Ordination & PCA

Introduction to Ordination
- Purpose & types
- Shepard diagrams
Principal Components Analysis (PCA)
- Properties
- Computing eigenvalues
- Computing principal components
- Biplots
- Covariance vs. Correlation
- Types of PCAs
- Meaningful Components
- Misuses of PCA

PCA Session / Software Tutorial

Ordination

Ordination (from the Latin *ordinatio* and German *Ordnung*) is the arrangement of units in some order (Goodall 1954).

It consists of plotting object-points along an axis representing an ordered relationship, or forming a scatter diagram with two or more axes.

The actual term "ordination" seems to have originated in the ecological literature (and is generally not used by statisticians).

Ordination

Biologists are often interested in characterizing data trends of variation in the objects with respect to all descriptors, not just a few.

The ordination approach permits the construction of a multidimensional space whereby each axis represents a descriptor in the study.

This multidimensional space is then reduced to two or three dimensions for graphical interpretation and communication, thus permitting the examination of relationship among objects.
Ordination

Ordination in reduced space is often referred to as factor analysis (in non-biological disciplines) since it is based on the extraction of the eigenvectors or factors of the association matrix. In actuality, there is a fundamental difference between FA and the other ordination procedures (treated later).

The domains of application of the techniques we will discuss are covered in the following table and include:

- PCA = Principal Components Analysis
- PCO = Principal Coordinates Analysis
- NMDS = Nonmetric Multidimensional Scaling
- CA = Correspondence Analysis
- FA = Factor Analysis

Domains of Ordination

<table>
<thead>
<tr>
<th>Method</th>
<th>Distance Preserved</th>
<th>Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA</td>
<td>Euclidean distance</td>
<td>Quantitative data, linear relationships, beware the double-zero</td>
</tr>
<tr>
<td>PCO</td>
<td>Any distance measure</td>
<td>Quantitative, semi-quantitative, qualitative, or mixed</td>
</tr>
<tr>
<td>NMDS</td>
<td>Any distance measure</td>
<td>Quantitative, semi-quantitative, qualitative, or mixed</td>
</tr>
<tr>
<td>CA</td>
<td>$\chi^2$ distance</td>
<td>Non-negative, homog. quant. data or binary data</td>
</tr>
<tr>
<td>FA</td>
<td>Euclidean distance</td>
<td>Quantitative data, linear relationships, beware the double-zero</td>
</tr>
</tbody>
</table>

Reduced Space

If the goal of ordination is then to reduce the dimensionality of a data set, and represent the final product in say $d = 2$ dimensional space, an obvious question is: "to what extent does the reduced space preserve the distance relationships among objects?"

To answer this, you need to compute the distance between all pairs of objects, both in the multidimensional space and the reduced space. The resulting values are plotted in a scatter diagram. When the projection in reduced space accounts for a high fraction of the variance, the spaces are similar. This is called a Shepard diagram.
Shepard Diagram

The Shepard diagram (1962) can be used to estimate the representativeness of ordinations obtained using any reduced-space ordination method.

In PCA, the distances among objects, in both the multidimensional space and the reduced space, are calculated using Euclidean distances. The $F$ matrix of principal components gives the coordinates of the objects in reduced space.

In PCO & NMDS, Euclidean distances among the objects in reduced space are compared to distances $D_h$ found in matrix $D$ used as the basis for computing the ordination. (CA uses chi-square distances as the abscissa.)

Ordination vs. Classification

Ordination and classification are often used as complements to each other in the evaluation of EEB-related questions. With regards to multivariate data, they both (1) show relationships, (2) reduce noise, (3) identify outliers, and (4) summarize redundancy.

However, they have slightly different applications. Clustering investigates pairwise distances among objects, and often produces a hierarchy of relatedness. Ordination considers the variability of the whole association matrix and emphasizes gradients and relationships.

Unlike direct gradient analysis, ordination & classification procedures rely solely on object-descriptor matrices. Environmental interpretations are made post-hoc as a separate step (in most cases).
Ordination in EEB

Mike Palmer at Oklahoma State University (the other OSU!) maintains an ordination web page that is an excellent resource:

http://ordination.okstate.edu/

Here, all of the ordination methods are explained, vocabulary defined, references provided, links to other resources & software, a listserv, etc.

