9 Multivariate Statistics

9.1 Introduction

Multivariate analysis aims to understand and describe the relationship between an arbitrary number of variables. Earth scientists often deal with multivariate data sets, such as microfossil assemblages, geochemical fingerprints of volcanic ashes or clay mineral contents of sedimentary sequences. If there are complex relationships between the different parameters, univariate statistics ignores the information content of the data. There is a number of methods, however, for investigating the scaling properties of multivariate data.

A multivariate data set consists of measurements of p variables on n objects. Such data sets are usually stored in n-by-p arrays:

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

The columns of the array represent the p variables, the rows represent the n objects. The characteristics of the 2nd object in the suite of samples is described by the vector in the second row of the data array:

$$X_2 = (x_{21} \quad x_{22} \quad \ldots \quad x_{2p})$$

As an example, assume the microprobe analysis on glass shards from volcanic ashes in a tephrochronology project. Then, the variables represent the p chemical elements, the objects are the n ash samples. The aim of the study is to correlate ashes by means of their geochemical fingerprints.

Most of the multi-parameter methods simply try to overcome the main difficulty associated with multivariate data sets. This problem relates to the data visualization. Whereas the character of an univariate or bivariate data

M. H. Trauth, *MATLAB® Recipes for Earth Sciences*

set can easily be explored by visual inspection of a 2D histogram or an *xy* plot (Chapter 3), the graphical display of a three variable data set requires a projection of the 3D distribution of data points into 2D. It is impossible to imagine or display a higher number of variables. One solution to the problem of visualization of high-dimensional data sets is the reduction of dimensionality. A number of methods group highly-correlated variables contained in the data set and then explore a smaller number of groups.

The classic methods to reduce dimensionality are the *principal component analysis* (PCA) and the *factor analysis* (FA). These methods seek the directions of maximum variance in the data set and use these as new coordinate axes. The advantage of replacing the variables by new groups of variables is that the groups are uncorrelated. Moreover, these groups often help to interpret the multivariate data set since they often contain valuable information on process itself that generated the distribution of data points. In a geochemical analysis of magmatic rocks, the groups defined by the method usually contain chemical elements with similar ion size that are observed in similar locations in the lattice of certain minerals. Examples for such behavior are Si^{4+} and Al^{3+}, and Fe^{2+} and Mg^{2+} in silicates, respectively.

The second important suite of multivariate methods aims to group objects by their similarity. As an example, *cluster analysis* (CA) is often applied to correlate volcanic ashes as described in the above example. Tephrochronology tries to correlate tephra by means of their geochemical fingerprint. In combination with a few radiometric age determinations of the key ashes, this method allows to correlate sedimentary sequences that contain these ashes (e.g., Westgate 1998, Hermanns et al. 2000). More examples for the application of cluster analysis come from the field of micropaleontology. In this context, multivariate methods are employed to compare microfossil assemblages such as pollen, foraminifera or diatoms (e.g., Birks and Gordon 1985).

The following text introduces the most important techniques of multivariate statistics, principal component analysis and cluster analysis (Chapter 9.2 and 9.4). A nonlinear extension of the PCA is the *independent component analysis* (ICA) (Chapter 9.3). First, the chapters provide an introduction to the theory behind the techniques. Subsequently, the use of these methods in analyzing earth sciences data is illustrated with MATLAB functions.

9.2 Principal Component Analysis

The principal component analysis (PCA) detects linear dependencies between variables and replaces groups of correlated variables by new uncor-

related variables, the *principal components* (PC). The performance of the PCA is better illustrated with help of a bivariate data set than a multivariate one. Figure 9.1 shows a bivariate data set that exhibits a strong linear correlation between the two variables x and y in an orthogonal xy coordinate system. The two variables have their univariate means and variances (Chapter 3). The bivariate data set can be described by the bivariate sample mean and the covariance (Chapter 4). The xy coordinate system can be replaced by a new orthogonal coordinate system, where the first axis passes through the long axis of the data scatter and the new origin is the bivariate mean. This new reference frame has the advantage that the first axis can be used to describe most of the variance, while the second axis contributes only a little. Originally, two axes were needed to describe the data set prior to the transformation. Therefore, it is possible to reduce the data dimension by dropping the second axis without losing much information as shown in Figure 9.1.

