Chapter 16. Correspondence Analysis (CA)

1. CA was first proposed to analyze contingency table. Contingency table is a common form to summarize species data (e.g., a 2×2 table classified by site and species). Data in each cell of the table is frequency of a species. The values of the frequency must be nonnegative.

   In general, correspondence analysis may be applied to any data table that is dimensionally homogeneous (i.e., the physical dimensions of all variables are the same so that at least addition makes sense here.). Certainly, all of the values must be ≥ 0. Site×species tables are such tables in which cell frequencies can be **presence/absence** or **abundance**.

   CA is appropriate for two situations: (1) To analyze tables with a lot of zeros because the χ^2 **distance** can easily handle (remove) double-zero (see below); (2) if there is a long gradient.

2. A typical site×species data table:

> dune

	Belper	Empnig	Junbuf	Junart	Airpra	Elepal	Rumace	Viclat	Brarut	Ranfla	Cirarv
X2	3	0	0	0	0	0	0	0	0	0	0
X13	0	0	3	0	0	0	0	0	0	2	0
X4	2	0	0	0	0	0	0	0	2	0	2
X16	0	0	0	3	0	8	0	0	4	2	0
X6	0	0	0	0	0	0	6	0	6	0	0
X1	0	0	0	0	0	0	0	0	0	0	0
X8	0	0	0	4	0	4	0	0	2	2	0
X5	2	0	0	0	0	0	5	0	2	0	0
X17	0	0	0	0	2	0	0	0	0	0	0
X15	0	0	0	3	0	5	0	0	4	2	0
X10	2	0	0	0	0	0	0	1	2	0	0
X11	0	0	0	0	0	0	0	2	4	0	0
X9	0	0	4	4	0	0	2	0	2	0	0
X18	2	0	0	0	0	0	0	1	6	0	0
X3	2	0	0	0	0	0	0	0	2	0	0
X20	0	0	0	4	0	4	0	0	4	4	0
X14	0	0	0	0	0	4	0	0	0	2	0
X19	0	2	0	0	3	0	0	0	3	0	0
X12	0	0	4	0	0	0	2	0	4	0	0
X7	0	0	2	0	0	0	3	0	2	0	0

(1) χ^2 **distance** – is used to calculate distance among sites using species abundance or other frequency data. No negative values are allowed in the data.

$$D(1,2) = \sqrt{\sum_{j=1}^{p} \frac{1}{x_{+j}/x_{++}} \left(\frac{x_{1j}}{x_{1+}} - \frac{x_{2j}}{x_{2+}} \right)^2} = \sqrt{x_{++}} \sqrt{\sum_{j=1}^{p} \frac{1}{x_{+j}} \left(\frac{x_{1j}}{x_{1+}} - \frac{x_{2j}}{x_{2+}} \right)^2}$$

where x_{1j} is cell frequency in a frequency table for site 1, while x_{2j} is cell frequency in a frequency table for site 2. x_{1+} and x_{2+} are row totals. x_{+j} is column total. x_{++} is the total sum of the frequency. The χ^2 distance is the difference between two profiles weighted by column sum (x_{+j}) and then times a constant $\sqrt{x_{++}}$. Or it can be interpreted as the difference between two profiles and then weighted by relative frequency (a probability) $\sqrt{\frac{1}{x_{+j}/x_{++}}}$.

This χ^2 **distance** can remove the effect of double zero.

3. CA is primarily a method of ordination. As such, it is similar to PCA, but it preserves, in principal axes (i.e., after rotation), the Euclidean distance between profiles of weighted conditional probabilities. This is equivalent to the χ^2 distance between the row and columns of the contingency table.

   Besides the role of as an ordination and dimension reduction method, CA may be used for studying the proximities between rows (or columns) of a contingency table.

4. **CA computation** – 3 major steps are involved:
   (1) Data standardization (the contingency table will be transformed into a table of contributions to the Pearson χ^2 statistic after fitting the null model of row-column independence to the contingency table.
   (2) Singular value decomposition is applied in order to compute eigenvalues and eigenvectors, as in PCA.
   (3) Further matrix operations will lead to various tables that are needed for plotting useful output diagrams.

