in the original space because the latter only has two dimensions.

In the reduced-space plane, the angles between descriptors are *projections* of their true covariance angles. It is thus important to consider only the descriptors that are well represented in the projection plane. To do so, one must recognize, in the multidimensional space, the descriptors that form small angles with the reduced plane; they are the descriptors whose projections approach their real lengths s in the multidimensional space. Since the length of the *projection* of a descriptor-axis j is equal to or shorter than s_j, one must choose a criterion to assess the value of the representations in the projection plane.

If a descriptor j was equally associated with each of the p principal axes, all elements of row j (which is of length s_j) of matrix $\mathbf{U}_{sc2}$ would be equal, their values being $s_j\sqrt{1/p}$. The length of the descriptor-axis would be $[p\,(s_j\sqrt{1/p})^2]^{1/2} = s_j$ in multidimensional space. The length of the projection of this descriptor-axis in a reduced space with d dimensions would therefore be $s_j\sqrt{d/p}$. The latter expression defines, in the d-dimensional space, a measure of the *equilibrium contribution of a descriptor* to the various axes of the whole multidimensional space. When applied to scaling 1, where the length of the projection of a descriptor in p-dimensional space is 1 (eq. 9.7) instead of s_j (eq. 9.10), the formula for the equilibrium projection of p orthogonal axes in d-dimensional space becomes $\sqrt{d/p}$, as shown above.

Equilibrium contribution

The actual length of a descriptor in reduced space can be compared to that measure, to help judge whether the contribution of the descriptor to the reduced space is larger or smaller than it would be under the hypothesis of an equal contribution to all principal axes. For the numerical example, the lengths of the rows of matrix $\mathbf{U}_{sc2}$ (eq. 9.8), in two-dimensional space, are:

$$\text{length of the first descriptor (row)} = \sqrt{2.6833^2 + (-1.0000)^2} = 2.8636 = s_1$$

$$\text{length of the second descriptor (row)} = \sqrt{1.3416^2 + 2.0000^2} = 2.4083 = s_2$$

Because this simple numerical example has two dimensions only, these lengths are equal to their equilibrium contributions in the two-dimensional space. This is easily verified, using the variances of the descriptors, which are known (Subsection 9.1.1):

$$\text{equilibrium projection of first descriptor} = s_1\sqrt{2/2} = \sqrt{8.2}\sqrt{2/2} = 2.8636$$

$$\text{equilibrium projection of second descriptor} = s_2\sqrt{2/2} = \sqrt{5.8}\sqrt{2/2} = 2.4083$$

In real studies, where ecological data sets are multidimensional, the lengths of descriptors in the reduced space are not equal to their equilibrium contributions.

In scaling 1 above, the angular interpretation of the product of two descriptor-axes was simple: the descriptor-axes were at right angles in multidimensional space. In scaling 2, the angle between two descriptors can be found by applying eq. 2.9 to the rows of the matrix of eigenvectors: the scalar product of two rows of matrix $\mathbf{U}_{sc2}$ (eq. 9.10) divided by the product of the lengths of the rows (which are the standard deviations s_j), gives the cosine of that angle.

The scalar product of two rows of matrix $\mathbf{U}_{sc2}$ is related to the correlation coefficient of the corresponding descriptors. The angles among all descriptors are obtained by reducing to unity (i.e. = 1) the lengths of the row vectors of matrix $\mathbf{U}_{sc2} = \mathbf{U}\boldsymbol{\Lambda}^{1/2}$, then computing the matrix of scalar products among the rows:

$$[\mathbf{D}(s)^{-1}\mathbf{U}\boldsymbol{\Lambda}^{1/2}]\,[\mathbf{D}(s)^{-1}\mathbf{U}\boldsymbol{\Lambda}^{1/2}]' = \mathbf{D}(s)^{-1}\underbrace{\mathbf{U}\boldsymbol{\Lambda}\mathbf{U}'}_{\mathbf{S}}\mathbf{D}(s)^{-1} = \mathbf{D}(s)^{-1}\,\mathbf{S}\,\mathbf{D}(s)^{-1} = \mathbf{R} \qquad \textbf{(9.11)}$$

The result of this equation is the correlation matrix among the descriptors. In the last step of the equation, the correlation matrix $\mathbf{R}$ is connected to the dispersion matrix $\mathbf{S}$ by the diagonal matrix of standard deviations $\mathbf{D}(s)$, following eq. 4.10.

The cosine of the angle α_{jl} between two descriptors $\mathbf{y}_j$ and $\mathbf{y}_l$, in multidimensional space, is therefore related to their *correlation* (r_{jl}); it can actually be shown that $\cos(\alpha_{jl}) = r_{jl}$. This angle is the same as that of the *covariance* because standardization of the rows to unit lengths has only changed the lengths of the descriptor-axes and not their positions in multidimensional space. For the numerical example, the correlation between the two descriptors is equal to $1.6/\sqrt{8.2 \times 5.8} = 0.232$ (eq. 4.7). The angle

corresponding to this correlation is (arc cos 0.232) = 76°35', which is the same as the angle of the covariance in Fig. 9.3b.

In the same way, the angle between a descriptor j and a principal axis k, in multidimensional space, is the arc cosine of the correlation between descriptor j and principal component k. The correlation r_{jk} is element jk of the matrix of eigenvectors (eq. 9.10) normalized by row (the length of a row vector in eq. 9.10 is s_j):

$$r_{jk} = u_{jk}\sqrt{\lambda_k}/s_j \qquad \textbf{(9.12)}$$

In other words, the correlation is calculated by weighting the element of the eigenvector by the ratio of the standard deviation of the principal component to that of the descriptor. For the numerical example, these correlations and corresponding angles are computed using matrix $\mathbf{U\Lambda}^{1/2}$ (calculated above) and the standard deviations of the two descriptors ($s_1 = 2.8636$, $s_2 = 2.4083$):

$$[r_{jk}] = [u_{jk}\sqrt{\lambda_k}/s_j] = \begin{bmatrix} 0.9370 & -0.3492 \\ 0.5571 & 0.8305 \end{bmatrix} \xrightarrow{\text{arc cos}} \begin{bmatrix} 20°26' & 110°26' \\ 56°09' & 33°51' \end{bmatrix}$$

The values of angles in Fig. 9.3b are thus: $\beta = 20°26'$, $\gamma = 56°09'$, $\delta = 33°51'$. These correlations may be used to study the contributions of the descriptors to the various components, the scale factors of the descriptors being removed. The highest correlations (absolute values), in the correlation matrix between descriptors and components, identify the descriptors that contribute most to each eigenvector. The significance of the correlations between descriptors and components cannot be tested using a standard test for Pearson correlation coefficients, however, because the principal components are linear combinations of the descriptors themselves.

When the descriptor-axes of matrix $\mathbf{U\Lambda}^{1/2}$ are scaled to unit lengths, which is done by computing $[\mathbf{D}(s)^{-1}\mathbf{U\Lambda}^{1/2}]$ as in eq. 9.11, drawing their projections in the principal axes space is not recommended. This is because the rescaled eigenvectors are not necessarily orthogonal and may be of any lengths:

$$[\mathbf{D}\,(s)^{-1}\,\mathbf{U\Lambda}^{1/2}]\,[\mathbf{D}\,(s)^{-1}\,\mathbf{U\Lambda}^{1/2}] \neq \mathbf{I} \qquad \textbf{(9.13)}$$

The principal axes are therefore not necessarily at right angles.

The projections of the descriptor-axes of matrix $\mathbf{U\Lambda}^{1/2}$ may be examined, in particular, with respect to the following points:

- The coordinates of the projection of a descriptor-axis specify the position of the apex of this descriptor-axis in the reduced space. It is recommended to use arrows to represent projections of descriptor-axes. Some authors call them point-descriptors or point-variables and represent them by *points* in the reduced space. This representation is ambiguous and misleading. It is acceptable only if the nature of the point-descriptors

is respected; they actually are *apices of descriptor-axes,* so that the relationships among them are defined in terms of angles representing their correlations, not in terms of proximities (Fig. 9.5).

• The projection $u_{jk}\sqrt{\lambda_k}$ of a descriptor-axis j on a principal axis k shows its covariance with the principal axis and, consequently, its positive or negative contribution to the position of the objects along the axis. It follows that a principal axis may often be qualified by the names of the descriptors that are mostly contributing, and in a preferential way, to its formation. Thus, in Fig. 9.5, principal axis I is formed mainly by descriptors 6 to 10 and axis II by descriptors 1 to 4.

• The descriptors that contribute most to the formation of the reduced space are those whose projected lengths reach or exceed the values of their equilibrium contributions. Descriptor-axes that are clearly shorter than these values contribute little to the formation of the reduced space under study and, therefore, contribute little to the structure that may be found in the projection of the *objects* in that reduced space.

• The correlations among descriptors are expressed by *the angles between descriptor-axes, not by the proximities between their apices*. In the reduced space, one can often identify groups of descriptor-axes that form small angles with one another, or have angles close to 180° ($\cos 180° = -1$, which would reflect a perfect negative correlation). One must remember, however, that projections of correlation angles in a reduced space do not render the complete correlations among variables. Thus, it may be informative to cluster descriptors by cluster analysis (Chapter 8) of a distance matrix computed as the one-complement of the correlations ($\mathbf{D} = 1 - \text{cor}(\mathbf{Y})$) or the one-complement of the absolute values of the correlations.

• Objects in scaling 2 (or correlation) biplots can be projected at right angles onto the descriptor-axes to approximate their values along the descriptors (Fig. 9.3b). The distances among objects in a scaling 2 biplot *are not* approximations of their Euclidean distances; they approximate their Mahalanobis distances (Subsection 9.1.4).

The main properties of a principal component analysis of centred descriptors are summarized in Table 9.2.

4. *Cumulative fit tables*. — How well is the variance of each descriptor explained, or *fitted*, by 1, 2, or more axes of the PCA solution? To obtain that information, R^2 coefficients (eq. 10.20) can be computed between the descriptors (which have possibly been standardized or transformed in some other way prior to PCA) and principal components 1, 2, …, k found in matrices **F** or **G** (Subsection 9.1.4). Then, for each descriptor, a table of *Cumulative fit per descriptor* is created. The R^2 coefficient for axis 1 alone is written in column 1; the cumulated sum of the R^2 coefficients over axes 1 and 2 is written in column 2; the cumulated sum of the R^2 coefficients over axes 1, 2 and 3 is written in column 3; and so on. For the numerical example data of

Cumulative fit per descriptor

Table 9.2 Principal component analysis. Main properties for centred descriptors j.

	Scaling 1 (distance biplot)	**Scaling 2** (correlation biplot)
Length of the scaled eigenvectors	1	$\sqrt{\lambda_k}$
Length of descriptor j in $\mathbf{U}$ or $\mathbf{U}_{sc2}$	1	s_j
Angles in reduced space	90°, i.e. rigid rotation of the system of axes	projections of covariances (correlations)
Length of equilibrium contribution	circle with radius $\sqrt{d/p}$	$s_j\sqrt{d/p}$
Projection on principal axis k	u_{jk} i.e. proportional to the covariance with k	$u_{jk}\sqrt{\lambda_k}$ i.e. covariance with component k
Correlation with principal component k	$u_{jk}\sqrt{\lambda_k}/s_j$	$u_{jk}\sqrt{\lambda_k}/s_j$

the present section, which produce a PCA solution in two dimensions, the table has two columns only:

	Cumul. axis 1	**Cumul. axis 2**
Descriptor $\mathbf{y}_1$	0.8780	1.0000
Descriptor $\mathbf{y}_2$	0.3103	1.0000

The values of R^2 in the last column are always 1 in PCA. Identical results can be computed directly from matrix $\mathbf{U}_{sc2}$: the coefficients found in that matrix (eq. 9.8) are squared and summed cumulatively from left to right; then the cumulated sum for row j is divided by the total variance of descriptor j. That table proves very useful for interpretation of analyses involving many variables, in particular in the case of species-rich assemblages in community studies: it allows one to decide which species are well fitted and should be represented, for example, in a two-dimensional PCA biplot (Subsection 9.1.4). This output table is available in program CANOCO where it is called "Cumulative fit per species as fraction of variance of species".

Objects are vectors in multivariate A-space (Fig. 7.2) and vectors have lengths (Section 2.4). The squared length of each object, computed as the sum of the squared values in matrix $\mathbf{Y}_c$ subjected to PCA, is the reference value. An identical total squared length can be computed using matrix $\mathbf{F}$ instead of $\mathbf{Y}_c$. Use matrix $\mathbf{F}$ to compute the

squared length of each object in 1, 2, 3 … PCA dimensions. For example, the total squared length of object 2 of the numerical example is $(-2.2^2 + 1.4^2) = 6.8$ and the squared length of the projection of object 2 on PCA axis I, read from matrix **F** (Subsection 9.1.2), is $(f_{21})^2 = (-1.342)^2 = 1.8$. Hence the proportion of fit for that object along PCA axis I is 1.8/6.8 = 0.2647. The squared residual length of object 2 after fitting it along axis I is 6.8 – 1.8 = 5. For the numerical example data, the table of *cumulative fit of the objects* is:

Cumulative fit of objects

	Cumul. axis 1	**Cumul. axis 2**
Object 1	1.0000	1.0000
Object 2	0.2647	1.0000
Object 3	0.2647	1.0000
Object 4	0.6622	1.0000
Object 5	0.6622	1.0000

This type of output table is useful to decide which objects are well represented in a PCA plot: the distances between well-represented objects can be trusted and interpreted. A related table called "Squared residual length per sample" is available in the output of program CANOCO; it gives squared residual values instead of relative values or percent fit.

4 — PCA biplots

The previous two subsections have shown that, in principal component analysis, both the descriptor-axes and object-vectors can be plotted in the reduced space. This led Jolicoeur & Mosimann (1960) to plot these projections together in the same diagram. Gabriel (1971) proposed the name *biplot* for these diagrams and developed the theory of biplots in Gabriel (1971, 1982). Other important contributors to the theory of biplots are ter Braak (1983 and other papers) and Gower (1990 and other papers). Mathematical details about the theory of PCA biplots are found in Greenacre (2010) and Gower *et al*. (2011); these books offer R functions to produce various types of biplots.

Two types of biplots may be used to represent PCA results (Gabriel, 1982; ter Braak, 1994). *Distance biplots* graph together matrices **U** (eigenvectors scaled to lengths 1) and **F** (eq. 9.4); in **F**, the variance of principal component (column) k is λ_k. *Correlation biplots* use matrix $\mathbf{U}_{sc2}$ for descriptors, where eigenvector k is scaled to length $\sqrt{\lambda_k}$, and matrix **G** for objects, where

$$\mathbf{G} = \mathbf{F}\boldsymbol{\Lambda}^{-1/2} \qquad \textbf{(9.14)}$$

The columns of matrix **G** have unit variances. The Euclidean distances among the objects in matrix **G** are equal to the Mahalanobis distances (D_5, eq. 7.38) among the objects in the original data matrix **Y**, so that the distances in a correlation biplot are

Mahalanobis distance

projections of these Mahalanobis distances, not of the original Euclidean distances. Since Mahalanobis distances are independent of the scaling of descriptors, it follows that the Euclidean distances among objects in matrix **G** are the same for a PCA conducted on unstandardized or standardized descriptors when considering all axes.

Matrices **F** and **U**, or **G** and $\mathbf{U}_{sc2}$, can be used together in biplots because the products of the eigenvectors with the object score matrices reconstruct the original (centred) matrix **Y** perfectly:

$$\mathbf{F}\mathbf{U}' = \mathbf{Y} \quad \text{and} \quad \mathbf{G}(\mathbf{U}\boldsymbol{\Lambda}^{1/2})' = \mathbf{Y}.$$

Actually, the eigenvectors and object score vectors may be multiplied by any constant without changing the interpretation of a PCA biplot.

Distance biplot (scaling 1)

• *Distance biplot, scaling 1 (Fig. 9.3a).* — The main features of a distance biplot are the following: (1) Distances among objects in the biplot are approximations of their Euclidean distances in multidimensional space. (2) Projecting an object at right angle on a descriptor approximates the position of the object along that descriptor. (3) Since descriptors have lengths of 1 in the full-dimensional space (eq. 9.7), the length of the projection of a descriptor in reduced space indicates how much it contributes to the formation of that space. (4) The angles among descriptor-axes are meaningless.

Correlation biplot (scaling 2)

• *Correlation biplot, scaling 2 (Fig. 9.3b).* — The main features of a correlation biplot are the following: (1) Distances among objects in the biplot are approximations of their Mahalanobis distances in multidimensional space; they *are not* approximations of their Euclidean distances. (2) Projecting an object at right angle on a descriptor approximates the position of the object along that descriptor. (3) Since descriptors have lengths s_j in full-dimensional space (eq. 9.10), the length of the projection of a descriptor in reduced space is an approximation of its standard deviation. (4) The angles between descriptors in the biplot reflect their correlations. (5) When the distance relationships among objects are important for interpretation, this type of biplot is inadequate; a distance biplot should be used.

For the numerical example, matrix **G** is computed from eq. 9.14 as follows:

$$\mathbf{G} = \mathbf{F}\boldsymbol{\Lambda}^{-1/2} = \begin{bmatrix} -3.578 & 0 \\ -1.342 & 2.236 \\ -1.342 & -2.236 \\ 3.130 & 2.236 \\ 3.130 & -2.236 \end{bmatrix} \begin{bmatrix} 0.3333 & 0 \\ 0 & 0.4472 \end{bmatrix} = \begin{bmatrix} -1.193 & 0.000 \\ -0.447 & 1.000 \\ -0.447 & -1.000 \\ 1.044 & 1.000 \\ 1.044 & -1.000 \end{bmatrix}$$

One can check that the columns of **G** have unit variances. In this particular example, the relationships between objects and descriptors are fully represented in a two-dimensional space. Readers are invited to repeat the PCA using standardized descriptors and verify the fact that the Euclidean distances among objects in matrix **G**

are the same for unstandardized and standardized descriptors: in both cases, they are the Mahalanobis distances among the objects in matrix **Y**.

The descriptor coordinates must often be multiplied by a constant to produce a clear visual display. In Fig. 9.3a for instance, the lengths of the descriptor arrows would be too short for visual appraisal if they were plotted in the same system of coordinates as the objects. In computer software, rescaling of the descriptors is done either by the PCA function or by the plotting function. Some researchers have been tempted to interpret the relationships between objects and descriptors in terms of their proximity in the reduced space, whereas a correct interpretation requires the projection of the objects on the descriptor-axes (centred with respect to the scatter of points) or on their extensions (Fig. 9.3a). In Fig. 9.2a for example, it would not come to mind to interpret the relationship between the objects and descriptor $\mathbf{y}_1$ in terms of the distance between the object-points and the apex (head of the arrow) of axis $\mathbf{y}_1$. In Fig. 9.3a, the position of the apex of $\mathbf{y}_1$ is arbitrary and depends on the multiplicative constant used. Projections of objects onto an axis specify the coordinates of the objects with respect to that descriptor-axis, taking the multiplicative constant into account.

5 — Principal components of a correlation matrix

Principal component analysis performs a partitioning of the total variance of matrix **Y**. If the variables are not in the same physical dimensions, their variances, which are expressed in the squared units of the variables, cannot be added (Section 3.2). As a consequence, before adding the variances of the p variables of **Y**, one must make sure that the variables are dimensionally homogeneous, i.e. expressed in the same physical dimensions. If they are not, they must be standardized (eq. 1.12). The analysis is then described as a PCA carried out on a correlation matrix **R** since correlations are covariances of standardized descriptors (Section 4.2).

In an **R** matrix, all diagonal elements are equal to 1. It follows that the sum of eigenvalues, which corresponds to the total variance of the dispersion matrix, is equal to the order of **R**, given by the number of descriptors p. Before computing the principal components, it may be a sound practice to check that $\mathbf{R} \neq \mathbf{I}$ (eq. 4.14).

Principal components extracted from correlation matrices are not the same as those computed from dispersion matrices. [Beware: some computer packages only allow the computation of principal components from correlation matrices; this is inappropriate for many studies.] Consider the basic equation for the eigenvalues and eigenvectors, $(\mathbf{S} - \lambda_k \mathbf{I})\, u_k = 0$. The sum of the eigenvalues of **S** is equal to the sum of variances s^2, whereas the sum of eigenvalues of **R** is equal to p, so that the eigenvalues of the two matrices, and therefore also their eigenvectors, are necessarily different. This is due to the fact that distances between objects are not the same in the two analyses.

In the case of correlations, the descriptors are standardized. It follows that the distances among objects are independent of the measurement units, whereas those in

the space of the original descriptors vary according to measurement scales. When the descriptors are all of the same type and order of magnitude, and have the same units, it is clear that matrix **S** must be used to compute PCA. In that case, the eigenvectors, on the one hand, and the correlation coefficients between descriptors and components, on the other hand, provide complementary information. The former give the loadings of descriptors and the latter quantify their relative importance. When the descriptors are of different types or orders of magnitude, or have different units, one must conduct PCA on matrix **R** instead of matrix **S**.

S or **R** matrix?

Ecologists who wish to study the relationships among objects in a reduced space of principal components may base their decision of conducting the analysis on **S** or **R** on the answer to the following question:

- If one wanted to cluster the objects in the reduced space, should the clustering be done with respect to the original descriptors (or any transformation of these descriptors; Section 1.5), thus preserving their differences in magnitude? Or, should all descriptors contribute equally to the clustering of objects, independently of the variance exhibited by each one? In the second instance, one should proceed from the correlation matrix. An alternative in this case is to transform the descriptors by ranging, using eq. 1.10 for relative-scale descriptors and eq. 1.11 for interval-scale descriptors, and carry out the analysis on matrix **S** of the transformed descriptors.

Another way to look at the same problem was suggested by Gower (1966):

- The Euclidean distance (eq. 7.32) is the distance preserved among objects through principal component analysis. Is it with the raw data (covariances) or with the standardized data (correlations) that the spatial configuration of the objects, in terms of Euclidean distances, is the most interesting for interpretation? In the first case, conduct PCA on matrix **S**; in the second case, use matrix **R**.

The principal components of a correlation matrix are computed from matrix **U** of the eigenvectors of **R** and the matrix of standardized observations:

$$\mathbf{F} = \left[\frac{y - \bar{y}}{s_y}\right]\mathbf{U} \qquad \textbf{(9.15)}$$

Principal component analysis is still only a rotation of the system of axes (Subsection 9.1.2). However, since the descriptors are now *standardized*, the objects are not positioned in the same way as if the descriptors had simply been *centred* (i.e. principal components computed from matrix **S** in the previous subsections).

As far as the representation of descriptors in the reduced space computed from matrix **R** is concerned, the conclusions of Subsection 9.1.3, which concerned matrix **S**, can be used here, after replacing *covariance* by *correlation*, s_{jl} by r_{jl}, and *dispersion matrix* **S** by *correlation matrix* **R**.

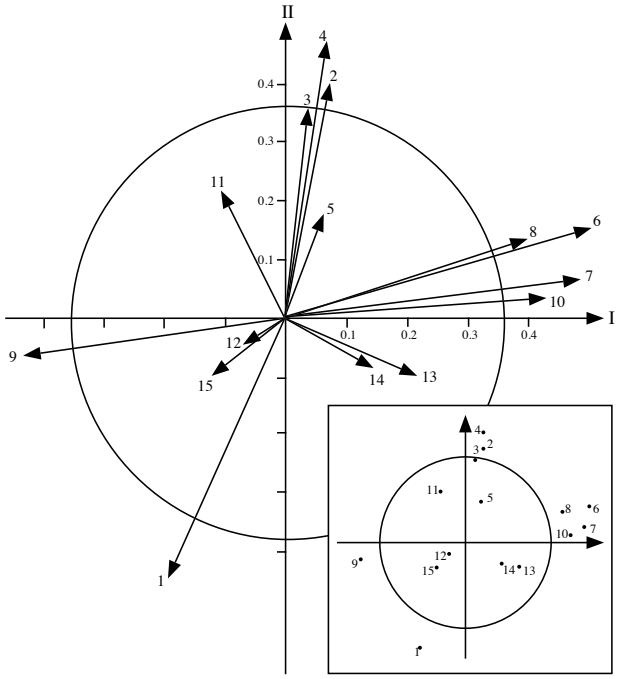


Figure 9.5 Fifteen descriptors plotted in the plane determined by the first two principal axes. The coordinates of each descriptor are the first two elements of the corresponding row of matrix $\mathbf{U}\boldsymbol{\Lambda}^{1/2}$ (i.e. the eigenvectors of $\mathbf{R}$ scaled to $\sqrt{\lambda}$). The circle of equilibrium descriptor contribution is drawn at $\sqrt{2/15} = 0.365$. The inset figure shows the same descriptor-axes using only the apices of the vectors. This representation, which is sometimes encountered in the ecological literature, must be avoided because of possible confusion with point-objects.

The variances, and therefore also the standard deviations, of the *standardized* descriptors are equal to unity (i.e. = 1), which leads to some special properties for the $\mathbf{U}\boldsymbol{\Lambda}^{1/2}$ matrix. First, $\mathbf{D}(s) = \mathbf{I}$, so that $\mathbf{U}\boldsymbol{\Lambda}^{1/2} = \mathbf{D}(s)^{-1}\mathbf{U}\boldsymbol{\Lambda}^{1/2}$, i.e. the coefficients $u_{jk}\sqrt{\lambda_k}$ are the correlation coefficients between descriptors j and components k. In addition, the equilibrium contribution corresponding to each descriptor, in the reduced space of $\mathbf{U}\boldsymbol{\Lambda}^{1/2}$, is $s_j\sqrt{d/p} = \sqrt{d/p}$ (since $s_i = 1$). It is therefore possible to judge whether the contribution of each descriptor to the reduced space is greater or smaller than expected under the hypothesis of an equal contribution to all principal axes, by comparing the lengths of their projections to an equilibrium circle with radius $\sqrt{d/p}$ (Fig. 9.5).

The main properties for standardized descriptors are summarized in Table 9.3, which parallels Table 9.2 for centred descriptors.

Table 9.3 Principal component analysis. Main properties for standardized descriptors j.

	Scaling 1 (distance biplot)	**Scaling 2** (correlation biplot)
Length of the scaled eigenvectors	1	$\sqrt{\lambda_k}$
Length of descriptor j in $\mathbf{U}$ or $\mathbf{U}_{sc2}$	1	1
Angles in reduced space	90°, i.e. rigid rotation of the system of axes	projections of correlations
Radius of equilibrium contribution circle	$\sqrt{d/p}$	$\sqrt{d/p}$
Projection on principal axis k	u_{jk} i.e. proportional to the correlation with k	$u_{jk}\sqrt{\lambda_k}$ i.e. correlation with component k
Correlation with principal component k	$u_{jk}\sqrt{\lambda_k}$	$u_{jk}\sqrt{\lambda_k}$

6 — *The meaningful components*

The successive principal components correspond to progressively smaller fractions of the total variance. One problem is therefore to determine how many components are meaningful in ecological terms or, in other words, what should be the number of dimensions of the reduced space. The best approach may be to visually check the representativeness of the projections in reduced space for two, three, or more dimensions, using Shepard diagrams (Fig. 9.1). However, principal component analysis being a form of variance partitioning, researchers may wish to test the significance of the variance associated with the successive principal axes.

There are a number of classical statistical approaches to this question, such as Bartlett's (1950) test of sphericity. These approaches have been reviewed by Burt (1952) and Jackson (1993). The problem is that these formal tests require normality of all descriptors, a condition that is rarely met by ecological data.

Kaiser-Guttman criterion

There is an empirical rule suggesting that one should only interpret a principal component if the corresponding eigenvalue λ is larger than the mean of the λ's. In the particular case of standardized data, where $\mathbf{S}$ is a correlation matrix, the mean of the λ's is 1 so that, according to the rule, only the components whose λ's are larger than 1 should be interpreted. This is the so-called Kaiser-Guttman criterion. Ibanez (1973) has provided a theoretical framework for this empirical rule. He showed that, if a variable made of randomly selected numbers is introduced among the descriptors, it is

not possible to interpret the eigenvectors that follow the one on which this random-number variable has the highest loading. One can show that this random-number variable, which has covariances near zero with all the other descriptors, introduces in the analysis an eigenvalue of 1 if the descriptors have been standardized. For non-standardized descriptors, this eigenvalue is the mean of the λ's if the variance of the random-number variable is made equal to the mean variance of the other descriptors.

Broken stick

Frontier (1976) proposed to compare the list of decreasing eigenvalues to the decreasing values of the broken stick model (Subsection 6.5.2). This comparison is based on the following idea. Consider the variance shared among the principal axes to be a resource embedded in a stick of unit length. If principal component analysis had divided the variance at random among the principal axes, the fractions of total variation explained by the various axes would be about the same as the relative lengths of the pieces obtained by breaking the unit stick at random into as many pieces as there are axes. If a unit stick is broken at random into $p = 2, 3, \ldots$ pieces, the expected values (E) of the relative lengths of the successively smaller pieces (j) are given by eq. 6.50:

$$E(\text{piece}_j) = \frac{1}{p}\sum_{x=j}^{p}\frac{1}{x} \qquad \textbf{(9.16)}$$

The expected values are equal to the lengths that would be obtained by breaking the stick at random a large number of times and calculating the mean length of the longest pieces, the second longest pieces, etc. A stick of unit length may be broken at random into p pieces by placing on the stick $(p - 1)$ random break points selected using a uniform [0, 1] random number generator. An R function is available to compute the expected values of the broken stick distribution for any number of pieces (Section 9.5).

Coming back to the eigenvalues, it would be meaningless to interpret the principal axes that explain a fraction of the variance as small as or smaller than that predicted by the broken stick null model. The test may be carried out in two ways. One may compare individual eigenvalues to individual predictions of the broken stick model and select for interpretation only the eigenvalues that are larger than the values predicted by the model. Or, to decide whether eigenvalue λ_k should be interpreted, one may compare the *sum of eigenvalues*, from 1 to k, to the sum of the values from 1 to k predicted by the model. This test usually recognizes the first two or three principal components as meaningful, which corresponds to the experience of ecologists.