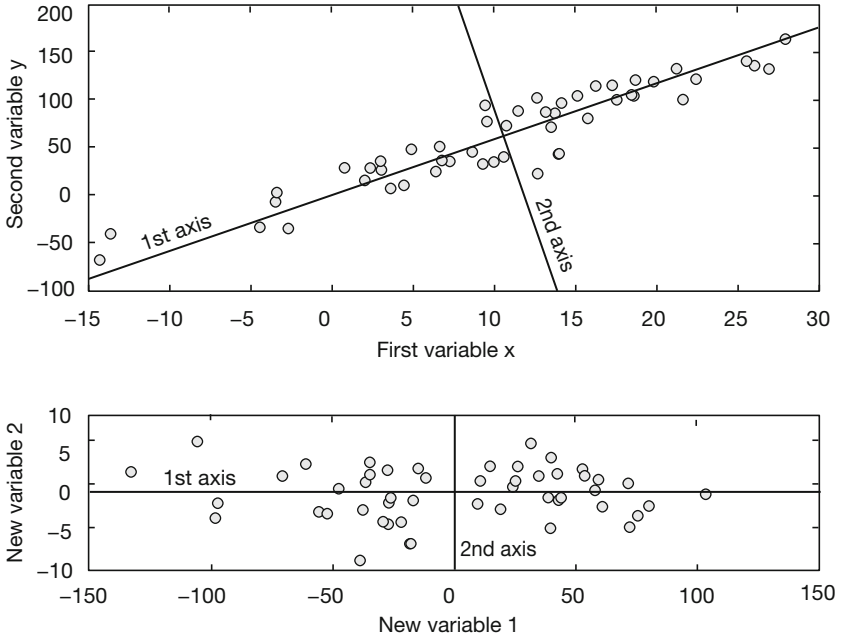

Fig. 9.1 Principal component analysis (PCA) illustrated on a bivariate scatter. The original xy coordinate system is replaced by a new orthogonal system, where the first axis passes through the long axis of the data scatter and the new origin is the bivariate mean. We can now reduce dimensionality by dropping the second axis without losing much information.

This is now expanded to an arbitrary number of variables and samples. Suppose a data set of measurements of p parameters on n samples stored in an n-by-p array.

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

The columns of the array represent the p variables, the rows represent the n samples. After rotating the axis and moving the origin, the new coordinates Y_j can be computed by

$$\begin{aligned} Y_1 &= a_{11}X_1 + a_{12}X_2 + \ldots + a_{1p}X_p \\ Y_2 &= a_{21}X_1 + a_{22}X_2 + \ldots + a_{2p}X_p \\ &\vdots \\ Y_n &= a_{n1}X_1 + a_{n2}X_2 + \ldots + a_{np}X_p \end{aligned}$$

The first principle component PC_1 denoted by Y_1 contains the greatest variance, PC_2 the second highest variance and so forth. All PCs together contain the full variance of the data set. The variance is concentrated in the first few PCs, which explain most of the information content of the data set. The last PCs are generally ignored to reduce the data dimension. The factors a_{ij} in the above equations are the *principal component loads*. The values of these factors represent the relative contribution of the original variables to the new PCs. If the load a_{ij} of a variable X_j in PC_1 is close to zero, the influence of this variable is low. A high positive or negative a_{ij} suggests a strong contribution of the variable X_j. The new values Y_j of the variables computed from the linear combinations of the original variables X_j weighted by the loads are called the *principal component scores*.

In the following, a synthetic data set is used to illustrate the use of the function `princomp` included in the Statistics Toolbox. Our data set contains the percentage of various minerals contained in sediment samples. The sediments are sourced from three rock types: a magmatic rock contains amphibole (*amp*), pyroxene (*pyr*) and plagioclase (*pla*), a hydrothermal vein characterized by the occurrence of fluorite (*flu*), sphalerite (*sph*) and galenite (*gal*), as well as some feldspars (plagioclase and potassium feldspar, *ksp*) and quartz (*qtz*), and a sandstone unit containing feldspars, quartz and clay minerals (*cla*).

Ten samples were taken from various levels of this sedimentary sequence

containing varying amounts of these minerals. The PCA is used to verify the influence of the three different source rocks and to estimate their relative contribution. First, the data are loaded by typing

```
data = load('sediments.txt');
```

Next, we define labels for the various graphs created by the PCA. We number the samples 1 to 10, whereas the minerals are characterized by three-character abbreviations.

```
for i = 1:10
   sample(i,:) = ['sample',sprintf('%02.0f',i)];
end
clear i

minerals = ['amp';'pyr';'pla';'ksp';'qtz';'cla';'flu';'sph';'gal']
```