Steps:

(1) Standardized contingency table using the Pearson χ^2

$$\chi_{ij} = \frac{O_{ij} - E_{ij}}{E_{ij}} = x_{++} \frac{p_{ij} - p_{i+}p_{+j}}{\sqrt{p_{i+}p_{+j}}}$$

where χ_{ij} is the χ-value of every ij cell, x_{++} is the sum total of the contingency table

$p_{ij} = \frac{x_{ij}}{x_{++}}$, $p_{i+} = \frac{x_{i+}}{x_{++}}$, and $p_{+j} = \frac{x_{+j}}{x_{++}}$.

CA is based on a matrix called $\overline{\boldsymbol{Q}}$ ($r \times c$) and $r \geq c$. $\overline{\boldsymbol{Q}}$ is calculated as follows.

$\overline{\boldsymbol{Q}} = \left[\bar{q}_{ij}\right] = \left[\frac{p_{ij} - p_{i+}p_{+j}}{\sqrt{p_{i+}p_{+j}}}\right]_{r \times c}$, where $\sum \bar{q}_{ij}^2$ measures the total inertia (variation) in $\overline{\boldsymbol{Q}}$. It equals the sum of all the eigenvalues to be extracted by eigenanalysis of $\overline{\boldsymbol{Q}}$.

Example:

$$\boldsymbol{X}_{r \times c} = \begin{bmatrix} 10 & 10 & 20 \\ 10 & 15 & 10 \\ 15 & 5 & 5 \end{bmatrix}$$

$$\boldsymbol{Q} = \begin{bmatrix} 0.10 & 0.10 & 0.20 \\ 0.10 & 0.15 & 0.10 \\ 0.15 & 0.05 & 0.05 \end{bmatrix} \quad \begin{bmatrix} p_{i+} \end{bmatrix} = \begin{bmatrix} 0.4 \\ 0.35 \\ 0.25 \end{bmatrix}$$

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

We have:

$$\overline{\boldsymbol{Q}} = \begin{bmatrix} -0.10690 & -0.05774 & 0.16036 \\ -0.06429 & 0.13887 & -0.06429 \\ 0.21129 & -0.09129 & -0.12677 \end{bmatrix}$$

$$\tilde{\boldsymbol{Q}} = \left[\tilde{q}_{ij}\right] = \left[\frac{p_{ij}}{\sqrt{p_{i+}p_{+j}}}\right] = \begin{bmatrix} 0.26726 & 0.28868 & 0.53452 \\ 0.28571 & 0.46291 & 0.28571 \\ 0.50709 & 0.18257 & 0.16903 \end{bmatrix}$$

(2) Apply SVD to matrix $\overline{\boldsymbol{Q}}$. We have

$\overline{\boldsymbol{Q}} = \hat{\boldsymbol{U}}\boldsymbol{W}\boldsymbol{U}'$, where $\boldsymbol{U}_{r\times c}$ and $\hat{\boldsymbol{U}}_{c\times c}$ are column-orthonormal matrices (i.e., column vectors are normalized and orthogonal to one another. "normalized" means the length of the eigenvector is normalized to be 1.), $\boldsymbol{W}_{c\times c}$ is a diagonal matrix with diagonal values w_i (nonnegative). These values w_i are the singular values of $\overline{\boldsymbol{Q}}$.

Because $\overline{\boldsymbol{Q}} = \hat{\boldsymbol{U}}\boldsymbol{W}\boldsymbol{U}'$, the multiplication of $\overline{\boldsymbol{Q}}'\overline{\boldsymbol{Q}}$ gives $\overline{\boldsymbol{Q}}'\overline{\boldsymbol{Q}} = \boldsymbol{U}\boldsymbol{W}'(\hat{\boldsymbol{U}}'\boldsymbol{U})\boldsymbol{W}\boldsymbol{U}'$.

