17 Common trends and sudden changes

In this chapter, we start discussing various methods to estimate long-term patterns in time series. We call these patterns 'trends', but it should be noted that they are not restricted to being straight lines. Some of the methods can be applied on univariate time series and others require multiple time series. If the data are available on a monthly basis, one should make a distinction between seasonal variation and long-term patterns. In order to do this, the seasonal component needs to be determined and dealt with in some way. We will discuss three methods, of increasing mathematical complexity, for estimating common patterns in time series. In the last section, we discuss a technique that can be used to identify sudden changes.

17.1 Repeated LOESS smoothing

Figure 16.3 showed the annual time series of CPUE of the lobster species Nephrops at 11 stations in the Atlantic Ocean south of Iceland, between 1960 and 1999. We noted that most time series follow a similar pattern over time. To focus on the question of whether there are any common trends in the data, various methods can be used. The conceptually easiest approach is based on repeated LOESS smoothing. More difficult methods are discussed in the following two sections.

To visualise the general pattern in the CPUE time series, we will fit a LOESS smoother (Chapter 7) through each time series, and plot these smoothers in one graph. This process was applied on each time series and the results are plotted in Figure 17.1. The thick line shows the mean value of the smoothing curves. It was calculated by averaging the smoothing values in each year. It may be an option to standardise or mean delete the series before doing this. In our case, the mean curve seems to capture the patterns of most smoothing curves reasonably well. We used a span width of 0.5 for each LOESS smoother. This means that 0.5×40 years = 20 years are used in the windows around each target value. A small span width results in a curve that fits the data reasonably well, but it may not be smooth. On the other hand, a large span gives a smoother curve, but it may not follow the patterns of the data. The decision of which span width to use is discussed later.

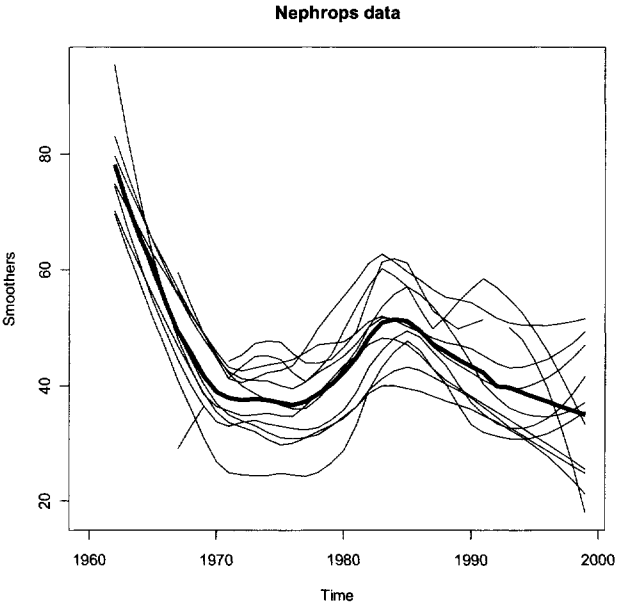

Figure 17.1. Eleven smoothing curves for the Nephrops time series data. The thick line represents the mean smoothing curve.

Mathematically, each smoothing curve in Figure 17.1 was estimated using the following model:

$$Y_t = f(\text{Time}_t, \lambda_1) + e_t$$

where $f(.)$ is a function obtained by the LOESS smoothing method; see Cleveland (1993) for details. The amount of smoothing is also called the degrees of freedom λ_1. Once a smoothing curve has been estimated for a univariate time series, the residuals can be calculated using the equation:

$$e_t = Y_t - f(\text{Time}_t, \lambda_1)$$

The function $f(.)$ can also be called the trend because it captures the long-term pattern in the time series. To capture the shorter-term variation, LOESS smoothing can be applied again, but now on the residuals, alias detrended series. Figure 17.2 shows the 11 LOESS curves capturing the shorter-term variation in the residuals. A smaller span of 0.2 was used. All the LOESS curves show a cyclic pattern, which is also captured by the mean value of the smoothing curves.

This repeated application of LOESS smoothing is discussed in Section 3.11 of Cleveland (1993). He suggested iterating the repeated LOESS smoothing process a couple of times because the smoothing curves may compete for the same variation. In the first step of the iteration process, LOESS smoothing is applied to the original time series with a span width of λ_1, followed by LOESS smoothing on the

residuals with a span width of λ_2. In the second step of the iteration, the short-term smoother is subtracted from the data, and the long-term smoother is estimated again. This process is repeated a couple of times, and in practise convergence is obtained quickly. Convergence itself can be assessed by comparing the changes in smoothing curves of two sequential iterations. It is important that λ_1 is relatively larger than λ_2, and that the time series are of approximately the same length.

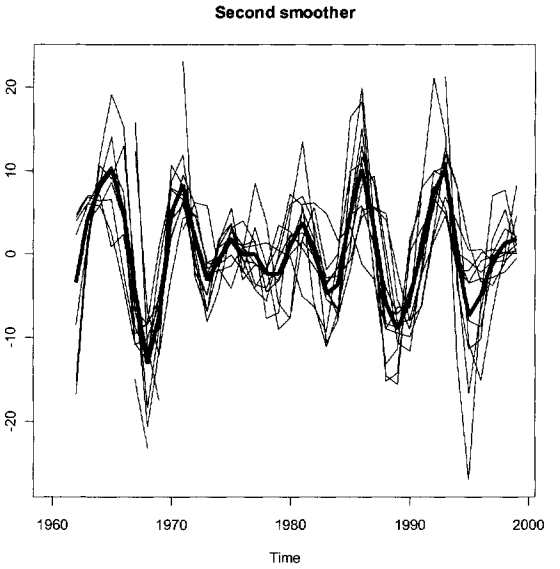

Figure 17.2. Eleven smoothing curves for the Nephrops time series data. These curves represent the short-term variation. The thick line is the mean smoothing curve.