A successful PCA requires linear correlations between variables. The *correlation matrix* provides a technique for exploring such dependencies in the data set (Chapter 4). The elements of the correlation matrix are Pearson's correlation coefficients for each pair of variables as shown in Figure 9.2. Here, the variables are minerals.

```
corrmatrix = corrcoef(data);
corrmatrix = flipud(corrmatrix);

imagesc(corrmatrix), colormap(hot)
title('Correlation Matrix')
axis square, colorbar, hold
set(gca,'XTickLabel',minerals,'YTickLabel',flipud(minerals))
```

This pseudocolor plot of the correlation coefficients shows strong positive correlations between the minerals *amp*, *pyr* and *pla*, the minerals *ksp*, *qtz* and *cla*, and the minerals *flu*, *sph* and *gal*, respectively. Moreover, some of the minerals show negative correlations. We also observe no dependency between some of the variables, for instance between the potassium feldspar and the vein minerals. From the observed dependencies, we expect interesting results from the application of the PCA.

Various methods exist for scaling the original data before applying the PCA, such as *mean centering* (zero means) or *autoscaling* (mean zero and standard deviation equals one). However, we use the original data for computing the PCA. The output of the function `princomp` includes the principal component loads `pcs`, the scores `newdata` and the variances `variances`.

```
[pcs,newdata,variances] = princomp(data);
```

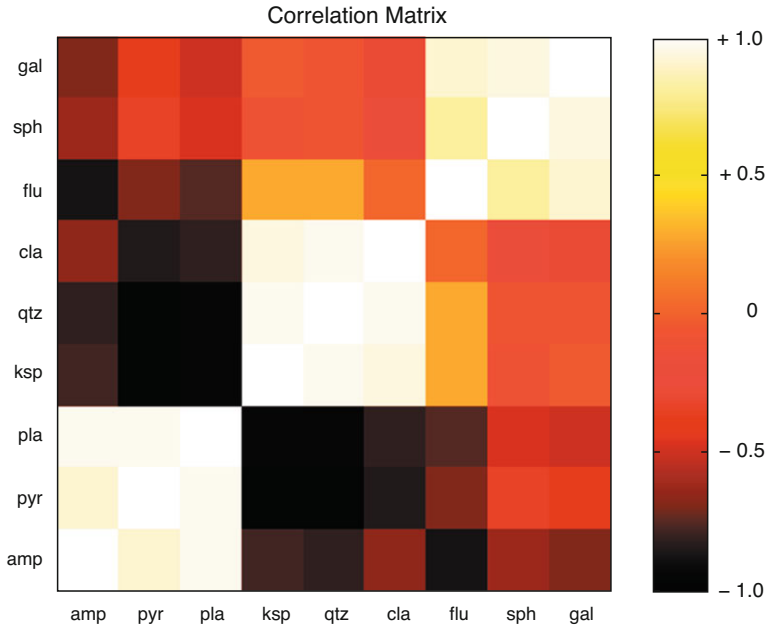

Fig. 9.2 Correlation matrix containing Pearson's correlation coefficients for each pair of variables, such as minerals in a sediment sample. Light colors represent strong positive linear correlations, whereas dark colors document negative correlations. Orange suggests no correlation.

The loads of the first five principal components PC_1 to PC_5 can be shown by typing

```
pcs(:,1:5)

ans =
   -0.3303    0.2963   -0.4100   -0.5971    0.1380
   -0.3557    0.0377    0.6225    0.2131    0.5251
   -0.5311    0.1865   -0.2591    0.4665   -0.3010
    0.1410    0.1033   -0.0175    0.0689   -0.3367
    0.6334    0.4666   -0.0351    0.1629    0.1794
    0.1608    0.2097    0.2386   -0.0513   -0.2503
    0.1673   -0.4879   -0.4978    0.2287    0.4756
    0.0375   -0.2722    0.2392   -0.5403   -0.0068
    0.0771   -0.5399    0.1173    0.0480   -0.4246
```

We observe that PC_1 (first column) has high negative loads in the first three variables *amp*, *pyr* and *pla* (first to third row), and a high positive load in the fifth variable *qtz* (fifth row). PC_2 (second column) has high negative loads in the vein minerals *flu*, *sph* and *gal*, and again a positive load in *qtz*. We create a number of plots of the PCs.

```
subplot(2,2,1), plot(1:9,pcs(:,1),'o'), axis([1 9 -1 1])
text((1:9)+0.2,pcs(:,1),minerals,'FontSize',8), hold
plot(1:9,zeros(9,1),'r'), title('PC 1')

subplot(2,2,2), plot(1:9,pcs(:,2),'o'), axis([1 9 -1 1])
text((1:9)+0.2,pcs(:,2),minerals,'FontSize',8), hold
plot(1:9,zeros(9,1),'r'), title('PC 2')

subplot(2,2,3), plot(1:9,pcs(:,3),'o'), axis([1 9 -1 1])
text((1:9)+0.2,pcs(:,3),minerals,'FontSize',8), hold
plot(1:9,zeros(9,1),'r'), title('PC 3')

subplot(2,2,4), plot(1:9,pcs(:,4),'o'), axis([1 9 -1 1])
text((1:9)+0.2,pcs(:,4),minerals,'FontSize',8), hold
plot(1:9,zeros(9,1),'r'), title('PC 4')
```

