1.3. *Principal coordinate analysis*

Pierre Legendre
Département de sciences biologiques
Université de Montréal

© Pierre Legendre 2018
Outline of the presentation

1. Definition of principal coordinate analysis
2. PCoA: Computation steps
3. PCoA of non-Euclidean dissimilarities
4. PCoA biplots
5. PCoA for R-mode analysis
6. Comparison of PCoA and nMDS
7. PCoA algorithm and functions
8. References

Principal coordinate analysis
Definition of principal coordinate analysis

Principal coordinate analysis (PCoA)
An ordination method preserving any dissimilarity measure $D$ among the objects, except nonmetric indices $^1$.

Also called classical multidimensional scaling (cMDScale) or metric multidimensional scaling, by opposition to nonmetric multidimensional scaling (nMDS).

Mathematical properties of data
Data can be of any mathematical type: quantitative, semi-quantitative, qualitative, or mixed.

$^1$ Nonmetric $D$ indices, not used in ecology, can produce $-\text{Inf}$ values due to division by 0.
The objective of PCoA is

• to represent the points in a full-dimensional Euclidean space

• that allows one to fully reconstruct the original dissimilarities among the objects.

Applications of PCoA

• Produce ordinations of the objects in reduced 2-D (or sometimes 3-D) space.

• Act as a data transformation after computation of an appropriately chosen dissimilarity measure. The coordinates of the objects in full-dimensional PCoA space represent the transformed data. They can be used as input to RDA or other methods of data analysis.
Euclidean space, Euclidean property

Definitions

• **Euclidean space**: a space in which the distance among points is measured by the Euclidean distance (Pythagora’s formula).

Synonym: *Cartesian* space.

• **Euclidean property**: a dissimilarity coefficient is Euclidean if any resulting dissimilarity matrix can be fully represented in a Euclidean space without distortion (Gower & Legendre 1986).

*Criterion*: Principal coordinate analysis (PCoA) of such a dissimilarity matrix does not produce negative eigenvalues.
Example –

The same data as for PCA:

\[
Y = \begin{bmatrix}
2 & 1 \\
3 & 4 \\
5 & 0 \\
7 & 6 \\
9 & 2
\end{bmatrix}
\]

Compute a Euclidean distance matrix among the objects (rows of \(Y\))

\[
D = \begin{bmatrix}
0.000 & 3.162 & 3.162 & 7.071 & 7.071 \\
3.162 & 0.000 & 4.472 & 4.472 & 6.325 \\
3.162 & 4.472 & 0.000 & 6.325 & 4.472 \\
7.071 & 4.472 & 6.325 & 0.000 & 4.472 \\
7.071 & 6.325 & 4.472 & 4.472 & 0.000
\end{bmatrix}
\]

PCoA can analyse any other type of dissimilarity matrix.
Apply the transformation and centring proposed by Gower:

1. Transform each value $D_{ij}$ into $-0.5D_{ij}^2$
2. Centring: make the sums of rows and columns equal to 0

$$G = \begin{bmatrix}
12.8 & 4.8 & 4.8 & -11.2 & -11.2 \\
4.8 & 6.8 & -3.2 & 0.8 & -9.2 \\
4.8 & -3.2 & 6.8 & -9.2 & 0.8 \\
-11.2 & 0.8 & -9.2 & 14.8 & 4.8 \\
-11.2 & -9.2 & 0.8 & 4.8 & 14.8 
\end{bmatrix}$$
Eigen-decomposition of matrix $G$:

Eigenvalues:

$\lambda_1 = 36, \lambda_2 = 20$

Eigenvectors

$U = \begin{bmatrix} 0.596 & 0.000 \\ 0.224 & 5.000 \\ -0.522 & 5.000 \\ -0.522 & -5.000 \end{bmatrix}$

Norm each eigenvector to the square root of its eigenvalue: multiply the values in each eigenvector by $\sqrt{\text{eigenvalue}}$

Matrix of principal coordinates: $\text{Pr.coo} = \begin{bmatrix} 3.578 & 0.000 \\ 1.342 & 2.236 \\ 1.342 & -2.236 \\ -3.130 & 2.236 \\ -3.130 & -2.236 \end{bmatrix}$
Matrix of principal coordinates:

\[
\text{Pr.coo} = \begin{bmatrix}
3.578 & 0.000 \\
1.342 & 2.236 \\
1.342 & -2.236 \\
-3.130 & 2.236 \\
-3.130 & -2.236 \\
\end{bmatrix}
\]

Observation –

In the centred matrix \( \mathbf{G} \), the diagonal values are the squares of the distances of the points to the origin in the PCoA plot.