History of Ordination in EEB

• 1901 - Pearson develops PCA as a regression technique
• 1927 - Spearman applies factor analysis to psychology
• 1930 - Ramaswky uses informal ordination technique & term "Ordnung" in ecology
• 1933 - Hotelling develops PCA for understanding the correlation matrix
• 1936 - Curtis and McIntosh employ the "continuum index" approach
• 1952 - Williams uses Correspondence Analysis
• 1954 - Goodall uses the term "ordination" for PCA
• 1957 - Bray-Curtis (Polar) ordination
• 1964 - Kendal develops NMDS
• 1970s - Whittaker develops theoretical foundations of gradient analysis
• 1973 - Hill revives Correspondence Analysis
• 1976 - Canonical Correlation introduced to ecology
• 1977 - Fasham, Prentice use NMDS
• 1979 - DCA introduced by Hill and Gauch
• 1982 - Gauch's "Multivariate Analysis in Community Ecology"
• 1986 - CCA introduced by ter Braak
• 1986 - Fuzzy set ordination introduced by Roberts
• 1988 - ter Braak and Prentice's "Theory of Gradient Analysis"

Principal Component Analysis (PCA)

In a multinormal distribution, the first principal axis is the line that goes through the greatest dimension of the concentration ellipsoid describing the distribution.

In the same way, the following principal axes (orthogonal to one another; i.e., at right angles & successively shorter) go through the following greatest dimensions of the $p$-dimensional ellipsoid.

A maximum of $p$ principal axes may be derived from a data table containing $p$ variables.
Principal Axes

The principal axes of a dispersion matrix $S$ are found by solving:

$$(S - \lambda I)u_k = 0$$

whose characteristic equation

$$|S - \lambda I| = 0$$

is used to compute the eigenvalues $\lambda_k$. The eigenvectors $u_k$ associated with the $\lambda_k$ are found by putting the different $\lambda_k$ values in turn into the first equation. These eigenvectors are the principal axes of the dispersion matrix $S$; the eigenvectors are normalized (scaled to unit length) before computing the principal components, which gives the coordinates of the objects on the successive principal axes.

Vocabulary

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

**First principal axis.** Line passing through the greatest dimension of the ellipsoid; major axis of the ellipsoid.

**Principal components.** New variates specified by the axes of rigid rotation of the original system of co-ordinates, and corresponding to the rigid rotation of original coordinates. Gives the positions of the objects in the new system of coordinates.

**Principal component axes.** System of axes (aka, principal axes) resulting from the rotation just described.

Principal Component Analysis (PCA)

PCA was first described by Hotelling (1933) and more clearly articulated in a seminal paper by Rao (1964). PCA is a powerful technique in EEB because of its properties:

1. Since any dispersion matrix $S$ is symmetric, its principal axes $u_k$ are orthogonal to one another. They correspond to linearly independent directions in the concentration ellipsoid of the distribution of objects.

2. The eigenvalues $\lambda_k$ of a dispersion matrix $S$ give the amount of variance corresponding to successive principal axes.

3. Because of (1) & (2), PCA is usually capable of summarizing a dispersion matrix containing many descriptors in just 2 or 3 dimensions.
Principal Component Analysis (PCA)

Let’s develop a simple numerical example involving 5 objects and 2 quantitative descriptors:

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

After centering on the column means,

\[
Y_{centered} = \begin{bmatrix}
-3.2 & -1.6 \\
-2.2 & 1.4 \\
-0.2 & -2.6 \\
1.8 & 3.4 \\
3.8 & -0.6 \\
\end{bmatrix}
\]

NB: in practice, PCA would never be used with 2 descriptors because one could simply do a bivariate scatter plot.

Principal Component Analysis (PCA)

- Simple Graphical Interpretation -

(a) 5 objects plotted WRT 2 descriptors, y₁ and y₂.
(b) After centering the data, objects are plotted with respect to means (dashed lines).
(c) Objects plotted WRT principal axes I & II, which are centered.
(d) Two systems of axes b & c are superimposed after a rotation of 26°34’.