The loads of the index minerals and their relationship to the PCs can be used to interpret the relative influence of the source rocks. PC_1 characterized by strong contributions of *amp*, *pyr* and *pla*, and a contribution with an opposite sign of *qtz* probably describes the amount of magmatic rock clasts in the sediment. The second principal component PC_2 is clearly dominated by hydrothermal minerals hence suggesting the detrital input from the vein. PC_3 and PC_4 show a mixed and contradictory pattern of loads and are therefore not easy to interpret. We will later see that this observation is in line with a rather weak and mixed signal from the sandstone source on the sediments.

An alternative way to plot of the loads is a bivariate plot of two principal components. We ignore PC_3 and PC_4 at this point and concentrate on PC_1 and PC_2.

```
plot(pcs(:,1),pcs(:,2),'o')
text(pcs(:,1)+0.02,pcs(:,2),minerals,'FontSize',14), hold
x = get(gca,'XLim'); y = get(gca,'YLim');
plot(x,zeros(size(x)),'r')
plot(zeros(size(y)),y,'r')
xlabel('First Principal Component Loads')
ylabel('Second Principal Component Loads')
```

Here, we observe the same relationships on a single plot that were previously shown on several graphs (Fig. 9.3). It is also possible to plot the data set as functions of the new variables. This needs the second output of `princomp` containing the principal component scores.

```
plot(newdata(:,1),newdata(:,2),'+')
text(newdata(:,1)+0.01,newdata(:,2),sample), hold
x = get(gca,'XLim'); y = get(gca,'YLim');
plot(x,zeros(size(x)),'r')
plot(zeros(size(y)),y,'r')
xlabel('First Principal Component Scores')
ylabel('Second Principal Component Scores')
```

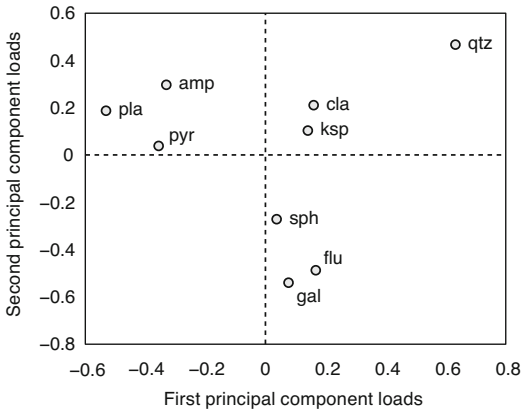

Fig. 9.3 Principal components loads suggesting that the PCs are influenced by different minerals. See text for detailed interpretation of the PCs.

This plot clearly defines groups of samples with similar influences. The samples 1, 2, 8 to 10 dominated by magmatic influences cluster in the left half of the diagram, the samples 3 to 5 dominated by the hydrothermal vein group in the lower part of the right half, whereas the two sandstone dominated samples 6 and 7 fall in the upper right corner.

Next, we use the third output of the function `princomp` to compute the variances of the corresponding PCs.

```
percent_explained = 100*variances/sum(variances)

percent_explained =
   80.9623
   17.1584
    0.8805
    0.4100
    0.2875
    0.1868
    0.1049
    0.0096
    0.0000
```

We see that more than 80% of the total variance is contained in PC_1, around 17% is described by PC_2, whereas all other PCs do not play any role. This means that most of the variability in the data set can be described by two new variables only.

9.3 Independent Component Analysis (by N. Marwan)

The principal component analysis (PCA) is the standard method for separating mixed signals. Such analysis provides signals that are linearly uncorrelated. This method is also called *whitening* since this property is characteristic for white noise. Although the separated signals are uncorrelated, they could still can be dependent, i.e., nonlinear correlation remains. The *independent component analysis* (ICA) was developed to investigate such data. It separates mixed signals into independent signals, which are then nonlinearly uncorrelated. Fast ICA algorithms use a criterion which estimates how gaussian distributed the joint distribution of the independent components is. The less gaussian this distribution is, the more independent the individual components are.