Repeated LOESS smoothing applied on the SST and NAO index

As a second example, repeated LOESS smoothing is applied on the Scottish SST time series and the NAO index. (Chapter 16). Because of the clear patterns in the CPUE *Nephrops* time series and their relative short length, it was relatively easy to find a span width for the first and second smoothers. If the series are longer and have a less clear pattern, then this can be more difficult. Figure 17.3 shows the smoothing curve for the SST time series using different span widths. There are minor differences in the curves with span widths between 0.5 and 0.9. The curve with a span of 0.5 shows more small-scale variation. The question is what we consider as the long-term trend. The curves with the large span widths clearly indicate a rise in SST since the early 1980s. The curves with intermediate span width show three bumps and the curve with span = 0.1 shows cyclic behaviour. We decided to choose $\lambda_1 = 0.6$. The same process was applied on the detrended series, and we chose $\lambda_2 = 0.3$. The resulting long-term and short-term smoothers are presented in Figure 17.4. The long-term smoothers for the SST and

NAO series both show a general decrease followed by an increase, but the NAO trend decreases again in the late 1980s. The shorter term trends (right panel) show similar patterns over time. As a follow-up analysis, the shorter term trends of the SST and NAO can be compared with each other using auto-correlation functions, ARIMAX models, MAFA or dynamic factor analysis.

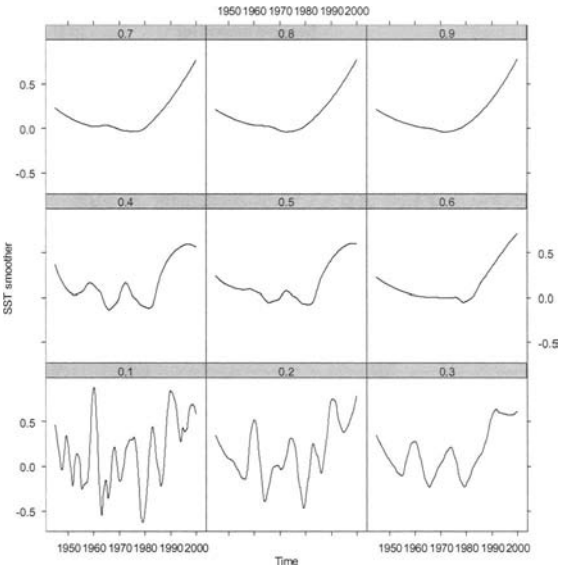

Figure 17.3. Smoothing curves for the SST time series using different span widths. The lower left panel shows the smoothing curve using a span of 0.1, and the upper right of 0.9.

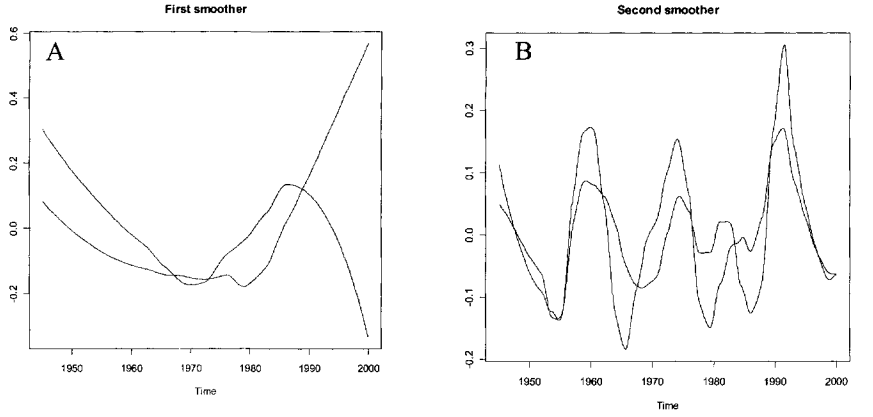

Figure 17.4. Smoothing curves for SST and NAO series obtained by repeated LOESS smoothing. A: Long-term smoothers obtained with $\lambda_1 = 0.6$. B: Short-term smoothers obtained with $\lambda_2 = 0.3$.

Neat et al. (2006) used repeated LOESS smoothing on a group of tagged cod time series in the coastal areas of the Shetland Isles in the northern part of the North Sea. Time series on depth and temperature on approximately 20 cod were available on a 10-minute basis throughout the year. Long-term, intermediate and short-term patterns were extracted using repeated LOESS smoothing for the depth time series and compared with each other. Various fish showed similar depth patterns for some of these components. The cyclic patterns may have been linked to seasonal and diet variation in prey resources and local variation in seabed substrate and bottom-depth.

The repeated LOESS smoothing method, as discussed here, can be extended in various ways. Two relatively easy to implement extensions are (i) automatic selection of the amount of smoothing per component (e.g., using cross-validation; see also Chapter 7), and (ii) applying bootstrapping methods to obtain confidence bands around the estimated components (Davison and Hinkley 1997).