Computing Eigenvectors

The dispersion (covariance) matrix \( S \) can be computed directly, by multiplying the matrix of centered data with its transpose:

\[
S = \frac{1}{n-1} \{Y - \bar{Y}\} \{Y - \bar{Y}\}^T = \begin{bmatrix} 8.2 & 1.6 \\ 1.6 & 5.8 \end{bmatrix}
\]

The corresponding characteristic equation is:

\[
[S - \lambda I] \begin{bmatrix} \lambda_1 \\ 0 \end{bmatrix} = 0
\]
Computing Eigenvectors

Solving for the characteristic polynomial, the eigenvalues are \( \lambda_1 = 9 \) and \( \lambda_2 = 5 \).

The total variance remains the same, but it is partitioned in a different way: the sum of the variances on the main diagonal of matrix \( S \) is \( (8.2 + 5.8 = 14) \), while the sum of the eigenvalues is \( (9 + 5 = 14) \). \( \lambda_1 = 9 \) accounts for 64.3% of the variance and \( \lambda_2 \) makes up the difference (35.7%).

There are always as many eigenvalues as there are descriptors. The successive eigenvalues account for progressively smaller fractions of variance.

Computing Eigenvectors

Now, introducing in turn the \( \lambda_k \)'s into the matrix equation:

\[
(S - \lambda I)u_k = 0
\]

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

\[
\begin{bmatrix}
0.8944 & -0.4472 \\
0.4472 & 0.8944 \\
\end{bmatrix}
\]

One can easily verify orthogonality of the eigenvectors:

\[
\mathbf{u}_1 \cdot \mathbf{u}_2 = (0.8944 \times (-0.4472)) + (0.4472 \times 0.8944) = 0
\]

NB: \( \cos^{-1} 0.8944 = 26^\circ34' \) [angle of rotation!]

Computing Principal Components

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.

The positions of all objects are given by matrix \( \mathbf{F} \) of the transformed variables, and is called the matrix of component scores:

\[
\mathbf{F} = [\mathbf{y} - \bar{\mathbf{y}}] \mathbf{U}
\]

where \( \mathbf{U} \) is the matrix of eigenvectors and \( [\mathbf{y} - \bar{\mathbf{y}}] \) the matrix of centered observations.
Computing Principal Components

NB: this would not be the case if \( U \) had been multiplied by \( Y \) (instead of the centered matrix) as in some special forms of PCA (i.e., non-centered PCA).

Now, for our numerical example:

\[
F = \begin{bmatrix}
-3.2 & -1.6 \\
-2.2 & 1.4 \\
-0.2 & -2.6 \\
1.8 & 3.4 \\
3.8 & -0.6 \\
\end{bmatrix}
\begin{bmatrix}
0.8944 & 0.4472 & 0.2 & 2.6 \\
0.4472 & 0.8944 & 3.8 & 0.6 \\
\end{bmatrix}
\begin{bmatrix}
-3.578 & 0 \\
-1.342 & 2.236 \\
-1.342 & -2.236 \\
3.180 & 2.236 \\
3.130 & -2.236 \\
\end{bmatrix}
\]

NB: this would not be the case if \( U \) had been multiplied by \( Y \) (instead of the centered matrix) as in some special forms of PCA (i.e., non-centered PCA).

Now, for our numerical example:

Computing Principal Components

Since the two columns of the matrix of component scores are the coordinates of the five objects WRT the principal axes, they can be used to plot the objects WRT principal axes I and II. PCA has simply rotated the axes by 26°34' in such a way that the new axes correspond to the two main components of variation.

NB1: The relative positions of the objects in the rotated \( p \)-dimensional space of principal components are the same as the \( p \)-dimensional space of the original descriptors.

NB2: This means that Euclidean distances among objects have been preserved through the rotation of axes.

NB3: This is one of the important properties of PCA discussed previously.