Because $\hat{\boldsymbol{U}}'\hat{\boldsymbol{U}} = \hat{\boldsymbol{U}}\hat{\boldsymbol{U}}' = \boldsymbol{I}$, we have:

$$\overline{\boldsymbol{Q}}'\overline{\boldsymbol{Q}} = \boldsymbol{U}\boldsymbol{W}'\boldsymbol{W}\boldsymbol{U}'$$

It is easy to show (see page 453-454 of Legendre's Numerical Ecology) the diagonal matrix $\boldsymbol{W}'\boldsymbol{W}$, which contains squared singular values on its diagonal, is the diagonal matrix of the eigenvalues of $\overline{\boldsymbol{Q}}'\overline{\boldsymbol{Q}}$. Furthermore, $\boldsymbol{U}$ is the eigenvectors of $\overline{\boldsymbol{Q}}'\overline{\boldsymbol{Q}}$, containing the **loadings** of the **columns** of the contingency table.

A similar application to matrix $\overline{\boldsymbol{Q}}\overline{\boldsymbol{Q}}'$ ($r\times r$) shows that the orthonormal $\hat{\boldsymbol{U}}'$, produced by the singular value decomposition is the matrix of eigenvectors of $\overline{\boldsymbol{Q}}\overline{\boldsymbol{Q}}'$, containing the **loadings** of the **rows** of the contingency table.

R-code of singular value decomposition: **> svd(Q).**

Note: This **svd** will produce the column and row eigenvectors ($\boldsymbol{U}$ and $\hat{\boldsymbol{U}}'$ respectively) and diagonal matrix $\boldsymbol{W}$. Eigenvalues can then be easily calculated by $\boldsymbol{W}'\boldsymbol{W}$.

Results identical to those of SVD can be obtained by applying to eigenvalue analysis either to the covariance matrix $\bar{\boldsymbol{Q}}'\bar{\boldsymbol{Q}}$, which would produce the matrix $\boldsymbol{U}$ (column eigenvectors), or to matrix $\bar{\boldsymbol{Q}}\bar{\boldsymbol{Q}}'$, which would produce $\hat{\boldsymbol{U}}'$ (row eigenvetors). $\bar{\boldsymbol{Q}}'\bar{\boldsymbol{Q}}$ and $\bar{\boldsymbol{Q}}\bar{\boldsymbol{Q}}'$ have the same eigenvalues which are $\boldsymbol{W}'\boldsymbol{W}$.

Important note: SVD or eigenvalue analysis of matrix $\bar{\boldsymbol{Q}}'\bar{\boldsymbol{Q}}$ or $\bar{\boldsymbol{Q}}\bar{\boldsymbol{Q}}'$ always produce one null eigenvalue. This is due to the centralization in calculating $\bar{\boldsymbol{Q}} = [\bar{q}_{ij}] = \left[\frac{p_{ij} - p_{i+}p_{+j}}{\sqrt{p_{i+}p_{+j}}}\right]_{r\times c}$, where ($p_{i+}$, p_{+j}) is substracted from each value p_{ij}. Therefore, the number of eigenvalues equals min(r, c) - 1. That is the number of rows or columns (whoever is smaller) minus one. Furthermore, all eigenvalues must be smaller than one due to the nature of $\bar{\boldsymbol{Q}}$ (compare $\bar{q}_{ij}$ with χ_{ij}. This difference makes eigenvalues < 1.).

Example:

(i) Data:

$$\bar{\boldsymbol{Q}} = \begin{bmatrix} -0.10690 & -0.05774 & 0.16036 \\ -0.06429 & 0.13887 & -0.06429 \\ 0.21129 & -0.09129 & -0.12677 \end{bmatrix}.$$ # **Qbar16.dat in R**

(ii) SVD:

> svd(Qbar16.dat)

$d

[1] 3.100518e-01 2.023462e-01 2.242471e-06

$u # $\hat{\boldsymbol{U}}'$, normalized row eigenvectors of $\bar{\boldsymbol{Q}}\bar{\boldsymbol{Q}}'$. The third column has no meaning and should be discarded.

[,1] [,2] [,3]

[1,] -0.5369188 0.5583332 0.6324415

[2,] -0.1304297 -0.7955875 0.5916321

[3,] 0.8334904 0.2351692 0.4999891

$v # $\boldsymbol{U}$, normalized column eigenvectors of $\overline{\boldsymbol{Q}}'\overline{\boldsymbol{Q}}$. Discard the third column.