After an empirical study using a variety of matrix types, using simulated and real ecological data, Jackson (1993) concluded that two methods consistently pointed to the correct number of ecologically meaningful components in data sets: the broken-stick model and a bootstrapped eigenvalue-eigenvector method proposed in his paper.

Chapter 10 will discuss how to use explanatory variables to ecologically interpret the first few principal components that are considered to be meaningful according to one of the criteria mentioned in the present subsection.

7 — Misuses of principal component analysis

Given the power of principal component analysis, some applications have used it in ways that exceed the limits of the model. Some of these limits may be transgressed without much consequences, while others are more critical. The most common errors are: the use of descriptors for which a measure of covariance is not appropriate, and the interpretation of relationships between descriptors, in reduced space, based on the relative positions of the apices of axes instead of the angles between them.

Principal component analysis was originally defined for data with *multinormal distributions* (Section 4.4), so that its optimal use (Cassie & Michael, 1968) calls for normalization of the data (Subsection 1.5.6). Deviations from normality do not necessarily bias the analysis, however (Ibanez, 1971). It is only important to make sure that the descriptors' distributions are reasonably unskewed. Typically, in analyses conducted with strongly skewed distributions, the first few principal components only separate a few objects with extreme values from the remaining objects, instead of displaying the main axes of variation of all objects in the study.

A full-rank dispersion matrix **S** cannot be estimated using a number of observations n smaller than or equal to the number of descriptors p. When $n \leq p$, since there are $n - 1$ degrees of freedom in total, the rank of the resulting dispersion matrix of order p is $(n - 1)$. In such a case, the eigen-decomposition of **S** produces $(n - 1)$ real and $[p - (n - 1)]$ null eigenvalues. Indeed, positioning n objects while respecting their distances requires $(n - 1)$ dimensions only. The PCA of a data matrix where $n \leq p$ produces $(n - 1)$ eigenvalues larger than 0 and the $(n - 1)$ corresponding eigenvectors and principal components. To obtain a full-rank dispersion matrix **S** and p principal components, the number of objects n must be larger than p.

Principal components are computed from the eigenvectors of a dispersion matrix. This means that the method is to be used on a matrix of covariances (or possibly correlations) with the following properties: matrix **S** (or **R**) has been computed among descriptors that are quantitative, and for which valid estimates of the covariances (or correlations) may be obtained. These conditions are violated in the following cases:

1) Some authors have transposed the data matrix and computed correlations among the *objects* (i.e. Q mode) instead of among the descriptors (R mode). Their aim was to position the descriptors in the reduced space of the objects. There are several reasons why this operation is incorrect, the least being that it is useless considering that principal component analysis provides information about the relationships among both objects and descriptors. The reasons why correlations should not be computed in Q-mode are explained in Box 7.1, where points 1, 2 and 4 also apply to covariances.

In the literature, the expression "components in Q mode" may sometimes designate a rightful analysis conducted on an R matrix. This expression comes from the fact that one can use principal component analysis primarily as a method for positioning objects

in reduced space. The meanings of "Q mode" and "R mode" are variable in the scientific literature; their meanings in numerical ecology are defined in Section 7.1.

Rao (1964), Gower (1966), and Orlóci (1967a) have shown that, *as a computational technique*, principal components can be obtained by computing the eigenvalues and eigenvectors of a Q-mode matrix. The steps are the following:

- Starting with matrix **Y** centred by columns, $\mathbf{Y}_c$, compute matrix $\mathbf{C}_{np} = \mathbf{Y}_c / \sqrt{n-1}$. This matrix is such that $\mathbf{C}'\mathbf{C} = \mathbf{S}_{pp}$, which is the usual variance-covariance matrix of **Y**.
- Compute the cross-product matrix $\mathbf{Q}_{nn} = \mathbf{C}\mathbf{C}'$ instead of $\mathbf{S}_{pp} = \mathbf{C}'\mathbf{C}$.
- Determine the non-zero eigenvalues of **Q** and their associated eigenvectors.
- Scale each eigenvector k to length $\sqrt{\lambda_k}$, then multiply each value by $\sqrt{n-1}$.
- The eigenvalues of matrix **Q** are the same as those of matrix **S**, and the scaled eigenvectors are matrix **F** of the principal components of **Y**. This perfectly valid computational technique is different from the approach criticised in the previous paragraph.

2) Covariances and correlations are defined for quantitative descriptors only (Section 7.5). This implies, in particular, that one must not use multistate qualitative descriptors in analyses based upon covariance matrices, because means and variances computed from non-ordered states are meaningless.

Precision of data

Principal component analysis is very robust, however, to variations in the *precision of data*. Variables may be recoded into a few classes without noticeable change to the results (Frontier & Ibanez, 1974; Dévaux & Millerioux, 1976a).

Spearman correlation

Pearson correlation coefficients calculated using semiquantitative data are equivalent to Spearman's rank correlation coefficients (eq. 5.3). In a discussion of principal component analysis computed using semiquantitative data, Lebart *et al.* (1979) provide, for various numbers of objects and descriptors, values above which the λ's of the first two principal components may be considered significant. Gower (1966) has also shown that, with binary descriptors, principal component analysis positions the objects, in multidimensional space, at distances that are proportional to the square roots of the complements of simple matching coefficients, i.e. $D = \sqrt{1 - S_1}$ (S_1: eq. 7.1).

3) When calculated over data sets with many double-zeros, coefficients such as the covariance and correlation lead to PCA ordinations with inadequate estimates of the distances among sampling sites. The problem arises from the fact that the principal-component rotation preserves the Euclidean distance among objects (Table 9.1, Fig. 9.2d). The double-zero problem has been discussed in Subsection 7.2.2 and the paradox associated with the Euclidean distance has been presented after eq. 7.32. With untransformed species abundance data, principal component analysis should only be used when the sampling sites cover short gradients (see Subsection 9.1.10). For longer ecological gradients, the species data must be transformed using one of the

transformations of Section 7.7. Else, ordinations can be obtained using correspondence analysis (CA, Section 9.2) when the chi-square distance is appropriate, or by principal coordinate analysis (PCoA, Section 9.3) or nonmetric multidimensional scaling (nMDS, Section 9.4) using other adequate distances.

This last remark explains why, in the ecological literature, principal component analysis has at times not provided interesting results, for example in studies of species associations (e.g. Margalef & Gonzalez Bernaldez, 1969; Ibanez, 1972; Reyssac & Roux, 1972). This problem had also been noted by Whittaker & Gauch (1973). The search for species association is discussed in Section 8.9.

Attempts to interpret the *proximities between the apices* of species-axes in the reduced space, instead of considering the angles separating these descriptor-axes (e.g. Fig. 9.5, inset), may also led to incorrect conclusions and useless results.

Table 9.4 summarizes, with reference to the text, the various questions that may be addressed in the course of a principal component analysis.

8 — Ecological applications

Ecological application 9.1a

From 1953 to 1960, pitfall traps were set at 100 sites in four valleys in the Meijendel dune area north of the Hague, in The Netherlands. They were visited weekly during 365 weeks. In the 36500 relevés, approximately 425 animal species were identified, about 90% of them being arthropods. Aart (1973) studied the wolf spiders (*Lycosidea* and *Pisauridae*: 45030 specimens) to assess how lycosid species shared the multidimensional space of resources (see Section 1.0 for the concept of niche). The Aart (1973) paper reports a PCA based on a data table of 100 sites × 12 species obtained by adding the values from the different week catches for each trap; two of the 14 species were eliminated because they had been found only twice and once, respectively. PCA was applied to the standardized species data, which contained about 30% zero values. Previous editions of the present book reproduced the separate PCA plots of species and sites found in the Aart (1973) paper.

Another set of pitfall traps were set at 100 sites for 60 weeks, in 1969-1970, in Bierlap, one of the dune valleys of the previous survey. Eleven of the 12 spider species were the same as in the Aart (1973) paper. Environmental descriptors were obtained for 28 of these sites. The spider data (28 sites × 12 species, data cumulated over the weeks) were analysed by Aart & Smeenk-Enserink (1975) and related to the environmental variables using canonical correlation analysis (CCorA, Section 11.4); these data are reanalyzed in Ecological application 11.1b using redundancy analysis (RDA, Section 11.1) instead of CCorA.

The spider data from the 28 sites* are analysed here by PCA to illustrate the interest of data transformations. Figure 9.6a shows the results of the analysis of the raw species abundance data and Fig. 9.6b the biplot resulting from the analysis of the same data after a $\log(y + 1)$ transformation, the same transformation that had been used by Aart & Smeenk-Enserink (1975).

* The species data file is available electronically. See footnote in Ecological application 11.1b.

Table 9.4 Questions that can be addressed in the course of a principal component analysis and the answers found in Section 9.1.

Before starting a principal component analysis	*Pages*
1) Are the descriptors appropriate?	
⇒ Quantitative descriptors; multinormality; not too many zeros.	450-452
2) Are the descriptors dimensionally homogeneous?	
⇒ If YES, conduct the analysis on the dispersion matrix	442, 446
⇒ If NO, conduct the analysis on the correlation matrix	445-448
3) Purpose of the ordination in reduced space:	
⇒ To preserve and display the relative positions of the objects: scale the eigenvectors to unit lengths to obtain matrix $\mathbf{U}$	432
Draw a distance biplot (descriptors: $\mathbf{U}$; objects: $\mathbf{F} = \mathbf{YU}$)	444
⇒ To display the correlations among descriptors: scale the eigenvectors to $\sqrt{\lambda}$ to obtain matrix $\mathbf{U}_{sc2}$	436
Draw a correlation biplot (descriptors: $\mathbf{U}_{sc2}$; objects: $\mathbf{G} = \mathbf{F\Lambda}^{-1/2}$) (beware: Euclidean distances among objects are not preserved)	444
While examining the results of a principal component analysis	
1) How informative is a representation of the objects in an m-dimensional reduced space?	
⇒ Compute eq. 9.5	433
2) Are the distances among objects well preserved in the reduced space?	
⇒ Compare Euclidean distances using a Shepard diagram	427-428
3) Which eigenvalues are important?	
⇒ Is λ_k larger than the mean of the λ's?	448
⇒ Is the percentage of the variance corresponding to λ_k larger than the corresponding value in the broken stick model?	449
4) What are the descriptors that contribute the most to the formation of the reduced space?	
⇒ Compute the equilibrium contribution of descriptors and, when appropriate, draw the circle	437, 439, 442, 448
⇒ Compute correlations between descriptors and principal axes	440, 442, 448
⇒ Compute the table of "Cumulative fit per descriptor"	441-442
5) How to represent the objects in the reduced space?	
⇒ Scaling 1: $\mathbf{F} = \mathbf{Y}_c\mathbf{U}$; scaling 2: $\mathbf{G} = \mathbf{F\Lambda}^{-1/2}$	433-435, 443-444
⇒ Compute the table of "Cumulative percent fit of the objects"	443

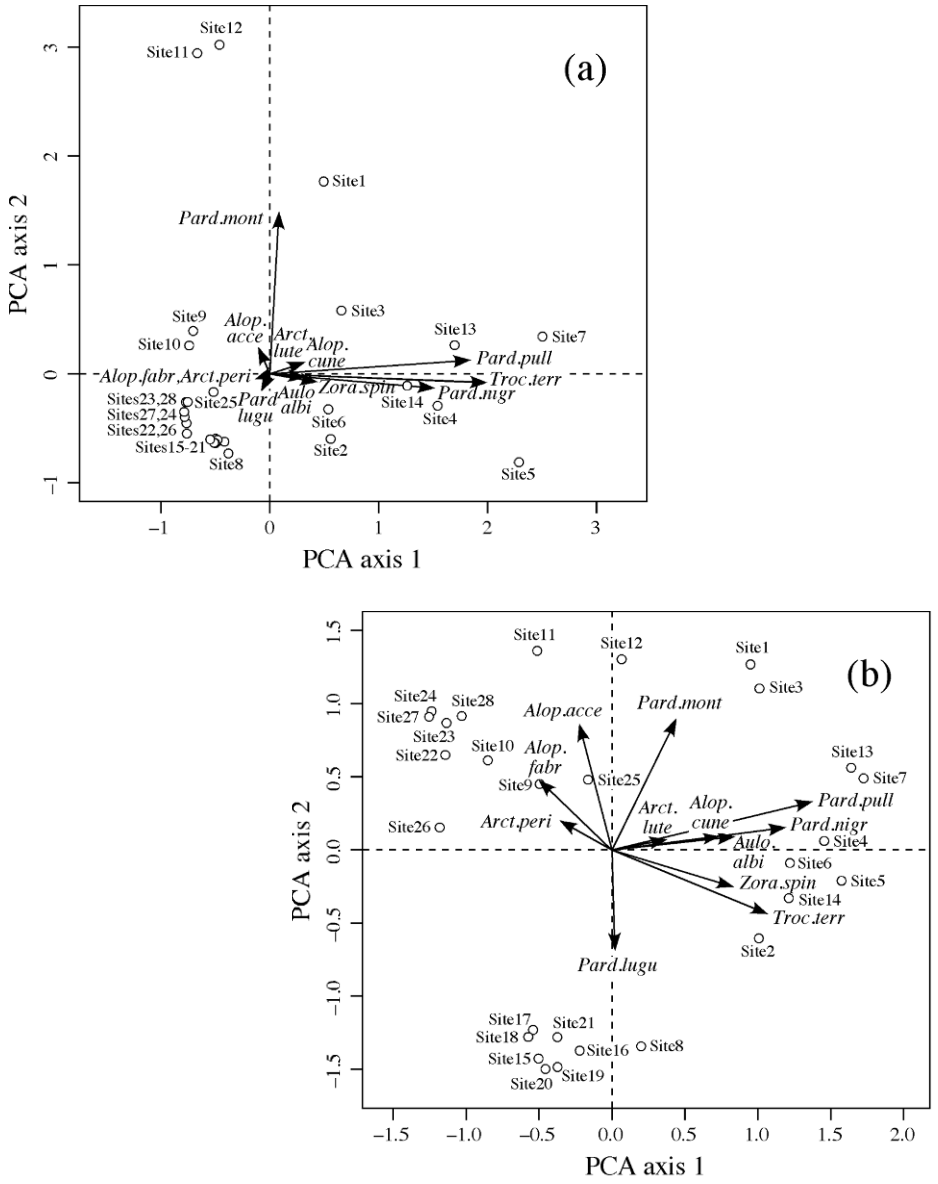


Figure 9.6 PCA correlation biplots of (a) the raw (untransformed) and (b) the log-transformed spider data (28 sites, 12 species). Scaling type 2 was used in both biplots to emphasize the covariances among species. The species are: *Alopecosa accentuata* (abbreviation: *Alop.acce*), *Alopecosa cuneata* (*Alop.cune*), *Alopecosa fabrilis* (*Alop.fabr*), *Arctosa lutetiana* (*Arct.lute*), *Arctosa perita* (*Arct.peri*), *Aulonia albimana* (*Aulo.albi*), *Pardosa lugubris* (*Pard.lugu*), *Pardosa monticola* (*Pard.mont*), *Pardosa nigriceps* (*Pard.nigr*), *Pardosa pullata* (*Pard.pull*), *Trochosa terricola* (*Troc.terr*) and *Zora spinimana* (*Zora.spin*).

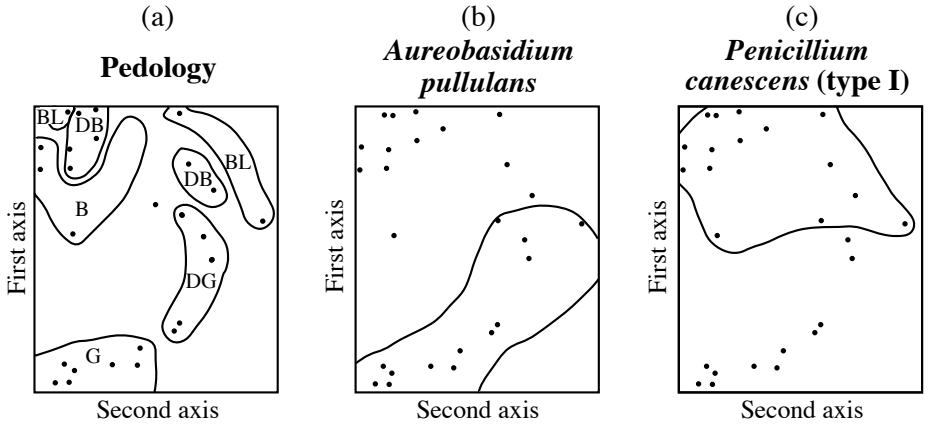


Figure 9.7 Principal component analysis computed from presence-absence of 51 soil microfungi. (a) Pedological information drawn on the ordination of the 26 sampling sites, plotted in the reduced space of the first two principal components. From north to south, soil types are: G = grey, DG = dark grey, BL = black, DB = dark brown, B = brown. (b) and (c) Distributions of the sites (envelopes) where two microflora species were present. Modified from Morrall (1974).

It is clear that Fig. 9.6b is easier to interpret than Fig. 9.6a. The reason is found by examining the "Cumulative fit per descriptor" table described in Subsection 9.1.3: in the raw data analysis, principal components I and II explained more than 60% of the variation for only four of the 12 species, whereas in the analysis of the log-transformed data, the first two principal components explained more than 60% of the variation for 10 of the 12 species.

Readers are invited to compare Fig. 9.6b to the results on the canonical analysis (RDA) in Fig. 11.7; the latter figure provides an interpretation of the site and species clusters using the environmental variables. Compare also the species clusters (species with small angles) in Fig. 9.6b to the species associations described in Ecological application 11.1b.

Ecological application 9.1b

A study of soil microfungi living in association with the aspen *Populus tremuloides* Michx. provides another type of utilization of principal component analysis. This study by Morrall (1974) covered 26 stations with 6 sites each, scattered throughout the Province of Saskatchewan (Canada). It evidenced relationships between the distributions of some species and soil types.

Among the 205 species or taxonomic entities that were identified, 51 were included in the ordination study. The others were not, considering criteria aimed at eliminating rare taxa which could have been either ephemeral constituents of the soil microflora or even contaminants of the laboratory cultures. Observations were transformed into presence-absence data.

Following principal component analysis, the 26 sampling sites were plotted in the reduced space of the first two principal components, onto which information about the nature of the soils was superimposed (Fig. 9.7a). Soils of Saskatchewan may be classified into 5 types, i.e. (G) the

grey wooded soils of the northern boreal forest, followed southward by the dark grey (DG) transition soils and the black soils (BL). Further south are dark brown soils (DB), which give way to the brown soils (B) of the grasslands. Since the principal components were computed from presence-absence data, the distribution of the sites in the reduced space is expected to reflect that of the fungus species. The author tested this for the most abundant species in the study, by plotting, in the reduced space, distributions of the sites where some fungus species were present; two examples are given in Figs. 9.7b and c. The author could then compare these distributions to that of the soil types.

9 — Algorithms

Three different methods are available for computing the eigenvalues and eigenvectors of a real, symmetric matrix, such as a covariance matrix **S**.

Householder

1. The most widely used method of eigen-decomposition is Householder reduction. This is the method implemented in function ***eigen()*** of R. It is very efficient for cases in which *all* eigenvalues and eigenvectors must be computed.

2. Clint & Jennings (1970) published a pioneering paper describing how to compute a *subset* only of the eigenvalues and corresponding eigenvectors of a real symmetric matrix, using an iterative method. Hill (1973b) used this idea to develop a "reciprocal averaging" algorithm for correspondence analysis; Hill's work will be further discussed in Section 9.2 on correspondence analysis. Building on these bases, ter Braak (1987c) proposed a *two-way weighted summation algorithm* (TWWS) for principal component analysis. This algorithm is described in detail here for three reasons: (1) it is closely linked to the basic equations of the PCA method, so that it may help readers understand them; (2) it is easy to program; (3) using it, one can compute the first few components only, when these are the ones of interest. The algorithm is summarized in Table 9.5.

TWWS algorithm

The numerical example worked out in Table 9.6 should help understand how the algorithm computes the principal components, the eigenvectors, and the eigenvalues. The data are those of the numerical example presented at the beginning of Section 9.1 and used in Subsections 9.1.1 to 9.1.4. The procedure starts with the matrix of centred data, $\mathbf{Y}_c$, shown in a box in the upper left-hand corner of Table 9.6.

To estimate principal component I, arbitrary scores are first assigned to the rows of the centred data matrix (Table 9.6, column R0); values $[f_{i1}] = [1\ 2\ 3\ 4\ 5]'$ are used here. Any other initial choice would lead to the same estimate for the first principal component $[f_{i1}]$ although the number of iterations necessary to reach it may differ. The only choice to avoid is to make all initial f_{i1} values equal. From these, column scores are found by multiplying the transpose of the data matrix by the row scores (Table 9.6, row C1):

$$[\text{column scores}_{1j}] = [y - \bar{y}]' [f_{i1}] \qquad \textbf{(9.17)}$$

Table 9.5 Two-way weighted summation (TWWS) algorithm for PCA. Modified from ter Braak (1987c).

a) Iterative estimation procedure

Step 1: Consider a table of n objects (rows) × p variables (columns). Centre each variable (column) on its mean.

Decide how many eigenvectors are needed and, for each one. **DO** the following:

Step 2: Take the row order as the arbitrary initial object scores (1, 2, …). Set the initial eigenvalue estimate to 0.

Iterative procedure begins

Step 3: Compute new variable loadings: $colscore(j) = \sum y(i,j) \times rowscore(i)$

Step 4: Compute new object scores: $rowscore(i) = \sum y(i,j) \times colscore(j)$

Step 5: For the second and higher-order axes, make the object scores uncorrelated with all previous axes (Gram-Schmidt orthogonalization procedure: see *b* below).

Step 6: Scale the vector of object scores to length 1 (normalization procedure *c*, below); obtain S.

Step 7: Upon convergence, the eigenvalue is $S/(n-1)$ where n is the number of objects. So, at the end of each iteration, $S/(n-1)$ provides an estimate of the eigenvalue. If this estimate does not differ from that of the previous iteration by more than a small quantity ("tolerance", set by the user), go to step 8. If the difference is larger than the tolerance value, go to step 3.

End of iterative procedure

Step 8: Normalize the eigenvector (variable loadings), i.e. scale it to length 1 (procedure *c*, below). Rescale the principal component (object scores) to variance = eigenvalue.

Step 9: If more eigenvectors are to be computed, go to step 2. If not, continue with step 10.

Step 10: Return the eigenvalue, % variance, cumulative % variance, eigenvector (variable loadings), and principal component (object scores).

b) Gram-Schmidt orthogonalization procedure

DO the following, in turn, for all previously computed principal components k:

Step 5.1: Compute the scalar product $SP = \sum [rowscore(i) \times v(i,k)]$ of the current object score vector estimate with previous component k, where vector $v(i,k)$ contains the object scores of component k, scaled to length 1. This product varies between 0 (if the vectors are orthogonal) and 1.

Step 5.2: Compute new values of $rowscore(i)$ such that vector *rowscore* becomes orthogonal to vector $v(i,k)$: $rowscore(i) = rowscore(i) - (SP \times v(i,k))$.

c) Normalization procedure

Step 6.1: Compute the sum of squares of the object scores: $S^2 = \sum rowscore(i)^2$, and the length $S = \sqrt{S^2}$.

Step 6.2: Compute the normalized object scores: $rowscore(i) = rowscore(i)/S$.

Subscript 1 designates the first iteration. At the end of the iteration process, the column scores will provide estimates of the first column of matrix **U**. The rationale for this operation comes from the basic equation of eigenanalysis (eq. 2.27) applied to matrix **S**:

$$\mathbf{S\,U = U\,\Lambda}$$

Replacing **S** by its value in the definition of the covariance matrix (eq. 4.6), $\mathbf{S} = (n-1)^{-1}\,[y-\bar{y}]'\,[y-\bar{y}]$, one obtains:

$$[y-\bar{y}]'\,[y-\bar{y}]\,\mathbf{U} = (n-1)\,\mathbf{U}\,\mathbf{\Lambda}$$

Since $\mathbf{F} = [y-\bar{y}]\,\mathbf{U}$ (eq. 9.4), it follows that:

$$[y-\bar{y}]'\,\mathbf{F} = (n-1)\,\mathbf{U}\,\mathbf{\Lambda}$$

Hence, the column scores obtained from eq. 9.17 are the values of the first eigenvector (first column of matrix **U**) multiplied by eigenvalue λ_1 (which is the first diagonal element of matrix **Λ**) and by $(n-1)$.

From the first estimate of column scores, a new estimate of row scores is computed using eq. 9.4, $\mathbf{F} = [y-\bar{y}]\,\mathbf{U}$:

$$[\text{row scores}_{i1}] = [y-\bar{y}]\,[u_{i1}] \qquad \textbf{(9.18)}$$

The algorithm alternates between estimating row scores and column scores until convergence. At each step, the row scores (columns called R in Table 9.6) are scaled to length 1 in order to prevent the scores from becoming too large for the computer to handle, which they may easily do. Before this normalization, the length of the row score vector, divided by $(n-1)$, provides the current estimate of the eigenvalue. This length actually measures the amount of "stretching" the row score vector has incurred during an iteration.

This description suggests one of several possible stopping criteria (Table 9.5, step 7): if the estimate of the eigenvalue has not changed, during the previous iteration, by more than a preselected tolerance value, the iteration process is stopped. Tolerance values between 10^{-10} and 10^{-12} produce satisfactory estimates when computing all the eigenvectors of large matrices, whereas values between 10^{-6} and 10^{-8} are sufficient to compute only the first two or three eigenvectors. Another possible stopping criterion would be a minimum percentage of change in the estimate of the eigenvalue.

At the end of the iterative estimation process (Table 9.5, step 8),

- the eigenvector (Table 9.6, line C13) is normalized (i.e. scaled to unit length), and
- the principal component is scaled to length $\sqrt{(n-1)\,\lambda_1}$. This makes its variance equal to its eigenvalue.

Table 9.6 Estimation of axes I (top) and II (bottom) for the centred data of the numerical example (values in boxes), using the "two-way weighted summation" algorithm (TWWS, Table 9.5). Iterations 1 to 13: estimates of the row scores (R1 to R13) and column scores (C1 to C13).

Objects ⇓	Var. 1	Var. 2	R0 (arbitrary)	R1	R1 length=1	R2	R2 length=1	R3	R3 … R9 length=1	R9 … R13 length=1	R13 length=1	R13 scaled (var=λ)
$\mathbf{x}_1$	**–3.2**	**–1.6**	1	–64.000	–0.586	–21.103	–0.593	–21.352	–0.595	–21.466	–0.596	–3.578
$\mathbf{x}_2$	**–2.2**	**1.4**	2	–34.000	–0.311	–9.745	–0.274	–9.037	–0.252	–8.080	–0.224	–1.342
$\mathbf{x}_3$	**–0.2**	**–2.6**	3	–14.000	–0.128	–6.082	–0.171	–6.977	–0.195	–8.019	–0.224	–1.341
$\mathbf{x}_4$	**1.8**	**3.4**	4	46.000	0.421	16.633	0.467	17.653	0.492	18.752	0.522	3.130
$\mathbf{x}_5$	**3.8**	**–0.6**	5	66.000	0.605	20.297	0.570	19.713	0.550	18.813	0.523	3.131
	Estimates of $\lambda_1 \Rightarrow$				27.295		8.895		8.967		9.000	
C1	18.000	4.000										
C2	5.642	1.905										
C3	5.544	2.257										
C4	5.473	2.449										
C5	5.428	2.554										
C6	5.401	2.612										
C7	5.386	2.644										
C8	5.377	2.661										
C9	5.373	2.671										
C10	5.370	2.677										
C11	5.368	2.680										
C12	5.368	2.681										
C13	5.367	2.682										
C13 length=1	0.895	0.447										

Objects ⇓	Var. 1	Var. 2	R0 (arbitrary)	R1	R1 ortho*	R1 length=1	R2	R2 ortho*	R2 length=1	R2 scaled (var=λ)
$\mathbf{x}_1$	**–3.2**	**–1.6**	1	–64.000	0.001	0.000	0.002	0.001	0.000	0.000
$\mathbf{x}_2$	**–2.2**	**1.4**	2	–34.000	–9.995	–0.500	–9.999	–10.000	–0.500	–2.236
$\mathbf{x}_3$	**–0.2**	**–2.6**	3	–14.000	9.996	0.500	10.001	10.000	0.500	2.236
$\mathbf{x}_4$	**1.8**	**3.4**	4	46.000	–9.996	–0.500	–10.002	–10.001	–0.500	–2.236
$\mathbf{x}_5$	**3.8**	**–0.6**	5	66.000	9.994	0.500	9.998	9.999	0.500	2.236
	Estimates of $\lambda_2 \Rightarrow$					4.998			5.000	
C1	18.000	4.000								
C2	2.000	–4.000								
C2 length=1	0.447	–0.894								

* ortho: scores are made orthogonal to R13 found in the upper portion of the table.

Note that the eigenvalues, eigenvectors, and principal components obtained using this iterative procedure and shown in Table 9.6 are the same as in Subsections 9.1.1 and 9.1.2, except for the signs of the second eigenvector and principal component, which are all changed in this example. One may arbitrarily change all signs of an eigenvector and the corresponding principal component, since signs result from an arbitrary decision made when computing the eigenvectors (Section 2.9). This is equivalent to turning the ordination diagram by 180° if signs are changed on both the first and second principal components, or looking at it from the back of the page, or in a mirror if signs are changed for one axis only.