17.2 Identifying the seasonal component

If one is looking for common trends in multiple time series that were measured monthly, then the main part of the variation may be related to seasonal fluctuation. It may be an option to remove the seasonal pattern in each time series and focus on the remaining information instead. Suppose we have a univariate time series Y_t, where measurements were taken monthly (or quarterly or indeed anything from which we know the length of the period). Y_t can be modelled as

$$Y_t = \text{Trend}_t + \text{Seasonal}_t + \text{remainder}_t \qquad (17.1)$$

Hence, the univariate time series is decomposed into a trend, a seasonal component and residual information. The method is described in Cleveland (1993) and is sketched here. Repeated LOESS smoothing (Section 17.2) is used to estimate these components. Suppose that the monthly time series Y_t was measured from January 1980 to December 2000. Instead of storing it as one long vector of length 252 (= 12 months × 21 years), we can also present it as in Table 17.1. To identify the three components in equation (17.1), a two-step algorithm is applied. In the first step of the algorithm, the mean value per month is calculated. This gives a January average, a February average, etc. Once the 12 monthly averages are estimated, they are concatenated to form a univariate time series of length 252. In the second step of the algorithm, the seasonal time series is subtracted from the original time series Y_t. To identify the 'Trend' a LOESS smoother with a large span size is applied on the 'deseasonalised' data. This gives the long-term trend. However, there is one problem, the seasonal component may contain part of the long-term trend signal. We simulated a small dataset to visualise this; see Figure 17.5. The simulated data contains a strong linear trend and a seasonal component. For these data, the algorithm will first calculate a January average, then a February average, etc. However, the January data during the first few years have much lower values than in later years. And therefore, the mean value of January will contain

some information of the upward trend and the same holds for the other months! Starting the other way around, estimating first a trend and then a seasonal component will give a general upward trend with some seasonal information in it. The solution to this problem is as in repeated LOESS smoothing. The algorithm (i) estimates the seasonal component, (ii) subtracts the seasonal component from the data, (iii) estimates the long-term pattern, (iv) subtracts the long-term pattern from the data, and (v) estimates a seasonal component. The process of estimating 12 monthly averages and a long-term trend is iterated a couple of times until convergence is reached and provides the terms Trend$_t$, Seasonalt and Remainder$_t$. We can either decide to apply subsequent analyses on only the trend or on the Trend+Remainder component.

Table 17.1. Monthly time series measured in 21 year.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1980	...	...	...	...	...	...	...	...	...	...	...	...
1981	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...
1999	...	...	...	...	...	...	...	...	...	...	...	...
2000	...	...	...	...	...	...	...	...	...	...	...	...

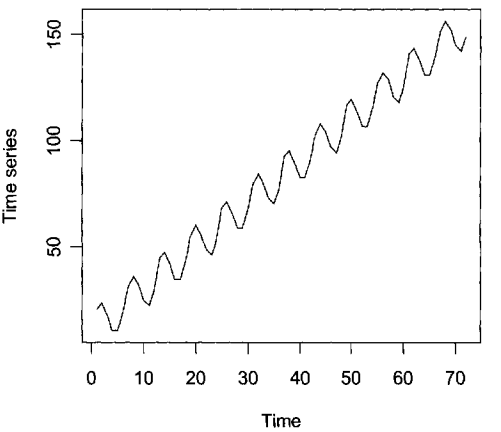

Figure 17.5. Simulated monthly data. The response variable was obtained by $Y = 10 + 2X + 10\sin(X)$, where X takes values between 1 and 72 (six years).

The problem with this approach is that the seasonal components for each month are always the same. This is fine if the seasonal pattern does not change over time. However, it is rather easy to extend the above algorithm to allow for changes in

seasonal patterns. Instead of calculating a mean value for all January data, LOESS smoothing can be applied on the January time series. The same can be done for the other 11 monthly time series. As a result, the seasonal components are allowed to change slowly over time. For multiple time series, this process can be applied on each univariate time series and corresponding components can be plotted in the same graph. It is also possible to predict missing values.

Other methods to remove (or better: identify) the seasonal component can be found in Makridakis et al. (1998). Instead of the decomposition in equation (17.1), it is also possible to use a decomposition of the form

$$Y_t = \text{Trend}_t \times \text{Seasonal}_t \times \text{Remainder}_t \tag{17.2}$$

This decomposition is useful if the variation in the seasonal component varies with the trend. The trend and seasonal components are multiplied allowing for larger seasonal fluctuation in Y_t if the trend is high and small seasonal variation if the trend is close to zero. To estimate the individual components, a similar algorithm as for the repeated LOESS smoothing is used except that instead of subtracting a term from Y_t, we now divide by it: $Y_t/(\text{Seasonal}_t \times \text{Remainder}_t) = \text{Trend}_t$. This gives immediately a problem if one tries to apply the decomposition in equation (17.2) on for example monthly zooplankton data. During most months measured values tend to be low, if not zero (which causes the problem as you cannot divide by something that is zero) and the peak is in spring. An interesting solution may be the pseudo-additive decomposition (Baxter 1994):

$$\begin{aligned} Y_t &= \text{Trend}_t \times (\text{Seasonal}_t + \text{Remainder}_t - 1) \\ &= \text{Trend}_t \times \text{Remainder}_t + \text{Trend}_t\,(\text{Seasonal}_t - 1) \end{aligned} \tag{17.3}$$

Further details and a justification of this approach can be found in Findley et al. (1997). There is actually a whole range of methods available (enough to write a book about) that estimate the trend, seasonal and remainder components, and a few names we want to mention are the X-11, X-11 ARIMA and X-12 ARIMA. They are all variations of each other, and a more detailed description can be found in Findley et al. (1997), Makridakis et al. (1998), among many others.

An example

The seasonal decomposition is illustrated for the North American SST data. Figure 17.6 shows a lattice plot containing auto-correlation functions for all SST time series and the NAO index. The shape of the auto-correlation functions indicates strong seasonal patterns for all SST time series. Figure 17.7 and Figure 17.8 show the estimated trend and seasonal components for the time series at the coordinates 22–24. The difference between the two figures is that in the first one, the mean value per month is used and in the second graph the LOESS smoother is applied on monthly time series to allow for changes over time in the seasonal components. The seasonal component in Figure 17.8 had a span width of 11 months. Figure 17.9 shows the trends obtained by this method for all time series. These

trends can be further analysed by the multivariate time series techniques to estimate common trends, which are discussed later in this chapter.