[,1] [,2] [,3]

[1,] 0.7801605 0.2033713 -0.5915993

[2,] -0.2038383 -0.8114309 -0.5477498

[3,] -0.5914385 0.5479233 -0.5915915

(iii) Computing eigenvalues:

> d=svd(Qbar16.dat)

> w=diag(d)

> w%*%t(w)

Note: you can derive the above eigenvalues and eigenvectors, respectively, using $\overline{\boldsymbol{Q}}\overline{\boldsymbol{Q}}'$ and $\overline{\boldsymbol{Q}}'\overline{\boldsymbol{Q}}$. Try it using **eigen**.

Some programs may compute SVD based on the following matrix (without centralizing data).

$$\tilde{\boldsymbol{Q}} = \left[\tilde{q}_{ij}\right] = \left[\frac{p_{ij}}{\sqrt{p_{i+}p_{+j}}}\right] = \begin{bmatrix} 0.26726 & 0.28868 & 0.53452 \\ 0.28571 & 0.46291 & 0.28571 \\ 0.50709 & 0.18257 & 0.16903 \end{bmatrix}$$

The consequence of without centralization is that it will produce min(r, c) eigenvalues (one more than that based on $\overline{\boldsymbol{Q}}$), with all other results being the same as those based on $\overline{\boldsymbol{Q}}$. This extra eigenvalues is easy to recognize because its value is 1. This eigenvalue (and its corresponding eigevector) has no meaning and should be discarded. It only reflects the distance between the centre of mass of the data points in the ordination space (i.e., the origin of the new coordinate system).

(3) Biplot (joint plot)

Eigenvector matrices $\boldsymbol{U}$ and $\hat{\boldsymbol{U}}'$ are the loadings of the "principal axes" (recall PCA loadings?). Using these loadings we can compute scores (i.e., positions) of the row and column vectors in ordination spaces (one for row, one for column, separately). However, to facilitate interpretation, we can scale these scores so that to plot them (rows and columns) together in a single ordination space.

First, compute $\boldsymbol{V}$ and $\hat{\boldsymbol{V}}'$ matrices:

$\boldsymbol{V}_{c\times c} = diag(p_{+j})^{-1/2}\boldsymbol{U}$ # weighted by the inverse of the square roots of the column scores. Equivalent to the step of computing scores in PCA, i.e., principal axis, for the columns.

$\hat{\boldsymbol{V}}_{r\times c} = diag(p_{i+})^{-1/2}\hat{\boldsymbol{U}}$ # weighted by the inverse of the square roots of the row scores. Equivalent to the step of computing scores in PCA, i.e., principal axis, for the rows.

Note: $\boldsymbol{V}$ and $\hat{\boldsymbol{V}}'$ are the positions of columns and rows, respectively in CA space. If Columns and rows are ordinated in separated space, $\boldsymbol{V}$ and $\hat{\boldsymbol{V}}'$ are used. However, in order to produce a joint plot, the following score matrices are needed.

Second, compute $\boldsymbol{F}$ and $\boldsymbol{F}'$ matrices:

$\boldsymbol{F}_{r\times c} = \hat{\boldsymbol{V}}\boldsymbol{\Lambda}^{-1/2} = diag(p_{i+})^{-1}\boldsymbol{Q}\boldsymbol{V}$ # This gives the positions of the **rows** of the contingency table in the CA space. Obtained from the transformed matrix of eigenvectors $\boldsymbol{V}$, which gives the positions of the columns in that space.

$\hat{\boldsymbol{F}}_{c\times c} = \boldsymbol{V}\boldsymbol{\Lambda}^{-1/2} = diag(p_{+j})^{-1}\boldsymbol{Q}'\boldsymbol{V}$ # This gives the positions of the **columns** of the contingency table in the CA space. Obtained from the transformed matrix of

eigenvectors $\hat{\boldsymbol{V}}$, which gives the positions of the rows in that space.

With this scaling, matrices ***F*** and ***V*** form a pair such that the rows (given by ***F***) are at the centroid (centre of mass or "barycenter") of the columns in matrix ***V***. Similarly, matrices $\hat{\boldsymbol{F}}$ and $\hat{\boldsymbol{V}}$ form a pair such that the columns (given by matrix $\hat{\boldsymbol{F}}$) are at the centroids of the rows in matrix $\hat{\boldsymbol{V}}$.