To estimate the second principal component, eigenvalue, and eigenvector, row scores are again assigned arbitrarily at the beginning of the iterative process. In Table 9.6 (bottom part), the same values were actually chosen as for axis I, as stated in step 2 of the algorithm (Table 9.5). Iterations proceed in the same way as above, with the exception that, during each iteration, the row scores are made orthogonal to the final estimate obtained for the first principal component (column R13 in the upper portion of Table 9.6). This follows from the basic rule that principal components must be linearly independent (i.e. orthogonal) of one another. For the third and following principal axes, the vectors estimating row scores are made orthogonal, in turn, to *all* previously computed principal components.

The algorithm converges fairly rapidly, even with small tolerance values. For the example of Table 9.6, it took 13 iterations to reach convergence for axis I, and 2 iterations only for axis II, using a tolerance value of 10^{-6}. With a tolerance value of 10^{-10}, it took 21 and 2 iterations, respectively. The initial, arbitrary values assigned to the row scores also have an (unpredictable) effect on the number of iterations; e.g. with a different set of initial values [2 5 4 3 1], it took 14 iterations instead of 13 to reach convergence for the first axis (tolerance = 10^{-6}).

Supplementary object and variable

Supplementary objects or variables may easily be incorporated in the calculations using this algorithm. These are objects or variables that have not been used to compute the eigenvalues and eigenvectors of the ordination space, but whose positions are sought with respect to the original set of objects and variables that were used to compute the eigenvalues and eigenvectors. In Ecological application 9.1a for example, where the principal component analysis was computed using the species abundance data, the environmental descriptors used in Ecological application 11.1b could have been added to the analysis as supplementary variables. In addition, since there were 100 pitfall traps of spider observations, the traps that were excluded from the analysis could have been added to the ordination plot as supplementary objects. Preliminary transformations are required: (1) the supplementary variables must be centred on their respective means; (2) for each variable used in the original PCA (e.g. the 12 spider species), supplementary objects must be centred using the mean value of that variable calculated for the original set of objects. When the algorithm has reached convergence for an axis using the original set of objects, it is a simple matter to compute the column scores of the supplementary variables using eq. 9.17 and the row scores of the supplementary objects using eq. 9.18. The final step consists in applying to the

supplementary variable scores the scaling that was applied to the terms of the eigenvector corresponding to the original set of variables and, to the supplementary object scores, the scaling that was applied to the original set of objects.

SVD

3. Another way of computing principal components involves *singular value decomposition* (SVD, Section 2.11). SVD is also a widely used approach to compute correspondence analysis (Section 9.2).

The relationship with principal component analysis is the following. Centre the column vectors of $\mathbf{Y}$ on their respective means, forming matrix $\mathbf{Y}_c$, and compute the covariance matrix $\mathbf{S} = (n-1)^{-1}\mathbf{Y}_c'\mathbf{Y}_c$ (eq. 4.6). Carry out a singular value decomposition: $\mathbf{Y}_c = \mathbf{V}\ \mathbf{W}\ \mathbf{U}_{svd}'$ (eq. 2.31). The following reasoning will show that matrix $\mathbf{U}$ produced by SVD, $\mathbf{U}_{svd}$, is equal to matrix $\mathbf{U}$ computed by eigen-decomposition, $\mathbf{U}_{eigen}$. Use these SVD result to reconstruct $\mathbf{S}$:

$$\mathbf{S} = \frac{1}{n-1}\mathbf{Y}_c'\mathbf{Y}_c = \frac{1}{n-1}(\mathbf{U}_{svd}\mathbf{W}'\ \mathbf{V}')(\mathbf{V}\ \mathbf{W}\ \mathbf{U}_{svd}')$$

Since $\mathbf{V}$ is orthonormal (Section 2.11), $\mathbf{V}'\mathbf{V} = \mathbf{I}$ and one obtains:

$$\mathbf{S} = \frac{1}{n-1}\mathbf{Y}_c'\mathbf{Y}_c = \frac{1}{n-1}\mathbf{U}_{svd}\ \mathbf{W}'\mathbf{W}\ \mathbf{U}_{svd}' \qquad \textbf{(9.19)}$$

In the theory of eigen-decomposition, eq. 2.28 states that $\mathbf{S} = \mathbf{U}_{eigen}\mathbf{\Lambda}\mathbf{U}_{eigen}^{-1}$. Because $\mathbf{U}_{eigen}$ is orthonormal, $\mathbf{U}_{eigen}^{-1} = \mathbf{U}_{eigen}'$ (property 7 of inverses, Section 2.8) and the equation can be rewritten:

$$\mathbf{S} = \mathbf{U}_{eigen}\mathbf{\Lambda}\mathbf{U}_{eigen}'$$

Combining the latter equation with eq. 9.19 shows that

$$\mathbf{U}_{eigen}\mathbf{\Lambda}\mathbf{U}_{eigen}' = \mathbf{U}_{svd}\left(\frac{1}{n-1}\mathbf{W}'\mathbf{W}\right)\ \mathbf{U}_{svd}'$$

hence: $\mathbf{U}_{eigen} = \mathbf{U}_{svd}$ and $\mathbf{\Lambda} = \frac{1}{n-1}\mathbf{W}'\mathbf{W} = \frac{1}{n-1}[w_j^2]$ **(9.20)**

These correspondences can readily be verified, for the numerical example data, by singular value decomposition of the matrix of centred data $\mathbf{Y}_c$:

$$\begin{matrix}\mathbf{Y}_c & = & \mathbf{V} & \mathbf{W} & \mathbf{U}_{svd}'\end{matrix}$$

$$\begin{bmatrix}-3.2 & -1.6\\ -2.2 & 1.4\\ -0.2 & -2.6\\ 1.8 & 3.4\\ 3.8 & -0.6\end{bmatrix} = \begin{bmatrix}-0.5963 & 0.0000\\ -0.2236 & 5.0000\\ -0.2236 & -5.0000\\ 0.5217 & 5.0000\\ 0.5217 & -5.0000\end{bmatrix}\begin{bmatrix}6.0000 & 0\\ 0 & 4.4721\end{bmatrix}\begin{bmatrix}0.8944 & 0.4472\\ -0.4472 & 0.8944\end{bmatrix}$$

(As for eigenvalue decomposition, different SVD functions can revert the signs of some columns of **V** and **U**.) One can check that the squared singular values divided by $(n-1)$ are the eigenvalues, $\lambda_1 = 9$ and $\lambda_2 = 5$, and that $\mathbf{U}_{svd} = \mathbf{U}_{eigen}$ computed in Subsection 9.1.1.

Matrix **G**, which gives the object positions in the correlation biplot (scaling 2), is obtained from **V** as follows:

$$\mathbf{G} = \sqrt{n-1}\ \mathbf{V} \qquad \textbf{(9.21)}$$

Matrix **F**, which gives the object positions in the distance biplot (scaling 1), can be computed from **V** in two different ways:

$$\mathbf{F} = \mathbf{VW} \qquad \text{or} \qquad \mathbf{F} = \sqrt{n-1}\ \mathbf{V}\boldsymbol{\Lambda}^{1/2} \qquad \textbf{(9.22)}$$

When there are as many, or more variables than there are objects (i.e. $p \geq n$, for example in species-rich communities), eigenvalues and eigenvectors can still be computed using any of the three methods described above: Householder reduction, the TWWS algorithm, or singular value decomposition. The covariance matrix is positive semidefinite in such cases, so that null eigenvalues are produced (Table 2.2). When p is much larger than n and all eigenvalues and eigenvectors must be computed, important savings in computer time can be made by applying Householder reduction or singular value decomposition to the cross-product matrix [**YY**'], which is of size $(n \times n)$, instead of [**Y**'**Y**]* which is of size $(p \times p)$ and is thus much larger; **Y** is centred by columns. The eigenvalues of [**YY**'] are the same as the non-zero eigenvalues of [**Y**'**Y**]. Matrix **U** of the eigenvectors of [**Y**'**Y**] can be found from matrix **V** of the eigenvectors of [**YY**'] using the transformation $\mathbf{U} = \mathbf{Y}'\mathbf{V}\boldsymbol{\Lambda}^{-1/2}$. Matrix **F** of the principal components is found from the equation $\mathbf{F} = \mathbf{V}\boldsymbol{\Lambda}^{1/2}$.

Negative eigenvalues may occur in principal component analysis due to the handling of missing values. Pairwise deletion of missing data (Subsection 1.6.2), in particular, creates covariances computed with different numbers of degrees of freedom; this situation can make the covariance matrix indefinite (Table 2.2). A Householder algorithm should be used in such a case because negative eigenvalues come out of SVD as positive singular values (Section 2.11, Application 2).

10 — Metric ordination of community composition data

Different approaches are available to obtain metric ordinations of community composition (species) data (Fig. 9.8): the classical ordination approaches (PCA, this section, and CA, Section 9.2), the transformation-based PCA (tb-PCA), and the

* Matrix **S** differs from the cross-product matrix [**Y**'**Y**] by the division of the cross-products by $(n-1)$ in **S**. The eigenvalues of [**Y**'**Y**] are larger than those of **S** by this factor $(n-1)$, but the eigenvectors are identical.

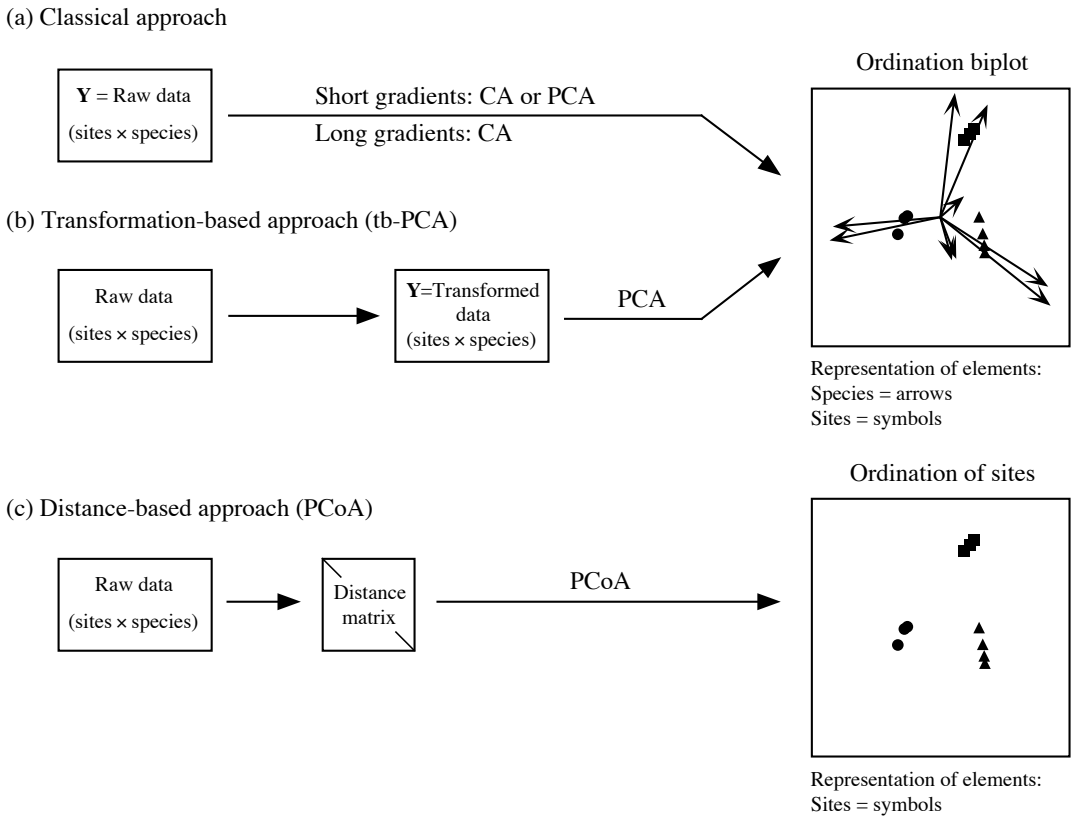


Figure 9.8 Different approaches are available for metric ordination of community composition data: (a) classical PCA and CA, (b) the transformation-based approach, and (c) the distance-based approach (PCoA). Metric ordination methods produce ordinations that fully preserve the distances among sites, as specified in Table 9.1. Modified from Legendre & Gallagher (2001).

distance-based method of principal coordinate analysis (PCoA, Section 9.3). These metric ordination methods produce ordinations that fully preserve the distances among sites. They are discussed here in turn. The distance preserved by each method is specified in Table 9.1. The non-metric method of nMDS (Section 9.4) is not mentioned in Fig. 9.8 because the ordinations that it produces distort the distances among sites.

In the classical approach (Fig. 9.8a), the species-environment relationship is analysed by PCA (this Section) or by CA (Section 9.2). In the early applications of PCA to community ecology, CA was considered preferable to PCA for species data tables sampled in highly diversified regions ("long gradients") because these tables contain many zeros. This is the case, for example, when sampling communities along

extensive spatial or temporal gradients, where the species composition may change greatly along the gradient. For groups of sites that were fairly homogeneous in species composition ("short gradients"), PCA was considered appropriate. A wider array of options is now available.

PCA can be made to preserve some distance that is appropriate for the study of composition data in highly diversified regions, e.g. along gradients, instead of the Euclidean distance D_1 (Fig. 9.8b). Composition data can be transformed using the transformations described in Section 7.7, leading to the transformation-based PCA, or tb-PCA, approach. PCA computed on data transformed using these equations will actually preserve the chord, profile, Hellinger, or chi-square distance, or the chi-square metric among sites, depending on the transformation used. Note that the corresponding distances (D_3 and D_{15} to D_{18} in Table 7.3) have the property of being Euclidean.

tb-PCA

One can also (Fig. 9.8c) compute directly one of the distance functions appropriate for community composition data (Table 7.4) and carry out a principal coordinate analysis (PCoA, Section 9.3) of the distance matrix to obtain an ordination. This is the distance-based approach. PCoA obtains metric ordinations from **D** matrices, whereas nonmetric multidimensional scaling (nMDS, Section 9.4) produces non-metric ordinations that distort the distances among sites. These methods should be used in analyses involving distance functions that cannot be obtained by a data transformation followed by PCA (tb-PCA approach, Fig. 9.8b). Among the distances developed specifically for species data (Table 7.4) are most of the coefficients designed for binary data, e.g. Jaccard ($\sqrt{1 - S_7}$) and Sørensen (D_{13} or $\sqrt{1 - S_8}$), as well as quantitative distance measures like the asymmetric Gower coefficient ($\sqrt{1 - S_{19}}$), the geodesic metric (D_4), Whittaker (D_9), Canberra (D_{10}), Clark (D_{11}), percentage difference (D_{14}), and mean character difference modified for species data D_{19}.

9.2 Correspondence analysis (CA)

Correspondence analysis (CA) was developed independently by several authors. It was first proposed for the analysis of contingency tables by Hirschfeld (1935), Fisher (1940), Benzécri (1969), and others. In a historical review of the subject, Nishisato (1980) traces its origin back to 1933. It was applied in ecology to the analysis of sites × species tables by Roux & Roux (1967), Hatheway (1971), Ibanez & Séguin (1972), Hill (1973b, 1974), Orlóci (1975), and others. Its use was generalized to other types of data tables by Benzécri and his collaborators (Escofier-Cordier, 1969; Benzécri and coll., 1973). Other important books on correspondence analysis are those of Nishisato (1980), Greenacre (1983, 2007), ter Braak (1988), and van Rijckevorsel & de Leeuw (1988). In the course of its history, the method was successively designated under the English names *contingency table analysis* (Fisher, 1940), RQ-*technique* (Hatheway, 1971), *reciprocal averaging* (Hill, 1973b), *correspondence analysis* (Hill, 1974), *reciprocal ordering* (Orlóci, 1975), *dual scaling* (Nishisato,

1980), and *homogeneity analysis* (Meulman, 1982), while it is known in French as *analyse factorielle des correspondances* (Cordier, 1965; Escofier-Cordier, 1969).

Contingency table

Correspondence analysis was first proposed for analysing two-way contingency tables (Section 6.2). In such tables, the states of a first descriptor (rows) are compared to the states of a second descriptor (columns). Data in each cell of the table are

Frequencies

frequencies, i.e. numbers of objects coded with a combination of states of the two descriptors. These frequencies are positive integers or zeros. The most common application of CA in ecology is the analysis of community composition (species presence-absence or abundance values) at sampling sites (Subsection 9.2.4). The rows and columns of the data table then correspond to sites and species, respectively. Such a table is analogous to a contingency table because the data are frequencies.

In general, correspondence analysis can be applied to any data table that is dimensionally homogeneous, meaning that the physical dimensions of all variables are the same (Chapter 3), and that does not contain negative values (i.e. only positive integers or zeros are allowed). The values have to be additive in rows and columns (additivity: see Subsection 1.4.2) to allow computation of row and column sums and transformation of the data table into matrix $\overline{\mathbf{Q}}$ (eq. 9.24). Frequency data have these characteristics. The χ^2 distance (D_{16}, eq. 7.55), which is a coefficient that excludes double-zeros, is used to quantify the relationships among rows and columns in CA (Table 9.1).

Correspondence analysis can also be conducted on contingency tables that compare two *groups* of descriptors. The method is then called *multiple correspondence*

Multiple corresp. analysis

analysis (MCA). For example, the rows of the table could be different species, each divided into a few abundance classes, and the columns, different descriptors of the physical environment with, for each, a number of columns equal to the number of its states. Each site (object) then contributes to several frequencies of the table, but this does not invalidate the results because of the transformations described in the next subsection. The analysis can be done using a standard CA function. Special programs and R functions also exist for MCA, which is not described further in this section. For a table of species × environmental variables, a better way of comparing species to environmental data is canonical correspondence analysis (CCA, Section 11.2), which does not require that the species and environmental data be recoded into a few classes.

Correspondence analysis is primarily an ordination method. As such, it is similar to principal component analysis; it preserves, in the space of the principal axes (i.e. after rotation), the Euclidean distance between *profiles of weighted conditional probabilities*. This is equivalent to preserving the χ^2 distance (D_{16}, eq. 7.55) between the rows or columns of the contingency table. The relationships between correspondence analysis and principal component analysis will be further described in the next subsections.

1 — Computation

This description of correspondence analysis will proceed in three steps. (1) The contingency (or community composition) table will be transformed into a table of contributions to the Pearson chi-square statistic after fitting a null model to the frequency data. (2) The transformed data table will be decomposed to obtain the eigenvalues and eigenvectors, as in PCA. (3) Further matrix operations will lead to the various tables needed for plotting useful diagrams. Besides its role as an ordination method, CA may be used for studying the proximities between the rows (or the columns) of a contingency table, as well as the correspondence between rows and columns as in Section 6.4.

Consider a contingency table with r rows and c columns, as in Section 6.2. Assume that the table is written in such a way that $r \geq c$; the table may be transposed to meet this condition, since the rows and columns of a contingency table play identical roles. The symbolism is as follows:

- Absolute frequencies are represented by f_{ij} and relative frequencies ("probabilities" or "proportions") by p_{ij}.

- p_{ij} is the frequency f_{ij} in cell ij divided by the sum f_{++} of the f_{ij}'s over the whole table. The table containing the relative frequencies p_{ij} is called **Q**; its size is (r×c).

- The weight attached to row i is $p_{i+} = f_{i+}/f_{++}$, where f_{i+} is the sum of the values in row i. Vector $[p_{i+}]$ is of size r = number of rows.

- Likewise, the weight attached to column j is $p_{+j} = f_{+j}/f_{++}$, where f_{+j} is the sum of values in column j. Vector $[p_{+j}]$ is of size c = number of columns.

The computation steps are as follows:

1. *Transform the data table*. — The Pearson chi-square statistic, χ^2_P (eq. 6.5), is a sum of squared χ_{ij} values, computed for every cell ij of the contingency table. Each χ_{ij} value is the standardized residual of a frequency f_{ij} after fitting a null model to the contingency table. The null model states that there is no relationship between the rows and columns of the table (eq. 6.4). Simple algebra shows that the component of χ_{ij} for each cell (eq. 6.26) is:

Component of chi-square

$$\chi_{ij} = \frac{O_{ij} - E_{ij}}{\sqrt{E_{ij}}} = \sqrt{f_{++}} \left[\frac{p_{ij} - p_{i+}p_{+j}}{\sqrt{p_{i+}p_{+j}}} \right] \qquad \textbf{(9.23)}$$

Correspondence analysis is based upon a matrix called $\bar{\mathbf{Q}}$ (r×c) in this book:

$$\bar{\mathbf{Q}} = [\bar{q}_{ij}] = \left[\frac{p_{ij} - p_{i+}p_{+j}}{\sqrt{p_{i+}p_{+j}}} \right] \qquad \textbf{(9.24)}$$

Values $\bar{q}_{ij}$, which are at the basis of correspondence analysis, only differ from the χ_{ij} values by the numerical constant $\sqrt{f_{++}}$: $\bar{q}_{ij} = \chi_{ij} / \sqrt{f_{++}}$. This difference causes all the eigenvalues to be smaller than or equal to 1, as shown below. Values $\bar{q}_{ij}$ can also be calculated directly from the f_{ij}'s:

$$\bar{q}_{ij} = \frac{f_{ij} f_{++} - f_{i+} f_{+j}}{f_{++} \sqrt{f_{i+} f_{+j}}} \qquad \textbf{(9.25)}$$

Total inertia

The sum of squares of all values in matrix $\bar{\mathbf{Q}}$, $\sum \bar{q}_{ij}^2$, measures the *total inertia* in $\bar{\mathbf{Q}}$. It is also equal to the sum of all eigenvalues to be extracted by eigenanalysis of $\bar{\mathbf{Q}}$.

SVD

2. *Decomposition of* $\bar{\mathbf{Q}}$. — Singular value decomposition (SVD, eq. 2.31) can be applied to matrix $\bar{\mathbf{Q}}$, with the following result (the symbolism is slightly modified compared to Section 2.11):

$$\bar{\mathbf{Q}}\,(r \times c) = \hat{\mathbf{U}}\,(r \times c)\ \mathbf{W}(\text{diagonal}, c \times c)\ \mathbf{U}'(c \times c) \qquad \textbf{(9.26)}$$

where both $\mathbf{U}$ and $\hat{\mathbf{U}}$ are orthonormal matrices (i.e. matrices containing column vectors that are normalized and orthogonal to one another; Section 4.4) and $\mathbf{W}$ is a diagonal matrix $\mathbf{D}(w_i)$. The diagonal values w_i in $\mathbf{W}$, which are all non-negative, are the singular values of $\bar{\mathbf{Q}}$.

Because $\bar{\mathbf{Q}} = \hat{\mathbf{U}}\,\mathbf{W}\mathbf{U}'$ (eq. 9.26), the multiplication $\bar{\mathbf{Q}}'\bar{\mathbf{Q}}$ gives the following result:

$$\bar{\mathbf{Q}}'\bar{\mathbf{Q}}\,(c \times c) = \mathbf{U}\mathbf{W}'\,(\hat{\mathbf{U}}'\hat{\mathbf{U}})\,\mathbf{W}\,\mathbf{U}' \qquad \textbf{(9.27)}$$

Since $\hat{\mathbf{U}}$ is orthonormal, $\hat{\mathbf{U}}'\hat{\mathbf{U}} = \hat{\mathbf{U}}\hat{\mathbf{U}}' = \mathbf{I}$, hence:

$$\bar{\mathbf{Q}}'\bar{\mathbf{Q}} = \mathbf{U}\,\mathbf{W}'\mathbf{W}\,\mathbf{U}' \qquad \textbf{(9.28)}$$

Equation 2.28 shows that the eigenvalues (forming diagonal matrix $\mathbf{\Lambda}$) and eigenvectors (matrix $\mathbf{U}$) of a square matrix $\mathbf{A}$ obey the relationship:

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$$

If the vectors in $\mathbf{U}$ are normalized, as they are here, $\mathbf{U}$ is an orthonormal matrix with the property $\mathbf{U}^{-1} = \mathbf{U}'$. As a consequence, eq. 2.28 may be rewritten as

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}' \qquad \textbf{(9.29)}$$

It follows that the diagonal matrix $[\mathbf{W}'\mathbf{W}]$, which contains squared singular values on its diagonal, is the diagonal matrix $\mathbf{\Lambda}(c \times c)$ of the eigenvalues of $\bar{\mathbf{Q}}'\bar{\mathbf{Q}}$. Similarly, the orthonormal matrix $\mathbf{U}$ of eqs. 9.27 and 9.28 is the same as matrix $\mathbf{U}$ of eq. 9.29; it is the matrix of eigenvectors of $\bar{\mathbf{Q}}'\bar{\mathbf{Q}}\,(c \times c)$, containing the loadings of the *columns* of the contingency table. A similar reasoning applied to matrix $\bar{\mathbf{Q}}\bar{\mathbf{Q}}'\,(r \times r)$ shows that the

orthonormal matrix $\hat{\mathbf{U}}$ produced by singular value decomposition is the matrix of eigenvectors of $\overline{\mathbf{Q}}\overline{\mathbf{Q}}'$, containing the loadings of the *rows* of the contingency table.

The relationship between eq. 9.26 and eigenvalue decomposition (eq. 2.22) is the same as in principal component analysis (Subsection 9.1.9). Prior to eigenvalue decomposition, a square matrix of sums of squares and cross products $\overline{\mathbf{Q}}'\overline{\mathbf{Q}}$ is computed. This is similar to using the matrix of sums of squares and cross products $\mathbf{Y}'\mathbf{Y}$ for eigenvalue decomposition in PCA; $\mathbf{Y}'\mathbf{Y}$ is the covariance matrix $\mathbf{S}$ multiplied by the constant $(n - 1)$. In PCA, matrix $\mathbf{Y}$ was centred on the column means prior to computing $\mathbf{Y}'\mathbf{Y}$ whereas, in CA, matrix $\overline{\mathbf{Q}}$ is centred by the operation $(O_{ij} - E_{ij})$ (eqs. 9.23 and 9.24). In spite of this centring operation, the sums of the rows and columns of $\overline{\mathbf{Q}}$ are not equal to zero.

Eigen-analysis

Results identical to those of SVD would be obtained by applying eigenvalue decomposition (eqs. 2.22 and 9.1) either to the covariance matrix $\overline{\mathbf{Q}}'\overline{\mathbf{Q}}$, which would produce the matrices of eigenvalues $\mathbf{\Lambda}$ and eigenvectors $\mathbf{U}$, or to matrix $\overline{\mathbf{Q}}\overline{\mathbf{Q}}'$, which would provide the matrices of eigenvalues $\mathbf{\Lambda}$ and eigenvectors $\hat{\mathbf{U}}$. Actually, it is not necessary to repeat the eigenanalysis to obtain $\mathbf{U}$ and $\hat{\mathbf{U}}$, because:

$$\hat{\mathbf{U}}\,(r\times c) = \overline{\mathbf{Q}}\mathbf{U}\,\mathbf{\Lambda}^{-1/2} \qquad \textbf{(9.30)}$$

and

$$\mathbf{U}(c\times c) = \overline{\mathbf{Q}}'\hat{\mathbf{U}}\,\mathbf{\Lambda}^{-1/2} \qquad \textbf{(9.31)}$$

In the sequel, all matrices derived from $\mathbf{U}$ will be without a hat and all matrices derived from $\hat{\mathbf{U}}$ will bear a hat.

Singular value decomposition of matrix $\overline{\mathbf{Q}}$, or eigenvalue analysis of matrix $\overline{\mathbf{Q}}'\overline{\mathbf{Q}}$, always yields one null eigenvalue. This is due to the centring in eq. 9.24, where $(p_{i+}p_{+j})$ is subtracted from each value p_{ij}. The number of positive eigenvalues is $\min(r - 1, c - 1)$. Hence, when $r \geq c$, there are $(c - 1)$ positive eigenvalues. The part of matrix $\mathbf{U}$ that is considered for interpretation is of size $c \times (c - 1)$; likewise, the part of $\hat{\mathbf{U}}$ that is considered is of size $r \times (c - 1)$.

The analysis, by either SVD or eigenvalue decomposition, is usually performed on matrix $\overline{\mathbf{Q}}$ with $r \geq c$, for convenience. The reason is that not all SVD programs can handle matrices with $r < c$ (function ***svd()*** of R does not present that problem). In addition, when using eigenanalysis, the computation is shorter when performed on the smallest of the two possible covariance matrices, both solutions leading to identical results. If one proceeds from a matrix such that $r < c$, the first $r - 1$ eigenvalues are the same as in the analysis of the transposed matrix, the remaining eigenvalues being zero.

Consider now the non-centred matrix $\tilde{\mathbf{Q}}\,(r\times c)$ in which $(p_{i+}p_{+j})$ is not subtracted from each term p_{ij} in the numerator:

$$\tilde{\mathbf{Q}} = [\tilde{q}_{ij}] = \left[\frac{p_{ij}}{\sqrt{p_{i+}p_{+j}}}\right] = \left[\frac{f_{ij}}{\sqrt{f_{i+}f_{+j}}}\right] \qquad \textbf{(9.32)}$$

What happens if the analysis is based on matrix $\tilde{\mathbf{Q}}$ instead of $\overline{\mathbf{Q}}$ (eq. 9.24)? The only difference is that decomposition of $\tilde{\mathbf{Q}}$ produces one extra eigenvalue; all the other results are identical. This extra eigenvalue is easy to recognize because its value is 1 in correspondence analysis. This eigenvalue is meaningless because it only reflects the distance between the centre of mass of the data points in the ordination space and the origin of the system of axes. In other words, it reflects the lack of centring of the scatter of points on the origin (Hill, 1974); it explains none of the dispersion (Lebart & Fénelon, 1971). There are computer programs that do not make the centring; in that case, the first eigenvalue ($\lambda_1 = 1$) and eigenvector must be discarded. All programs that carry out the calculations on matrix $\overline{\mathbf{Q}}$ produce one eigenvalue less than $\min[r, c]$; if the data table $\mathbf{Q}$ is such that $r \geq c$, correspondence analysis yields $(c - 1)$ non-null and positive eigenvalues.