Ex. \( \sqrt{12.8} = 3.58 \)

\( \sqrt{6.8} = 2.61 \)

\( \sqrt{14.8} = 3.85 \)
The dissimilarities $D$ are fully reconstructed by the principal coordinates.

Can we verify that property?  \textit{(Demonstrated by Gower, 1966)}

Compute the original dissimilarities. The chosen dissimilarity function is the Euclidean distance: $\text{dist}(Y)$

Compute the Euclidean distance among the rows of matrix $\text{Pr.coo}$ (positions of the objects in the PCoA ordination): $\text{dist}(\text{Pr.coo})$

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3.162</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3.162</td>
<td>4.472</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>7.071</td>
<td>4.472</td>
<td>6.325</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>7.071</td>
<td>6.325</td>
<td>4.472</td>
<td>4.472</td>
</tr>
</tbody>
</table>
Matrix of principal coordinates:

\[
\text{Pr.coo} = \begin{bmatrix}
3.578 & 0.000 \\
1.342 & 2.236 \\
1.342 & -2.236 \\
-3.130 & 2.236 \\
-3.130 & -2.236 \\
\end{bmatrix}
\]

Compare to matrix F of PCA:

\[
F = \begin{bmatrix}
-3.578 & 0.000 \\
-1.342 & 2.236 \\
-1.342 & -2.236 \\
3.130 & 2.236 \\
3.130 & -2.236 \\
\end{bmatrix}
\]
Compare the PCA and PCoA plots

- Axes may be inverted; no consequence for interpretation.
- The distances among the points in the PCA and PCoA plots are the same.

Reason:

The *Euclidean distance* was computed among the objects before the PCoA.

If another dissimilarity function had been computed, the distances among objects would not be the same in the PCoA plot (other D) and the PCA biplot (Euclidean D).
**Principal coordinate analysis** was developed to its present form in 1966 by Prof. John C. Gower who made extensive contributions to statistical methodology, computing and applications, especially to taxonomy and ecology.

John Gower was Head of the Biomathematics Division of the *Rothamsted Experimental Station* (England) from 1985 to 1990.

This picture was taken during the *NATO Advanced Research Workshop on Numerical Ecology* held in Roscoff, France, in June 1986.

---

*Photo P. Legendre*
PCoA of non-Euclidean dissimilarities

Metric property

The attributes of a metric dissimilarity are the following\(^1\):

1. Minimum 0: if \( a = b \), then \( D(a, b) = 0 \);
2. Positiveness: if \( a \neq b \), then \( D(a, b) > 0 \);
3. Symmetry: \( D(a, b) = D(b, a) \);
4. Triangle inequality: \( D(a, b) + D(b, c) \geq D(a, c) \).

The sum of two sides of a triangle drawn in ordinary Euclidean space is equal to or larger than the third side.

\(^1\) Attributes also described in the course on dissimilarity functions.

*Note:* A metric dissimilarity function is also called a *distance*. 
Some of the $D$ functions used to analyse community composition data are semimetrics\(^1\), meaning that the triangle inequality does not always hold in $D$ matrices computed with these functions.

*Examples:* percentage difference (aka Bray-Curtis dissimilarity), Whittaker and Kulczynski indices, and the one-complements of some widely used binary coefficients (Sørensen, Ochiai, etc.)

---

\(^1\) See Legendre & Legendre (2012, Tables 7.2 and 7.3).
The percentage difference (aka Bray-Curtis dissimilarity) is a semimetric. It can produce triangles that do not close.

*Example*: spider data, 3 sites, 7 species

```r
# Percentage difference, spiders
library(adespatial)  # Contains function dist.ldc()
D.mat <- dist.ldc(spiders[c(1,3,5),1:7], "percent")
```

Indeed: $0.341 + 0.195 = 0.536$, which is *smaller* than 0.583 so that the triangle does not close.
Euclidean property

A dissimilarity coefficient is Euclidean if any resulting dissimilarity matrix can be fully represented in a Euclidean space without distortion (Gower & Legendre 1986).