Computing Principal Components

The quality of the representation in reduced Euclidean space with \( m \) dimensions only (\( m \times p \)) may be assessed by an "R^2-like ratio" (analogous to regression):

\[
\frac{\sum \lambda_i}{\sum \lambda_i}
\]

NB: the denominator is the trace of matrix \( S \).

Given our example, we would find 9/(9+5) = 0.643 of the total variance along the first principal component (a confirmation of our previous summing of eigenvalues).
Contributions of Descriptors

PCA 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 original descriptors in the reduced space.

These can be described via projection in a reduced space matrix \((U\Lambda^{1/2})\) resulting in scalars originating from the centered projection. The result is often portrayed as a biplot where both observations and descriptors are graphed on the same plot.

**Biplot Example**

Legendre et al. 1984

Time series of 10 observations from a Canadian river.

12 descriptors including 5 species of benthic gastropods and 7 environmental variables.

NB: spp. & env. descriptors scores all multiplied by 5 prior to plotting.

Contributions of Descriptors

One approach to studying the relationships among descriptors consists of scaling the eigenvectors in such a way that the cosines of the angles between the descriptor-axes be proportional to their covariances.

In this approach, the angles between the descriptor-axes are between 0° (max. pos. cov.) and 180° (max. neg. cov.); and 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 \(\sqrt{\lambda_k}\). NB: Using this scaling, ED among objects is NOT preserved.
Contributions of Descriptors

Using the diagonal matrix of eigenvalues \( \Lambda \), the new matrix of eigenvectors can be directly computed by means of expression \( U \Lambda^{1/2} \).

Thus, for our numerical example:

\[
\begin{bmatrix}
0.8944 & -0.4472 & \sqrt{3} & 0 \\
0.4472 & 0.8944 & \sqrt{2} & 0 \\
\end{bmatrix} \begin{bmatrix}
\sqrt{3} & 0 \\
0 & \sqrt{2} \\
\end{bmatrix} = \begin{bmatrix}
2.6333 & -1.000 \\
1.3416 & 2.000 \\
\end{bmatrix}
\]

Principal Components of a Correlation Matrix

Even though PCA is defined for a dispersion matrix \( S \), it can also be carried out on a correlation matrix \( R \) since correlations are covariances of standardized descriptors.

In an \( R \) matrix, all of the diagonal elements are one. It follows that the sum of the eigenvalues, which corresponds to the total variance of the dispersion matrix, is equal to the order of \( R \), which is given by the number of descriptors \( p \).

PCs extracted from correlation matrices are not the same as those computed from dispersion matrices. **Beware**: some software apps only allow computation from a correlation matrix and this may be wholly inappropriate under certain situations!

Principal Components of a Correlation Matrix

In the case of correlations, the descriptors are standardized. Thus, the distances are independent of measurement units, whereas those in the space of the original descriptors vary according to scales.

When the descriptors are all of the same kind and order of magnitude, and have the same units, it is clear that the \( S \) matrix must be used.

When the descriptors are of a heterogeneous nature, it is more appropriate to use an \( R \) matrix.
Principal Components of a Correlation Matrix

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

$$ F = \left[ \frac{y - \bar{y}}{s_y} \right] U $$

where $s_y$ is the standard deviation of $y$.

Principal component analysis is still only a rotation of the system of axes. However, since the descriptors are now standardized, the objects are not positioned in the same way as if the descriptors had simply been centered (i.e., PCA from $S$).

Standardized vs. Unstandardized

In community studies standardization is often desirable when a small number of species are dominant across all of your samples (i.e., Simpson's dominance is great). It prevents "swamping" of the uncommon species. Although, in certain cases this may be seen as inappropriate.

Standardization must be done when the quantities of different descriptors are measured in different units.

Data matrices whose elements are the values of incomparable environmental elements should be standardized.

Centered vs. Uncentered

In addition to standardization, one must also consider whether the data should be "centered." The vast majority of published EEB studies are on centered PCAs, but, it may not always be the best approach to visualizing the data.

An uncentered PCA is called for when the data exhibit between-axes heterogeneity, i.e., when there are clusters of data points such that each cluster has negligible projections on some subset of the axes; a different subset of axes is required for each cluster.