Alternatively, what happens if the analysis is based on the matrix of χ_{ij} values (eq. 9.23) instead of matrix $\overline{\mathbf{Q}}$? Since values $\chi_{ij} = \sqrt{f_{++}}\ \bar{q}_{ij}$, it follows that the total variance in matrix $[\chi_{ij}]$ is larger than that of matrix $\overline{\mathbf{Q}}$ by a factor $(\sqrt{f_{++}})^2 = f_{++}$; hence, all eigenvalues obtained by analysing matrix $[\chi_{ij}]$ are larger than those of $\overline{\mathbf{Q}}$ by a factor f_{++}. The normalized eigenvectors in matrices $\mathbf{U}$ and $\hat{\mathbf{U}}$ remain unaffected. When the analysis is carried out on matrix $\overline{\mathbf{Q}}$, all eigenvalues are smaller than or equal to 1, which is convenient.

Biplot

3. *Compute matrices for biplots*. — Matrices $\mathbf{U}$ and $\hat{\mathbf{U}}$ may be used to plot the positions of the row and column vectors in two separate scatter diagrams. For *biplots*, which are joint plots of the rows and column vectors, various scalings have been proposed. First, matrices $\mathbf{U}$ and $\hat{\mathbf{U}}$ can be weighted by the inverse of the square roots of the column and row weights, written out in diagonal matrices $\mathbf{D}(p_{+j})^{-1/2}$(size $c \times c$) and $\mathbf{D}(p_{i+})^{-1/2}$(size $r \times r$), respectively:

$$\mathbf{V}(c \times c) = \mathbf{D}(p_{+j})^{-1/2}\ \mathbf{U} \qquad \textbf{(9.33)}$$

$$\hat{\mathbf{V}}\ (r \times c) = \mathbf{D}(p_{i+})^{-1/2}\ \hat{\mathbf{U}} \qquad \textbf{(9.34)}$$

Discarding the null eigenvalue, the part of matrix $\mathbf{V}$ to consider for interpretation is of size $c \times (c - 1)$ and the part of matrix $\hat{\mathbf{V}}$ to consider is of size $r \times (c - 1)$.

Matrix $\mathbf{F}$, which gives the positions of the *rows* of the contingency table in the correspondence analysis space, is obtained from the transformed matrix of eigenvectors $\mathbf{V}$, which gives the positions of the *columns* in that space. This is done by applying the usual equation for component scores (eq. 9.4) to data matrix $\mathbf{Q}$, with division by the row weights:

$$\mathbf{F}(r \times c) = \hat{\mathbf{V}}\ \boldsymbol{\Lambda}^{1/2} \qquad \textbf{(9.35a)}$$

or

$$\mathbf{F}(r \times c) = \mathbf{D}(p_{i+})^{-1}\mathbf{Q}\mathbf{V} \qquad \textbf{(9.35b)}$$

In the same way, matrix $\hat{\mathbf{F}}$, which gives the positions of the *columns* of the contingency table in the correspondence analysis space, is obtained from the transformed matrix of eigenvectors $\hat{\mathbf{V}}$, which gives the positions of the *rows* in that space. The equation is the same as above, except that division here is by the column weights:

$$\hat{\mathbf{F}}\,(c\times c) = \mathbf{V}\boldsymbol{\Lambda}^{1/2} \qquad \textbf{(9.36a)}$$

or

$$\hat{\mathbf{F}}\,(c\times c) = \mathbf{D}(\mathrm{p}_{+j})^{-1}\mathbf{Q}'\hat{\mathbf{V}} \qquad \textbf{(9.36b)}$$

Discarding the null eigenvalue, the part of matrix **F** to consider for interpretation is of size $r \times (c-1)$ and the part of matrix $\hat{\mathbf{F}}$ to consider is of size $c \times (c-1)$. With this scaling, matrices **F** and **V** form a pair such that the rows (given by matrix **F**) are at the centroid (also called centre of mass, or "barycentre", from the Greek βαρυς, pronounced "barus", heavy) of the columns in matrix **V**. In the same way, matrices $\hat{\mathbf{F}}$ and $\hat{\mathbf{V}}$ form a pair such that the columns (given by matrix $\hat{\mathbf{F}}$) are at the centroids of the rows in matrix $\hat{\mathbf{V}}$. This property is illustrated in the numerical example below.

Biplots of the rows (e.g. sites) and columns (e.g. species) can be drawn using different combinations of the matrix scalings described above. Scaling types 1 and 2, described below, are the most commonly used by ecologists when analysing community composition data (ter Braak, 1990).

Scalings in CA

• Scaling type 1. — Draw a joint plot with the sites (matrix **F**) at the centroids of the species (matrix **V**). For sites × species data tables, this scaling is the most appropriate if one is primarily interested in representing the distance relationships among the sites because, in matrix **F**, the distances among sites are projections of their χ^2 distances (D_{16}) (ter Braak, 1987c; see Numerical example, Subsection 9.2.2).

• Scaling type 2. — Draw a joint plot with the species (matrix $\hat{\mathbf{F}}$) at the centroids of the sites (matrix $\hat{\mathbf{V}}$). For sites × species data tables, this scaling is the most appropriate if one is primarily interested in representing the distance relationships among the species because, in matrix $\hat{\mathbf{F}}$, the distances among species are projections of their χ^2 distances (see Numerical example, Subsection 9.2.2).

• Scaling type 3. — This is a compromise between scalings 1 and 2. This scaling, called "symmetric" in program CANOCO, does not preserve the chi-square distances among the species or among the site scores. It is obtained by drawing together matrices $\hat{\mathbf{V}}\,\boldsymbol{\Lambda}^{1/4}$ (or $\mathbf{F}\boldsymbol{\Lambda}^{-1/4}$) for sites and $\mathbf{V}\boldsymbol{\Lambda}^{1/4}$ (or $\hat{\mathbf{F}}\,\boldsymbol{\Lambda}^{-1/4}$) for species.

• Scaling type 4. — This scaling is useful in the correspondence analysis of a contingency table crossing two qualitative descriptors or two factors. Draw a joint plot using **F**, which preserves the chi-square distances among the rows, and $\hat{\mathbf{F}}$ which preserves the chi-square distances among the columns of the contingency table. This hybrid scaling correctly represents the chi-square distance relationships among the states of the two qualitative descriptors. In this scaling, the relative positions of the row

and column symbols *along each axis of the plot* are the same as in scaling 3. The range of axis k in scaling 3 multiplied by $\lambda_k^{1/4}$ gives the range of that axis in scaling type 4. The axes in scaling 4 are thus compressed compared to the corresponding axes in scaling 3 because the eigenvalues are always smaller than 1 in CA. The compression is not isotropic, however, because the eigenvalues differ among axes.

Other possible, but less often used scaling methods are discussed by ter Braak (1987c, 1990).

4. *Cumulative fit tables*. — Tables of cumulative fit for columns and rows can be computed in CA, as it was the case in PCA (Subsection 9.1.3.4). The sum of squares of the values in row j of matrix $\hat{\mathbf{F}}$ gives the total variance of column (descriptor) j in the multidimensional ordination space. The relative cumulative fit of descriptor j is found by computing the sum of squared values for axis 1, axes 1 and 2, axes 1, 2 and 3, and so on, and dividing it by the total variance of j. This statistic represents the fit of descriptor j in 1, 2, or more dimensions; it can be interpreted like a coefficient of multiple determination (R^2).

The sum of squared values of row i of matrix $\mathbf{F}$ gives the squared length of the vector representing object i in the multidimensional ordination space. Use matrix $\mathbf{F}$ to compute the squared length of each object vector i in 1, 2, 3 … CA dimensions and divide these lengths by the total square length of object vector i. See example in the next subsection. Squared residual lengths can be computed by subtracting from the length of i the sum of squared values for axis 1, axes 1 and 2, axes 1, 2 and 3, etc.

2 — Numerical example

The following numerical example illustrates the calculations involved in correspondence analysis. This example assumes that three species have been observed in three lakes (Table 9.6)*. The justification for analysing community composition data by CA is provided in Subsection 9.2.4. The data table is of small size (3 × 3) to allow readers to follow or repeat the calculations easily.

Matrix $\mathbf{Q}$ contains the proportions p_{ij}, from which the marginal distributions of the rows and columns, p_{i+} and p_{+j}, are computed. The row and column identifiers are as in Table 9.6:

$$\mathbf{Q} = [p_{ij}] = \begin{array}{c} \\ L1 \\ L2 \\ L3 \end{array} \begin{array}{c} \begin{array}{ccc} Sp1 & Sp2 & Sp3 \end{array} \\ \begin{bmatrix} 0.10 & 0.10 & 0.20 \\ 0.10 & 0.15 & 0.10 \\ 0.15 & 0.05 & 0.05 \end{bmatrix} \end{array} \quad \begin{array}{c} [p_{i+}] \\ \begin{bmatrix} 0.40 \\ 0.35 \\ 0.25 \end{bmatrix} \end{array}$$

$$[p_{+j}] = \begin{bmatrix} 0.35 & 0.30 & 0.35 \end{bmatrix}$$

* Table 9.6 could also represent a contingency table crossing the states of two qualitative descriptors, to illustrate CA of a contingency table. A biplot of the results would use scaling type 4 (Subsection 9.2.1.3).

Table 9.6 Numerical example: a site-by-species data table.

	Species 1 (Sp1)	Species 2 (Sp2)	Species 3 (Sp3)	**Row sums**
Lake 1 (L1)	10	10	20	40
Lake 2 (L2)	10	15	10	35
Lake 3 (L3)	15	5	5	25
Column sums	35	30	35	100

Matrix $\overline{\mathbf{Q}}$ is computed with eq. 9.24:

$$\overline{\mathbf{Q}} = [\bar{q}_{ij}] = \left[\frac{p_{ij} - p_{i+}p_{+j}}{\sqrt{p_{i+}p_{+j}}}\right] = \begin{array}{c} \\ \text{L1} \\ \text{L2} \\ \text{L3} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Sp1} & \text{Sp2} & \text{Sp3} \end{array} \\ \begin{bmatrix} -0.10690 & -0.05774 & 0.16036 \\ -0.06429 & 0.13887 & -0.06429 \\ 0.21129 & -0.09129 & -0.12677 \end{bmatrix} \end{array}$$

and matrix $\tilde{\mathbf{Q}}$ with eq. 9.32:

$$\tilde{\mathbf{Q}} = [\tilde{q}_{ij}] = \left[\frac{p_{ij}}{\sqrt{p_{i+}p_{+j}}}\right] = \begin{array}{c} \\ \text{L1} \\ \text{L2} \\ \text{L3} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Sp1} & \text{Sp2} & \text{Sp3} \end{array} \\ \begin{bmatrix} 0.26726 & 0.28868 & 0.53452 \\ 0.28571 & 0.46291 & 0.28571 \\ 0.50709 & 0.18257 & 0.16903 \end{bmatrix} \end{array}$$

The eigenvalues of $\overline{\mathbf{Q}}'\overline{\mathbf{Q}}$ are $\lambda_1 = 0.09613$ (70.1%), $\lambda_2 = 0.04094$ (29.9%), and $\lambda_3 = 0$ (because of the centring). The first two eigenvalues are also eigenvalues of $\tilde{\mathbf{Q}}'\tilde{\mathbf{Q}}$, its third eigenvalue being 1 because $\tilde{\mathbf{Q}}$ is not centred (eq. 9.32). The normalized eigenvectors of $\overline{\mathbf{Q}}'\overline{\mathbf{Q}}$, corresponding to λ_1 and λ_2, are (in columns):

$$\mathbf{U} = \begin{array}{c} \\ \text{Sp1} \\ \text{Sp2} \\ \text{Sp3} \end{array} \begin{array}{c} \begin{array}{cc} (\lambda_1) & (\lambda_2) \end{array} \\ \begin{bmatrix} 0.78016 & -0.20336 \\ -0.20383 & 0.81145 \\ -0.59144 & -0.54790 \end{bmatrix} \end{array}$$

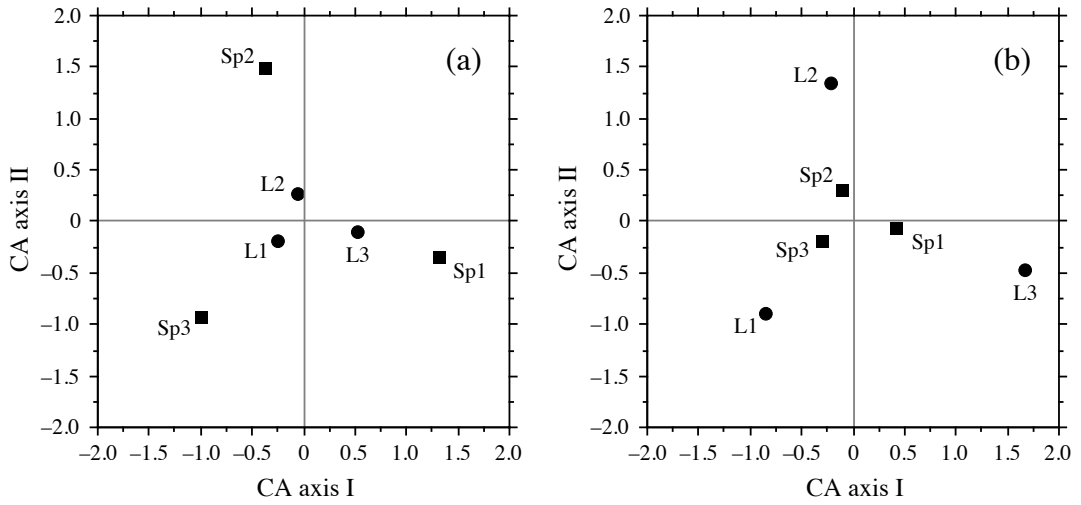


Figure 9.9 Correspondence analysis biplots. (a) Scaling type 1: the rows of the data table (lakes L1 to L3 represented by circles, matrix **F**) are at the centroids (barycentres) of the columns (species Sp1 to Sp3 represented by squares, matrix **V**). (b) Scaling type 2: the species (squares, matrix $\hat{\mathbf{F}}$) are at the centroids (barycentres) of the lakes (circles, matrix $\hat{\mathbf{V}}$).

The normalized eigenvectors of $\overline{\mathbf{Q}}\overline{\mathbf{Q}}'$ are (in columns):

$$\hat{\mathbf{U}} = \begin{array}{c} \\ \text{L1} \\ \text{L2} \\ \text{L3} \end{array} \begin{array}{c} \begin{array}{cc} (\lambda_1) & (\lambda_2) \end{array} \\ \begin{bmatrix} -0.53693 & -0.55831 \\ -0.13043 & 0.79561 \\ 0.83349 & -0.23516 \end{bmatrix} \end{array}$$

The third eigenvector is of no use and is therefore not given. Most programs do not compute it.

In scaling type 1 (Fig. 9.9a), the rows of the data matrix (L1, L2 and L3 in the example), whose coordinates will be stored in matrix **F**, are to be plotted at the centroids of the columns (Sp1, Sp2 and Sp3 in the example). The scaling for the columns is obtained using eq. 9.33:

$$\mathbf{V} = \mathbf{D}(p_{+j})^{-1/2}\,\mathbf{U} = \begin{array}{c} \\ \text{Sp1} \\ \text{Sp2} \\ \text{Sp3} \end{array} \begin{array}{c} \begin{array}{cc} (\lambda_1) & (\lambda_2) \end{array} \\ \begin{bmatrix} 1.31871 & -0.34374 \\ -0.37215 & 1.48150 \\ -0.99972 & -0.92612 \end{bmatrix} \end{array}$$

To put the rows (matrix **F**) at the centroids of the columns (matrix **V**), the position of each row along an ordination axis is computed as the mean of the column positions, weighted by the relative frequencies of the observations in the various columns of that row. Consider the first row of the data table (Table 9.6), for example. The relative frequencies of the three columns in

that row are 0.25, 0.25, and 0.50. Multiplying matrix **V** by that vector provides the coordinates of the first row in the ordination diagram:

$$[0.25 \;\; 0.25 \;\; 0.50] \begin{bmatrix} 1.31871 & -0.34374 \\ -0.37215 & 1.48150 \\ -0.99972 & -0.92612 \end{bmatrix} = [-0.26322 \;\; -0.17862]$$

These coordinates put the first row at the centroid of the columns in Fig. 9.9a; they are stored in the first row of matrix **F**. The row-conditional probabilities for the whole data table are found using the matrix operation $\mathbf{D}(p_{i+})^{-1}\mathbf{Q}$, so that matrix **F** is computed using eq. 9.35b:

$$\mathbf{F} = \mathbf{D}(p_{i+})^{-1}\mathbf{QV} = \begin{array}{c} \\ \text{L1} \\ \text{L2} \\ \text{L3} \end{array} \begin{array}{c} \begin{array}{cc} (\lambda_1) & (\lambda_2) \end{array} \\ \begin{bmatrix} -0.26322 & -0.17862 \\ -0.06835 & 0.27211 \\ 0.51685 & -0.09517 \end{bmatrix} \end{array}$$

Using the formulae for the Euclidean (D_1, eq. 7.32) and χ^2 (D_{16}, eq. 7.55) distances, one can verify that the Euclidean distances among the rows of matrix **F** are equal to the χ^2 distances among the rows of the original data table (Table 9.6):

$$\mathbf{D} = \begin{array}{c} \\ \text{L1} \\ \text{L2} \\ \text{L3} \end{array} \begin{array}{c} \begin{array}{ccc} \text{L1} & \text{L2} & \text{L3} \end{array} \\ \begin{bmatrix} 0 & & \\ 0.49105 & 0 & \\ 0.78452 & 0.69091 & 0 \end{bmatrix} \end{array}$$

Matrix **F** thus provides a proper ordination of the rows of the original data matrix (temperatures in the numerical example).

In scaling type 2 (Fig. 9.9b), the columns, whose coordinates will be stored in matrix $\hat{\mathbf{F}}$, are to be plotted at the centroids of the rows (matrix $\hat{\mathbf{V}}$). The scaling for matrix $\hat{\mathbf{V}}$ is obtained using eq. 9.34:

$$\hat{\mathbf{V}} = \mathbf{D}(p_{i+})^{-1/2}\,\hat{\mathbf{U}} = \begin{array}{c} \\ \text{L1} \\ \text{L2} \\ \text{L3} \end{array} \begin{array}{c} \begin{array}{cc} (\lambda_1) & (\lambda_2) \end{array} \\ \begin{bmatrix} -0.84896 & -0.88276 \\ -0.22046 & 1.34482 \\ 1.66697 & -0.47032 \end{bmatrix} \end{array}$$

To put the columns (matrix $\hat{\mathbf{F}}$) at the centroids of the rows (matrix $\hat{\mathbf{V}}$), the position of each column along an ordination axis is computed as the mean of the row positions, weighted by the relative frequencies of the observations in the various rows of that column. Consider the first column of the data table (Table 9.6), for example. The relative frequencies of the three rows in that column are (10/35 = 0.28571), (10/35 = 0.28571) and (15/35 = 0.42857). Multiplying matrix $\hat{\mathbf{V}}$ by that vector provides the coordinates of the first column in the ordination diagram:

$$[0.28571 \;\; 0.28571 \;\; 0.42857] \begin{bmatrix} -0.84896 & -0.88276 \\ -0.22046 & 1.34482 \\ 1.66697 & -0.47032 \end{bmatrix} = [0.40887 \;\; -0.06955]$$

These coordinates put the first column at the centroid of the rows in Fig. 9.9a; they are stored in the first row of matrix $\hat{\mathbf{F}}$. The column-conditional probabilities for the whole data table are found through matrix operation $\mathbf{D}(p_{+j})^{-1}\mathbf{Q}'$, so that matrix $\hat{\mathbf{F}}$ is computed using eq. 9.36a or 9.36b:

$$\hat{\mathbf{F}} = \mathbf{V}\boldsymbol{\Lambda}^{1/2} = \mathbf{D}(p_{+j})^{-1}\mathbf{Q}'\hat{\mathbf{V}} = \begin{array}{c} \\ \text{Sp1} \\ \text{Sp2} \\ \text{Sp3} \end{array} \begin{array}{c} \begin{array}{cc} (\lambda_1) & (\lambda_2) \end{array} \\ \begin{bmatrix} 0.40887 & -0.06955 \\ -0.11539 & 0.29977 \\ -0.30997 & -0.18739 \end{bmatrix} \end{array}$$

Using the formulae for the Euclidean (D_1, eq. 7.32) and χ^2 (D_{16}, eq. 7.55) distances, one can verify that the Euclidean distances among the rows of matrix $\hat{\mathbf{F}}$ are equal to the χ^2 distances among the columns of the original data table (Table 9.6):

$$\mathbf{D} = \begin{array}{c} \\ \text{Sp1} \\ \text{Sp2} \\ \text{Sp3} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Sp1} & \text{Sp2} & \text{Sp3} \end{array} \\ \begin{bmatrix} 0 & & \\ 0.64128 & 0 & \\ 0.72843 & 0.52458 & 0 \end{bmatrix} \end{array}$$

Matrix $\hat{\mathbf{F}}$ thus provides a proper ordination of the columns of the original data matrix (species abundance classes in the numerical example).

For the numerical example, the table of *Cumulative fit per species* (3 species) is of size (3 × 2) because the CA solution has two dimensions (i.e. two positive eigenvalues) only:

	Cumul. axis 1	**Cumul. axis 2**
Sp1	0.9719	1.0000
Sp2	0.1290	1.0000
Sp3	0.7323	1.0000

In the column that corresponds to the last eigenvalue, the values are always 1 in CA. The table of *Cumulative fit of the objects* (3 lakes) is also of size (3 × 2) in this numerical example:

	Cumul. axis 1	**Cumul. axis 2**
L1	0.6847	1.0000
L2	0.0594	1.0000
L3	0.9672	1.0000

The two tables indicate that species 2 and lake 2 are poorly fitted along axis 1, as can be observed in Fig. 9.9, and that all species and lakes are perfectly represented in 2 dimensions.

3 — Interpretation

The relationship between matrices $\mathbf{V}$ and $\hat{\mathbf{V}}$, which provide the ordinations of the columns and rows of the species data (or contingency) table, respectively, is found by combining eqs. 9.30, 9.33 and 9.34 in the following expression:

$$\hat{\mathbf{V}}\,\boldsymbol{\Lambda}^{1/2} = \mathbf{D}(p_{i+})^{-1/2}\,\overline{\mathbf{Q}}\,\mathbf{D}(p_{+j})^{1/2}\mathbf{V} \qquad \textbf{(9.37)}$$

This equation means that the ordination of the rows (matrix $\hat{\mathbf{V}}$) is related to the ordination of the columns (matrix $\mathbf{V}$), along principal axis h, by the value $\sqrt{\lambda_h}$ which is a measure of the "correlation" between these two ordinations. The value $(1 - \lambda_h)$ actually measures the difficulty of ordering, along principal axis h, the rows of the contingency table from an ordination of the columns, or the converse (Orlóci, 1978). The highest eigenvalue (0.096 in the above numerical example), or its square root ($\sqrt{\lambda_1} = 0.31$), is thus a measure of dependence between two unordered descriptors, to be added to the measures described in Chapter 6. Williams (1952) discusses different methods for testing the significance of $R^2 = \lambda_h$.

Joint plots (e.g. Fig. 9.9) can be used to draw conclusions about the ecological relationships displayed by the data.

• With scaling type 1, (a) the distances among rows (or sites in the case of a species × sites data table) in reduced space approximate their χ^2 distances, and (b) the rows (sites) are at the centroids of the columns (species). Positions of the centroids are calculated using weights equal to the relative frequencies of the columns (species); columns (species) that are absent from a row (site) have null weights and do not contribute to the position of that row (site). Thus, the ordination of rows (sites) is meaningful. In addition, any row (site) found near the point representing a column (species) is likely to have a high contribution of that column (species); for binary (or species presence-absence) data, the row (site) is more likely to possess the state of that column (or contain that species).

• With scaling type 2, it is the distances among columns (species) in reduced space that approximate their χ^2 distances, whereas columns (species) are at the centroids of the rows (sites). Consequently, (a) the ordination of columns (species) is meaningful, and (b) any column (species) that lies close to the point representing a row (site) is more likely to be found in the state of that row (site), or with higher frequency (abundance) than in rows (sites) that are further away in the joint plot.

For species presence-absence or abundance data, insofar as a species has a unimodal (i.e. bell-shaped) response curve along the axes of ecological variation corresponding to the ordination axes, the optimum for that species should be close to the point representing it in the ordination diagram and its frequency of occurrence or abundance should decrease with the distance from that point. Species that are absent at most sites often appear at the edge of the scatter plot, near the point representing a site where they happen to be present — by chance, or because they are favoured by some

rare condition occurring at that site. Such species have little influence on the analysis because their numerical contributions are small (column sums in Table 9.6). Finally, species that lie near the centre of the ordination diagram may have their optimum in that area of the plot, or have two or several optima (bi- or multi-modal species), or else be unrelated to the pair of ordination axes under consideration. Species of the latter group may express themselves along some other axis or axes; close examination of the raw data table may be required in that case. It is the species found away from the centre of the diagram, but not near the edges, that are the most likely to display clear relationships with the ordination axes (ter Braak, 1987c).

4 — Site × species data tables

Correspondence analysis has been applied to data tables other than contingency tables. Justification is provided by Benzécri and coll. (1973). Notice, however, that the elements of a table to be analysed by correspondence analysis must be *dimensionally homogeneous* (i.e. same physical units, so that they can be added), *non-negative* (≥ 0, so that they can be transformed into probabilities or proportions), and additive so that the sums of rows and columns, f_{i+} and f_{+j}, make sense (additivity: see Subsection 1.4.2). Several types of data possess these characteristics, such as (bio)mass values, concentrations, financial data (in \$, €, ¥, etc.), and species abundances.

Other types of data may be recoded to make the descriptors dimensionally homogeneous and positive; the most widely used data transformations are discussed in Section 1.5. For descriptors with different physical units, the data may, for example, be standardized (which makes them dimensionless; eq. 1.12) and made positive by translation, i.e. by subtracting the most negative value; or they may be divided by the maximum or by the range of values (eqs. 1.10 and 1.11). Data may also be recoded into ordered classes. Regardless of the method, recoding is then a critical step of correspondence analysis. Consult Benzécri and coll. (1973) on this matter.

Several authors, mentioned at the beginning of this section, have applied correspondence analysis to the analysis of site × species matrices containing species presence/absence or abundance data. This generalization of the method is based on the following *sampling model*. If sampling had been designed in such a way as to collect individual organisms (which is usually not the case, the sampled elements being, most often, sampling sites), each organism could be described by two descriptors: the site where it was collected and the taxon to which it belongs. These two descriptors may be recorded in an *inflated data matrix*, which has as many rows as there are individual organism, and two columns identifying the site and the taxon of the individual (qualitative descriptors). The familiar site × species data table is the contingency table resulting from crossing the two descriptors of the inflated data matrix, i.e. the sites and taxa. That table could be analysed using any of the methods applicable to contingency tables. Most methods involving tests of statistical significance cannot be used, however, because the hypothesis of independence of the individual organisms, following the sampling model described above, is not met by species presence-absence

Inflated data matrix

or abundance data collected at sampling sites. An inflated data matrix will be used again in the description of canonical correspondence analysis, Subsection 11.2.1.

Niche theory tells us that species have ecological preferences, meaning that they are found at sites where they encounter favourable conditions. This statement is rooted in the idea that species have unimodal distributions along environmental variables (Fig. 9.10), more individuals being found near some environmental value which is "optimal" for the given species. This has been formalised by Hutchinson (1957) in his *fundamental niche* model. Furthermore, Gause's (1935) competitive exclusion principle suggests that, in their micro-evolution, species should have developed non-overlapping niches. These two principles indicate together that species should be roughly equally spaced in the n-dimensional space of resources. This model has been used by ter Braak (1985) to justify the use of correspondence analysis on presence-absence or abundance data tables; he showed that the χ^2 distance preserved through correspondence analysis (Table 9.1) is an appropriate model for species with unimodal distributions along environmental gradients.

Niche

Let us follow the path travelled by Hill (1973b), who rediscovered correspondence analysis while exploring the analysis of vegetation variation along environmental gradients; he called his method "reciprocal averaging" before realizing that this was correspondence analysis (Hill, 1974). Hill started from the simpler method of *gradient analysis*, proposed by Whittaker (1960, 1967) to analyse site × species data tables. Gradient analysis uses a matrix **Y** (site × species) and an initial vector **v** of values v_j which are ascribed to the various species j as indicators of the physical *gradient* to be evidenced. For example, a score (scale from 1 to 10) could be given to the each species for its preference with respect to soil moisture. These coefficients are used to calculate the positions of the sites along the gradient. The score $\hat{v}_i$ of a site i is calculated as the average score of the species ($j = 1 \ldots p$) present at that site, using the formula:

Reciprocal averaging

$$\hat{v}_i = \frac{\sum_{j=1}^{p} y_{ij} v_j}{y_{i+}} \qquad \textbf{(9.38)}$$

where y_{ij} is the abundance of species j at site i and y_{i+} is the sum of the organisms at this site (i.e. the sum of values in row i of matrix **Y**).

Gradient analysis produces a vector $\hat{\mathbf{v}}$ of the positions of the sites along the gradient under study. Hill (1973b, 1974) suggested to continue the analysis, using now vector $\hat{\mathbf{v}}$ of the ordination of sites to compute a new ordination (**v**) of the species:

$$v_j = \frac{\sum_{i=1}^{n} y_{ij} \hat{v}_i}{y_{+j}} \qquad \textbf{(9.39)}$$

in which y_{+j} is the sum of values in column j of matrix **Y**. Alternating between **v** and $\hat{\mathbf{v}}$ (scaling the vectors at each step as shown in step 6 of Table 9.8) defines an iterative procedure that Hill (1973b) called "reciprocal averaging". This procedure converges towards a unique unidimensional ordination of the species and sites, which is independent of the values initially given to the v_j's; different initial guesses as to the values v_j may however change the number of steps required to reach convergence. Being aware of the work of Clint & Jennings (1970), Hill realized that he had discovered an eigenvector method for gradient analysis, hence the title of his 1973b paper. It so happens that Hill's method produces the barycentred vectors **v** and $\hat{\mathbf{v}}$ for species and sites, which correspond to the first eigenvector of a correspondence analysis. Hill (1973b) showed how to calculate the eigenvalue (λ) corresponding to these ordinations and how to find the other eigenvalues and eigenvectors. Hill thus created a simple algorithm, described in Subsection 9.2.7, for correspondence analysis.