All approaches discussed so far in this section are based on the assumption that the signal measured in January in year 1 is the same signal as in year 2, 3, etc. Hence, extracting all the January data and obtaining a mean value or smoother makes sense. But what if the data are monthly zooplankton values, and the big peak is in April in year 1, March in year 2 and May in year 3? The answer is simple: The method falls down. Analysing anomalies obtained from fitting a moving average may be an option.

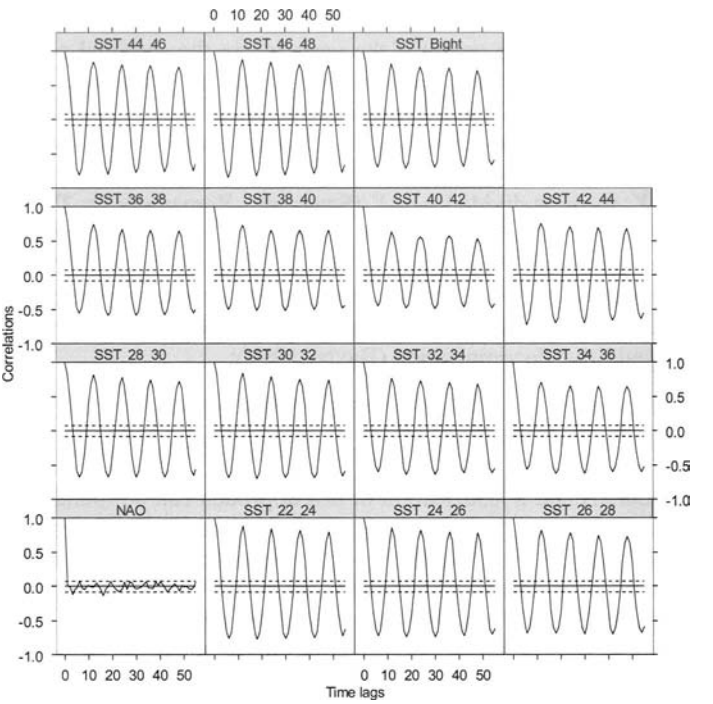

Figure 17.6. Auto-correlation functions for the North American SST. The x-axis shows time lags (months) and the y-axis the auto-correlations.

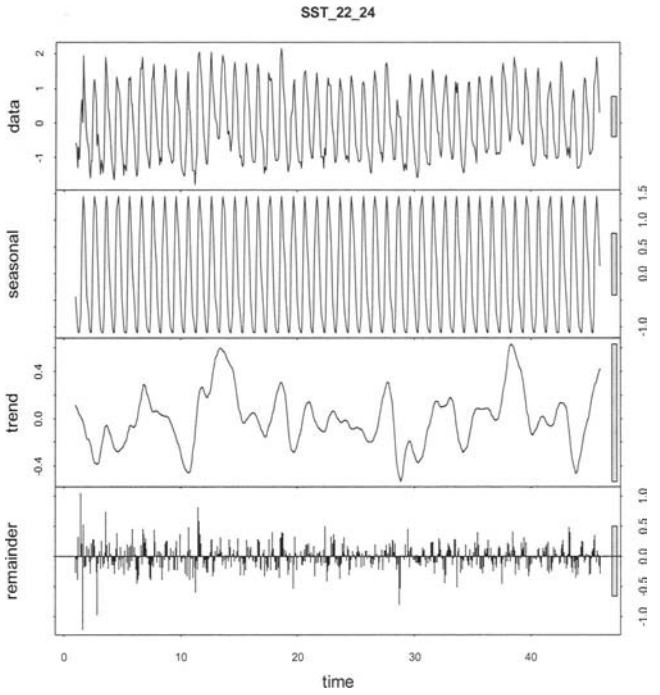

Figure 17.7. Decomposition of a SST time series into seasonal, trend and remainder components. The seasonal component was obtained by taking the average per month. The original time series were normalised.

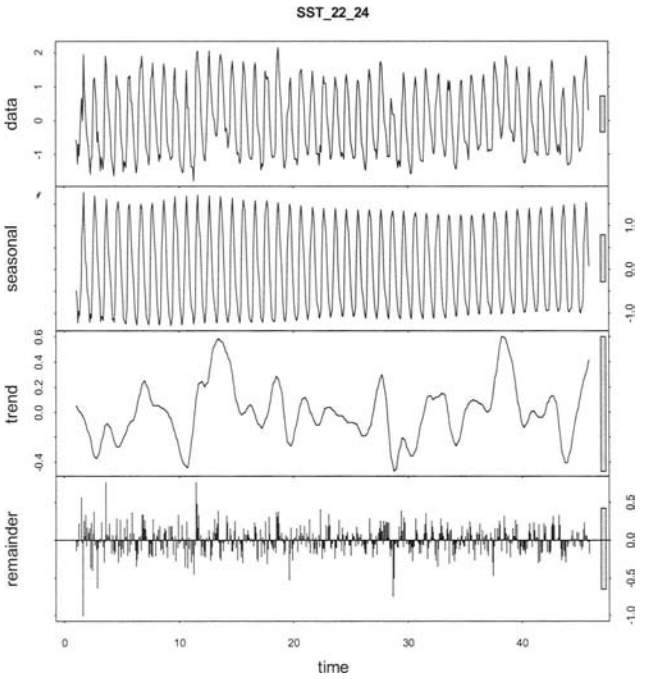

Figure 17.8. Decomposition of an SST time series into seasonal, trend and remainder components. The seasonal component was estimated by a LOESS smoother using a span width of 11 points. The time series were normalised.