A centered PCA is appropriate when the data exhibit little or no between-axes heterogeneity; i.e., the data points have appreciable projections on all axes.
Centered vs. Uncentered

In practice, data are often obtained for which it is not immediately obvious whether the between-axes heterogeneity exceeds the within-axes heterogeneity or vice versa.

When this happens, the recommendation is to do BOTH a centered and uncentered PCA.

NB: There are many analytical approaches to data centering and standardization in PCA. We have just covered the basics here.

### 4 Types of PCA

We have now essentially defined four basic types of PCA:
- Unstandardized Uncentered PCA
- Standardized Uncentered PCA
- Unstandardized Centered PCA
- Standardized Centered PCA

To evaluate, consider a matrix $X$ (from Pielou 1984) with two descriptors and ten objects:

$$
X = \begin{bmatrix}
22 & 20 \\
5 & 30 \\
3 & 13 \\
4 & 30 \\
2 & 17 \\
5 & 42 \\
6 & 27 \\
5 & 25 \\
0 & 25 \\
6 & 43 \\
\end{bmatrix}
$$

Confirm for yourself the following:

<table>
<thead>
<tr>
<th></th>
<th>Unstandardized Uncentered PCA</th>
<th>Unstandardized Centered PCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{n}XX'$</td>
<td>[ \begin{bmatrix} 3644 &amp; 1641 \ 1641 &amp; 889 \end{bmatrix} ]</td>
<td>[ \begin{bmatrix} 781.9 &amp; 132.3 \ 132.3 &amp; 93.8 \end{bmatrix} ]</td>
</tr>
<tr>
<td>$U = \frac{1}{\sqrt{n}}XX'\Lambda^{1/2}$</td>
<td>[ \begin{bmatrix} 0.906 &amp; 0.423 \ -0.423 &amp; 0.906 \end{bmatrix} ]</td>
<td>[ \begin{bmatrix} 0.993 &amp; 0.183 \ -0.183 &amp; 0.993 \end{bmatrix} ]</td>
</tr>
<tr>
<td>$\Lambda = \begin{bmatrix} 4409 &amp; 0 \ 0 &amp; 124 \end{bmatrix}$</td>
<td></td>
<td>[ \begin{bmatrix} 806.4 &amp; 0 \ 0 &amp; 69.2 \end{bmatrix} ]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Standardized Uncentered PCA</th>
<th>Standardized Centered PCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{n}ZZ'$</td>
<td>[ \begin{bmatrix} 6.06 &amp; 6.06 \ 6.06 &amp; 9.48 \end{bmatrix} ]</td>
<td>[ \begin{bmatrix} 1 &amp; 0.488 \ 0.488 &amp; 1 \end{bmatrix} ]</td>
</tr>
<tr>
<td>$U = \frac{1}{\sqrt{n}}ZZ'\Lambda^{1/2}$</td>
<td>[ \begin{bmatrix} 0.561 &amp; 0.828 \ -0.828 &amp; 0.561 \end{bmatrix} ]</td>
<td>[ \begin{bmatrix} 0.707 &amp; 0.707 \ -0.707 &amp; 0.707 \end{bmatrix} ]</td>
</tr>
<tr>
<td>$\Lambda = \begin{bmatrix} 13.593 &amp; 0 \ 0 &amp; 0.548 \end{bmatrix}$</td>
<td></td>
<td>[ \begin{bmatrix} 1.488 &amp; 0 \ 0 &amp; 0.512 \end{bmatrix} ]</td>
</tr>
</tbody>
</table>
Meaningful Components

Since the principal components correspond to progressively smaller fractions of the total variance, one must determine how many components are biologically meaningful (i.e., what is the dimensionality of the reduced space?).

Shepard diagrams are one approach, but there are others, perhaps better.

An empirical rule-of-thumb (aka Kaiser-Guttman criterion) when using the S-matrix is that one should interpret a principal component if the corresponding $\lambda$ eigenvalue is larger than the mean of the $\lambda$'s. For the R-matrix, meaningful components $> 1$.

A scree plot (Cattell 1966) is often useful in determining $d$. This is simply a rank-order plot of the eigenvalues in decreasing order.