Criterion: Principal coordinate analysis (PCoA) of a Euclidean $D$ matrix does not produce negative eigenvalues.

PCoA of a non-Euclidean $D$ matrix produces some negative eigenvalues and complex principal coordinates.

Consider an eigenvector that has a negative eigenvalue. Multiplying that eigenvector by the square root of the negative eigenvalue produces a vector of complex numbers, because the square root of a negative eigenvalue is an imaginary number.
A semimetric coefficient cannot be Euclidean since some triangles do not close.

The reverse statement is not true: a metric $D$ matrix is not necessarily Euclidean.

**Example of a metric $D$ matrix with 4 points** –

Each triangle closes, yet point $x_4$ has an undefined position in the ordination because some edges of the graph are too short for the structure to close.

Negative eigenvalues are produced.
The presence of negative eigenvalues in the PCoA solution is the criterion to recognize non-Euclidean dissimilarity matrices.

Function `is.euclid()` of \{ade4\} carries out a PCoA and looks for the presence of negative eigenvalues in the solution.

*Note:* a PCoA ordination occupies at most \((n - 1)\) dimensions.
Eliminate negative eigenvalues

Three methods are available to eliminate negative eigenvalues produced during PCoA of a non-Euclidean dissimilarity coefficient.

The principle is to increase the small dissimilarities more than the large ones.

1. The easiest method, which works in most cases, is to take the square roots of the dissimilarities before PCoA.

This transformation also turns semimetric coefficients to metric (all triangles are closing).

Tables 7.2 and 7.3 of Legendre & Legendre (2012) show which coefficients can be made Euclidean using that method.
Example for the open triangle shown before:

$D(1,3) = 0.341$

$D(1,5) = 0.583$

$D(3,5) = 0.195$

$sqrt(D)$

The square-rooted dissimilarities are metric.
Why does this method work?

The square-root transformation increases the small $D$ values more than the large ones. Example for $D$ values $\leq 1$:

<table>
<thead>
<tr>
<th>$D$</th>
<th>$\sqrt{D}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0.0</td>
</tr>
<tr>
<td>0.01</td>
<td>0.1</td>
</tr>
<tr>
<td>0.04</td>
<td>0.2</td>
</tr>
<tr>
<td>0.09</td>
<td>0.3</td>
</tr>
<tr>
<td>0.16</td>
<td>0.4</td>
</tr>
<tr>
<td>0.25</td>
<td>0.5</td>
</tr>
<tr>
<td>0.36</td>
<td>0.6</td>
</tr>
<tr>
<td>0.49</td>
<td>0.7</td>
</tr>
<tr>
<td>0.64</td>
<td>0.8</td>
</tr>
<tr>
<td>0.81</td>
<td>0.9</td>
</tr>
<tr>
<td>1.00</td>
<td>1.0</td>
</tr>
</tbody>
</table>
Compute PCoA on the spider data, percentage difference $D$

```r
# Percentage difference D, spider data
# Functions dist.ldc() in adespatial, pcoa() in ape
library(adespatial); library(ape)
D.mat <- dist.ldc(spiders, method="percentdiff")
res1 <- pcoa(D.mat)  # Original D values
res2 <- pcoa(sqrt(D.mat))  # Square-rooted D values

# Plot the eigenvalues
par(mfrow=c(1,2))
values <- res1$values$Eigenvalues
eig.pos <- values[values >= 0]
for (value in eig.pos) {
  points(1:20, value, pch=20, col="blue")
}
for (value in eig.neg) {
  points(21:28, value, pch=20, col="red")
}
plot(res2$values$Eigenvalues, pch=20, col="blue")
```

```r
abline(h=0, col="grey50")
```
Examine the two series of eigenvalues:

**Original $D$ values**

19 positive, 1 null, 8 negative eigenvalues

$\Rightarrow$ Non-Euclidean matrix

**Square-rooted $D$ values**

27 positive eigenvalues, 1 null (not shown)

$\Rightarrow$ Euclidean matrix
Correct for negative eigenvalues

2. Lingoes method – Correct the dissimilarities by adding a constant to the squared dissimilarities:

\[ \hat{D}_{hi} = \sqrt{D_{hi}^2 + 2c_1} \]

Constant \( c_1 \) is chosen to make sure that all triangles close.