When interpreting the results of correspondence analysis, one should keep in mind that the simultaneous ordination of species and sites aims at determining how useful the ordination of species is, as a whole, for predicting the ordination of the sites. In other words, it seeks the predictive value of one ordination with respect to the other. Subsection 9.2.3 has shown that, for any given dimension h, $(1 - \lambda_h)$ measures the difficulty of ordering, along principal axis h, the row states of the contingency table from an ordination of the column states, or the converse. The interpretation of the relationship between the two ordinations must be done with reference to this statistic.

When it is used as an ordination method, correspondence analysis provides an ordination of the sites which is somewhat similar to that resulting from a principal component analysis of the correlation matrix among species (standardized data). This is to be expected since the first step in the calculation actually consists in weighting each datum by the sums (or the relative frequencies) of the corresponding row and column (eq. 9.24 and 9.25), which eliminates the effects due to the large variances that certain rows or columns may have. In the case of steep gradients (i.e. many zeros in the data matrix), correspondence analysis should produce a better ordination than PCA (Hill, 1973b). This was also shown by Gauch *et al*. (1977) using simulated and observational floristic data. This result logically follows from the fact that the χ^2 distance (D_{16}) is a coefficient that excludes double-zeros from the estimation of resemblance. This is not the case with the Euclidean distance (eq. 7.32), which is the distance preserved in principal component analysis. For this reason, correspondence analysis is one of the methods recommended in Fig. 9.8 for reduced-space ordination of species abundances when the data contain a large number of null values; this situation is encountered when sampling environmental gradients that are long enough for species to replace one another. Data tables with many zeros may contain rare species. Box 9.2 describes a procedure for handling rare species in CA.

As mentioned in Section 8.9, for the clustering of species into associations, correspondence analysis does not seem to escape the problems encountered with principal component analysis (Reyssac & Roux, 1972; Ibanez & Séguin, 1972; Binet *et al*., 1972). The most serious problem is that the species, which are multidimensional

Rare species in CA — Box 9.2

In Chapter 7, it was shown that rare species contribute heavily to the chi-square distance (D_{16}, eq. 7.55), which is the distance preserved in correspondence analysis. The present discussion focuses on the species with *small occurrence values*; they only occur in a small fraction of the study sites. These species generate a large number of zeros in the data matrix. Because zeros have high leverage, they contribute heavily to the total inertia of matrix $\overline{\mathbf{Q}}$ (eq. 9.24). These species contribute very little to the first few CA ordination axes, but they are highly conspicuous in biplots because they are found at the periphery of the graph. Should we keep rare species in CA? If not, which ones should be eliminated?

On the one hand, ecologists who see what they are collecting (e.g. vegetation) may consider rare species as potential indicators of special environmental conditions, but it is not the role of CA to display these conditions. The primary purpose of CA is to display the main axes of variation of the data, not to deal with exceptions. On the other hand, ecologists who sample blindly often consider the occurrence of rare species a chance event which should not be heavily weighted in the analysis. In the case of mobile animals, the presence of an animal at a site is no indication that the site provides favourable conditions for that species.

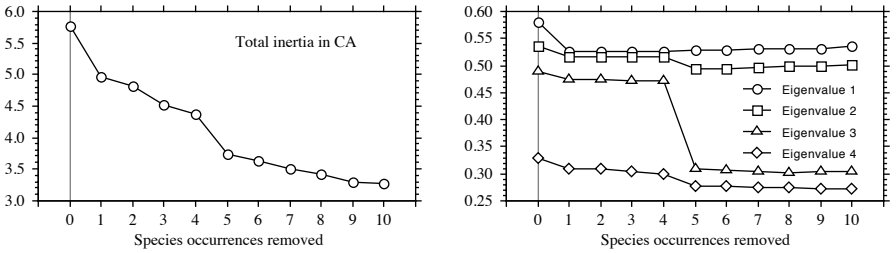
Empirical methods for down-weighting rare species have been proposed and are available in some computer programs, but these methods lack strong ecological foundations. It is better to simply eliminate the rarest species from CA. The following stepwise method has been developed by Daniel Borcard (personal communication):

- For convenience, order the species in the data table in increasing or decreasing occurrences. That will facilitate the stepwise elimination of the species with small occurrence values.
- Carry out a first CA. Note the total inertia as well as the first few eigenvalues (e.g. 4).
- Repeat that step after removing the species with occurrence 1; the species with occurrence 1 and 2; the species with occurrence 1 to 3; and so on. After each analysis, note the total inertia as well as the first few eigenvalues.
- Plot these results. A jump should be observed in total inertia and in some of the eigenvalues. The jump indicates that one has gone too far in removing rare species. [*Continued next page*.]

descriptor-axes, are projected in a low-dimensional space by both PCA and CA. This explains the tendency for the species to form a more or less uniformly dense scatter centred on the origin except in simple situations. It may nevertheless be interesting to superimpose a clustering of species, determined using the methods of Section 8.9, on a reduced-space ordination obtained by correspondence analysis.

Box 9.2 (*continued*)

The following example concerns fish biomass data (47 underwater transects, 156 fish species) collected by researchers Pierre Labrosse and Eric Clua (*Secretariat of the Pacific Community*) near the village of Manuka in the Tonga Islands, under the *DemEcoFish* project funded by the MacArthur Foundation (data used here with permission of the authors).



The curves show that the 61 species with occurrences 1 to 4 can be removed from the analysis with little effects on the first four eigenvalues (right-hand graph). These species generate 24% of the inertia in matrix $\overline{\mathbf{Q}}$ (left-hand graph) subjected to eigenvalue decomposition in CA.

For comparison, the same data were submitted to PCA after Hellinger transformation (Section 7.7). The total variance decreased by only 4.6% after removing the 61 species with occurrences 1 to 4 and the first four eigenvalues were not affected at all by the removal of rare species. The Hellinger transformation was recalculated after each step of species removal.

When sites (objects) and species (descriptors) are plotted together, the joint plot must be interpreted with care. The practice that consists in only associating species with neighbouring species in the plot often gives good results, although it may overlook indications of avoidance of sites by certain species. An interesting complement to correspondence analysis is the direct analysis of the site × species table by the Freeman-Tukey deviates and standardized residuals methods described in Section 6.4. These methods are better at evidencing all the correspondences between sites and species (attraction and avoidance). Applying contingency table analysis to sites × species tables is justified by the same logic that allows correspondence analysis to be applied to such data matrices.

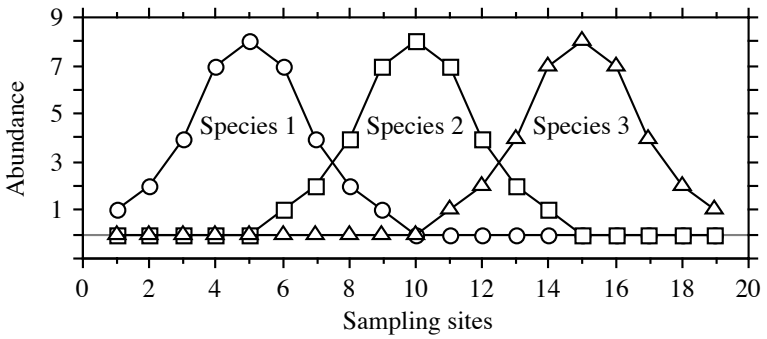


Figure 9.10 Distributions of three species at 19 sampling sites along a hypothetical environmental gradient These artificial data are found in Table 9.7.

5 — Arch effect and detrended correspondence analysis

Environmental gradient

Environmental or temporal gradients often support spatial or temporal succession of species. Since the species that are controlled by environmental factors (*versus* population dynamics, historical events, etc.) generally have unimodal distributions along gradients, the effect of gradients on the distance relationships among sites, calculated on species presence-absence or abundance data, is necessarily nonlinear.

Numerical example 1. A data set was created (Fig. 9.10; Table 9.7) to represent the abundances of three hypothetical species at 19 sites over an environmental gradient along which the species were assumed to have unimodal distributions (Whittaker, 1967).

The three species in Fig. 9.10 have unimodal distributions; each one shows a well-defined mode along the gradient represented by sites 1 to 19. Ordination methods aim at rendering this non-linear phenomenon in a Euclidean space, in particular in two-dimensional plots. In such plots, non-linearities end up being represented by curves,

Table 9.7 Artificial data illustrated in Fig. 9.10.

Sampling sites	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Species 1	1	2	4	7	8	7	4	2	1	0	0	0	0	0	0	0	0	0	0
Species 2	0	0	0	0	0	1	2	4	7	8	7	4	2	1	0	0	0	0	0
Species 3	0	0	0	0	0	0	0	0	0	0	1	2	4	7	8	7	4	2	1

called *arches* or *horseshoes*, described in the next paragraph. While most ecologists are content with interpreting the ordination plots for the information they display about distances among sites, some feel that they should try to reconstruct the original gradient underlying the observed data. Hence their concern with *detrending*, which is an operation carried out on the ordination axes of correspondence analysis. In that operation, the arch is unbent to display the gradient as a linear arrangement of the sites.

Detrending

Euclidean distances calculated on the species data of Fig. 9.10, between site 1 and sites 2, 3, etc., do not increase monotonically from one end of the gradient to the other. These distances form the first row of the Euclidean distance matrix among sites; they are reported on the first row of Table 9.11 in Subsection 9.3.5. Distances from site 1 increase up to site 5, after which they decrease; they increase again up to site 10, then decrease; they increase up to site 15 and decrease again. The other rows of the Euclidean distance matrix display equally complex patterns; they are not shown in Table 9.11 to save space. A PCA algorithm is facing the task of representing these complex patterns in at most three dimensions because PCA ordinations cannot have more axes than the number of original variables (i.e. three species in Fig. 9.10). The result is illustrated in Fig. 9.11, panels a and b. The most dramatic effect is found at the ends of the transect, which are folded inwards along axis I. This is because the Euclidean distance formula considers the extreme sites to be very near each other (small distances due to double-zeros for species 2). This shape is called a *horseshoe*. Figure 9.11b shows that the end sites also go "down" along the third axis. In correspondence analysis on the contrary, extremities of the gradient are, in most instances, not folded inwards in the plot (but see Wartenberg *et al*., 1987, Fig. 3, for a case where this occurs); a bent ordination plot with extremities not folded inwards is called an *arch*, e.g. Fig. 9.11c.

Horseshoe

Arch

The presence in ordination plots of a *bow* (Swan, 1970), *horseshoe* (Kendall, 1971), or *arch* (Gauch, 1982) had already been noted by ecologist Goodall (1954). Benzécri and coll. (1973) discuss the arch under the name *Guttman effect*. Several authors have explained the nature of this mathematical construct, which occurs when the species composition of the sites progressively changes along an environmental gradient. *Detrended correspondence analysis* (DCA; Hill & Gauch, 1980; Gauch, 1982) aims at eliminating the arch effect.

Detrended CA (DCA)

Figure 9.11c helps in understanding the meaning of CA joint plots. This joint plot has been produced using scaling type 1 to preserve the χ^2 distances (D_{16}) among sites; in that respect, this plot is comparable to the PCA ordination shown in Fig. 9.11a. The ordination is two-dimensional since the data set only contains three species. The species (black squares) occupy the edges of a triangle; heavy lines are drawn to materialize their distances to the centre of the plot. Sites 1-5, 10, and 15-19, which only have one species present, occupy the same position as the point representing that species because sites are at the barycentres (centroids) of the species; CA does not spread apart sites that possess a single and same species, even in different amounts. Sites 6-9 and 11-14, which possess two species in various combinations, lie on a line between the two species; their positions along that line depend on the relative

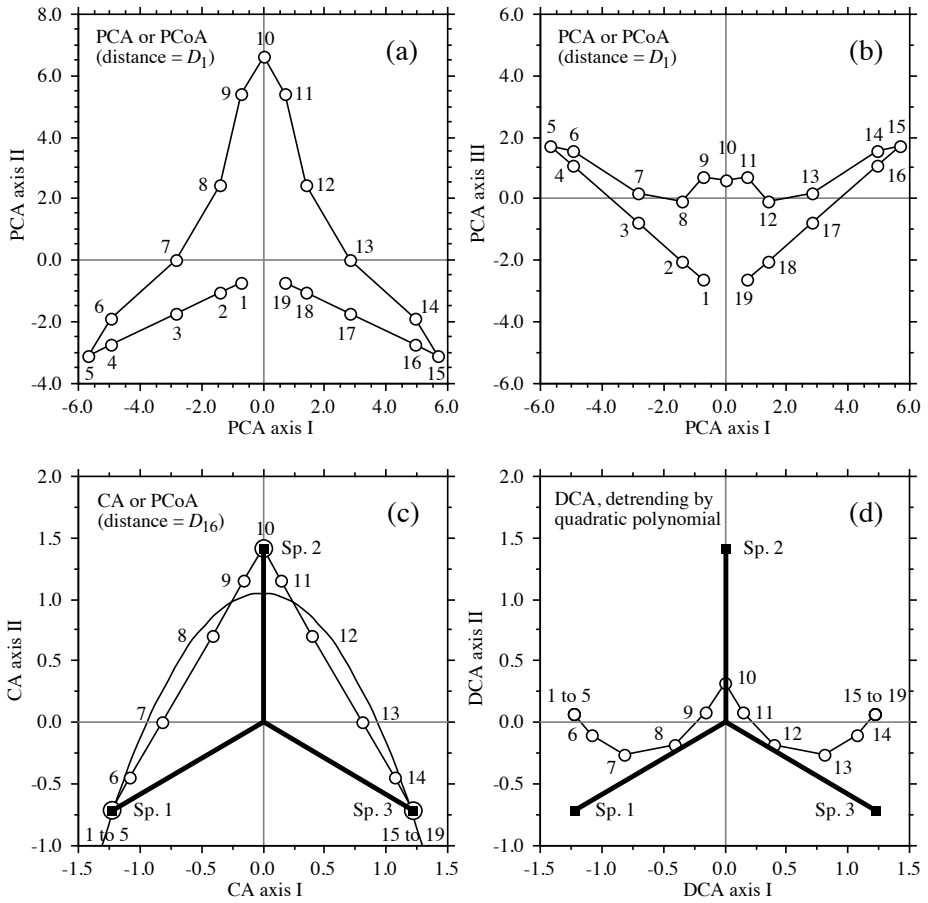


Figure 9.11 Ordinations of the data from Fig. 9.10 and Table 9.7. Circles are sites, and squares in panels c and d are species. Principal component analysis, scaling 1: (a) PCA axes I and II ($\lambda_1 = 50.1\%$, $\lambda_2 = 40.6\%$), (b) axes I and III ($\lambda_1 = 50.1\%$, $\lambda_3 = 9.3\%$). (c) Correspondence analysis, scaling 1, CA axes I and II ($\lambda_1 = 58.1\%$, $\lambda_2 = 41.9\%$). A quadratic polynomial function of axis I is also shown (convex curve): (axis II) = $1.056 - 1.204$ (axis I)^2. (d) Detrended correspondence analysis (scaling type 1, detrending by quadratic polynomial), DCA axes I and II ($\lambda_1 = 58.1\%$, $\lambda_2 = 1.6\%$). (c) and (d) Bold lines drawn from the centres of the plots represent the species axes.

abundances of the two species at each site. No site has three species in this example, so that no point lies inside the triangular shape of the scatter of sites. Considering site 1 (lower left in Fig. 9.11c), examine its distances (D_{16}) to all the other sites in the last row of Table 9.11: they increase from site 6 to 10, after which they remain constant. This corresponds to the relative positions of the sites in the figure. Had the example contained more species, the site points would have displayed a rounded shape.

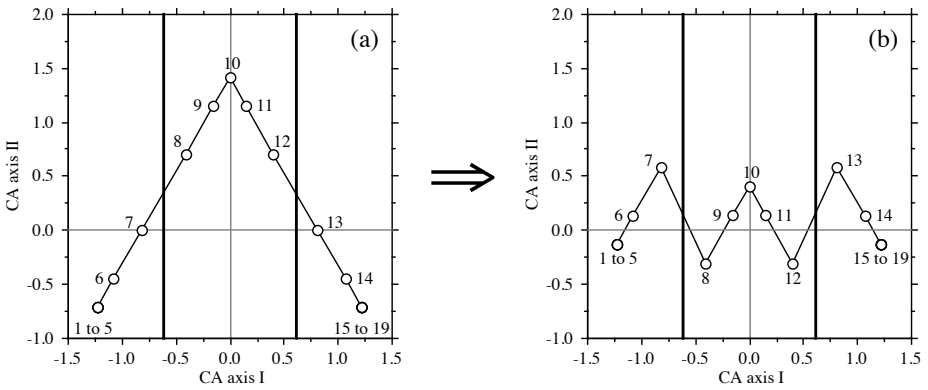


Figure 9.12 Detrending by segments. (a) Three arbitrarily defined segments are delimited by vertical lines in the CA ordination (from Fig. 9.11c). (b) After detrending, the mean of the points in each segment is zero.

Two approaches have been proposed to remove arches in CA, producing detrended correspondence analysis: detrending by segments and by polynomials.

1. When *detrending by segments* (Hill & Gauch, 1980), axis I is divided into a number of *segments* and, within each one, the mean of the scores along axis II is made equal to zero; in other words, data points in each segment are moved along axis II to make their mean coincide with the abscissa. Figure 9.12b shows the result of detrending the ordination of Fig. 9.11c using the three segments defined in Fig. 9.12a. The bottom line is that scores along detrended axis II are meaningless. Proximities among points should in no case be interpreted ecologically, because segmenting generates large differences in scores for points that are near each other in the original ordination but happen to be on either side of segment divisions (Fig. 9.12). The number of segments is arbitrary; different segmentations lead to different ordinations along axis II.

The method is only used with a fairly large number of segments. Programs DECORANA (Hill, 1979b) and CANOCO use a minimum of 10 and a maximum of 46 segments, 26 being the default value that users often take to be the 'recommended' number. This requires a number of data points larger than that in Fig. 9.10.

In order to deal with the contraction of the ends of the gradient when the sites are projected onto the first axis, nonlinear rescaling of the axes is often performed following detrending. An extreme case is represented by Fig. 9.11c where sites 1 to 5 and 15 to 19 each occupy a single point along axis I. To equalize the breadths of the species response curves, the axis is divided into small segments and segments with small within-group variances are expanded, whereas segments with large within-group variances are contracted (Hill, 1979b). Figure 5.5 of ter Braak (1987c) provides a good illustration of the process; ter Braak (1987c) advises *against* the routine use of nonlinear rescaling.

Length of gradient

After detrending by segments and nonlinear rescaling of the axes, the DCA ordination has the interesting property that the axes are scaled in units of the average standard deviation (SD) of species turnover (Gauch, 1982). Along a regular gradient, a species appears, rises to its modal value, and disappears over a distance of about 4 SD; similarly, a complete turnover in species composition occurs, over the sites, in about 4 SD units. A half-change in species composition occurs within about 1 to 1.4 SD units. Thus the length of the first DCA axis is an approximate measure of the length of the ecological gradient, measured in species turnover units. In this respect, DCA with nonlinear rescaling of the axes is a useful method to estimate the lengths of ecological gradients. The length of a gradient revealed by a pilot study may help determine the *extent* (Section 13.0) to be given to a subsequent full-scale study.

2. *Detrending by polynomials* (Hill & Gauch, 1980; ter Braak, 1987c) directly follows from the fact that an arch is produced when a gradient of sufficient length is present in data. When a sufficient number of species are present and replace each other along the gradient, the second CA axis approaches a quadratic function of the first one (i.e. a second-degree polynomial), and so on for the subsequent axes. This is clearly not the case with the data of Table 9.7, which consist of three species only. Figure 9.11c shows that the 'arch' is reduced to a triangular shape in that case.

The arch effect is removed by imposing, in the CA algorithm, the constraint that axis II be uncorrelated not only to axis I (orthogonalization procedure in Table 9.8), but also to its square, its cube, and so on; the degree of the polynomial function is chosen by the user. In the same way, axis III is made uncorrelated to the 1st, 2nd, 3rd … *k*-th degree polynomial of axes I and II. And so forth. When detrending is sought, detrending by polynomial is an attractive method. The result is a continuous function of the previous axes, without the discontinuities generated by detrending-by-segments. However, detrending by polynomials imposes a specific model onto the data, so that the success of the operation depends on how closely the polynomial model corresponds to the data. Detrending by polynomial does not solve the problem of compression of the sites at the ends of the ordination axes.

Detrending by quadratic polynomial was applied to the test data. Figure 9.11c shows the quadratic polynomial (convex curve; among the terms of the quadratic polynomial, only the $(\text{axis I})^2$ term was significant) that was fitted to the CA ordination, which has a triangular shape in the present example. Detrending involves computing and plotting the vertical (residual) distances between the data points and the fitted polynomial. The detrended ordination is shown in Fig. 9.11d. The regression residuals display an elegant but meaningless shape along axis II.

The controversy about detrending raged in the literature for more than 10 years. Key papers are those of Wartenberg *et al*. (1987), Peet *et al*. (1988), and Jackson & Somers (1991b). Wartenberg *et al*. (1987) argued that the arch is an important and inherent attribute of the distances among sites, not a mathematical artifact. The only effect of DCA is to flatten the distribution of points onto axis I without affecting the ordination of sites along that axis. They also pointed out that detrending-by-segments is an arbitrary method for which no theoretical justification has been offered. Similarly, the nonlinear rescaling procedure assumes that, on average, each species appears and disappears at the same rate along the transect and that the parametric variance is an adequate measure of that rate; these assumptions have not been substantiated. Despite these criticisms, Peet *et al*. (1988) still supported DCA on the ground that detrending

and rescaling may facilitate ecological interpretation. Jackson & Somers (1991b) showed that the DCA ordination of sites greatly varied with the number of segments one arbitrarily decides to use, so that the ecological interpretation of the results may vary widely, as do the correlations one can calculate with environmental variables. One should always try different numbers of segments if one decides to use DCA.

Simulation studies involving DCA have been conducted on artificial data representing unimodal species responses to environmental gradients in one (*coenoclines*) or two (*coenoplanes*) dimensions, following the method pioneered by Swan (1970). Kenkel & Orlóci (1986) report that DCA did not perform particularly well in recovering complex gradients. Using Procrustes statistics (Subsection 11.5.2) as measures of structure recovery, Minchin (1987) showed that DCA did not perform well with complex response models and non-regular sampling schemes. Both studies concurred that nMDS (Section 9.4) was a better method than DCA for recovering complex gradients.

Present evidence indicates that detrending should be avoided except for the specific purpose of estimating the lengths of gradients; such estimates remain subject to the condition that the assumptions of the model are true. In particular, DCA should be avoided when analysing data that represent complex ecological gradients. Most ordination techniques are able to recover simple, one-dimensional environmental gradients. When there is a single gradient in the data, detrending is useless since the gradient is well represented by CA axis I.

Satisfactory mathematical solutions to the problem of detrending remain to be found. In the meantime, ordination results should be interpreted with caution and in the light of the type of distance preserved by each method.

6 — Ecological applications

Ecological application 9.2a

The spider data (28 sites × 12 species) of Aart & Smeenk-Enserink (1975) that have been analysed by principal component analysis (PCA) in Ecological application 9.1a are reanalysed here by correspondence analysis (CA). Figure 9.13 presents two CA biplots (scaling types 1 and 2) obtained for these data. Compare the species groups and site ordinations with the PCA biplot presented in Fig. 9.6b (log-transformed data).

Ecological application 9.2b

Cadoret *et al*. (1995) investigated the species composition (presence/absence and abundance) of chaetodontid fish assemblages off Moorea Island, French Polynesia, in order to describe the spatial distribution of the butterflyfishes and determine their relationships with groups of benthic organisms. Sampling was conducted in four areas around the island: (a) Opunohu Bay, (b) Cook Bay, (c) the Tiahura transect across the reef in the northwestern part of the island, and (d) the Afareaitu transect across the reef in the eastern part of the island.

Correspondence analysis (Fig. 9.14) showed that the fish assemblages responded to the main environmental gradients that characterized the sampling sites. For areas c and d (transects across the reef), axis I corresponded to a gradient from the coastline to the ocean; from left to right, in

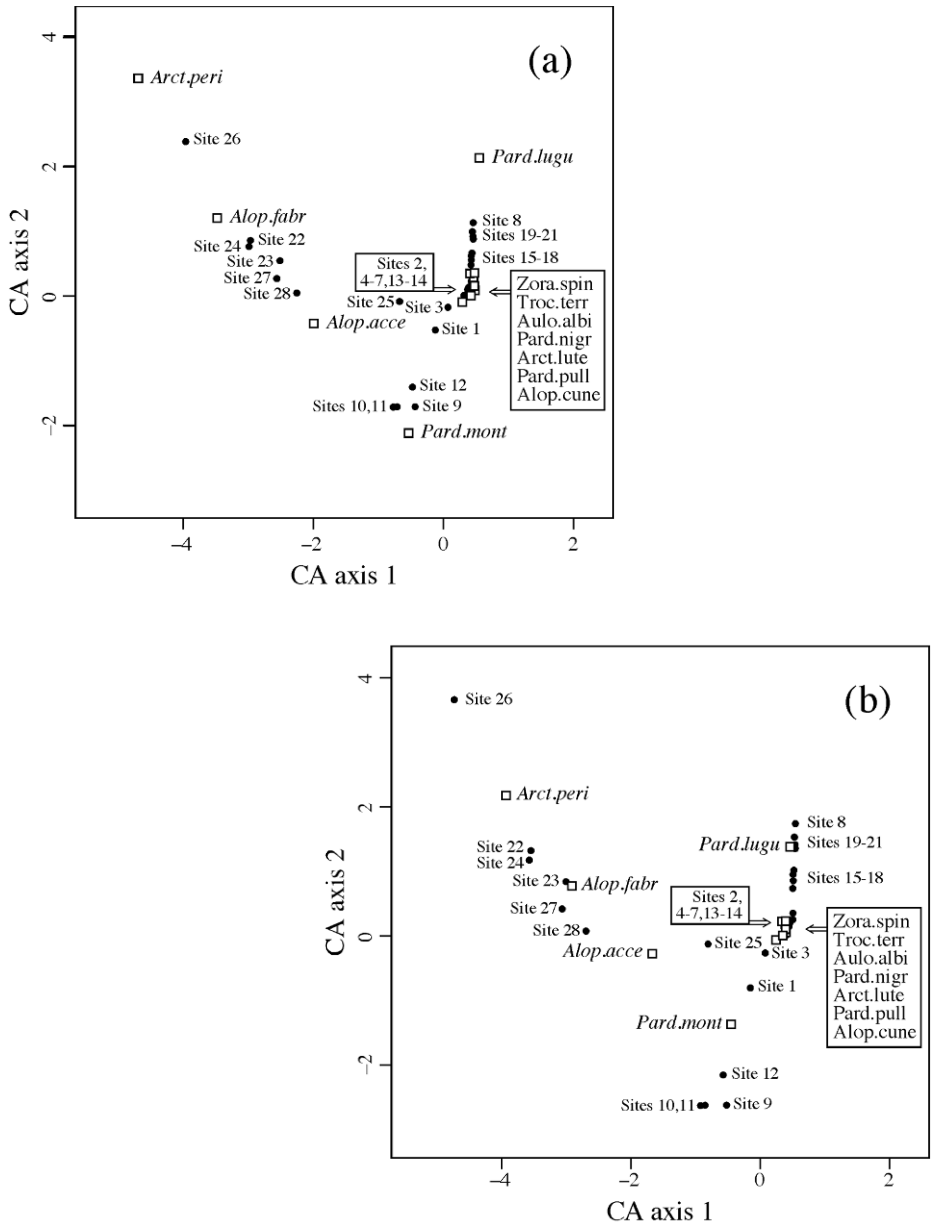


Figure 9.13 Correspondence analysis biplots of the spider data. (a) Scaling type 1: the sites (solid circles) are at the centroids of the species (open squares). (b) Scaling type 2: the species (open squares) are at the centroids of the sites (solid circles). Species abbreviations: see Fig. 9.6.

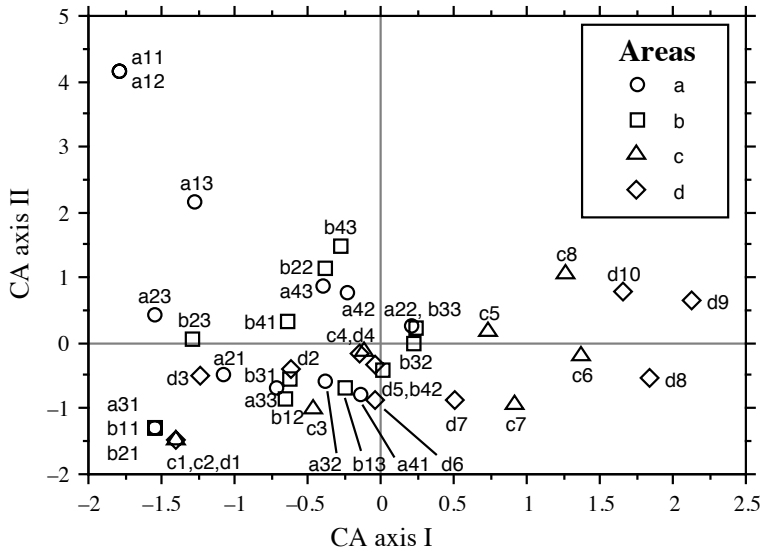


Figure 9.14 Correspondence analysis (CA): ordination of sampling sites with respect to axes I and II from presence/absence observations of butterflyfishes (21 species) in four areas (different symbols) around Moorea Island. Axes I and II explain together 29% of the variation among sites. Species are not drawn; they would have overloaded the plot. Modified from Cadoret *et al*. (1995).

the plot, are the sites of the fringing reef, the shallows (found only in sector c), the barrier reef, and the outer slope. Sites from the bays (areas a and b) are also found in the left-hand part of the graph. Axis II separates the sites located in the upper reaches of Opunohu Bay (a11, a12 and a13, in the upper-left of the plot) from all the others. This application will be further developed, in Subsection 11.2.2, to identify species assemblages and evidence the relationships between species and environmental variables, using canonical correspondence analysis.