Note: mean $\lambda = 13.5$, therefore using K-G criterion, two PCs probably adequate for interpretation; smaller eigenvalues are mostly just noise. Alternatively, all values above a line fitted through the smallest values are meaningful.

Misuses of PCA

The power and general utility of PCA have encouraged some biologists to go beyond the limits of the model. Some transgressions have no effect, others have dramatic effects.

PCA was originally defined for data with multi-normal distributions, thus the data should really be normalized.

Deviations from normality do not necessarily bias the results, however, one should be particularly careful of the descriptors and try to ensure they are not skewed or have outliers.
Misuses of PCA

Technically, a dispersion matrix cannot be estimated using a number of observations \( n \) smaller than or equal to the number of descriptors \( p \). The number of objects must be larger than the number of descriptors.

Do not transpose a primary matrix and compute correlations among the objects instead of among the descriptors. (This is a bit odd especially since PCA provides information about the relationships of both objects and descriptors.)

Covariances and correlations are defined for quantitative descriptors only. Do not use multi-state qualitative descriptors—means and variances are useless.

---

Misuses of PCA

When calculated over data sets with many double-zeros, coefficients such as the covariance or correlation lead to ordinations that produce inadequate estimates of the distances among objects.

This makes PCA particularly inappropriate for analyzing many biological data sets containing samples \( \times \) species, but, it remains as an excellent analytical procedure for analyzing environmental, systematic, or morphometric data.

---

Misuses of PCA

In addition to the many zeroes problem, there is a fundamental assumption that the descriptors are linearly (or at least monotonically) related to each other (lines or planes).

While this may be true with certain types of data, it is rarely the case with community data where species abundances are being analyzed. Most species are unimodally distributed.
Consider this hypothetical coenocline. There are 3 species distributed along the gradient, each with a unimodal response function. What happens if you do a PCA on these data?

Certainly there is a nonlinearity problem here.

The red line (in 2-D space PC-1 vs PC-2) is what is known as the "horseshoe effect" (an extreme version of the "arch" effect), where axis 2 exhibits a parabolic curve & is not a true representation of the linear gradient.

PCA is such a widely used procedure that almost every major software application supports it. My preferences for PCA are NCSS, SAS, and R.

Let's look at a PCA example using 6 descriptors and 30 objects and work through a session in NCSS.
Missing Data

NCSS needs to know how you wish to handle missing data. (NB: a zero value is different than a missing value!) If data are missing there are several options available, the three most common of which are:

(a) delete the entire row from which this observation belongs (select "none" option)--this results in an obvious loss of data (sometimes large),

(b) the "mean option" just drops the mean of that descriptor in to the matrix for analysis (but, while simple, this causes estimation problems later).

(c) Estimate the covariance matrix $S$ and use these coefficients in a regression to estimate the missing datum based on the other data that are available; once each missing value is estimated, a new covariance matrix is calculated and the process is repeated until there is convergence. This convergence is measured by the trace of the covariance matrix. This is the recommended procedure.

Outliers

There are various ways to approach dealing with outliers (to which PCA is quite sensitive because of the distortion they cause in the variance-covariance structure):

(a) Start with doing a full univariate EDA of all of your descriptors. You may wish to winsorize or delete selected severe outliers. If you delete an observation (row) make sure you specify a procedure for handling missing data (previous).

(b) There are several algorithms that are used to construct a PCA. The two that NCSS supports are "regular" (what we learned) and "robust". The latter applies weights to outlying points to minimize their influence. Both $S$ and $R$ can be estimated robustly.
Rotations

In addition to the "normal" approach for constructing a PCA, there are various types of orthogonal rotation techniques. In other words, in order to reveal data structure and interpret the meaning of your axes, it may be advisable to provide an additional orthogonal rotation of your data.

Two options are available in NCSS: varimax and quartimax. In varimax, the axes are rotated to maximize the sum of the variances of the squared loadings within each column of the loadings matrix. In quartimax, the rows of the matrix containing the rotated factors are maximized rather than the columns (varimax).

Suggestion: start with normal, then try a rotation if necessary.