According to the model, n independent signals $x(t)$ are linearly mixed in m measurements.

$$x(t) = As(t)$$

and we are interested in the source signals s_i and in the mixing matrix A. For example, we can imagine that we are on a party and a lot of people talk independently with others. We hear a mixing of these talks and perhaps cannot distinguish the single talks. Now we could install some microphones and use these measurements to separate the single conversations. Hence, this dilemma is also called the *cocktail party problem*. Its correct term is *blind source separation* that is given by

$$s(t) = W^T x(t)$$

where W^T is the separation matrix in order to reverse the mixing and get the original signals. Let us consider a mixing of three signals s_1, s_2 and s_3 and their separation using PCA and ICA. First, we create three periodic signals

```
clear
i = (1:0.01:10 * pi)';
[dummy index] = sort(sin(i));

s1(index,1) = i/31; s1 = s1 - mean(s1);
s2 = abs(cos(1.89*i)); s2 = s2 - mean(s2);
s3 = sin(3.43*i);
```

```
subplot(3,2,1), plot(s1), ylabel('s_1'), title('Raw signals')
subplot(3,2,3), plot(s2), ylabel('s_2')
subplot(3,2,5), plot(s3), ylabel('s_3')
```

Now we mix these signals and add some observational noise. We get a three-column vector x which corresponds to our measurement (Fig. 9.4).

```
randn('state',1);

x = [.1*s1 + .8*s2 + .01*randn(length(i),1),...
     .4*s1 + .3*s2 + .01*randn(length(i),1),...
     .1*s1 +   s3  + .02*randn(length(i),1)];

subplot(3,2,2), plot(x(:,1)), ylabel('x_1'), title('Mixed signals')
subplot(3,2,4), plot(x(:,2)), ylabel('x_2')
subplot(3,2,6), plot(x(:,3)), ylabel('x_3')
```

We begin with the separation of the signals using the PCA. We calculate the

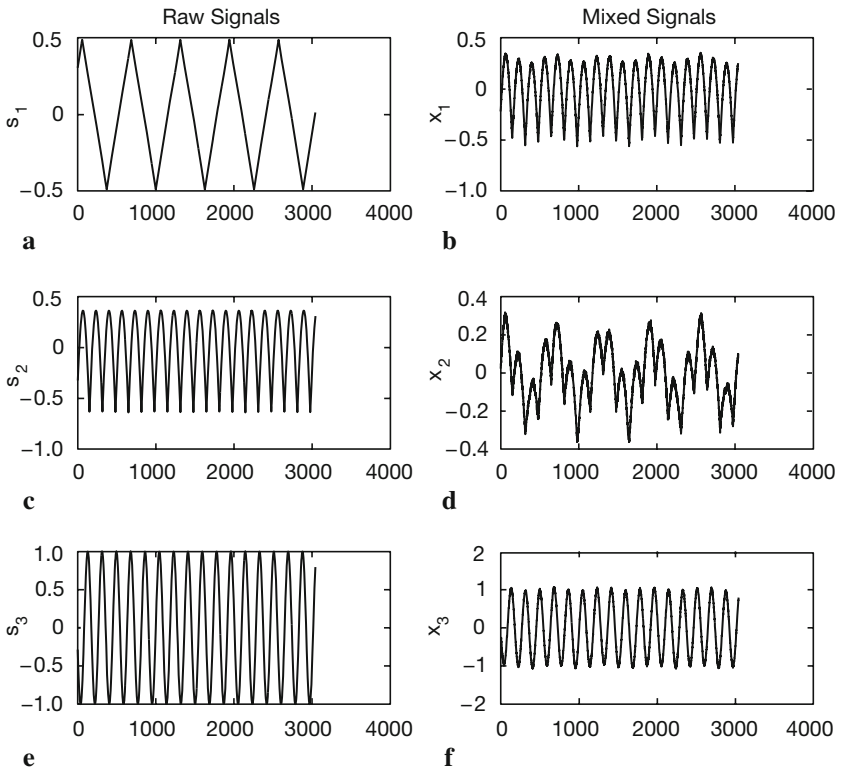

Fig. 9.4 Sample input for the independent component analysis. We first generate three period signals (**a**, **c**, **e**), mix the signals and add some gaussian noise (**b**, **d**, **f**).

principal components and the whitening matrix `W_PCA` with

```
sPCA = sPCA./repmat(std(sPCA),length(sPCA),1);
```

The PC scores `sPCA` are the linearly separated components of the mixed signals x (Fig. 9.5).

```
subplot(3,2,1), plot(sPCA(:,1))
ylabel('s_{PCA1}'), title('Separated signals - PCA')
subplot(3,2,3), plot(sPCA(:,2)), ylabel('s_{PCA2}')
subplot(3,2,5), plot(sPCA(:,3)), ylabel('s_{PCA3}')
```