Ecological application 9.2c

In a study on the vegetation dynamics of southern Wisconsin, Sharpe *et al*. (1987) undertook a systematic field survey of all forest tracts in two townships. Detrended correspondence analysis was used to display the relationships among stands with respect to species composition. The scores of the first ordination axes were used to construct three-dimensional maps. In the map of the first axis (Fig. 9.15), the scores were generally low in the southern and central portions of the area, and increased towards the west and north. Since the first axis showed a trend from forest tracts dominated by *Acer saccharum* to oak-dominated forests (not shown), Fig. 9.15 indicates that stands dominated by *A. saccharum* were located in the south-central portion of the area, whereas oak-dominated stands were to the west, north and, to a lesser extent, east. Such a mapping, using a 3- or 2-dimensional representation, is often a useful way of displaying synthetic information provided by the scores of objects along the first ordination axes.

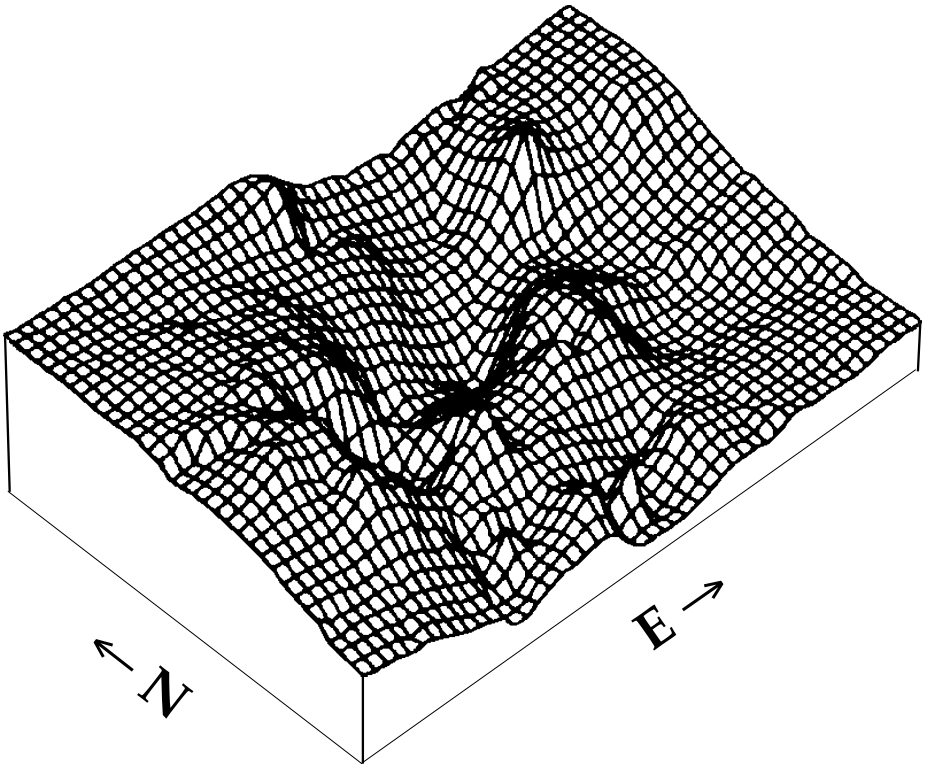


Figure 9.15 Three-dimensional map of the scores of the first ordination axis (detrended correspondence analysis), based on trees observed in 92 forest tracts of southern Wisconsin, U.S.A. (survey area: 11 × 17 km). Modified from Sharpe *et al*. (1987).

Maps like the one displayed in Fig. 9.15 may be produced for the ordination scores computed by any of the methods described in the present chapter; see Section 13.2.

7 — Algorithms

There are several computer programs and R functions available for correspondence analysis; see Section 9.5.

TWWA algorithm

CANOCO (ter Braak, 1988b, 1988c, 1990; ter Braak & Smilauer, 1998) uses Hill's *two-way weighted averaging* (TWWA) algorithm as summarized by ter Braak (1987c). This algorithm is described in Table 9.8. There are three main differences with the TWWS algorithm for PCA presented in Table 9.5: (1) variables are centred in PCA, not in CA; (2) in CA, the centroid of the site scores is not zero and must thus be estimated (step 6.1); (3) in CA, summations are standardized by the row sum, column

Table 9.8 Two-way weighted averaging (TWWA) algorithm for correspondence analysis. From Hill (1973b) and ter Braak (1987c).

a) Iterative estimation procedure

Step 1: Consider a table **Y** with n rows (sites) × p columns (species).
Do NOT centre the columns (species) on their means.

Determine how many eigenvectors are needed. For each one, **DO** the following:

Step 2: Take the row order as the arbitrary initial site scores. (1, 2, …)
Set the initial eigenvalue estimate to 0. In what follows, y_{i+} = row sum for site i, y_{+j} = column sum for species j, and y_{++} = grand total for the data table **Y**.

Iterative procedure begins

Step 3: Compute new species loadings: $colscore(j) = \sum y(i,j) \times rowscore(i)/y_{+j}$

Step 4: Compute new site scores: $rowscore(i) = \sum y(i,j) \times colscore(j)/y_{i+}$

Step 5: For the second and higher-order axes, make the site scores uncorrelated with all previous axes (Gram-Schmidt orthogonalization procedure: see *b* below).

Step 6: Normalize the vector of site scores (procedure *c*, below) and obtain an estimate of the eigenvalue. If this estimate does not differ from the previous one by more than the tolerance set by the user, go to step 7. If the difference is larger than the tolerance, go to step 3.

End of iterative procedure

Step 7: If more eigenvectors are to be computed, go to step 2. If not, continue with step 8.

Step 8: The row (site) scores correspond to matrix $\hat{\mathbf{V}}$. The column scores (species loadings) correspond to matrix $\hat{\mathbf{F}}$. Matrices $\hat{\mathbf{F}}$ and $\hat{\mathbf{V}}$ provide scaling type 2 (Subsection 9.2.1). Scalings 1 or 3 may be calculated if required. Return the eigenvalues, % variance, species loadings, and site scores.

b) Gram-Schmidt orthogonalization procedure

DO the following, in turn, for all previously computed components k:

Step 5.1: Compute the scalar product $SP = \sum (y_{i+} \times rowscore(i) \times v(i,k)/y_{++})$ of the current site score vector estimate with the previous component k. Vector $v(i,k)$ contains the site scores of component k scaled to length 1. This product is between 0 (if the vectors are orthogonal) and 1.

Step 5.2: Compute new values of $rowscore(i)$ such that vector *rowscore* becomes orthogonal to vector $v(i,k)$: $rowscore(i) = rowscore(i) - (SP \times v(i,k))$.

c) Normalization procedure[†]

Step 6.1: Compute the centroid of the site scores: $z = \sum (y_{i+} \times rowscore(i)/y_{++})$.

Step 6.2: Compute the sum of squares of the site scores: $S^2 = \sum (y_{i+} \times (rowscore(i) - z)^2/y_{++})$; $S = \sqrt{S^2}$.

Step 6.3: Compute the normalized site scores: $rowscore(i) = (rowscore(i) - z)/S$.

Step 6.4: At the end of each iteration, S, which measures the amount of shrinking during the iteration, provides an estimate of the eigenvalue. Upon convergence, the eigenvalue is S.

[†] Normalization in CA is such that the *weighted* sum of squares of the elements of the vector is equal to 1.

sum, or grand total, as appropriate, which produces shrinking of the ordination scores at the end of each iteration in CA (step 6.4), instead of stretching as in PCA.

SVD
Householder

R functions for CA use either singular value decomposition (SVD, function ***svd()*** of R) or Householder reduction (function ***eigen()*** of R). SVD and eigen-decomposition were both used to describe the CA algorithm in Subsection 9.2.1; they provide the eigenvalues as well as matrices $\mathbf{U}$ and $\hat{\mathbf{U}}$. The various matrices for the row and column scores used in scalings are then obtained using eqs. 9.33 to 9.36.

9.3 Principal coordinate analysis (PCoA)

Principal component analysis (PCA) is only applicable to data for which the Euclidean distance (D_1) is appropriate, whereas correspondence analysis (CA) is only applicable to frequency-like data for which the χ^2 distance (D_{16}) is appropriate. For other types of data, the relationships among objects are computed with one of the resemblance coefficients described in Chapter 7. The list includes coefficients that can handle binary data (S_1 to S_{14}, S_{24} to S_{27}) and mixtures of quantitative and qualitative descriptors (S_{15}, S_{16}). PCA cannot be applied to these data. CA can be used with presence-absence data for which double zeros must be excluded from object comparisons, but not with mixtures of quantitative and qualitative descriptors.

Euclidean representation

Gower (1966) described a method to obtain a Euclidean representation (i.e. a representation in a Cartesian coordinate system) of a set of objects whose relationships are measured by any distance coefficient chosen by users. This method, known as *principal coordinate analysis* (abbreviated PCoA), *metric multidimensional scaling* (in contrast to the nonmetric method described in Section 9.4), or *classical scaling* by reference to the pioneering work of Torgerson (1958), allows one to position objects in a space of reduced dimensionality while preserving their distance relationships as well as possible; see also Rao (1964).

Mixed precision

The interest of the PCoA method lies in the fact that it may be used with all types of descriptors — even data sets with descriptors of mixed levels of precision, provided that a coefficient appropriate to the data has been used to compute the resemblance matrix (e.g. S_{15} or S_{16}, Chapter 7). It will be shown that, if the distance matrix is metric, i.e. if it contains no violation of the triangle inequality, the relationships among objects can, in most cases, be fully represented in Euclidean space. In the case of violations of the triangle inequality, or when problems of "non-Euclideanarity" occur with metric distances (Gower, 1982; Fig. 9.16), negative eigenvalues are produced. In most cases, this does not impair the quality of the Euclidean representation obtained for the first few principal coordinates. It is also possible to transform the distance matrix, or use an alternative resemblance measure, to eliminate the problem. These topics are discussed in Subsection 9.3.4.

Euclidean model

One may look at principal coordinates as the equivalent of principal components. Principal components, on the one hand, are linear combinations of the original (or standardized) descriptors; *linear* is the key concept. Principal coordinates, on the other hand, are also functions of the original variables, but mediated through the distance function that has been computed among objects. In any case, PCoA can only embed (i.e. fully represent), in Euclidean space, the Euclidean part of a distance matrix. This is not a property of the data, but a result of the Euclidean model, which is forced upon the data because the objective is to draw scatter diagrams on sheets of paper. By doing so, one must accept that whatever is non-Euclidean cannot be drawn on paper. This may be viewed as the problem of drawing points separated by non-Euclidean distances into a Euclidean space.

Like PCoA, the method of nonmetric multidimensional scaling (nMDS, Section 9.4) produces ordinations of objects from any resemblance matrix. It compresses the distance relationships among objects into, say, two or three dimensions in a more efficient way than PCoA. nMDS always obtains a Euclidean representation, even from non-Euclidean-embeddable distances. However, nMDS compresses the distances in a non-linear way and its algorithm is computer-intensive, requiring more computing time than PCoA. The latter is faster for large distance matrices.

1 — Computation

Gower (1966) explained how to compute the principal coordinates of a distance matrix:

- The calculation starts with a distance matrix $\mathbf{D} = [D_{hi}]$. It is also possible to carry out the calculations from a similarity matrix $\mathbf{S} = [S_{hi}]$; the method is detailed in Subsection 9.3.3.

- Matrix **D** is transformed into a new matrix $\mathbf{A} = [a_{hi}]$ by defining:

$$a_{hi} = -\frac{1}{2}D_{hi}^2 \qquad \textbf{(9.40)}$$

The purpose of this transformation is explained in Subsection 9.3.3.

- Matrix **A** is centred to give matrix $\mathbf{\Delta}_1 = [\delta_{hi}]$, using the following equation:

$$\delta_{hi} = a_{hi} - \bar{a}_h - \bar{a}_i + \bar{a} \qquad \textbf{(9.41)}$$

where $\bar{a}_h$ and $\bar{a}_i$ are the means of the row and column corresponding to element a_{hi} of matrix **A**, respectively, and $\bar{a}$ is the mean of all a_{hi}'s in the matrix. The following matrix equation produces the centring described in eq. 9.41:

$$\mathbf{\Delta}_1 = \left(\mathbf{I} - \frac{\mathbf{11}'}{n}\right)\mathbf{A}\left(\mathbf{I} - \frac{\mathbf{11}'}{n}\right) \qquad \textbf{(9.42)}$$

where **I** is an identity matrix of order n and **1** is a column vector of length n containing values "1". Centring has the effect of positioning the origin of the new system of axes at the centroid of the scatter of objects without altering the distances among objects.

Euclidean distance

In the particular case of distances computed using the Euclidean distance coefficient (D_1, eq. 7.32), it is possible to obtain the Gower-centred matrix $\mathbf{\Delta}_1$ directly, i.e. without calculating a matrix **D** of Euclidean distances and going through eqs. 9.40 and 9.41, because $\mathbf{\Delta}_1 = \mathbf{Y}_c\mathbf{Y}_c'$, where $\mathbf{Y}_c$ is **Y** centred by columns. This may be verified using numerical examples. In that particular case, $\mathbf{\Delta}_1$ is always a positive semidefinite matrix (Table 2.2).

• The eigenvalues λ_k and normalized eigenvectors (matrix **U**) are computed and each eigenvector $\mathbf{u}_k$ is multiplied by the square root of its eigenvalue. As a result, the eigenvectors are scaled to lengths equal to the square roots of their eigenvalues:

$$\sqrt{\mathbf{u}'_k\mathbf{u}_k} = \sqrt{\lambda_k}$$

Degenerate **D** matrix

Due to the centring, matrix $\mathbf{\Delta}_1$ always has at least one zero eigenvalue. The reason is that at most $(n - 1)$ real axes are necessary for representing n points in Euclidean space. There may be more than one zero eigenvalue if the distance matrix is degenerate, i.e. if the objects can be represented in fewer than $(n - 1)$ dimensions. In practice, there are c positive eigenvalues and c real axes forming the Euclidean representation of the data, the rule being that $c \leq n - 1$.

With the Euclidean distance (D_1), when there are more objects than descriptors ($n > p$), the maximum value of c is p; when $n \leq p$, then $c \leq n - 1$. Take as example a set of three objects or more, and two descriptors ($n > p$). The objects, as many as they are, may be represented in a two-dimensional space — for example, the scatter diagram of the two descriptors. Consider now the case where there are two objects and two descriptors ($n \leq p$); the two objects only require one dimension for representation.

• After scaling, if the eigenvectors are written as columns (e.g. Table 9.9), the *rows* of the resulting table are the *coordinates of the objects* in the space of principal coordinates, without any further transformation; they form matrix **PC** of the principal coordinates. Plotting the points on, say, the first two principal coordinates produces a reduced-space ordination diagram of the objects in two dimensions.

2 — Numerical example

Readers may get a better feeling of what principal coordinate analysis does by comparing it to principal component analysis. Consider a data matrix **Y** on which a principal component analysis (PCA) has been computed, with resulting eigenvalues, eigenvectors (matrix **U**), and principal components (matrix **F**). If one also computed a Euclidean distance matrix $\mathbf{D} = [D_{hi}]$ for the same n objects, the eigenvectors obtained by principal coordinate analysis would be exactly the same as the principal components. The eigenvalues of the PCoA are equal to the eigenvalues one would

Table 9.9 Principal coordinates of the objects (rows) are obtained by scaling the eigenvectors to $\sqrt{\lambda}$.

	Eigenvalues			
	λ_1	λ_2	$\cdots$	λ_c
Objects	**Eigenvectors**			
$\mathbf{x}_1$	u_{11}	u_{12}	$\cdots$	u_{1c}
$\mathbf{x}_2$	u_{21}	u_{22}	$\cdots$	u_{2c}
$\mathbf{x}_h$	u_{h1}	u_{h2}	$\cdots$	u_{hc}
$\mathbf{x}_i$	u_{i1}	u_{i2}	$\cdots$	u_{ic}
$\mathbf{x}_n$	u_{n1}	u_{n2}	$\cdots$	u_{nc}
Lengths: $\sqrt{\sum_i u_{ik}^2}$ =	$\sqrt{\lambda_1}$	$\sqrt{\lambda_2}$	$\cdots$	$\sqrt{\lambda_c}$
Centroid: $[\bar{u}_k]$ =	0	0	$\cdots$	0

obtain from a PCA conducted on the cross-product matrix $[y - \bar{y}]' [y - \bar{y}]$; these are larger than the eigenvalues of a PCA conducted on the covariance matrix **S** by factor $(n - 1)$ because $\mathbf{S} = (1/(n-1)) [y - \bar{y}]' [y - \bar{y}]$. Since PCA has been defined, in this book, as the eigenanalysis of the covariance matrix **S**, the same PCA eigenvalues can be obtained from a principal coordinate analysis computed on the Euclidean distance matrix among objects, and dividing the resulting PCoA eigenvalues by $(n - 1)$. If one is only interested in the *relative* magnitude of the eigenvalues, this scaling step is not necessary and may be ignored.

The previous paragraph does not mean that principal coordinate analysis is limited to Euclidean distance matrices. It can actually be computed for *any* distance matrix. If the distances cannot readily be embedded in Euclidean space, negative eigenvalues may be obtained, with consequences described in Subsection 9.3.4.

Numerical example 2. The numerical example for principal component analysis (Section 9.1) is used here to illustrate the main steps in the computation of principal coordinates. The example also shows that computing principal coordinates from a matrix of Euclidean distances $\mathbf{D} = [D_{hi}]$ gives the exact same results as a principal component analysis of the raw data, with the exception that the descriptor loadings are not obtained in PCoA. Indeed, information about the original descriptors is not passed on to the PCoA algorithm. Indeed, since PCoA is computed from a distance matrix among objects, it cannot give back the loadings of the descriptors. A method for computing them *a posteriori* is described in Subsection 9.3.3 (eq. 9.45).

1) The matrix of Euclidean distances among the 5 objects of data matrix **Y** used to illustrate Section 9.1 is:

$$\mathbf{D} = \begin{bmatrix} 0.00000 & 3.16228 & 3.16228 & 7.07107 & 7.07107 \\ 3.16228 & 0.00000 & 4.47214 & 4.47214 & 6.32456 \\ 3.16228 & 4.47214 & 0.00000 & 6.32456 & 4.47214 \\ 7.07107 & 4.47214 & 6.32456 & 0.00000 & 4.47214 \\ 7.07107 & 6.32456 & 4.47214 & 4.47214 & 0.00000 \end{bmatrix}$$

2) Matrix $\mathbf{\Delta}_1$ obtained by Gower's centring (eqs. 9.40 and 9.41) is:

$$\mathbf{\Delta}_1 = \begin{bmatrix} 12.8 & 4.8 & 4.8 & -11.2 & -11.2 \\ 4.8 & 6.8 & -3.2 & 0.8 & -9.2 \\ 4.8 & -3.2 & 6.8 & -9.2 & 0.8 \\ -11.2 & 0.8 & -9.2 & 14.8 & 4.8 \\ -11.2 & -9.2 & 0.8 & 4.8 & 14.8 \end{bmatrix}$$

The trace (sum of the diagonal elements) of this matrix is 56. This is $(n - 1) = 4$ times the trace of the covariance matrix computed in PCA, which was 14. The diagonal elements are the squared distances of the points to the multivariate centroid. Note that matrix $\mathbf{\Delta}_1$ could have been obtained directly from data matrix **Y** centred by columns ($\mathbf{Y}_c$), as mentioned in Subsection 9.3.1 for the particular case where **D** is computed using the Euclidean distance coefficient (D_1, eq. 7.32): $\mathbf{\Delta}_1 = \mathbf{Y}_c\mathbf{Y}_c'$. Readers can verify this property numerically for the example.

3) The eigenvalues and eigenvectors of matrix $\mathbf{\Delta}_1$, scaled to $\sqrt{\lambda}$, are shown in Table 9.10. There are only $c = 2$ eigenvalues different from zero; this was to be expected since the distances had been computed from $p = 2$ variables only ($c = p = 2$). The principal coordinates, which are the rescaled eigenvectors of the PCoA, are identical to the principal components (Subsection 9.1.2 and Table 9.6) in this example. Measures of resemblance other than the Euclidean distance may produce a different number of eigenvalues and principal coordinates and they would, of course, position the objects differently.

PCA and PCoA

While the numerical example illustrates the fact that a PCoA computed on a Euclidean distance matrix gives the same results as a PCA conducted on the original data, the converse is also true: taking the coordinates of the objects in the full space (all eigenvectors) obtained from a PCoA and using them as input of a principal component analysis would produce the same PCA eigenvalues as those of the original PCoA (to a

Table 9.10 Principal coordinates computed for the numerical example for PCA developed in Section 9.1. Compare with PCA results in Subsection 9.1.2 and Table 9.6.

	Eigenvalues	
	λ_1	λ_2
Objects	**Eigenvectors**	
$\mathbf{x}_1$	–3.578	0.000
$\mathbf{x}_2$	–1.342	–2.236
$\mathbf{x}_3$	–1.342	2.236
$\mathbf{x}_4$	3.130	–2.236
$\mathbf{x}_5$	3.130	2.236
Eigenvalues of PCoA	36.000	20.000
PCoA eigenvalues/$(n-1)$ = eigenvalues of corresponding PCA	9.000	5.000
Lengths: $\sqrt{\sum_i u_{ik}^2}$ =	6.000 = $\sqrt{36}$	4.472 = $\sqrt{20}$

factor $n-1$), and the principal components will be identical to the principal coordinates. All the signs of any one component may be inverted, though, as explained in Subsection 9.1.9; signs depend on an arbitrary decision made during execution of eigen-decomposition functions (Subsection 2.9.2). Because of this, before presenting their results, users of ordination methods are free to invert all the signs of any principal component or principal coordinate if that suits them better.

3 — Rationale of the method

Gower (1966) has shown that the distance relationships among objects are preserved in the full-dimensional principal coordinate space. His proof is summarized as follows.

• In the total space of the principal coordinates (i.e. all eigenvectors), the distance between objects h and i can be found by computing the Euclidean distance between rows h and i of Table 9.9:

$$D'_{hi} = \left[\sum_{k=1}^{c} (u_{hk} - u_{ik})^2\right]^{1/2} = \left[\sum_{k=1}^{c} u_{hk}^2 + \sum_{k=1}^{c} u_{ik}^2 - 2\sum_{k=1}^{c} u_{hk}u_{ik}\right]^{1/2} \quad \textbf{(9.43)}$$

• Since the eigenvectors are scaled in such a way that their lengths are $\sqrt{\lambda_k}$ (in other words, **U** is scaled here to $\mathbf{\Lambda}^{1/2}$), the eigenvectors have the property that $\mathbf{\Delta}_1 = \mathbf{U}\mathbf{U}'$. One can thus write:

$$\mathbf{\Delta}_1 = [\delta_{hi}] = \mathbf{u}_1\mathbf{u}_1' + \mathbf{u}_2\mathbf{u}_2' + \ldots + \mathbf{u}_c\mathbf{u}_c'$$

from which it can be shown, following eq. 9.43, that:

$$D'_{hi} = [\delta_{hh} + \delta_{ii} - 2\delta_{hi}]^{1/2}$$

Readers can verify this property on the above numerical example.

• Since $\delta_{hi} = a_{hi} - \bar{a}_h - \bar{a}_i + \bar{a}$ (eq. 9.41), replacing the values of δ in the right-hand member of the previous equation gives:

$$\delta_{hh} + \delta_{ii} - 2\delta_{hi} = a_{hh} + a_{ii} - 2a_{hi}$$

hence

$$D'_{hi} = [a_{hh} + a_{ii} - 2a_{hi}]^{1/2}$$

The transformation of **A** into $\mathbf{\Delta}_1$ is not essential. It is simply meant to eliminate one of the eigenvalues, which could be the largest and would only account for the distance between the centroid and the origin.

• The transformation of the matrix of original distances D_{hi} into **A** is such that distances are preserved in the course of the calculations. Actually, one can replace the a_{hi} terms in the previous equation by $-0.5\ D^2_{hi}$ (eq. 9.40), which produces the equation

$$D'_{hi} = \left[-\frac{1}{2}D^2_{hh} - \frac{1}{2}D^2_{ii} + D^2_{hi}\right]^{1/2}$$

and, since $D_{hh} = D_{ii} = 0$ (property of distances),

$$D'_{hi} = [D^2_{hi}]^{1/2}$$

Principal coordinate analysis thus preserves the original distances, regardless of the formula used to compute them. If the distances have been calculated from similarities, $D_{hi} = 1 - S_{hi}$ will be preserved in the full-dimensional principal coordinate space. If the transformation of similarities into distances was done by $D_{hi} = \sqrt{1 - S_{hi}}$ or $D_{hi} = \sqrt{1 - S^2_{hi}}$, then it is these distances that are preserved by the PCoA. As a corollary, these various representations in principal coordinate space should be as different from one another as are the distances themselves.

Gower (1966) has also shown that principal coordinates can be directly computed from a similarity matrix **S** instead of a distance matrix **D**, as follows: (1) make sure that the diagonal of matrix **S** contains 1's and not 0's before centring; (2) centre matrix **S** using eq. 9.41 or 9.42

without applying eq. 9.40 first; (3) compute the eigenvalues and eigenvectors; (4) multiply the elements of each eigenvector k by $\lambda_k^{0.5}$. The distances D'_{hi} among the reconstructed point-objects in the full-dimensional principal coordinate space are not the same as the distances $D_{hi} = (1 - S_{hi})$; they are distorted, being such that $D'_{hi} = \sqrt{2}\sqrt{D_{hi}}$. Looking at it from another viewpoint, the reconstructed distances D'_{hi} are larger than the distances $D_{hi} = (1 - S_{hi})^{0.5}$ by a factor $\sqrt{2}$ without further distortion. These relationships hold only if the centred matrix **S** is positive semidefinite, i.e. if its eigen-decomposition does not produce negative eigenvalues.

To summarize, principal coordinate analysis produces a representation of objects in Euclidean space that preserves the distance relationships computed using any measure selected by users. This is a major difference with PCA, where the distance among objects is always, by definition, the Euclidean distance (Table 9.1). In PCoA, the representation of objects in the reduced space of the first few principal coordinates forms the best possible Euclidean approximation of the original distances, because the sum of squared lengths of the objects in the selected subspace is maximum (Gower, 1982). The quality of a Euclidean representation in a space of principal coordinates can be assessed using a Shepard diagram (Fig. 9.1).

Contrary to principal component analysis, the relationships between the principal coordinates and the original descriptors are not provided by a principal coordinate analysis. Indeed the descriptors, from which distances were initially computed among the objects, do not play any role during the calculation of the PCoA from matrix **D**.

PCoA biplot

However, computing the projections of descriptors in the space of the principal coordinates to produce biplots is fairly simple:

$$\mathbf{S}_{pc} = \frac{1}{(n-1)} \mathbf{Y}_c' \mathbf{U}_{st} \qquad \textbf{(9.44)}$$

$$\mathbf{U}_{proj} = \sqrt{n-1}\ \mathbf{S}_{pc} \mathbf{\Lambda}^{-0.5} \qquad \textbf{(9.45)}$$

$\mathbf{Y}_c$ is the centred matrix of the original descriptors or any other set of explanatory variables that users wish to project in the PCoA biplot. **Y** may need to be transformed before it is centred and used in eq. 9.44; for example, dimensionally heterogeneous physical variables need to be standardized. $\mathbf{U}_{st}$ is the matrix of PCoA eigenvectors ($n \times c$) standardized by columns (eq. 1.12); it may contain a subset of the eigenvectors only, for example the first two. $\mathbf{S}_{pc}$ is the covariance matrix between **Y** and the standardized principal coordinates $\mathbf{U}_{st}$; in the computation of this covariance matrix, eq. 9.44 assumes that the descriptors in matrix **Y** are quantitative. The rows of matrix $\mathbf{U}_{proj}$ correspond to the p descriptors to be added to the biplot, and its columns correspond to the principal coordinates. For a PCoA conducted on a Euclidean distance matrix (D_1) computed from **Y**, the PCoA biplot with matrix **PC** for objects and matrix $\mathbf{U}_{proj}$ for descriptors is identical to a PCA distance biplot of **Y** (Subsection 9.1.4), notwithstanding possible changes of signs along some of the axes between the two analyses.

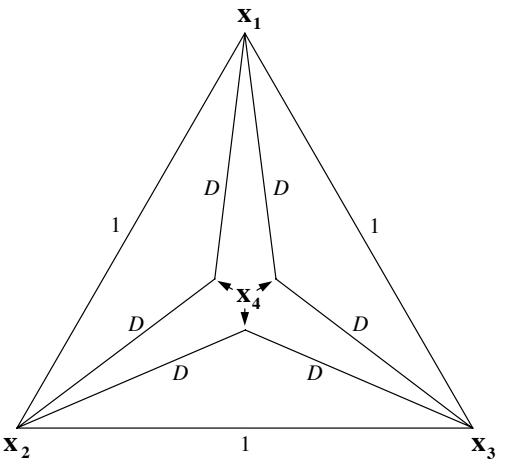


Figure 9.16 This figure, from Gower (1982), illustrates a case where the triangle inequality is not violated, yet no Euclidean representation of the four points ($\mathbf{x}_1$ to $\mathbf{x}_4$) is possible because the distances D, which are all equal, are too small for the three representations of the inner point ($\mathbf{x}_4$) to join in a single point. Assuming that the outer edges are all of lengths 1, the triangle inequality will be violated if D is smaller than 0.5. On the contrary, a two-dimensional Euclidean representation of the four points will be possible with $D = 1/\sqrt{3}$ because then the three representations of $\mathbf{x}_4$ will meet at the centroid. With $D > 1/\sqrt{3}$, the Euclidean representation of the four points, $\mathbf{x}_1$ to $\mathbf{x}_4$, will form a three-dimensional pyramid.