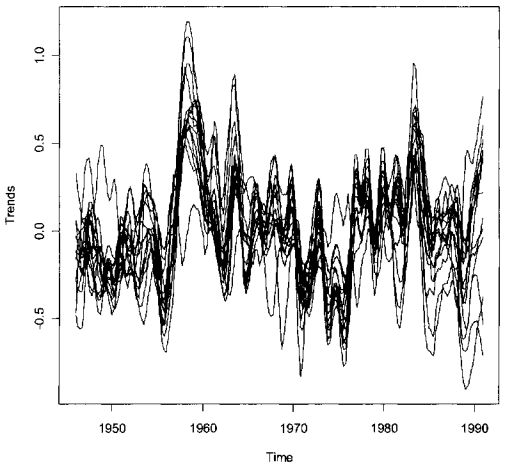

Figure 17.9. Trends for all time series. The seasonal components had a span width of 11. The original time series were normalised.

17.3 Common trends: MAFA

If the data contains a seasonal pattern, methods discussed in the previous section can be applied to remove it. If this is not done, methods to estimate common patterns will mainly pick up the seasonal patterns. We now discuss the first of two more formal methods (compared to repeated LOESS smoothing) to estimate common trends in a multivariate time series dataset.

What does it do?

MAFA (Solow 1994) stands for min/max auto-correlation factor analysis. It can be described in various ways, e.g., a type of principal component analysis for time series, a method for extracting trends from multiple time series, a method for estimating index functions from time series or a smoothing method. Let us recall the underlying formula for principal component analysis (Chapter 12):

$$Z_I = c_{I1} Y_{i1} + c_{I2} Y_{i2} + \ldots + c_{IN} Y_{iN} \qquad (17.1)$$

Y_{i1}, Y_{i2}, .., Y_{iN} contain the values of the N variables at site i and Z_I is the principal component. The factors c_{ij} are calculated in such a way that the variance of Z_1 is maximal. In discriminant analysis we have a similar underlying formula, but it uses a different optimisation criteria namely that observations from the same group were as close to each other as possible and maximum separation of group averages. In MAFA we use a different optimisation criteria namely Z_I should have maximum auto-correlation with time lag 1. Hence, in MAFA, the first axis has the highest auto-correlation with lag 1. The second axis has the second highest auto-correlation with time lag 1. The underlying idea is that a trend is associated with high auto-correlation with time lag 1. Therefore, the first MAFA axis represents the trend, or the main underlying pattern in the data. This axis can also be seen as an index function or smoothing curve that summarises the original N time series in the best possible way. The second MAFA is the second most important trend. Just as in PCA, the axes (or trends) are uncorrelated and estimating a second or third trend does not change the shape of the earlier estimated trends. One of the main differences is that we do not get eigenvalues that tell us how important is a trend. Instead, Solow (1994) presented a permutation test that determines whether the auto-correlation of the axes is significantly different from zero.

An example

We will use a zoobenthic dataset from an intertidal in the Dutch part of the Wadden Sea, namely the Balgzand data (Beukema 1974, 1979, 1992; Zuur et al. 2003a). The data used here consist of 15 zoobenthic species. The original data were measured at 15 locations on the Balgzand but for illustration purposes, total abundances over all sites were taken. The species used in this example are given in Table 17.2, and Figure 17.10 shows a plot of all 15 standardised time series.
Table 17.2. List of zoobenthic species used in the MAFA example.

No	Species	No	Species
1	*Arenicola marina*	9	*Mya arenaria*
2	*Cerastoderma edule*	10	*Mytilus edulis*
3	*Corophium spec.*	11	*Nephtys hombergii*
4	*Eteone longa*	12	*Nereis spec.*
5	*Heteromastus filiformis*	13	*Phyllodoce sp.*
6	*Hydrobia ulvae*	14	*Scoloplos armiger*
7	*Lanice conchilega*	15	*Tellina tenuis*
8	*Macoma balthica*		

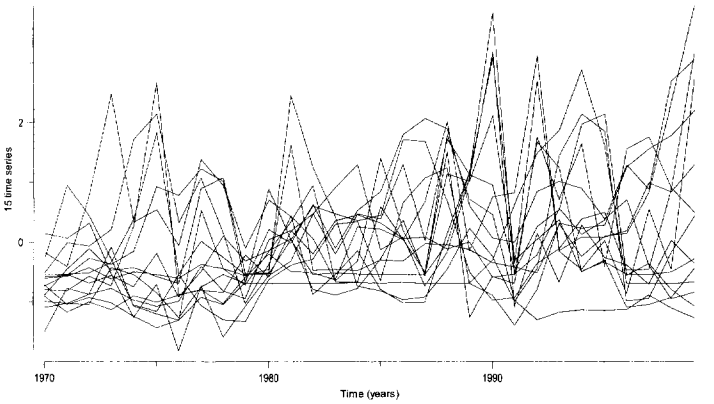

Figure 17.10. Time series plot of all 15 standardised zoobenthic species. The *x*-axis shows the year and the *y*-axis the value of the standardised series.

MAFA was applied on the 15 time series, and Figure 17.11 shows the first two MAFA axes. The first axis shows an increase between 1973–1982 and 1992–2000. Note that this is the main trend underlying the time series. The second axis shows an increase from 1970 until 1986 and a decrease there after (except for the period 1994–1995).

Just as in PCA, loadings are estimated. These can be used to infer which species are related to a particular MAFA axis. Another option, used here, is to calculate the cross-correlation between the MAFA axes and each of the original species time series. We called these canonical correlations (Figure 17.12). Results indicate that the first MAFA axis is important for *A. marine*, *H. ulvae*, *C. edule*, *E. directus*, *M. balthica*, *H. filiformis*, *M. edulis*, *P.* species, *M. arenaria* and *S. armiger*. Hence, all these species are characterised by a general increase (or decrease if the correlation was negative) in abundance. In Figure 17.13, the original time series of some of these species are highlighted, and one can (vaguely) see a general increase. A similar graph can be made for some of the species related to the second MAFA axis (not shown here).