The mixing matrix A can be found with

```
A_PCA = E * sqrt (D);
W_PCA = inv(sqrt(diag(D))) * E';
```

Next, we separate the signals into independent components. We will do

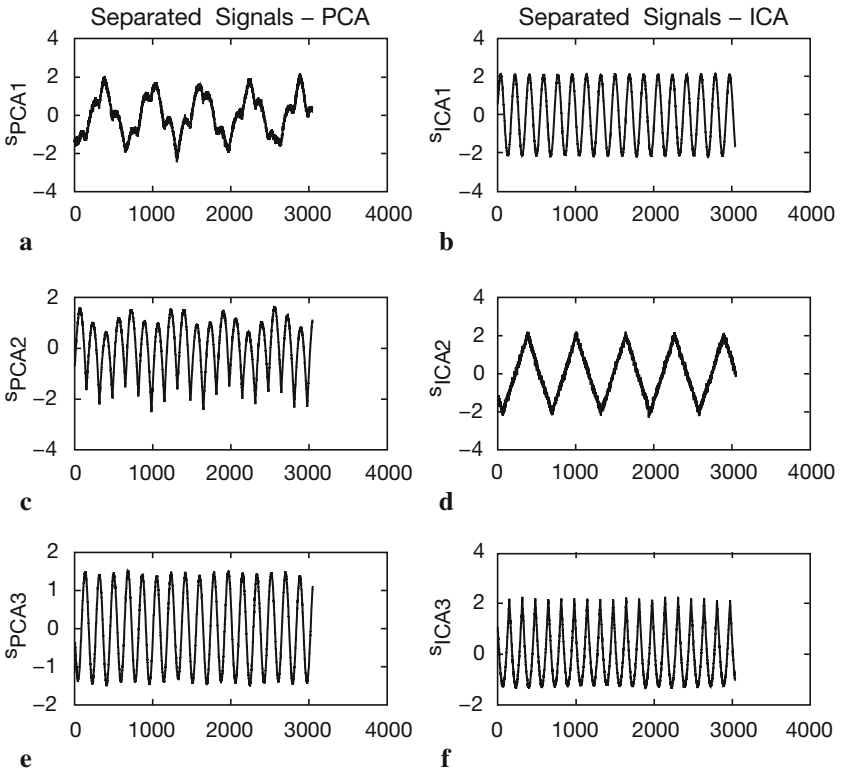

Fig. 9.5 Output of the principal component analysis (**a**, **c**, **e**) compared with the output of the independent component analysis (**b**, **d**, **f**). The PCA has not reliably separated the mixed signals, whereas the ICA found the source signals almost perfectly.

this by using a FastICA algorithm which is based on a fixed-point iteration scheme to find the maximum of the non-gaussianity of the independent components $W^T x$. As the nonlinearity function we use a power of three function for instance.

```
rand('state',1);

div = 0;
B = orth(rand(3, 3) - .5);
BOld = zeros(size(B));

while (1 - div) > eps
   B = B * real(inv(B' * B)^(1/2));
   div = min(abs(diag(B' * BOld)));
   BOld  = B;
   B = (sPCA' * ( sPCA * B) .^ 3) / length(sPCA) - 3 * B;
   sICA = sPCA * B;
end
```

We plot the separated components with (Fig. 9.5)

```
subplot(3,2,2), plot(sICA(:,1)), ylabel('s_{ICA1}'),
   title('Separated signals - ICA')
subplot(3,2,4), plot(sICA(:,2)), ylabel('s_{ICA2}')
subplot(3,2,6), plot(sICA(:,3)), ylabel('s_{ICA3}')
```

The PCA algorithm has not reliably separated the mixed signals. Especially the saw-tooth signal was not correctly found. In contrast, the ICA has found the source signals almost perfectly. The only remarkable differences are the noise, which came through the observation, the wrong sign and the wrong order of the signals. However, the sign and the order of the signals are not really important, because we have generally not the knowledge about the real sources nor their order. With

```
A_ICA = A_PCA * B;
W_ICA = B' * W_PCA;
```

we compute the mixing matrix A and the separation matrix W. The mixing matrix A can be used in order to estimate the portion of the separated signals on our measurements The components a_{ij} of the mixing matrix A correspond to the principal components loads as introduced in Chapter 9.2. A FastICA package is available for MATLAB and can be found at

```
http://www.cis.hut.fi/projects/ica/fastica/
```

9.4 Cluster Analysis

Cluster analysis creates groups of objects that are very similar compared to other objects or groups. It first computes the similarity between all pairs of objects, then it ranks the groups by their similarity, and finally creates a hierarchical tree visualized as a dendrogram. Examples for grouping objects in earth sciences are the correlations within volcanic ashes (Hermanns et al. 2000) and the comparison of microfossil assemblages (Birks and Gordon 1985).