4 — Negative eigenvalues

There are distance matrices that do not allow a full representation of the distance relationships among objects in Euclidean space (i.e. a set of real Cartesian coordinates).

• Problems of Euclidean representation may result from the use of a distance measure that violates the triangle inequality. Such distances are called *semimetric* and *nonmetric* in Tables 7.2 and 7.3.

• Such problems may also result from an imbalance in the distance matrix, due to the handling of missing values. See for instance how missing values are handled in coefficients S_{15}, S_{16}, S_{19}, and S_{20} of Chapter 7, using Kronecker delta functions.

Non-Euclideanarity

• Some *metric* distance matrices present problems of "non-Euclideanarity", as described by Gower (1982, 1985). Figure 9.16 illustrates such a case; the closing of all individual triangles (triangle inequality condition, Section 7.4) is a necessary, but not a sufficient condition to guarantee a full Euclidean representation of a set of objects. This "non–Euclideanarity", when present, translates itself into negative eigenvalues.

For instance, most of the metric distances resulting from the transformation of a similarity coefficient using the formula $D = 1 - S$ are non-Euclidean (Table 7.2). This does not mean that all distance matrices computed using these coefficients are non-Euclidean, but that cases can be found where PCoA produces negative eigenvalues. Among the metric coefficients described in Subsection 7.4.1, several were demonstrated to be Euclidean whereas others are not Euclidean (Table 7.3).

Tables 7.2 and 7.3 show that, for many coefficients, the distances $\sqrt{D}$ or $D = \sqrt{1 - S}$ are Euclidean even though the distances D or $D = (1 - S)$ are not Euclidean. The use of $\sqrt{D}$ or the transformation $D_{hi} = \sqrt{1 - S_{hi}}$ should thus be preferred before computing PCoA using those coefficients. This transformation solves the negative eigenvalue problem even for coefficients that are known to be semimetric. This is the case, for instance, with coefficients S_8, S_{17}, and the percentage difference ($D_{14} = 1 - S_{17}$), which are widely used by ecologists to analyse tables of species presence or abundance data. A square-root transformation of $D_{14} = 1 - S_{17}$, for example, eliminates negative eigenvalues in principal coordinate analysis; see Numerical example 1 (continued) in Subsection 9.3.5. In support of this statement, Gower & Legendre (1986) have shown that coefficient S_8, which is the binary form of S_{17}, is Euclidean when transformed into $D = \sqrt{1 - S_8}$, and simulations have never turned up cases where $D = \sqrt{1 - S_{17}}$ is non-Euclidean.

When one does not wish to apply a square root transformation to the distances, or when negative eigenvalue problems persist in spite of a square root transformation, Gower & Legendre (1986) have shown that the problem of "non–Euclideanarity", and of the negative eigenvalues that come with it, can be solved by adding a (large enough) constant to all values of a distance matrix that would not lend itself to full Euclidean representation. No correction is made along the diagonal, though, because the distance between an object and itself is always zero. Actually, adding some large constant would make the negative eigenvalues disappear and produce a fully Euclidean representation, but it would also create an extra dimensions (and eigenvalue) to express the additional variance so generated. In Fig. 9.17c, for instance, adding a large value, like 0.4, to all six distances among the four points in the graph would create a pyramid, requiring three dimensions for a full Euclidean representation instead of two.

The problem is to add just the right amount to all distances in the matrix to eliminate all negative eigenvalues and produce a Euclidean representation of the distance relationships among objects, without creating unnecessary extra dimensions. Following Gower & Legendre (1986, Theorem 7*), this result can be obtained by adding a constant c to either the squared distances D_{hi}^2 or the original distances D_{hi}. This provides two methods for adjusting the original distances and correcting for their non-Euclidean behaviour.

* The present subsection corrects two misprints in theorem 7 of Gower & Legendre (1986).

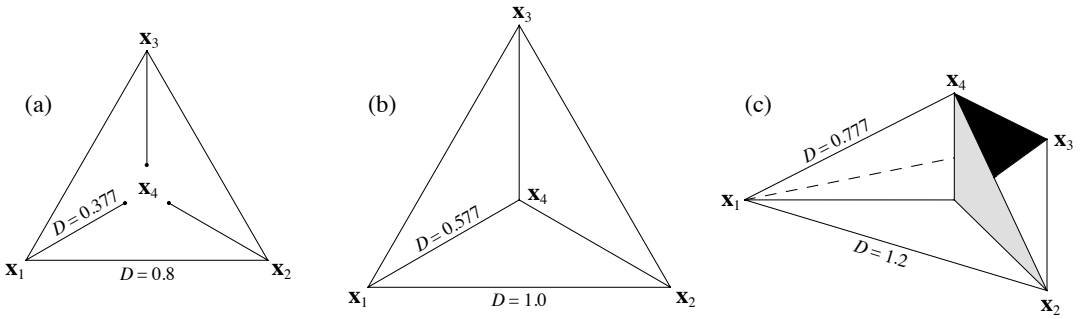


Figure 9.17 (a) Distances among four points constructed in such a way that the system cannot be represented in Euclidean space because the three lines going towards point $\mathbf{x}_4$ do not meet. (b) By adding a constant to all distances ($c_2 = 0.2$ in the present case), correction method 2 makes the system Euclidean; in this example, the distances can be associated with a representation of the points in two-dimensional space. (c) When increasing the distances further (adding again 0.2 to each distance in the present case), the system remains Euclidean but requires more dimensions for representation (three dimensions in this example).

• *Correction method 1* (derived from the work of Lingoes, 1971). — Add a constant to all squared distances D_{hi}^2, except those on the diagonal, creating a new matrix $\hat{\mathbf{D}}$ of distances $\hat{D}_{hi}$ through the following transformation:

$$\hat{D}_{hi} = \sqrt{D_{hi}^2 + 2c_1} \qquad \text{for } h \neq i \tag{9.46}$$

How to obtain c_1 is described a few lines below. Then proceed to the transformation of $\hat{\mathbf{D}}$ into matrix $\hat{\mathbf{A}}$ using eq. 9.40. The two operations may be combined into a single transformation producing the new matrix $\hat{\mathbf{A}} = [\hat{a}_{hi}]$ directly from the original distances D_{hi}:

$$\hat{a}_{hi} = -\frac{1}{2}\hat{D}_{hi}^2 = -\frac{1}{2}(D_{hi}^2 + 2c_1) = -\frac{1}{2}D_{hi}^2 - c_1 \qquad \text{for } h \neq i$$

Then, proceed with eq. 9.41 and recompute the PCoA. The constant to be added, c_1, is the *absolute value of the largest negative eigenvalue* obtained by analysing the original matrix $\mathbf{\Delta}_1$. Constant c_1 is also used in eq. 9.48 below. After correction, all non-zero eigenvalues are augmented by a value equal to c_1, so that the largest negative eigenvalue is now shifted to value 0. As a consequence, the corrected solution has two null eigenvalues (hence a maximum of $n - 2$ dimensions), or more if the matrix is degenerate. The constant c_1 is the smallest value that will produce the desired effect. Any value larger than c_1 would also eliminate all negative eigenvalues and make the system fully Euclidean, but it would also create a solution requiring more dimensions.

• *Correction method 2* (proposed by Cailliez, 1983). — Add a constant c_2 to all elements D_{hi} of matrix **D**, except those on the diagonal, creating a new matrix $\hat{\mathbf{D}}$ of distances $\hat{D}_{hi}$ through the transformation:

$$\hat{D}_{hi} = D_{hi} + c_2 \qquad \text{for } h \neq i \tag{9.47}$$

and proceed to the transformation of $\hat{\mathbf{D}}$ into matrix $\hat{\mathbf{A}}$ using eq. 9.40. The two operations may be combined into a single transformation producing the new matrix $\hat{\mathbf{A}} = [\hat{a}_{hi}]$ directly from the original distances D_{hi}:

$$\hat{a}_{hi} = -\frac{1}{2}(D_{hi} + c_2)^2 \qquad \text{for } h \neq i$$

Then, proceed with eq. 9.41 and recompute the PCoA. The constant to be added, c_2, is equal to the *largest positive eigenvalue* obtained by analysing the following special matrix, which is of order $2n$:

$$\begin{bmatrix} \mathbf{0} & 2\boldsymbol{\Delta}_1 \\ -\mathbf{I} & -4\boldsymbol{\Delta}_2 \end{bmatrix}$$

where **0** is a null matrix, **I** is an identity matrix, $\boldsymbol{\Delta}_1$ is the centred matrix defined by eqs. 9.40 and 9.41, and $\boldsymbol{\Delta}_2$ is a matrix containing values $(-0.5D_{hi})$ centred using eq. 9.41. The order of each of these matrices is n. Beware: the special matrix is asymmetric. Press *et al*. (2007) describe an algorithm to compute the eigenvalues of such a matrix. Function ***eigen()*** of R can also compute them. The solution has two null eigenvalues (hence a maximum of $n-2$ dimensions), or more if the matrix is degenerate. The constant c_2 is the smallest value that will produce the desired effect; any value larger than c_2 would also eliminate all negative eigenvalues and make the system fully Euclidean, but the solution would require more dimensions. Figure 9.17a-b shows the effect of adding constant c_2 to a non-Euclidean group of four points, and Fig. 9.17c shows the effect of adding a value larger than c_2.

The two correction methods do not produce the same Euclidean representation. This may be understood by examining the consequences of adding c_2 to the distances in **D**. When $\hat{\mathbf{D}}$ is transformed into $\hat{\mathbf{A}}$ (eq. 9.40), $(D_{hi} + c_2)$ becomes:

$$\hat{a}_{hi} = -0.5(D_{hi} + c_2)^2 = -0.5(D_{hi}^2 + 2c_2D_{hi} + c_2^2) \qquad \text{for } h \neq i$$

The effect on $\hat{a}_{hi}$ does not only depend on the value of c_2 but it also varies with each value D_{hi}. This is clearly not the same as subtracting a constant from all a_{hi} values (i.e. correction method 1). The eigenvectors resulting from one or the other correction also differ from those resulting from a PCoA without correction for negative eigenvalues. The two correction methods, and PCoA without correction, thus correspond to different partitions of the variation because the total variance, given by the trace of centred matrix $\boldsymbol{\Delta}_1$, differs among methods.

How large must constants c_1 and c_2 be for coefficient $D_{14} = 1 - S_{17}$, which is important for the analysis of species abundance data? To answer this question, Legendre & Anderson (1999) simulated species abundance data matrices. After computing distance D_{14}, the correction constants (c_1 for method 1, c_2 for method 2) increased nearly linearly with the ratio (*number of sites : number of species*). In extremely species-poor ecosystems, corrections were the largest; for instance, with a ratio 20:1 (e.g. 200 sites, 10 species), c_1 was near 0.4 and c_2 was near 0.8. When the ratio was near 1:1 (i.e. number of sites ≈ number of species), c_1 was about 0.06 and c_2 was about 0.2. In species-rich ecosystems, corrections were small, becoming smaller as the species richness increased for a constant number of sites; with a ratio 1:2 for example (e.g. 100 sites, 200 species), c_1 was near 0.02 and c_2 was about 0.1. Results also depended to some extent on the data generation parameters.

To summarize, all methods for eliminating negative eigenvalues operate by making the small distances larger, compared to the large distances, in order to allow all triangles to close (Figs. 9.16, 9.17a and b). As explained above, the first approach consists in taking the square root of all distances; this reduces the largest distances more than the small ones. The other two approaches (described above as correction methods 1 and 2) involve adding a constant to all non-diagonal distances; small distances are proportionally more augmented than large distances. In correction method 1, a constant ($2c_1$) is added to the squared distances D^2_{hi} whereas in method 2 a constant (c_2) is added to the distances D_{hi} themselves.*

Numerical example 3. Consider the numerical example used in Subsection 7.4.2 to demonstrate the semimetric nature of the percentage difference (D_{14}). The data matrix contained 3 objects and 5 species. Matrix **D**, matrix $\mathbf{A} = [-0.5 D^2_{hi}]$, and matrix $\mathbf{\Delta}_1$ are:

$$\mathbf{D} = \begin{bmatrix} 0.00000 & 0.05882 & 0.60000 \\ 0.05882 & 0.00000 & 0.53333 \\ 0.60000 & 0.53333 & 0.00000 \end{bmatrix} \quad \mathbf{A} = \begin{bmatrix} 0.00000 & -0.00173 & -0.18000 \\ -0.00173 & 0.00000 & -0.14222 \\ -0.18000 & -0.14222 & 0.00000 \end{bmatrix} \quad \mathbf{\Delta}_1 = \begin{bmatrix} 0.04916 & 0.03484 & -0.08401 \\ 0.03484 & 0.02398 & -0.05882 \\ -0.08401 & -0.05882 & 0.14283 \end{bmatrix}$$

The trace of $\mathbf{\Delta}_1$ is 0.21597. The eigenvalues are: $\lambda_1 = 0.21645$, $\lambda_2 = 0.00000$, and $\lambda_3 = -0.00049$. The sum of the eigenvalues is equal to the trace.

For correction method 1, value $c_1 = 0.00049$ is subtracted from all non-diagonal values of **A** to give $\hat{\mathbf{A}}$, which is then centred (eq. 9.41) to give the corrected matrix $\mathbf{\Delta}_1$:

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.00000 & -0.00222 & -0.18049 \\ -0.00222 & 0.00000 & -0.14271 \\ -0.18049 & -0.14271 & 0.00000 \end{bmatrix} \quad \mathbf{\Delta}_1 = \begin{bmatrix} 0.04949 & 0.03468 & -0.08417 \\ 0.03468 & 0.02430 & -0.05898 \\ -0.08417 & -0.05898 & 0.14315 \end{bmatrix}$$

The trace of the corrected matrix $\mathbf{\Delta}_1$ is 0.21694. The corrected eigenvalues are: $\lambda_1 = 0.21694$, $\lambda_2 = 0.00000$, and $\lambda_3 = 0.00000$. This Euclidean solution is one-dimensional.

* In the R language, function ***pcoa()*** in APE offers these two corrections.

For correction method 2, value $c_2 = 0.00784$, which is the largest eigenvalue of the special matrix, is added to all non-diagonal elements of matrix $\mathbf{D}$ to obtain $\hat{\mathbf{D}}$, which is then transformed into $\hat{\mathbf{A}}$ (eq. 9.40) and centred (eq. 9.41) to give the corrected matrix $\boldsymbol{\Delta}_1$:

$$\hat{\mathbf{D}} = \begin{bmatrix} 0.00000 & 0.06667 & 0.60784 \\ 0.06667 & 0.00000 & 0.54118 \\ 0.60784 & 0.54118 & 0.00000 \end{bmatrix} \quad \hat{\mathbf{A}} = \begin{bmatrix} 0.00000 & -0.00222 & -0.18474 \\ -0.00222 & 0.00000 & -0.14644 \\ -0.18474 & -0.14644 & 0.00000 \end{bmatrix} \quad \boldsymbol{\Delta}_1 = \begin{bmatrix} 0.05055 & 0.03556 & -0.08611 \\ 0.03556 & 0.02502 & -0.06058 \\ -0.08611 & -0.06058 & 0.14669 \end{bmatrix}$$

The trace of the corrected matrix $\boldsymbol{\Delta}_1$ is 0.22226. The corrected eigenvalues are: $\lambda_1 = 0.22226$, $\lambda_2 = 0.00000$, and $\lambda_3 = 0.00000$. This Euclidean solution is one-dimensional, as was the case with correction method 1.

Using the square root of coefficient D_{14}, matrices $\mathbf{D}$, $\mathbf{A}$ and $\boldsymbol{\Delta}_1$ are:

$$\mathbf{D} = \begin{bmatrix} 0.00000 & 0.24254 & 0.77460 \\ 0.24254 & 0.00000 & 0.73030 \\ 0.77460 & 0.73030 & 0.00000 \end{bmatrix} \quad \mathbf{A} = \begin{bmatrix} 0.00000 & -0.02941 & -0.30000 \\ -0.02941 & 0.00000 & -0.26667 \\ -0.30000 & -0.26667 & 0.00000 \end{bmatrix} \quad \boldsymbol{\Delta}_1 = \begin{bmatrix} 0.08715 & 0.04662 & -0.13377 \\ 0.04662 & 0.06492 & -0.11155 \\ -0.13377 & -0.11155 & 0.24532 \end{bmatrix}$$

The trace of $\boldsymbol{\Delta}_1$ is 0.39739. The eigenvalues are: $\lambda_1 = 0.36906$, $\lambda_2 = 0.02832$, and $\lambda_3 = 0.00000$. No negative eigenvalue is produced using this coefficient. This Euclidean solution is two-dimensional.

If negative eigenvalues are present in a full-dimensional PCoA solution and no correction is made to the distances to eliminate negative eigenvalues, problems of interpretation arise. Since the eigenvectors $\mathbf{u}_k$ are scaled to length $\sqrt{\lambda_k}$, it follows that the axes corresponding to negative eigenvalues are not real, but complex. Indeed, in order for the sum of squares of the u_{ik}'s in an eigenvector $\mathbf{u}_k$ to be negative, the coordinates u_{ik} must be imaginary numbers. When some of the axes of the reference space are complex, the distances cannot be fully represented in Euclidean space, as in the example of Figs. 9.16 and 9.17a.

It is, however, legitimate to investigate whether the Euclidean approximation corresponding to the positive eigenvalues (i.e. the non-imaginary principal coordinates) provides a good representation, when no correction for negative eigenvalues is applied. Cailliez & Pagès (1976) have shown that such a representation is meaningful as long as the largest negative eigenvalue is smaller, in absolute value, than any of the m positive eigenvalues of interest for representation in reduced space (usually, the first two or three).

When there are no negative eigenvalues, the quality of the representation in a reduced Euclidean space with m dimensions can be assessed, as in principal component analysis (eq. 9.5), by the R^2-like ratio:

R^2-like ratio

$$\left(\sum_{k=1}^{m} \lambda_k \right) \Big/ \left(\sum_{k=1}^{c} \lambda_k \right) \qquad (\mathbf{9.5})$$

where c is the number of positive eigenvalues. This comes from the fact that the eigenvalues of a PCoA are the same (to a factor $n - 1$) as those of a PCA performed on the coordinates of the same points in the full-dimensional space of the principal coordinates, e.g. the object coordinates in Table 9.10. Cailliez & Pagès (1976) further suggested that, when negative eigenvalues are present, a correct estimate of the quality of a reduced-space representation can be obtained by the corrected R^2-like ratio:

Corrected R^2-like ratio

$$\frac{\left(\sum_{k=1}^{m} \lambda_k\right) + mc_1}{\left(\sum_{k=1}^{n} \lambda_k\right) + (n-1)\,c_1} \qquad \textbf{(9.48)}$$

where m is the dimensionality of the reduced space, n is the order of the distance matrix (total number of objects), and c_1 is the absolute value of the largest negative eigenvalue; c_1 was found in correction method 1 above. Equation 9.48 gives the same value as if correction method 1 had been applied to the distance matrix, the PCoA had been recomputed, and the quality of the representation had been calculated using eq. 9.5. All non-zero eigenvalues would be augmented by a value equal to $c_1 = |\lambda_n|$, producing the same changes to the numerator and denominator as in eq. 9.48.

5 — Ecological applications

Principal coordinate analysis is an ordination method of great interest to ecologists because the nature of ecological descriptors often makes it necessary to use other measures of resemblance than the Euclidean distance preserved by principal component analysis or the χ^2 distance preserved by correspondence analysis (Table 9.1). Ordination methods such as principal coordinate analysis and nonmetric multidimensional scaling (Section 9.4) provide Euclidean representations of point-objects for any distance measure selected by users.

Numerical example 1 (continued from Subsection 9.2.5). From the data shown in Fig. 9.10 and Table 9.7, four Q-mode distance measures were computed among sites (Table 9.11) to illustrate some properties of principal coordinate analysis.

• Row 1 of Table 9.11 — The Euclidean distance D_1 is a symmetrical coefficient. It is not ideal for species abundance data, and it is only used here for comparison. A principal coordinate analysis of this matrix led to 19 eigenvalues: three positive (accounting for 50, 41, and 9% of the variation, respectively) and 16 null. This was expected since the original data matrix contained three variables.

• Row 2 — Distance D_{14} is often applied to species abundance data. Like its one-complement S_{17}, it excludes double-zeros. Principal coordinate analysis of this distance matrix led to 19 eigenvalues: 11 positive, one null, and 7 negative. The distance matrix was corrected using method 1 of Subsection 9.3.4, which makes use of the largest negative eigenvalue. PCoA produced 17 positive and two null eigenvalues, the largest one accounting for 31% of the variation. The distance matrix was also corrected using method 2 of Subsection 9.3.4, which

Table 9.11 Distance matrices computed from the artificial data in Fig. 9.10 and Table 9.7. Each row in this table corresponds to the first row of a distance matrix, comparing site 1 to itself and to the 18 other sites. The remaining rows of the distance matrices are not shown to save space; readers can compute these matrices from the data in Table 9.7. Values are rounded to a single decimal place.

Sampling sites	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
D_1 (Euclidean)	0.0	1.0	3.0	6.0	7.0	6.1	3.6	4.1	7.0	8.1	7.1	4.6	4.6	7.1	8.1	7.1	4.1	2.2	1.4
$D_{14} = (1 - S_{17})$	0.0	0.3	0.6	0.8	0.8	0.8	0.7	0.7	0.8	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
$\sqrt{D_{14}}$	0.0	0.6	0.8	0.9	0.9	0.9	0.8	0.8	0.9	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
D_{16} (χ^2 distance)	0.0	0.0	0.0	0.0	0.0	0.3	0.8	1.6	2.1	2.4	2.3	2.2	2.2	2.3	2.4	2.4	2.4	2.4	2.4

makes use of the largest eigenvalue of the special matrix. PCoA also produced 17 positive and two null eigenvalues, the largest one accounting for 34% of the variation.

• Row 3 — Principal coordinate analysis was also conducted using the square root of coefficient D_{14}. The analysis led to 18 positive, one null, and no negative eigenvalues, the largest one accounting for 35% of the variation.

• Row 4 — A fourth distance matrix was computed using the χ^2 distance D_{16}, which excludes double-zeros. Principal coordinate analysis produced 19 eigenvalues: two positive (accounting for 4 and 36% of the variation, respectively) and 17 null. The χ^2 distance (D_{16}) is the coefficient preserved in correspondence analysis (CA, Section 9.2), which would also produce two positive eigenvalues with these data. Indeed, CA always produces one dimension less than the original number of species, or fewer in the case of degenerate matrices.

This example shows that different distance measures may lead to very different numbers of dimensions of the Euclidean representations. In the analyses reported here, the numbers of dimensions obtained were 3 for distance D_1, 11 for uncorrected D_{14} (not counting the complex axes corresponding to negative eigenvalues), 17 for D_{14} after correction by the largest negative eigenvalue, 18 for the square root of D_{14}, and 2 for D_{16}.

For the example data, the PCA ordination (Fig. 9.11a, b) is identical to the ordination that would have been obtained from PCoA of a matrix of Euclidean distances among sites, as shown in Numerical example 2 (Subsection 9.3.2). In the same way, the ordination of sites in the CA plot (Fig. 9.11c), which used scaling type 1, is similar to a PCoA ordination that would be obtained from a matrix of χ^2 distances (D_{16}) among sites. The ordinations (Fig. 9.18) obtained from distance coefficients D_{14} and $\sqrt{D_{14}}$ are also of interest because these coefficients are often used to analyse community composition data; they are illustrated in Fig. 9.18a-d. The ordinations produced by these coefficients are quite similar to each other and present horseshoes, which are not as pronounced as in PCA because coefficients D_{14} and $\sqrt{D_{14}}$ exclude double-zeros from the calculations. In Fig. 9.18a (coefficient D_{14}), sites 6 to 14 form an arch depicting the three-species gradient, with arms extending in a perpendicular

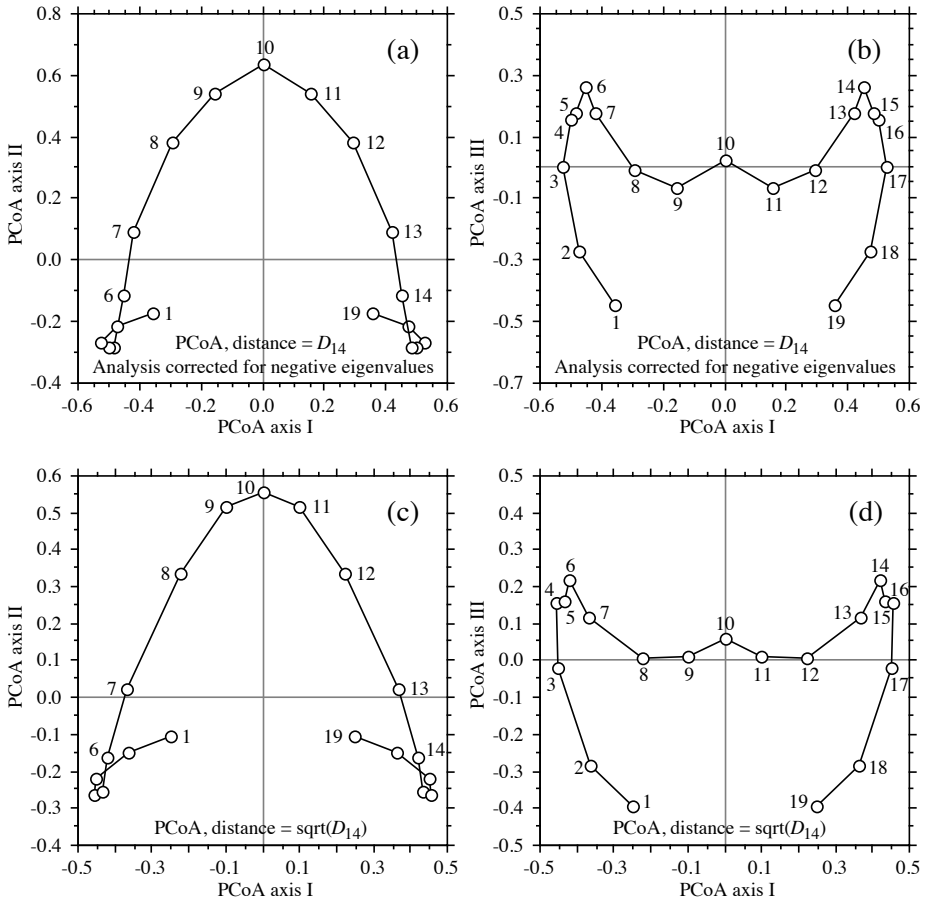


Figure 9.18 Principal coordinate ordinations of the data in Fig. 9.10 and Table 9.7. Distance D_{14}, analysis corrected for negative eigenvalues: (a) PCoA axes I and II ($\lambda_1 = 30.8\%$, $\lambda_2 = 18.6\%$), (b) axes I and III ($\lambda_1 = 30.8\%$, $\lambda_3 = 8.3\%$). Distance $\sqrt{D_{14}}$: (c) PCoA axes I and II ($\lambda_1 = 34.5\%$, $\lambda_2 = 22.9\%$), (d) axes I and III ($\lambda_1 = 34.5\%$, $\lambda_3 = 10.5\%$).

direction (axis III, Fig. 9.18b) to account for the dispersion of sites 1 to 5 and 15 to 19, each group containing one species only, as shown in Table 9.7. The ordination produced by coefficient $\sqrt{D_{14}}$ is very similar to the above (Fig. 9.18c-d). There are two advantages to $\sqrt{D_{14}}$ over D_{14}, though: $\sqrt{D_{14}}$ never produces negative eigenvalues and, in the present case at least, the ordination explains more variation than D_{14} in two or three dimensions.

Only the Euclidean distance and derived coefficients lead to a number of principal axes equal to the original number of descriptors. Other coefficients may produce fewer, or more axes. The dimensionality of a principal coordinate space is a function of the

number of original descriptors, mediated through the distance measure that was selected for the analysis.

There are many applications of principal coordinate analysis in the ecological literature. This method may be used in conjunction with clustering techniques; an example is presented in Ecological application 10.1. Direct applications of the method are summarized in Ecological applications 9.3a and 9.3b. The application of principal coordinate analysis to the clustering of large numbers of objects is discussed in Subsection 8.7.3.

Ecological application 9.3a

Field & Robb (1970) studied the molluscs and barnacles from a rocky shore (21 quadrats) in False Bay, South Africa, in order to determine the influence of factors *emergence* (the height on the shore relative to the mean sea level) and *wave* on these communities. Quadrats 1 to 10, on a transect parallel to the shoreline, differed in their exposure to wave action; quadrats 11 to 21, located on a transect at right angle to the shoreline, covered the spectrum between the mean high and mean low waters of spring tides. 79 species were enumerated, one reaching 10864 individuals in a single quadrat. When going up the shore, quadrats had progressively larger numbers of individuals and smaller numbers of species. This illustrates the principle that increasing environmental stress (here, the *emergence* factor) is accompanied by decreasing diversity. It also shows that the few species that can withstand a high degree of stress do not encounter much interspecific competition and may therefore become very abundant.

The same principal coordinate ordination could have been obtained by estimating species abundances with a lesser degree of precision, e.g. using classes of abundance. Table 7.4 gives the association measures that would have been appropriate for such data.