3. Cailliez method – Correct the dissimilarities by adding a constant to the dissimilarities:

\[ \hat{D}_{hi} = D_{hi} + c_2 \]

Constant \( c_2 \) is chosen to make sure that all triangles close.

Constants \( c_1 \) and \( c_2 \) increase the small \( D \) values relatively more than large ones. Details: see Gower & Legendre (1986); Legendre & Legendre (2012, Sect. 9.3.4).
Non-Euclidean \( D \) matrix

Euclidean solutions in \((n-2) = 2\) dimensions

Euclidean solution in \((n-1) = 3\) dimensions

**Non-Euclidean matrix \( D \)**

- \( D = 0.377 \)
- \( D = 0.8 \)

**Euclidean solutions in \((n-2) = 2\) dimensions**

- \( D = 0.499 \)
- \( D = 0.864 \)

**Euclidean solution in \((n-1) = 3\) dimensions**

- \( D = 0.614 \)
- \( D = 0.894 \)

**Non-Euclidean to Euclidean solutions**

\[ \hat{D}_{hi} = \sqrt{D_{hi}^2 + 2c_1} \]

**Original matrix \( D \)**

**Lingoes correction**

\[ \hat{D}_{hi} = \sqrt{D_{hi}^2 + 2c_1} \]

**Cailliez correction**

\[ \hat{D}_{hi} = D_{hi} + c_2 \]

**sqrt(\( D \))**

Constants \( c_1 \) and \( c_2 \) increase the small \( D \) values relatively more than the large ones.
**Example:** spider data, $n = 5$ sites, $p = 7$ species

```
spiders[1:5,1:7]

<table>
<thead>
<tr>
<th></th>
<th>Alop.acce</th>
<th>Alop.cune</th>
<th>Alop.fabr</th>
<th>Arct.lute</th>
<th>Arct.peri</th>
<th>Aulo.albi</th>
<th>Pard.lugu</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site1</td>
<td>25</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Site2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>30</td>
<td>1</td>
</tr>
<tr>
<td>Site3</td>
<td>15</td>
<td>20</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>Site4</td>
<td>2</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>24</td>
<td>1</td>
</tr>
<tr>
<td>Site5</td>
<td>1</td>
<td>20</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>1</td>
</tr>
</tbody>
</table>
```
# Percentage difference, spider data
library(adespatial); library(ade4); library(ape)
D.mat <- dist.ldc(spiders[1:5,1:7], method="percent")

is.euclid(D.mat)
[1] FALSE

res1 <- pcoa(D.mat)          # Original D values
res1$values$Eigenvalues
[1] 0.47842  0.14711  0.00620  0.00000 -0.01363

=> Non-Euclidean PCoA solution
   5 eigenvalues: 3 positive, 1 null, 1 negative
principal coordinate analysis is.euclid(sqrt(D.mat))
[1] TRUE

res2 <- pcoa(sqrt(D.mat))     # Square-rooted D values
res2$values$Eigenvalues
[1] 0.58515  0.29121 0.09053 0.06640

=> Euclidean PCoA solution in \((n - 1) = 4\) dimensions.
   The 5\(^{th}\) eigenvalue, which is null, is not shown.

Note: a PCoA ordination occupies at most \(n - 1\) dimensions.
res3 <- pcoa(D.mat, "lingoes")  # Original D values
res3$values$Corr_eig
[1] 0.49205 0.16074 0.01983 0.00000 0.00000

=> Euclidean PCoA solution in \((n - 2) = 3\) dimensions.

res4 <- pcoa(D.mat, "cailliez")  # Original D values
res4$values$Corr_eig
[1] 0.57279 0.19628 0.02371 0.00000 0.00000

=> A different Euclidean solution in \((n - 2) = 3\) dimensions.
The $\text{sqrt}(D)$, Lingoes and Cailliez correction methods produce ordination on axes 1 and 2 that are very similar to PCoA without correction …

… except when the absolute values of the negative eigenvalues are close to or larger than those of the first 2 positive eigenvalues.
The species data can be projected a posteriori onto a PCoA ordination, forming a biplot.