There are numerous methods for calculating the similarity between two data vectors. Let us define two data sets consisting of multiple measurements on the same object. These data can be described as the vectors:

$$X_1 = (x_{11} \quad x_{12} \quad \ldots \quad x_{1p})$$
$$X_2 = (x_{21} \quad x_{22} \quad \ldots \quad x_{2p})$$

The most popular measures of similarity of the two sample vectors are the

- *Euclidian distance* – This is simply the shortest distance between the two points in the multivariate space:

$$\Delta_{12} = \sqrt{(x_{11} - x_{21})^2 + (x_{12} - x_{22})^2 + \ldots + (x_{1p} - x_{2p})^2}$$

  The Euclidian distance is certainly the most intuitive measure for similarity. However, in heterogenic data sets consisting of a number of different types of variables, it should be replaced by the following measure.

- *Manhattan distance* – In the city of Manhattan, one must walk on perpendicular avenues instead of diagonal crossing blocks. The Manhattan distance is therefore the sum of all differences:

$$\Delta_{12} = \left[(x_{11} - x_{21}) + (x_{12} - x_{22}) + \ldots + (x_{1p} - x_{2p})\right]$$

- *Correlation similarity coefficient* – Here, we use Pearson's linear product-moment correlation coefficient to compute the similarity of two objects:

$$r_{x1x2} = \frac{\sum_{i=1}^{n}(x_{1i} - \bar{x}_1)(x_{2i} - \bar{x}_2)}{(n-1)s_{x1}s_{x2}}$$

This measure is used if one is interested in ratios between the variables measured on the objects. However, Pearson's correlation coefficient is highly sensitive to outliers and should be used with care (see also Chapter 4).

- *Inner-product similarity index* – Normalizing the data vectors to one and computing the inner product of these yield another important similarity index. This is often used in transfer function applications. In this example, a set of modern flora or fauna assemblages with known environmental preferences is compared with a fossil sample to reconstruct the environmental conditions in the past.

$$s_{12} = \frac{1}{|X_1|}\frac{1}{|X_2|}\left(x_{11}\ x_{12}\ldots x_{1p}\right)\begin{pmatrix} x_{21} \\ x_{22} \\ \vdots \\ x_{2p} \end{pmatrix}$$

The inner-product similarity varies between 0 and 1. A zero value suggests no similarity and a value of one represents maximum similarity.

The second step in performing a cluster analysis is to rank the groups by their similarity and build a hierarchical tree visualized as a dendrogram. Defining groups of objects with significant similarity and separating clusters depends on the internal similarity and the difference between the groups. Most clustering algorithms simply link the two objects with highest similarity. In the following steps, the most similar pairs of objects or clusters are linked iteratively. The difference between groups of objects forming a cluster is described in different ways depending on the type of data and application.

- *K-means clustering* – Here, the Euclidean distance between the multivariate means of the *K* clusters is used as a measure for the difference between the groups of objects. This distance is used if the data suggest that there is a true mean value surrounded by random noise.

- *K-nearest-neighbors clustering* – Alternatively, the Euclidean distance of the nearest neighbors is used as measure for this difference. This is used

if there is a natural heterogeneity in the data set that is not attributed to random noise.

It is important to evaluate the data properties prior to the application of a clustering algorithm. First, one should consider the absolute values of the variables. For example, a geochemical sample of volcanic ash might show SiO_2 contents of around 77% and Na_2O contents of 3.5%, although the Na_2O content is believed to be of great importance. Here, the data need to be transformed to zero means (*mean centering*). Differences in the variances *and* in the means are corrected by *autoscaling*, i.e., the data are standardized to zero means and variances that equal one. Artifacts arising from closed data, such as artificial negative correlations, are avoided by using *Aitchison's log-ratio transformation* (Aitchison 1984, 1986). This ensures data independence and avoids the constant sum normalization constraints. The log-ratio transformation is

$$x_{tr} = \log(x_i / x_d)$$

where x_{tr} denotes the transformed score (i=1, 2, 3, …, d–1) of some raw data x_i. The procedure is invariant under the group of permutations of the variables, and any variable can be used as divisor x_d.