Note: Scaling type 1 – Draw a biplot with the rows (matrix ***F***) at the centroids of the columns (matrix ***V***). For site×species data tables where sites are rows and species are columns, this scaling is the most appropriate if one is interested in the ordination of sites. Use scaling 1 if you are interested in preserving distance, e.g., the distance in sites. The locations of sites are measured by distance.

Scaling type 2 – Draw a biplot with the column (matrix $\hat{\boldsymbol{F}}$) at the centroids of the rows (matrix $\hat{\boldsymbol{V}}$). This scaling is the most appropriate if one is interested in the ordination of species. Use scaling 2 if your interest is to preserve correlation, e.g., correlation in species groups.

Scaling type 3 – Not appropriate for biplot. The eigenvectors in matrix ***U*** are normalized as in PCA. The scaling of ***F*** is such that the Euclidean distances among the rows of ***F*** are equal to the χ^2 distances among objects of the original data table.

vegan – has these three scaling options.

Example:

In scaling type 1, the states in the rows of the data matrix, whose coordinates will be stored in matrix ***F***, are to be plotted at the centroids of the columns states. The scaling for the columns is obtained using

$$diag(p_{+j}) = \begin{bmatrix} 0.35 & 0 & 0 \\ 0 & 0.30 & 0 \\ 0 & 0 & 0.35 \end{bmatrix}, \text{ and}$$

$$\boldsymbol{U} = \begin{bmatrix} 0.7801605 & 0.2033713 & -0.5915993 \\ -0.2038383 & -0.8114309 & -0.5477498 \\ -0.5914385 & 0.5479233 & -0.5915915 \end{bmatrix}$$

We have:

$$\boldsymbol{V} = diag(p_{+j})\boldsymbol{U} = \begin{bmatrix} 1.31871 & -0.34374 & -1 \\ -0.37215 & 1.48150 & -1 \\ -0.99972 & -0.92612 & -1 \end{bmatrix}$$

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. For example, consider the first row of original data:

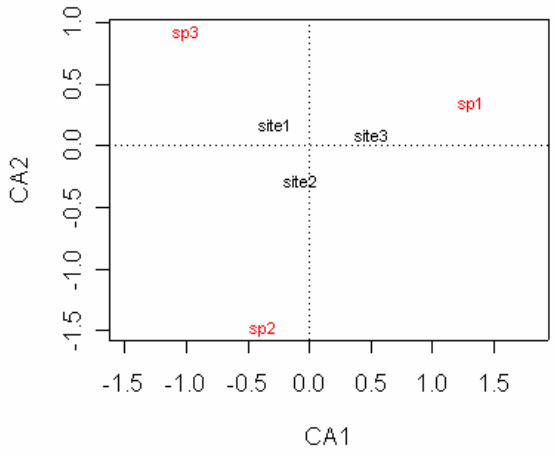
$$\boldsymbol{X} = \begin{bmatrix} 10 & 10 & 20 \\ 10 & 15 & 10 \\ 15 & 5 & 5 \end{bmatrix}.$$

The relative frequencies (i.e., conditional probabilities) of the three columns in that row are: 0.25 (10/40), 0.25 (10/40) and 0.5 (20/40). Multiplying matrix V by that vector provides the coordinates of the first row in the ordination diagram:

$$\begin{bmatrix} 0.25 & 0.25 & 0.5 \end{bmatrix} \begin{bmatrix} 1.31871 & -0.34374 \\ -0.37215 & 1.48150 \\ -0.99972 & -0.92612 \end{bmatrix} = \begin{bmatrix} -0.26322 & -0.17862 \end{bmatrix}$$

These coordinates put the first row at the centroid of the columns in CA ordination space. They are stored in the first row of matrix $\boldsymbol{F}$. The row-conditional probabilities for the whole data table are found using the matrix operation:

$$\boldsymbol{F} = \hat{\boldsymbol{V}}\boldsymbol{\Lambda}^{-1/2} = diag(p_{i+})^{-1}\boldsymbol{Q}\boldsymbol{V} = \begin{bmatrix} -0.26322 & -0.17862 \\ -0.06835 & 0.27211 \\ 0.51685 & -0.09517 \end{bmatrix}$$



Joint plot between sites and species. The scores of sites (in black) are from *F* matrix (first two columns) derived in the above. The scores of species (in red) are derived from ***V*** matrix (first two columns).