The species may have been transformed in a way that the user finds useful for proper representation of the species, e.g. log1p(). Example, spider. The species are log-transformed before plotting.

```r
# Percentage difference, spider data
library(adespatial); library(ape)
D.mat <- dist.ldc(spiders[1:5,1:7], method="percent")
res2 <- pcoa(sqrt(D.mat))   # Square-rooted D values

# Take the log of species data before plotting them
spi.log = log1p(spiders[1:5,c(1:4,6:7)])
biplot(res2, spi.log, rn=paste("Site",1:5))
```
A full biplot example: PCoA ordination of the spider data.

```r
# Spider data, percentage difference D
library(adespatial); library(ape)
D.mat <- dist.ldc(spiders, method="percent")
res2 <- pcoa(sqrt(D.mat))  # Square-rooted D values

# Take log of species data before plotting them
spi.log = log1p(spiders)
biplot(res2, spi.log, rn=paste("Site",1:28))
```

Principal coordinate analysis
PCoA biplot
Response variables projected as in PCA with scaling 1
What proportion of the species variance captured by the percentage difference index is expressed on the first PCoA axes?

Examine the eigenvalue table (the first 4 and last lines are shown).

<table>
<thead>
<tr>
<th>Eigenvalues</th>
<th>Relative_eig</th>
<th>Broken_stick</th>
<th>Cumul_eig</th>
<th>Cumul_br_stick</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.7899889</td>
<td>0.3494284</td>
<td>0.14412803</td>
<td>0.3494284</td>
</tr>
<tr>
<td>2</td>
<td>1.9004365</td>
<td>0.23801761</td>
<td>0.10709099</td>
<td>0.5874460</td>
</tr>
<tr>
<td>3</td>
<td>0.7469060</td>
<td>0.09354524</td>
<td>0.08857247</td>
<td>0.6809913</td>
</tr>
<tr>
<td>4</td>
<td>0.3514708</td>
<td>0.04401949</td>
<td>0.07622679</td>
<td>0.7250107</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>27</td>
<td>0.0184412</td>
<td>0.00230964</td>
<td>0.00137174</td>
<td>1.0000000</td>
</tr>
</tbody>
</table>

Column **Cumul_eig** lists the cumulative sums of the relative eigenvalues. These are pseudo-$R^2$ statistics. As in PCA, they indicate the proportion of inertia expressed in biplots drawn in 1, 2, 3, … dimensions.
PCoA can also be used to display relationships among variables, for example species.

Example using the spider data –

```r
# Percentage difference, spider data
# Hellinger transformation of the species data
spi.hel <- decostand(spiders, "hellinger")

# Compute correlations among the transformed species
spi.hel.cor <- cor(spi.hel)
# Transform the correlations to dissimilarities
spi.D <- as.dist(1-spi.hel.cor)

is.euclid(spi.D)  # FALSE: Not Euclidean
is.euclid(sqrt(spi.D))  # TRUE: Euclidean

res5 <- pcoa(sqrt(spi.D))  # Square-rooted D values
biplot(res5, main="PCoA of the spider species")
```
PCoA of the spider species

This plot would nicely complement a study of species associations.
Comparison of PCoA and nMDS

Principal coordinate analysis (PCoA, aka classical multidimensional scaling, cMDScale) and nonmetric multidimensional scaling (nMDS) share the ability to produce ordinations in space of reduced dimensionality (usually in 2 dimensions) from any type of dissimilarity index $D$, except nonmetric indices.

PCoA finds the solution by eigenvalue decomposition of the $D$ matrix whereas nMDS finds it through an iterative algorithm.

The following questions arise:

• Are the two methods equally suitable in all situations?
• Do they produce identical or fairly similar solutions?
• Is one method preferable over the other for certain objectives?
**nMDS**

1. The iterative algorithm may find different solutions depending on the starting point of the calculation, which, in most instances, is a randomly chosen configuration.

2. The dissimilarities are distorted (stretched or squeezed) during nMDS calculation. That is an acknowledged property of the method. The distances in the ordination solution do not exactly correspond to the starting dissimilarities.