As an example for performing a cluster analysis, the sediment data stored in *sediment.txt* are loaded and the plotting labels are defined.

```
data = load('sediments.txt');

for i = 1:10
  sample(i,:) = ['sample',sprintf('%02.0f',i)];
end
clear i

minerals= ['amp';'pyr';'pla';'ksp';'qtz';'cla';'flu';'sph';'gal'];
```

Subsequently, the distances between pairs of samples can be computed. The function `pdist` provides many ways for computing this distance, such as the Euclidian or Manhattan *city block* distance. We use the default setting which is the Euclidian distance.

```
Y = pdist(data);
```

The function `pdist` returns a vector `Y` containing the distances between each pair of observations in the original data matrix. We can visualize the distances on another pseudocolor plot.

```
squareform(Y);
imagesc(squareform(Y)), colormap(hot)
title('Euclidean distance between pairs of samples')
xlabel('First Sample No.')
ylabel('Second Sample No.')
colorbar
```

The function `squareform` converts `Y` into a symmetric, square format, so that the elements `(i,j)` of the matrix denote the distance between the `i` and `j` objects in the original data. Next, we rank and link the samples with respect to their inverse distance using the function `linkage`.

```
Z = linkage(Y);
```

In this 3-column array `Z`, each row identifies a link. The first two columns identify the objects (or samples) that have been linked, the third column contains the individual distance between these two objects. The first row (link) between objects (or samples) 1 and 2 has the smallest distance corresponding to the highest similarity. Finally, we visualize the hierarchical clusters as a dendrogram which is shown in Figure 9.6.

```
dendrogram(Z);
xlabel('Sample No.')
ylabel('Distance')
box on
```

Clustering finds the same groups as the principal component analysis. We observe clear groups consisting of samples 1, 2, 8 to 10 (the magmatic source rocks), samples 3 to 5 (the hydrothermal vein) and samples 6 and 7 (the sandstone). One way to test the validity of our clustering result is the *cophenet correlation coefficient*. The value of

```
cophenet(Z,Y)

ans =
    0.7579
```

looks convincing, since the closer this coefficient is to one, the better is the cluster solution.

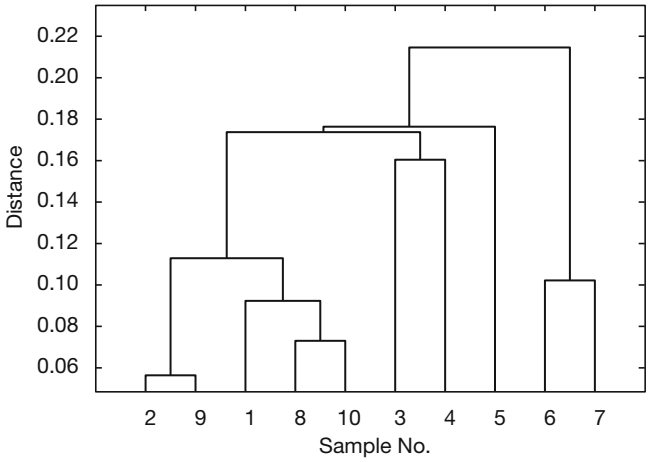

Fig. 9.6 Output of the cluster analysis. The dendrogram shows clear groups consisting of samples 1, 2, 8 to 10 (the magmatic source rocks), samples 3 to 5 (the magmatic dyke containing ore minerals) and samples 6 and 7 (the sandstone unit).

Recommended Reading

Aitchison J (1984) The Statistical Analysis of Geochemical Composition. Mathematical Geology 16(6):531–564

Aitchison J (1999) Logratios and Natural Laws in Compositional Data Analysis. Mathematical Geology 31(5):563–580

Birks HJB, Gordon AD (1985) Numerical Methods in Quaternary Pollen Analysis. Academic Press, London

Brown CE (1998) Applied Multivariate Statistics in Geohydrology and Related Sciences. Springer, Berlin Heidelberg New York

Hermanns R, Trauth MH, McWilliams M, Strecker M (2000) Tephrochronologic Constraints on Temporal Distribution of Large Landslides in NW-Argentina. Journal of Geology 108:35–52

Pawlowsky-Glahn V (2004) Geostatistical Analysis of Compositional Data – Studies in Mathematical Geology. Oxford University Press, Oxford

Reyment RA, Savazzi E (1999) Aspects of Multivariate Statistical Analysis in Geology. Elsevier Science, Amsterdam

Westgate JA, Shane PAR, Pearce NJG, Perkins WT, Korisettar R, Chesner CA, Williams MAJ, Acharyya SK (1998) All Toba Tephra Occurrences Across Peninsular India Belong to the 75,000 yr BP Eruption. Quaternary Research 50:107–112