The plot is produced using **vegan**:
>plot(example16.cca, scaling=1)

If **>plot(example16.cca, scaling=2)**, the coordinates of species use $\hat{\boldsymbol{V}}$, and sites coordinates use $\hat{\boldsymbol{F}}$.

In the plot, each site locates in the centroids of the species. Positions of the centroids are calculated using weights equal to the relative frequencies of the species. Because species frequencies are different in different sites, e.g., some species may be absent from a site and thus do not contribute to the position of that site, the sites do not crashed onto to one location (single centroid). See interpretation in 6(2)(ii) below.

Using the formulae for the Euclidean distances, one can verify that the Euclidean distances among the rows of matrix $\boldsymbol{F}$ equal the χ^2 distance among the rows of the original data table:

$$\boldsymbol{D} = \begin{bmatrix} 0 & & \\ 0.49105 & 0 & \\ 0.78452 & 0.69091 & 0 \end{bmatrix}.$$

Matrix $\boldsymbol{F}$ thus provides a proper ordination of the rows of the original data matrix.

5. Reciprocal averaging

Hill's (1973) paper (Reciprocal averaging: an eigenvector method of ordination, *J. Ecol*. 61:237-249) proposes the reciprocal averaging method for site×species contingency table. But it became clear later on that this method actually is CA.

Reciprocal averaging is derived from ecological perspective, based on the principle of *gradient analysis*. Gradient analysis uses a matrix $\boldsymbol{X}$ (site×species) and an initial vector $\boldsymbol{v}$ of values v_j which are ascribed to the various species j as indicators of the physical gradient to be evidenced. For example, a (arbitrary) score (scaled from 1 to 10) could be given to each species for its preference with respect to soil moisture. These coefficients are used to calculate the positions of the sites along the gradient.

(i) Site scores: The score $\hat{v}_i$ of a site i is the average score of the species (j = 1, 2, …, p) present at that site, using the formula:

$$\hat{v}_i = \frac{\sum_{j=1}^{p} x_{ij} v_j}{x_{i+}}.$$

where x_{ij} is the abundance of species j at site i and x_{i+} is the sum of the species at this site (i.e., the sum of the values in row i of matrix $\boldsymbol{X}$).

(ii) Species scores: The scores (positions) of species can be similarly derived, but using $\hat{v}$:

$v_j = \frac{\sum_{i=1}^{n} x_{ij} \hat{v}_i}{x_{+j}}$, where x_{+j} is the sum of values in column j of matrix $\boldsymbol{X}$.

(iii) Iteration: Use $\hat{v}_j$ calculated from step (ii) to update $\hat{v}_i$ in step (i). Alternating between steps (i) and (ii) defines an iterative procedure of "reciprocal averaging". This iteration eventually will converge to a unique unidimensional ordination of the species and sites, which is independent of initial values given to v_j's.

$\hat{v}_i$ and v_j are the scores (positions) of sites and species on the first axis. Hill (1973) also shows how to calculate the first eigenvalue corresponding to this first axis coordinate system. He also shows how to find other eigenvalues and eigenvectors.