NCSS Output

Without going through all of the output, I would like to draw your attention to several key points in the output of a single software application:

This is Bartlett's sphericity test (Bartlett, 1950) for testing the null hypothesis that the correlation matrix is an identity matrix (all correlations are zero). If you get a probability (P) value greater than 0.05, you should not perform a PCA on the data.

Scree plot suggests only first 2 axes are meaningful.
The eigenvectors are the weights that relate the scaled original variables to the factors. These coefficients may be used to determine the relative importance of each variable in forming the factor. Often, the eigenvectors are scaled so that the variances of the factor scores are equal to one. These scaled eigenvectors are given in the Score Coefficients section described later.

The communality is the proportion of the variation of a variable that is accounted for by the factors that are retained. It is the $R^2$ value that would be achieved if this variable were regressed on the retained factors. This table value gives the amount added to the communality by each factor.

This report is useful for detecting outliers—observations that are very different from the bulk of the data. To do this, two quantities are displayed: $T^2$ and $Q$. Both suggest rows 2&3 are having inordinate influence and need to be scrutinized.
This report presents the individual factor scores scaled so each column has a mean of zero and a standard deviation of one. These are the values that are plotted. Remember, there is one row of score values for each observation and one column for each factor that was kept.

Shown are plots of the first three axes. Note the value of plotting the third axis, even though only two are indicated as important. Observation 3 is clearly an outlier (as suggested by previous stat tests). This point needs to be dropped and analysis re-run.

Let’s look at another tutorial, this time using R. This will give you another perspective and a broader appreciation of what is available for this type of analysis.

There are two ways to perform PCA in R: `princomp()` and `prcomp()`.

The library `LabDSV` contains a third which is called `pca()` which essentially calls `princomp()`, but adds different computing and plotting options useful for EEB.
First, make sure to install and load BOTH packages LabDSV and vegan.

Next, access the Bryce Canyon data set from the vegan package:

```r
> library(vegan)
> library(labdsv)
> data(bryceveg)
```

Will run a PCA on the Bryce Canyon vegetation data set, using a correlation matrix and only calculating scores for the first 10 eigenvectors.

There are then four aspects that need to be considered:

1. Variance explained by eigenvector
2. Cumulative variance by eigenvector
3. Species loading by eigenvector
4. Plot scores by vector

Variance Explained

```r
> summary(pca.1, dim=3)
```

NB: For a reason peculiar to R, the first line is SD, so you must square it to get the variance.
Variance Explained

We can look at the same information graphically:

\[ \text{varplot.pca(pca.1)} \]

Species Loadings

By default, small values are depressed. You can see (from what is shown) that eigenvector 1 is negatively correlated with \texttt{arcpat} and \texttt{ceamar} and positively correlated with \texttt{chrvis}.

Plot Scores

Similar output can be obtained for the actual plot scores (here limited to just the first 3 dimensions, which is usually sufficient and a partial listing of stands):

\[
\begin{array}{cccc}
\text{PC1} & \text{PC2} & \text{PC3} \\
0.001 & -0.120 & 0.001 \\
0.001 & -0.050 & 0.001 \\
0.001 & -0.030 & 0.001 \\
0.001 & -0.020 & 0.001 \\
0.001 & -0.010 & 0.001 \\
0.001 & -0.000 & 0.001 \\
\end{array}
\]
Plot Scores

These plot scores are typically what are used to produce the final PCA plot that we usually want in an EEB application.

The defaults in LabDSV are designed to provide optimal output for most EEB applications (not so for the base R package), but can be altered as desired.

Alternatively, these scores can be copied into a separate graphics program and the plot constructed there.

```r
> plot(pca.1, title="Bryce Canyon")
```

The default for plot is PC-1 vs. PC-2; however, you can look at other dimensions, change symbols, colors, etc.:

```r
> plot(pca.1, ax=1, ay=3, col=3, pch=3, title="Bryce Canyon")
```

NB: R also supports interactive point highlighting. After creating a graph, enter:

```r
> plotid(pca.1)
```

Then click on points on your graph to see what happens!
R-Tutorial
- Summary -