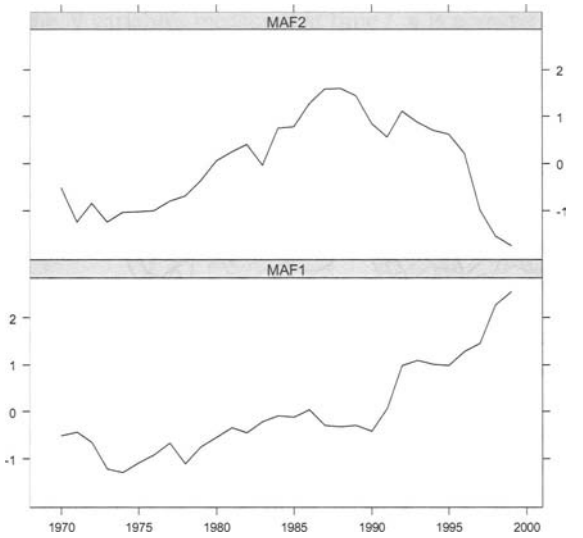

Figure 17.11. First and second MAFA trends. The *x*-axis shows time in years and the *y*-axis the value of the MAFA axis (which is unitless).

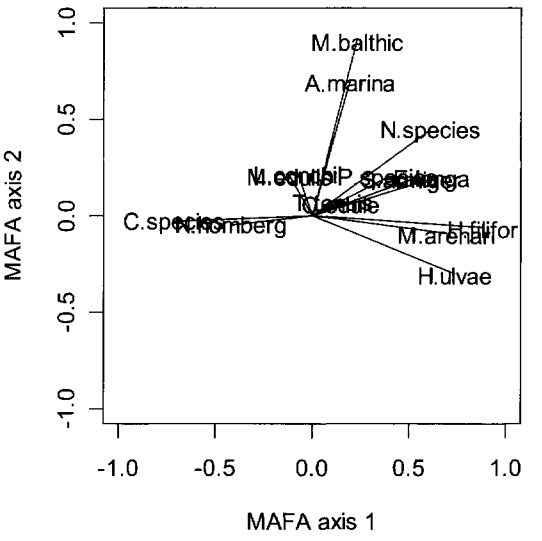

Figure 17.12. Canonical correlations for MAFA axis 1 and 2.

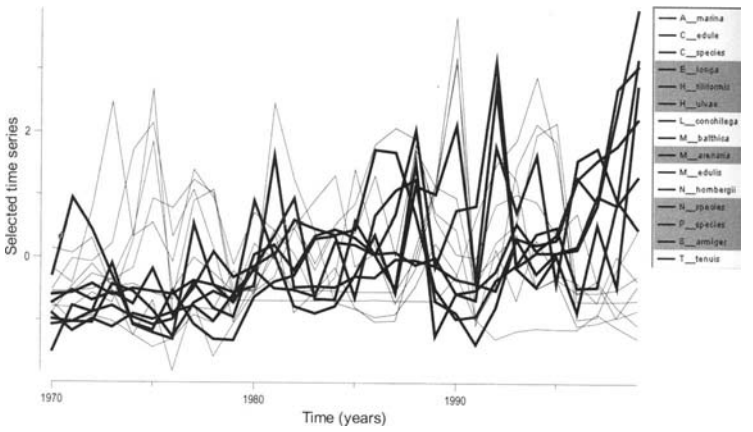

Figure 17.13. Highlighted time series are important for the first MAFA axis. The x-axis shows time in years and the y-axis the value of the standardized series (which are unitless).

If explanatory variables are available, one can estimate the correlations between the MAFA axes and all explanatory variables. For this dataset water temperature was measured also. The correlation between the temperature and the first MAFA axis is 0.095, and it is not significant at the 5% level. The correlation between temperature and the second MAFA axis is –0.062, and for the third axis it is –0.206. Summarising, temperature is not significantly related to any of the first three MAFA axes.

Solow (1994) described a randomisation process to obtain p-values for the MAFA axes. These can be used to decide how many axes to present. Results for the zoobenthic dataset are as follows:

Axis	Auto-correlation lag 1	p-value
1	0.953	<0.001
2	0.906	<0.001
3	0.879	0.006
4	0.727	0.047

These results indicate that the first three MAFA are important.

How does it work?

The reader not interested in the underlying mathematical details may skip this part. The mathematics underlying MAFA are described in two technical reports (Shapiro and Switzer 1989; Switzer and Green 1984) and Solow (1994). Here, a short sketch of the mathematics is presented. The underlying formula is similar to that in PCA:

$$\mathbf{z}_t = \mathbf{a}'\mathbf{y}_t$$

where $\mathbf{y}_t$ contains the N variables measured at time t, $\mathbf{a}$ is a vector of dimension N-by-1 containing the loadings, and $\mathbf{z}_t$ represents the first MAFA axis. In PCA, the loadings in $\mathbf{a}$ are chosen such that $\mathbf{z}_t$ has maximum variance. In MAFA, we chose them such that $\mathbf{z}_t$ has maximum auto-correlation with lag 1. So, how does it do this? It takes a couple of pages of formulae and a few hours, but at the end of the day it is plain high school algebra to show that the following relationship holds:

$$\frac{\mathbf{a}'\mathbf{V}\mathbf{a}}{\mathbf{a}'\mathbf{C}\mathbf{a}} = 2(1 - r_1)$$