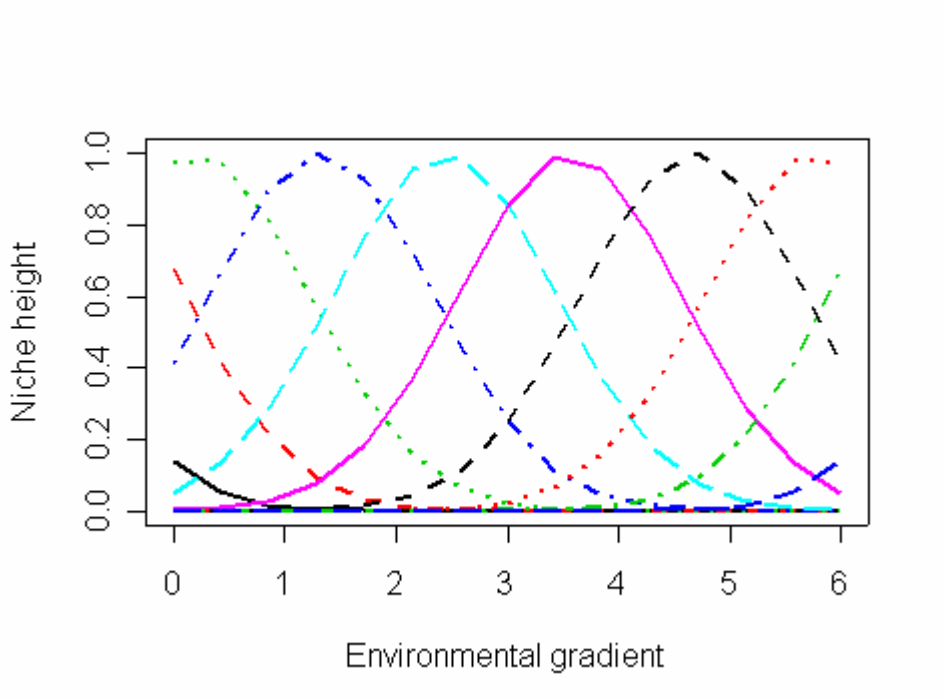
6. CA interpretation. The purposes of CA include: data reduction and pattern detection (site/species clustering classification and correlation between sites and species).
   (1) Eigenvalues λ_i (equivalent to R^2) measures the correlation between column scores (V) and row scores ($\hat{V}$) at the i^{th} principal axis. For example, in our **example16.dat**, the eigenvalue of the first axis is 0.096. Methods are available to test for the significance of $R^2 = \lambda$. In many CA programs, the test can be done using permutation test. (Remember a central question in contingency table analysis is to test for row and column independence. The eigenvalues measure that.)
   (2) Biplot allows one to conclude about the ecological relationships displayed by the data. With scaling type 1, we have:
      (i) The Euclidean distance among sites in reduced space approximate their χ^2 distances.
      (ii) The sites (rows) are at the centroids of the species (column). Positions of the centroids are calculated using weights equal to the relative frequencies of the species. Species that are absent from a site do not contribute to the position of that site.
      (iii) Therefore, any site found near the point representing a species is likely to receive a high contribution of that species; for binary data, the species is more likely to be present in that site. (Similar interpretation applies to the ordination of scaling type 2.)
   (3) Niche interpretation: If species are distributed in unimodal (i.e., bell-shapped) response curve along environmental gradient, the use CA is justified.
      (i) The position of a species in the ordination space is largely the optimum of the distribution of that species. Spatial proximity between site and species represents association between them.

(ii) Species that are not associated with most sites (absent from most sites) often appear at the edge of the scatter plot, near the point representing a site where the species occur by chance (or because they are favored by some rare condition occurring at the sites).

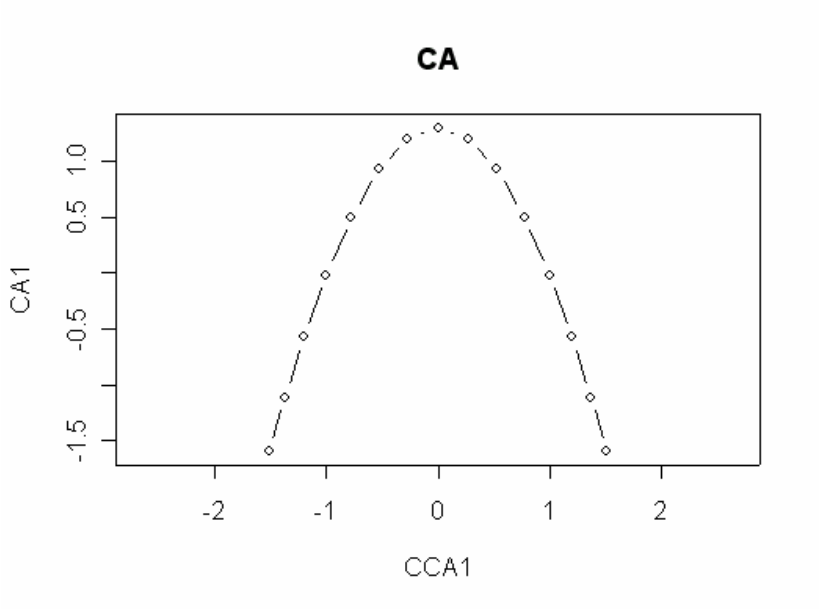
(iii) Species that lie near the centre of the ordination diagram may have their optimum in this area of the plot, or have two or several optima (bi- or multi-modal species), or be unrelated to the pair of axes under consideration.