Species abundances (y'_{ij}) were first normalized by logarithmic transformation $y''_{ij} = \log_e (y'_{ij} + 1)$, and centred ($y_{ij} = y''_{ij} - \bar{y}_i$), to form matrix $\mathbf{Y} = [y_{ij}]$ containing the data to be analysed. Scalar products among quadrat vectors were used as measures of similarity:

$$\mathbf{S}_{n \times n} = \mathbf{Y}_{n \times p} \mathbf{Y}'_{p \times n}$$

Principal coordinates were computed using a variant procedure proposed by Orlóci (1966). Figure 9.19 displays the ordination of quadrats 1 to 19 in the space of the first two principal coordinates. The ordination was also calculated including quadrats 20 and 21 but, since these came from the highest part of the shore, they introduced so much variation in the analysis that the factor *emergence* dominated the first two principal coordinates. For the present ecological application, only the ordination of quadrats 1 to 19 is shown. The authors looked for a relationship between this ordination and the two environmental factors, by calculating Spearman's rank correlation coefficients (eq. 5.3) between the ranks of the quadrats on each principal axis and their ranks on the two environmental factors. This showed that the first principal axis had a significant correlation with elevation with respect to the shoreline (*emergence*), and the second axis was significantly related to *wave action*. The authors concluded that PCoA is well adapted to the study of ecological gradients, provided that the data set is fairly homogeneous. (Correspondence analysis, described in Section 9.2, would have been another appropriate way of obtaining an ordination of these quadrats.)

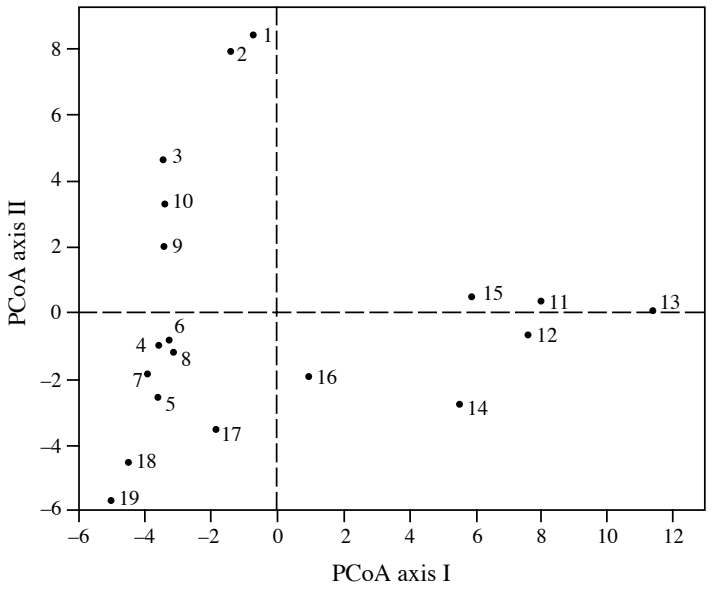


Figure 9.19 Ordination of quadrats 1 to 19 in the space of the first two principal coordinates (PCoA axes I and II). Modified from Field & Robb (1970).

Ecological application 9.3b

Ardisson *et al*. (1990) investigated the spatio-temporal organization of epibenthic communities in the Estuary and Gulf of St. Lawrence, an area ca. 1150 × 300 km. Quantitative data were obtained over 8 years, between 1975 and 1984, from 161 collectors (summer navigation buoys) moored yearly from May through November, and retrieved by the Canadian Coastguard before winter ice formation.

Each year was represented by a data table of 161 sites (buoys) × 5 dominant species (dry biomass). A similarity matrix among sites was computed for each year separately, using the asymmetrical form of the Gower similarity coefficient (S_{19}). The eight yearly matrices were compared to one another using Mantel statistics (Subsection 10.5.1). A principal coordinate analysis (Fig. 9.20) was conducted on the resulting matrix of (1 – Mantel) statistics to determine whether year-to-year differences were random or organized. The among-year pattern of dispersion suggested the existence of a cycle of variation whose length was about equal to the duration of the study. This cycle might represent the response of the Estuary-Gulf system, as an integrated unit, to external inputs of auxiliary energy, although the specific causal process, physical or biotic, remains unknown.

Mantel statistic

The same type of analysis as in this ecological application, i.e. comparing several data matrices about the same objects, could be based on RV coefficients (eq. 11.66) computed between all pairs of data matrices. The corresponding distance-like coefficients (1 – RV) would be assembled in a square distance matrix and PCoA would be computed on that matrix to obtain an ordination of the type illustrated in Fig. 9.20.

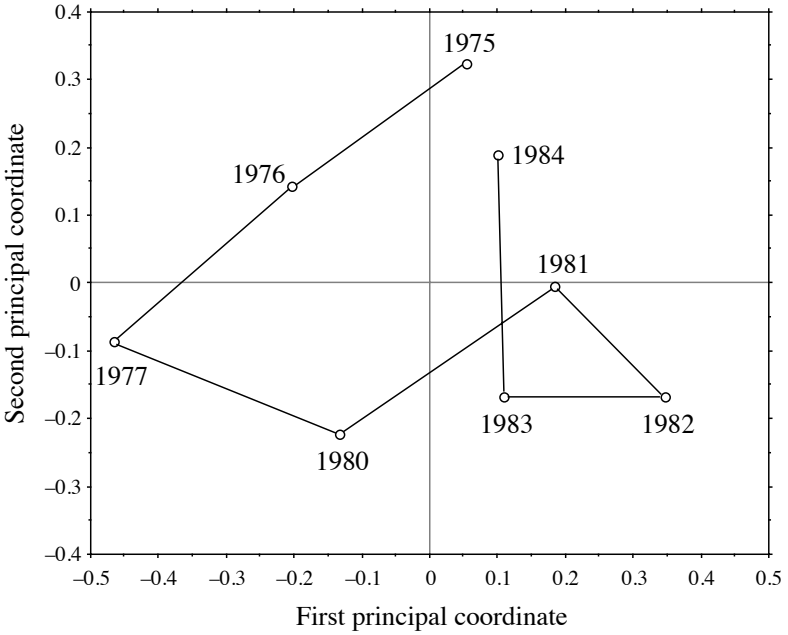


Figure 9.20 Among-year variability illustrated in the space of the first two principal coordinates, obtained from analysing a matrix of Mantel statistics comparing yearly similarity matrices. Recomputed from the Mantel statistic values provided in Fig. 8 of Ardisson *et al*. (1990).

This last ecological application showed that the usefulness of principal coordinate analysis is not limited to projecting, in reduced space, classical resemblance matrices among objects. In that example, the relationships among data tables, as expressed by Mantel statistics, were represented in a Euclidean space using PCoA. The method may actually be applied to any type of symmetric resemblance matrices. This includes cases where the measures of resemblance are obtained directly from observation (e.g. interaction matrices in behavioural studies) or from laboratory work (DNA hybridisation results, serological data, etc.). If the resulting matrix is non-symmetric, it may be decomposed into a symmetric and a skew-symmetric component (Section 2.3), which can be analysed separately by PCoA.

6 — Algorithm

Principal coordinate analysis is easy to compute for any distance matrix, using a standard eigenanalysis function. Follow the steps summarized in Table 9.12.

Table 9.12 Computing principal coordinates.

a) Centre the distance matrix

Transform and centre the distance matrix following Gower's method (eqs. 9.40 and 9.41).

b) Compute the eigenvalues and eigenvectors

Use an eigen-decomposition function†.

c) Final scaling

Scale each eigenvector k to length $\sqrt{\lambda_k}$ to obtain the principal coordinates.

† For a matrix of Euclidean distances D_1, the eigenvalues obtained from PCoA are larger than those of PCA by a factor $(n - 1)$.

9.4 Nonmetric multidimensional scaling (nMDS)

The reduced-space ordination methods of the previous sections produced an ordination (scaling) of the objects in full-dimensional space. Users could then select the first few dimensions and check how well the distance relationships among the objects were preserved in that reduced space. There are cases, however, where the exact preservation of the distances among objects is not of primary importance, the priority being instead the representation of the objects in a small and specified number of dimensions, usually two or three. In such cases, the objective is to plot dissimilar objects far apart in the ordination space and similar objects close to one another. This is called the *preservation of ordering relationships* among objects. The method to do so is called *nonmetric multidimensional scaling* (nMDS, or simply MDS). It was devised by psychometricians Shepard (1962, 1966) and Kruskal (1964a, b). Programs for nMDS were originally distributed by Bell Laboratories in New Jersey, where the method originated; see Carroll (1987) for a review. The method is now available in several major (SPSS, SAS, SYSTAT, etc.) and specialized computer packages as well as in R*. A useful reference is the book of Kruskal & Wish (1978). Relationships between nMDS and other forms of ordination have been described by Gower (1987). Extensions of nMDS to several matrices, weighted models, the analysis of preference data, etc. are discussed by Young (1985) and Carroll (1987). A form of *hybrid scaling*, combining metric and nonmetric scaling criteria, was proposed by Faith *et al*. (1987); it was further explained in Belbin (1991) and is available in packages DECODA (written by Peter R. Minchin) and PATN.

Like principal coordinate analysis (PCoA), nMDS is not limited to Euclidean distance matrices; it can produce ordinations of objects from any distance matrix. The method can also proceed with missing distances — actually, the more missing distances there are, the easier the computations — as long as there are enough measures left to position each object with respect to a few of the others. This feature makes it a method of choice for the analysis of matrices obtained by direct observation (e.g. behaviour studies) or laboratory assays, where missing pairwise distances often occur. Some programs can handle non-symmetric distance matrices, for which they provide a compromise solution between distances in the upper and lower triangular parts of the matrix. Contrary to PCA, PCoA, or CA, which are eigenvector-based methods, nMDS calculations do not maximize the variability associated with individual axes of the ordination; nMDS axes are arbitrary, so that plots may arbitrarily be rotated, centred, or inverted. Reasons for this will become clear from the presentation of the method.

Consider a distance matrix $\mathbf{D}_{n\times n} = [D_{hi}]$ computed using a measure appropriate to the data at hand (Chapter 7). Matrix **D** may also result from direct observations, e.g. affinities among individuals or species found in serological, DNA pairing, or behavioural studies; these matrices may be non-symmetric. Nonmetric multidimensional scaling of matrix **D** may be summarized in the following steps.

1) Specify the number *m* of dimensions chosen *a priori* for scaling the objects. The output will provide coordinates of the *n* objects on *m* axes. If several configurations for different numbers of dimensions are sought — say, 2, 3, 4, and 5 dimensions, they must be computed separately. Several programs actually allow solutions to cascade from high to low numbers of dimensions — for instance from 4 to 3 to 2 to 1.

2) Construct an initial configuration of the objects in *m* dimensions, to be used as a starting point for the iterative adjustment process of steps 3 to 7. The way this initial configuration is chosen is critical because the solution on which the algorithm eventually converges depends to some extent on the initial positions of the objects. The same problem was encountered with *K*-means partitioning (Section 8.8); the "space of

---

* nMDS is available in R functions listed in Section 9.5. It is also found in the following commercially available packages (list not exhaustive):

- NTSYSpc. Distribution: see footnote in Section 7.8.
- PATN. Distribution: see footnote in Section 7.8.
- PRIMER. That package was developed by M. R. Carr and K. R. Clarke at the Plymouth Marine Laboratory, Prospect Place, West Hoe, Plymouth PL1 3DH, Great Britain.
- PC-ORD. Distribution: see footnote in Section 11.7. Besides nMDS, PC-ORD contains many other methods of multivariate ecological data analysis.

Local minimum

solutions" may contain several local minima besides the overall minimum (Fig. 8.17). The usual solutions to this problem are the following:

- Run the program several times, starting from different random initial placements of the objects. The solution minimizing the objective function (step 5) is retained.

- Initiate the run from an ordination obtained using another method, e.g. PCoA.

- If the data are thought to be spatially structured and the geographic positions of the objects are known, these geographic positions may be used as the starting configuration for nMDS of a matrix **D** computed from the data.

- Work step by step from higher to lower dimensionality. Compute, for instance, a first nMDS solution in 5 dimensions from a random initial placement of the objects. Note the stress value (eqs. 9.49 to 9.51), which should be low because the high number of dimensions imposes little constraint to the distances. Use 4 of the 5 dimensions so obtained as the initial configuration for a run in 4 dimensions, and so forth until the desired number (m) of ordination dimensions is reached.

3) Calculate a matrix of fitted distances d_{hi} in the ordination space, using one of Minkowski's metrics (D_6, eq. 7.43). Most often, one chooses the second degree of Minkowski's metric, which is the Euclidean distance. (a) In the first iteration, distances d_{hi} are computed from the initial (often random) configuration. (b) In subsequent iterations, the configuration is that obtained in step 6.

Shepard diagram

4) Consider the Shepard diagram (Figs. 9.1 and 9.21b) comparing the fitted distances d_{hi} to the empirical (i.e. original) distances D_{hi}. Regress d_{hi} on D_{hi}. Values forecasted by the regression line are called $\hat{d}_{hi}$. The choice of the type of regression is left to the users, given the choices implemented in the computer program. Usual choices are the linear, polynomial, or monotone regressions (also called "nonparametric", although there are other types of nonparametric regression methods).

Monotone regression is a step-function constrained to always increase from left to right (Fig. 9.21b); this is a common choice in nMDS. A monotone regression is equivalent to a linear regression performed after monotonic transformation of the original distances D_{hi}, so as to maximize the linear relationship between D_{hi} and d_{hi}. The regression is fitted by least squares.

Tied values

If there are tied values among the empirical distances, Kruskal (1964a, b) proposed two approaches that may be followed in monotone regression. Ties are likely to occur when the empirical distances D_{hi} are computed from a table of raw data using one of the coefficients described in Chapter 7; they are less likely to occur when distances result from direct observations. In Fig. 9.21b, for instance, there are ties for several of the values on the abscissa; the largest number of ties is found at $D = D_{max} = 1$.

- In Kruskal's *primary approach*, one accepts the fact that, if an empirical distance D_{hi} corresponds to different fitted values d_{hi}, it also corresponds to different forecasted

values $\hat{d}_{hi}$. Hence the monotone regression line is allowed to go straight up in a column of tied values, subject to the constraint that the regression line is not allowed to decrease compared to the previous values D. The monotone regression line is not a mathematical function in that case, however. In order to insure monotonicity, the only constraint on the $\hat{d}_{hi}$ values is:

$$\text{when } D_{gi} < D_{hi}, \text{ then } \hat{d}_{gi} \leq \hat{d}_{hi}$$

• In the *secondary approach*, the forecasted value $\hat{d}_{hi}$ is the same for all fitted distances d_{hi} that are tied to a given empirical distance value D_{hi}. To insure monotonicity, the constraints on the $\hat{d}_{hi}$ values are:

$$\text{when } D_{gi} < D_{hi}, \text{ then } \hat{d}_{gi} \leq \hat{d}_{hi}$$

$$\text{when } D_{gi} = D_{hi}, \text{ then } \hat{d}_{gi} = \hat{d}_{hi}$$

In this approach, the least-squares solution for $\hat{d}_{hi}$ is the mean of the tied d_{hi}'s when considering a single value D_{hi}. The vertical difference in the diagram between d_{hi} and $\hat{d}_{hi}$ is used as the contribution of that point to the stress formula, below. In Fig. 9.21b, the secondary approach is applied to all tied values found for $D_{hi} < (D_{max} = 1)$, and the primary approach when $D_{hi} = D_{max} = 1$.

Computer programs may differ in the way they handle ties. This may cause major differences between reported stress values corresponding to the final solutions, although the final configurations of points are usually very similar from program to program, except when two programs identify different final solutions having very similar stress values.

A reduced-space scaling would be perfect if all points in the Shepard diagram fell exactly on the regression line (straight line, smooth curve, or step-function); the rank ordering of the fitted distances d_{hi} would be exactly the same as that of the original distances D_{hi} and the value of the objective function (step 5) would be zero.

Objective function

5) Measure the goodness-of-fit of the regression using an objective function. All objective functions used in nMDS are based on the sum of the squared differences between fitted values d_{hi} and the corresponding values $\hat{d}_{hi}$ forecasted by the regression function; this is the usual sum of squared residuals of regression analysis (least-squares criterion, Subsection 10.3.1). Several variants have been proposed and are used in nMDS programs:

$$\textit{Stress} \text{ (formula 1)} = \sqrt{\sum_{h,i} (d_{hi} - \hat{d}_{hi})^2 / \sum_{h,i} d_{hi}^2} \qquad \textbf{(9.49)}$$

$$\textit{Stress} \text{ (formula 2)} = \sqrt{\sum_{h,i} (d_{hi} - \hat{d}_{hi})^2 / \sum_{h,i} (d_{hi} - \bar{d})^2} \qquad \textbf{(9.50)}$$

$$Sstress = \sqrt{\sum_{h,i} (d_{hi}^2 - \hat{d}_{hi}^2)^2} \quad \textbf{(9.51)}$$

The denominators in the two *Stress* formulas (eqs. 9.49 and 9.50) are scaling terms that make the objective functions dimensionless and produce *Stress* values between 0 and 1. These objective functions may apply the square root, or not, without changing the issue; a configuration that minimizes these objective functions would also minimize the non-square-rooted forms. Other objective criteria, such as *Strain*, have been proposed. All objective functions measure how far the reduced-space configuration is from being monotonic to the original distances D_{hi}. Their values are only relative, measuring the decrease in lack-of-fit between iterations of the calculation procedure.

Steepest descent

6) Improve the configuration by moving it slightly in a direction of decreasing stress. This is done by a numerical optimization algorithm called the *method of steepest descent*; the method is explained, for instance, in *Numerical Recipes* (Press *et al*., 2007) and in Kruskal (1964b). The direction of steepest descent is the direction in the space of solutions along which stress is decreasing most rapidly. This direction is found by analysing the partial derivatives of the stress function (Carroll, 1987). The idea is to move points in the ordination plot to new positions that are likely to decrease the stress most rapidly.

7) Repeat steps 3 to 6 until the objective function reaches a small, predetermined value (tolerated lack-of-fit), or until convergence is achieved, i.e. until it reaches a minimum and no further progress can be made. The coordinates calculated at the last passage through step 6 become the coordinates of the *n* objects in the *m* dimensions of the multidimensional scaling ordination.

8) Most nMDS programs rotate the final solution using principal component analysis, for easier interpretation.

In most situations, users of nMDS decide that they want a representation of the objects in two or three dimensions, for illustration or other purpose. In some cases, however, one wonders what the "best" number of dimensions would be for a data set, i.e. what would be the best compromise between a summary of the data and an accurate representation of the distances. As pointed out by Kruskal & Wish (1978), determining the dimensionality of an nMDS ordination is as much a substantive as a statistical question. The substantive aspects concern the interpretability of the axes, ease of use, and stability of the solution. The statistical aspect is easier to approach since stress may be used as a guide to dimensionality. Plot the *stress* values as a function of *dimensionality* of the solutions, using one of the stress formulas above (eqs. 9.49 to 9.51). Since stress decreases as dimensionality increases, choose for the final solution the dimensionality where the change in stress becomes small.

For species count data, Faith *et al*. (1987) have shown, through simulations, that the following strategy yields informative ordination results: (1) standardize the data by

dividing each value by the maximum abundance for that species in the data set; (2) use the Steinhaus (S_{17}) or the Kulczynski (S_{18}) similarity measure; (3) compute the ordination by nMDS.

Besides the advantages mentioned above for the treatment of nonmetric distances or non-symmetric matrices (see also Sections 2.3 and 8.10 on this topic), Gower (1966) pointed out that nMDS can summarize distances in fewer dimensions than principal coordinate analysis (i.e. lower stress in, say, two dimensions). Results of the two methods may be compared by examining Shepard diagrams of the results obtained by PCoA and nMDS, respectively. If the scatter of points in the Shepard diagram for PCoA is narrow, as in Fig. 9.1a or b, the reduced-space ordination is useful in that it correctly reflects the relative positions of the objects. If the scatter is wide or nearly circular (Fig. 9.1c), the ordination diagram is of little use and one may try nMDS to find a more satisfactory solution in a few dimensions. A PCoA solution remains easier to compute in most cases, however, because it does not require multiple runs, and it is obtained using a direct eigenanalysis algorithm instead of an iterative procedure.

Numerical example 1 (continued from Subsections 9.2.5 and 9.3.5)**.** The percentage difference distance matrix (D_{14}) computed in Table 9.11 was subjected to nMDS analysis using the package DECODA written by Peter R. Minchin. This nMDS program uses Stress formula 1 (eq. 9.49). Repeated runs, using $m = 2$ dimensions but different random starting configurations, produced very similar results; the best one had a stress value of 0.0181 (Fig. 9.21a).

Kruskal's secondary approach, explained with computation step 4 above, was used in Fig. 9.21b for all tied values found when $D_{hi} < D_{max}$, while the primary approach was used when $D_{hi} = D_{max} = 1$. The rationale for this follows from the fact that the empirical distances D_{hi} are blocked by an artificial ceiling D_{max} of the distance function, over which they cannot increase. So, pairs of sites tied at distance $D_{max} = 1$, for which d_{hi} is larger than the previous value $\hat{d}$, are not expected to be the same distance apart in the ordination. Hence these values should not contribute to the stress despite their ties.

Using $\sqrt{D_{14}}$ as the distance measure, instead of D_{14}, produced an identical ordination, since nMDS is invariant to monotonic transformations of the distances. The stress value did not change either, because the square root transformation of D_{14} affects only the abscissa of Fig. 9.21b, whereas the stress is computed along the ordinate. The arch effect found in Fig. 9.18a does not appear in Fig. 9.21a. The horizontal axis of the nMDS ordination reproduces the original gradient almost perfectly in this example.

Points in an nMDS plot may be rotated, translated, inverted, or scaled *a posteriori* in any way considered appropriate to achieve maximum interpretability or to illustrate the results. This may be done either by hand or, for example, through canonical analysis of the nMDS axes with respect to a set of explanatory variables (Chapter 11).

With the present data, a one-dimensional ordination (stress = 0.1089) perfectly reconstructed the gradient of sites 1 to 19; the same ordination was always obtained when repeating the run from different random starting configurations and cascading from 3 to 2 to 1 dimensions. This configuration, and the low stress value, were hardly ever obtained when performing the nMDS ordination directly in one dimension, without the cascading procedure.

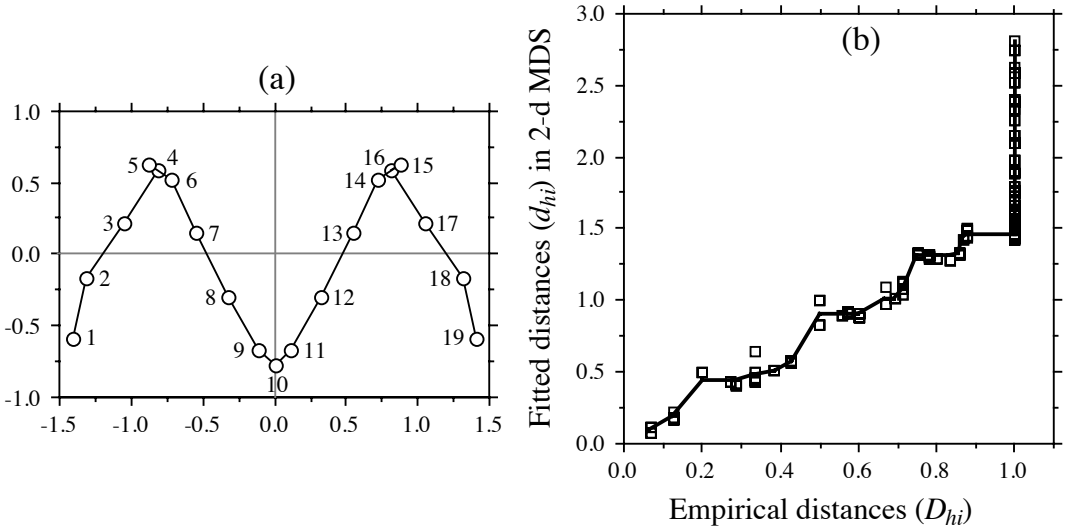


Figure 9.21 (a) nMDS ordination (2-dimensional) of the D_{14} distance matrix in Table 9.11. Sampling sites are numbered as in Fig. 9.10 and Table 9.7. (b) Shepard diagram of the final solution (2-dimensional) showing the monotone regression line fitted by nonparametric regression. The scatter about the line is measured by a stress function (eq. 9.49 to 9.51).

Ecological application 9.4a

Sprules (1980) used nonmetric multidimensional scaling to analyse seasonal changes in zooplankton assemblages at a site located in Lake Blelham, in the Lake District of northern England, and in two experimental enclosures built in that lake. The three sites were surveyed on a weekly basis from June to December 1976. nMDS was preferred to PCA because the responses of the species to environmental gradients could not be assumed to be linear.

For each site, points in the nMDS ordination diagram were connected in chronological order to reflect the seasonal changes in faunal composition. The plot (not reproduced here) is therefore of the same type as Fig. 12.24. In one of the enclosures, the assemblage oscillated about a mean value without any clear cycle; small-size species dominated the assemblage. In the other enclosure and in the lake, changes were more directional; at these sites, predators were more abundant. Based on available evidence, Sprules concluded that the differences observed between the two patterns of seasonal change were related to differences in predation intensity, quality of food available to herbivores, and nutrient dynamics.

Ecological application 9.4b

Redford *et al*. (2010) studied the bacteria living on the surfaces of leaves of 56 tree species found on the University of Colorado campus in Boulder, Colorado, USA. The bacterial communities were characterized by barcoded pyrosequencing. The authors analysed the intra-

and inter-tree-species variation in bacterial community composition. They found the bacterial communities to be more similar within a given tree species and among closely related species than among phylogenetically distant host species. The results were illustrated by nMDS plots computed from matrices of phylogenetic and community ecology distances among bacterial communities. The authors also compared the bacterial communities found on *Pinus ponderosa* needles from different locations with geographic distances ranging from 10 m to more than 10000 km; the bacterial floras of the leaves of three other tree species from Boulder were included in this analysis. The authors found that the bacterial communities were more similar among *P. ponderosa* trees from different locations than among different host species from the University of Colorado campus in Boulder.

Ecological application 9.4c

The relationship between snake community composition (46 sampling sites, 43 species) and environmental variables was investigated by de Fraga *et al*. (2011) in a 25 km^2 portion of Reserva Ducke near Manaus (Amazonas, Brazil) in the rain forest. The authors computed the first nMDS axis of a distance matrix based on the presence-absence of the species (Chao index, Subsection 7.3.4), and related it to the distance from streams in a dispersion diagram. The plot showed that snake community composition was fairly similar among sites close to streams, but varied much more for distances greater than 100 m from streams. Direct ordination of presence and absence data for 26 common snake species across all plots indicated a gradual substitution of species with distance from the streams.

Many ecological applications of nonmetric multidimensional scaling are found in the ecological literature. Two papers are especially interesting: Whittington & Hughes (1972; Ordovician biogeography from the analysis of trilobite fauna), and Fasham (1977; comparison of nonmetric multidimensional scaling, principal coordinate analysis and correspondence analysis for the ordination of simulated coenoclines and coenoplanes). Ecological application 12.6b (Subsection 12.6.5) features a nMDS plot.

9.5 Software

All general-purpose statistical packages offer principal component analysis, but not with options for the scalings used by ecologists. Among the specialized packages recommended for ecological analysis are CANOCO* (ter Braak & Smilauer, 1998 and later editions), PC-ORD and SYN-TAX 2000. For distribution of these programs, see Section 11.7.

* CANOCO was often referred to in this chapter. It was the first program to offer ecologists a whole array of simple and canonical ordination methods and it is still the reference for developers of ordination programs.

Several R-language packages offer functions for ordination of multivariate data.

1. Principal component analysis (PCA). — Functions ***dudi.pca()*** in ADE4, ***PCA()*** in FACTOMINER, ***pca()*** in LABDSV, ***prcomp()*** in STATS, and ***rda()*** in VEGAN compute PCA. Compare the PCA eigenvalues to the broken-stick model (eq. 9.16): ***PCAsignificance()*** in BIODIVERSITYR and ***bstick()*** in VEGAN. A PCA biplot with an equilibrium contribution circle is drawn by ***ordiequilibriumcircle()*** of BIODIVERSITYR. Fuzzy PCA and CA are available in ***dudi.fca()*** of ADE4.

2. Correspondence analysis (CA). — Functions ***dudi.coa()*** of ADE4, ***ca()*** of CA, ***CA()*** of FACTOMINER, ***corresp()*** of MASS and ***cca()*** of VEGAN compute CA. Multiple correspondence analysis (MCA): functions ***dudi.acm()*** of ADE4, ***mjca()*** of CA, ***MCA()*** of FACTOMINER, and ***mca()*** of MASS.

3. Principal coordinate analysis (PCoA). — Functions ***dudi.pco()*** of ADE4, ***pcoa()*** of APE, ***pco()*** of ECODIST, ***pco()*** of LABDSV, ***pcoa()*** and ***pcoa.all()*** of PCNM, ***cmdscale()*** of STATS, and ***wcmdscale()*** of VEGAN compute PCoA. Function ***pcoa.all()*** of package PCNM allows the computation of principal coordinates from distance matrices that have non-zero diagonals, which will be useful in Section 14.2. It also contains an option to output the eigenvectors corresponding to negative eigenvalues; in that case, the eigenvectors are not scaled to lengths of $\sqrt{\lambda_k}$ since that would produce complex vectors, but are kept normalized to lengths 1. Function ***is.euclid()*** of ADE4 checks the Euclidean nature of distance matrices; see Tables 7.2 and 7.3.

4. Nonmetric multidimensional scaling (nMDS). — Functions ***nmds()*** of ECODIST, ***nmds()*** of LABDSV, ***isoMDS()*** of MASS, and ***metaMDS()*** of VEGAN compute nMDS.

R functions for PCA and CA that follow and illustrate the algebra described in Sections 9.1 and 9.2 are available on the page http://numericalecology.com/rcode/.