$\mathbf{C}$ is the covariance matrix of the original time series $\mathbf{Y}_t$, $\mathbf{V}$ is the covariance matrix of the first-order difference $\mathbf{Y}_t' = \mathbf{Y}_t - \mathbf{Y}_{t-1}$, and r_1 is the auto-correlation coefficient of the first MAFA axis $\mathbf{z}_t$ at time lag 1. The aim is now to maximise r_1 as a function of the loadings $\mathbf{a}$. Just as in PCA (Chapter 12), we can take derivatives, set them to zero, solve the equations, etc. Using first year university algebra it can be shown that the solution is based on an eigenvalue equation of $\mathbf{C}^{-1}\mathbf{V}$. In fact, $\mathbf{a}$ is proportional to the eigenvector corresponding to the smallest eigenvalue of $\mathbf{C}^{-1}\mathbf{V}$. Just as in PCA, we have a certain flexibility in making the solution unique. Recall that in PCA the sum of squared loadings was set to 1. In MAFA, they are made unique by setting (i) the variance of $\mathbf{z}_1$ to 1, and (ii) the weights on $\mathbf{Y}$ to be positive. Just as in PCA we can derive a second MAFA axis. This is the second smoothest curve, and it is uncorrelated with the first MAFA axis. In fact, we can rewrite the MAFA formula as $\mathbf{Z}_t = \mathbf{A}\mathbf{y}_t$, where $\mathbf{Z}_t$ contains all MAFA axes and $\mathbf{A}$ the corresponding loadings.

All we need is an eigenvalue equation of the matrix $\mathbf{C}^{-1}\mathbf{V}$, but this is a bit tricky, as the matrix is not symmetric. For this reason, Shapiro and Switzer (1989) used an indirect method for extracting the eigenvectors of $\mathbf{C}^{-1}\mathbf{V}$. It involves PCA on centred data $\mathbf{Y}$, followed by a first-differencing on the principal components, and a second PCA on these differenced components to give the matrix $\mathbf{A}$. As a result, the MAFA axes are mutually uncorrelated with unit variance, and the axes have decreasing auto-correlation with time lag 1. A requirement for MAFA is that there are more time points than variables.

17.4 Common trends: Dynamic factor analysis

Dynamic factor analysis (DFA) is a method to estimate common trends, effects of explanatory variables and interactions in a multivariate time series dataset. DFA has been applied in many different fields, and all have their different flavours and modifications. Zuur et al. (2003a) presented a detailed mathematical derivation in a pilot paper with a zoobenthic example. This paper was followed by two applied papers (Zuur et al. 2003b; Zuur and Pierce 2004) in which fisheries data were used. Erzini (2005) and Erzini et al. (2005) applied both DFA and MAFA as complimentary methods to examine fisheries data. Other fisheries applications can be

found in Chen et al. (2006), among others. Hydrological applications can be found in Muñoz-Carpena et al. (2005) and Ritter and Muñoz-Carpena (2006). The underlying DFA method in all these papers was discussed in Harvey (1989), Shumway and Stoffer (1982) and Shumway and Stoffer (2000). Mendelssohn and Schwing (1997, 2002) used a slightly modified version of DFA to estimate common trends in large oceanographical time series datasets albeit their approach cannot easily deal with explanatory variables. DFA has also been used extensively in psychological related fields (Molenaar 1985; Molenaar et al. 1992, among others).

What does it do?

In Section 17.1, we used 11 Icelandic Nephrops CPUE annual time series. Let the vector $\mathbf{Y}_t=(Y_1,..,Y_{11})'$ contain the values at year t. The symbol ' is mathematical notation for the transpose, hence $\mathbf{Y}_t$ is of dimension 11-by-1. The main aim of DFA is to estimate underlying common trends. The simplest DFA model contains only one common trend and is given by

$$\mathbf{Y}_t = \mathbf{A}z_t + \boldsymbol{\varepsilon}_t \tag{17.2}$$

where $\mathbf{A}$ is a vector of dimension 11-by-1 with *unknown* loadings, z_t is the trend and $\boldsymbol{\varepsilon}_t$ is normally distributed noise (of dimension 11-by-1) with expectation $\mathbf{0}$ and variance $\sigma^2\mathbf{V}$ and where $\mathbf{V}$ is a positive definite diagonal matrix. The unknown parameters are the multiplication factors $\mathbf{A}$, the trend z_t and the variances. Later in this section we show that a special construction for z_t is used to ensure that it captures the long-term pattern. It is perhaps easier to present this model in full mathematical notation:

$$\begin{pmatrix} Y_{1,t} \\ Y_{2,t} \\ \vdots \\ Y_{11,t} \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_{11} \end{pmatrix} z_t + \begin{pmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \\ \vdots \\ \varepsilon_{11,t} \end{pmatrix}$$

The model with one common trend assumes that all the eleven time series follow the same pattern, namely that of z_t. To obtain the fitted value for each time series, we multiply the trend z_t by a loading. If the loading is relatively large and positive, we know that the corresponding time series follows the pattern of the trend. If the loading is close to zero, we know it does not follow this pattern. A loading that is relatively large and negative indicates that the time series follows the opposite pattern of the trend. These statements assume that the spread in the 11 time series is the same. Otherwise, it is more difficult to compare the factor loadings with each other! One way to ensure this is normalisation of the time series prior to the analysis. It is also an option to include an intercept:

$$\mathbf{Y}_t = \mathbf{c} + \mathbf{A}z_t + \boldsymbol{\varepsilon}_t \tag{17.3}$$

The intercept $\mathbf{c}$ is a vector of dimension 11-by-1 containing an intercept for each time series. Hence, we are saying that all time series follow the underlying

pattern (the trend), the factor loadings are used to determine the strength of this relationship, and each time series is shifted up or down by an intercept. If the time series are normalised or centred, the intercepts will all be close to zero. The model in equation (17.3) can be extended in various ways. The easiest option is to add explanatory variables, for example the NAO index:

$$\mathbf{Y}_t = \mathbf{c} + \mathbf{A}z_t + \boldsymbol{\beta}\mathrm{NAO}_t + \boldsymbol{\varepsilon}_t \tag{17.4}$$