7. Detrended correspondence analysis (DCA)

Species that are controlled by environmental factors often have unimodal bell-shaped distributions along the environmental gradient. The effect of gradients on the relationship among sites, calculated on species presence-absence or abundance data, is necessarily nonlinear. Ordination methods aim at rendering this nonlinear phenomenon in a Euclidean space, in particular as two-dimensional plots. In such plots, nonlinearities end up being represented by curves, called **arches** or **horseshoes**. This horseshoe effect causes problem in interpreting ordination results.

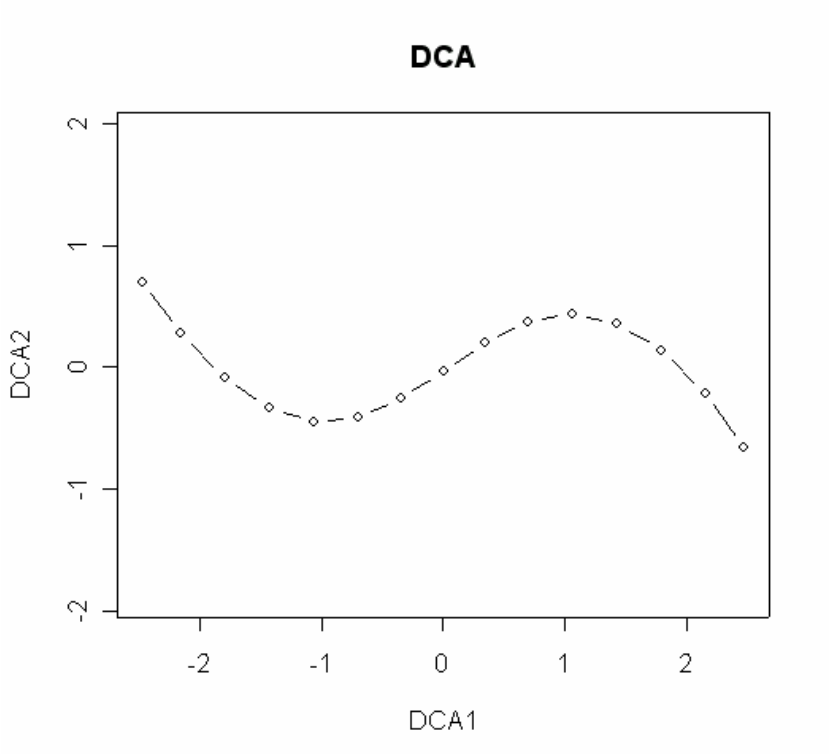


The result of CA, showing horseshoe effect:



Two approaches to remove the horseshoe effect:

(i) Detrending by segments – Axis I is divided into a number of "segments" and, within each segment 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.



(ii) Detrending by polynomials – This method 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 form of first one. The horseshoe effect can be removed by fitting a quadratic form the arch. This method is not widely use because real data do not usually have an ideal quadratic shape.

8. In **vegan**, DCA is implemented using **decorana**.

9. Try **dune** data – use both CA and DCA.

10. Some common ordination (indirect gradient analysis) methods:

Method	**Distance**	**Variables**
Principal component analysis	Euclidean distance	Quantitative data, linear relationships (beware of double zeros. Use CA for lot of zeros.)
Principal coordinate analysis	Any distance measure	Quantitative data, semiquantitative, qualitative, or mixed
Nonmetric multidimensional scaling (NMDS, MDS)	Any distance measure	Quantitative data, semiquantitative, qualitative, or mixed. Computing intensive, always obtains a Euclidean representation.
Correspondence analysis χ^2 distance	χ^2 distance	Non-negative, dimensionally homogeneous quantitative or binary data, species presence/absence, abundance data