3. The first nMDS axis is not bound to maximize the variance of the observations. However, most if not all nMDS programs nowadays compute a PCA rotation of the nMDS result, so that the first PCA axis maximizes the variance of the nMDS solution.

4. Different criteria are available in different nMDS functions to minimize the stress. The solution may differ depending on the criterion that one (or the R function) chooses to optimise.
5. R functions may contain several other arguments that may influence the solution. Except for highly experienced users, it is difficult to determine which combination of options is best for a particular data set.

6. Users must set $k$, the number of dimensions of the solution. All ordination axes can be produced, i.e. $\min[p, n - 1]$, but then the solution is not exact since the dissimilarities are distorted. It is recommended to set $k < (n - 1)/2$.

7. The stress statistic does not indicate the proportion of the variance of the data represented in the ordination solution. Instead, it indicates the amount of deformation of the original dissimilarities, which is very different.

$\Rightarrow$ Applications to ecological analysis – When it is necessary to squeeze in two dimensions a PCA or PCoA solution that requires 3 or 4 dimensions, nMDS is useful to represent well the main dissimilarity relationships among the sites in 2-D. In such a case, it is preferable to use the PCA or PCoA ordination axes as input into nMDS to make sure that the nMDS solution will not diverge markedly from the metric ordination.
**PCoA**

1. PCoA finds the optimal solution by eigenvalue decomposition. The PCoA solution is unique.

2. In a PCoA, the dissimilarities are not distorted in the ordination solution.

3. PCoA can be used to find the first ordination axes for a given \( D \) matrix. These axes are those that maximize the variance of the observations.

4. PCoA produces all ordination axes corresponding to a given \( D \) matrix. The ordination solution is exact.

5. The full matrix of PCoA axes allows one to precisely reconstruct the dissimilarities among objects.

*Reason*: the matrix of Euclidean distances among objects computed from the PCoA axes is strictly equal to the dissimilarity matrix subjected to PCoA, whatever the dissimilarity function that has been used. Proof of that property is found in Gower (1966) and in Legendre and Legendre (2012).
6. Hence, the set \{D function, ordination method\} produces a unique transformation of the data, and the solution is reproduced exactly if one runs PCoA again on the same data, irrespective of the user or program, except for possible inversion of the signs along any one axis.

7. A pseudo-\(R^2\) statistic is computed as the sum of the eigenvalues of the axes of interest (for example the first 2 axes) divided by the sum of all eigenvalues. It indicates the fraction of the variance of the data represented in the reduced-space ordination. This is a useful statistic to assess the ordination result.

Applications to ecological analysis –

- PCoA produces ordinations of the objects in reduced 2-D or 3-D space.
- PCoA can also act as a data transformation after computation of an appropriately chosen dissimilarity measure. The coordinates of the objects in full-dimensional PCoA space represent the transformed data. They can be used as starting point for new analyses, for example db-RDA or \(k\)-means partitioning.

=> Recommendation: use PCoA in most applications.
PCoA algorithm and functions

PCoA algorithms use eigenvalue decomposition of the transformed and centred $D$ matrix. The decomposition cannot safely be done with svd() because that method cannot identify negative eigenvalues, which may be produced in PCoA.

Several functions are available in R to compute principal coordinate analysis:

cmdsclae {stat}, dudi.pco {ade4}, pco {ecodist}, pco {labdsv}, pcoa {ape} and wcmdscale {vegan}.

Some of them offer corrections for negative eigenvalues:

cmdscale {stat}, pco {ecodist}, pcoa {ape}, wcmdscale {vegan}.
References


Principal coordinate analysis
End of section
1.4. *Metric ordination methods for community composition data:*
* A summary

Pierre Legendre
Département de sciences biologiques
Université de Montréal


© Pierre Legendre 2018
Unconstrained metric ordination of community composition data

(a) Classical approach

\[ Y = \text{Raw data} \quad (\text{sites x species}) \]

Short gradients: CA or PCA

Long gradients: CA

(b) Transformation-based approach (tb-PCA)

Raw data

\[ Y = \text{Transformed data} \quad (\text{sites x species}) \]

PCA

(c) Distance-based approach (PCoA)

Raw data

\[ \text{Distance matrix} \]

PCoA

© Legendre & Legendre 2012, Figure 9.8.
End of section