The term '$\boldsymbol{\beta}\mathrm{NAO}_t$' models the effect of the NAO index on the 11 time series. It is always useful to do a dimension check after a model specification. The NAO index is a global variable, and in this case, we have only one value per time unit (year). So, the term NAO_t is of dimension 1-by-1. The regression parameter $\boldsymbol{\beta}$ is of dimension 11-by-1; the NAO is allowed to have a different effect on each CPUE time series. Just as in linear regression, standard errors and t-values are obtained for the estimated values of $\boldsymbol{\beta}$ and these can be used to assess whether it is significantly different from 0. Another extension is modifying the error structure. In equation (17.4) we assumed that

$$\boldsymbol{\varepsilon}_t \sim N(\mathbf{0}, \sigma^2\mathbf{V}), \qquad \text{and } \mathbf{V} \text{ is diagonal and positive definite} \tag{17.5}$$

So each error component has a different variance. In the same way as the linear regression model was extended to generalised least squares (GLS) in Chapter 16, the error components of different time series of the DFA model can be allowed to covary. This can be done by using an unstructured (positive definite) error covariance matrix $\mathbf{V}$ of the form

$$\mathbf{V} = \begin{pmatrix} 1 & v_{1,2} & v_{1,3} & \cdots & \cdots & v_{1,N-1} \\ v_{2,1} & 1 & v_{2,3} & \cdots & \cdots & v_{2,N-2} \\ v_{3,1} & v_{3,2} & 1 & & & v_{3,N-3} \\ \vdots & v_{4,2} & v_{4,3} & \ddots & & \vdots \\ \vdots & \vdots & \vdots & & \ddots & v_{N-1,N} \\ v_{N-1,1} & v_{N-2,2} & v_{N-3,3} & \cdots & \cdots & 1 \end{pmatrix}$$

Let us just recap what we have so far. We have 11 CPUE time series, and the NAO may have a different effect on each of them. This is done in a regression style ($\mathbf{c} + \boldsymbol{\beta}\mathrm{NAO}_t$). A trend is used to capture the remaining common long-term variation. Factor loadings determine how important this trend is for each of the time series. On top of this, we allow for residual interactions between the time series via a non-diagonal error covariance matrix. Obviously, we have to estimate a lot of parameters, especially if a non-diagonal error covariance matrix $\mathbf{V}$ is used, and the technical aspects of this are discussed in Zuur et al. (2003a).

Hopefully, the explanatory variables explain most variation in the data as it makes interpretation of the model much easier. If this is not the case, the researcher has the task of explaining the meaning of the common trend. If the factor loadings show a clear grouping, then this can be easy as it indicates a grouping in the time series.

So far, we have discussed a DFA model with one common trend. But what happens if groups of time series are behaving differently over time? If only two common patterns behave oppositely, we can still model it with a model containing only one common trend; the model will use positive and negative loadings. But this does not work if the patterns are not opposite to each other, or if there are more than two common patterns. The solution is to extend the DFA model in equation (17.2) with an extra common trend:

$$\mathbf{Y}_t = \mathbf{A}\mathbf{Z}_t + \boldsymbol{\varepsilon}_t \tag{17.6}$$

The model specification looks the same as before, but the difference is the dimension of $\mathbf{A}$ and $\mathbf{Z}_t$. Suppose we have two common trends. The factor loading matrix $\mathbf{A}$ is now of dimension 11-by-2 and $\mathbf{Z}_t$ is of dimension 2-by-1. Writing out the full formula may clarify the model:

$$\begin{pmatrix} Y_{1,t} \\ Y_{2,t} \\ \vdots \\ Y_{11,t} \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{2,1} \\ a_{1,2} & a_{2,2} \\ \vdots & \\ a_{1,11} & a_{2,11} \end{pmatrix} \begin{pmatrix} z_{1,t} \\ z_{2,t} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \\ \vdots \\ \varepsilon_{11,t} \end{pmatrix}$$

There are now two common trends, and each one has factor loadings associated with it. Hence, each time series is modelled as the sum of (i) a factor loading multiplied with the first common trend plus another factor loading multiplied with the second common trend plus noise. It is now interesting to compare the factor loadings with each other. The signs and magnitudes will tell us which CPUE time series are driven by the first, the second, and by both trends. The model with two common trends can easily be extended to include explanatory variables, an intercept and a non-diagonal error covariance matrix $\mathbf{V}$:

$$\mathbf{Y}_t = \mathbf{c} + \mathbf{A}\mathbf{Z}_t + \boldsymbol{\beta}\mathrm{NAO}_t + \boldsymbol{\varepsilon}_t \qquad \boldsymbol{\varepsilon}_t \sim N(\mathbf{0}, \sigma^2\mathbf{V}) \tag{17.7}$$

One can add even more trends, but just as in PCA the interpretation of three or more axes (trends) becomes difficult. Further explanatory variables can also be used. Zuur et al. (2003a) used the AIC to find the optimal model in terms of (i) the number of common trends, (ii) which explanatory variables to select and (iii) what type of error covariance matrix $\mathbf{V}$ to use. Another aspect one should keep in mind is that in MAFA and PCA, the first axis does not change if two or three axes are calculated. In DFA, all trends are estimated simultaneously using the maximum likelihood method. This means that if two trends are used, the first trend may not be the same as the trend in a model with only one common trend. To get some idea of the importance of the trends, we advise the following: (i) Apply the model with one common trend, (ii) apply a model with two common trends, and (iii) compare trends of both models. In most cases, one of the trends in the model with two common trends will look similar (although not identical) to the trend obtained in the first model. This is the dominant pattern.

An example

To illustrate dynamic factor analysis, we use the CPUE Nephrops data. A time series plot was given in Figure 16.3 and showed that various time series followed a similar pattern over time. The DFA model was applied on the standardised CPUE series, and various combinations of number of trends, explanatory variable and error covariance matrix were investigated. Each time we obtained the AIC, and these are given in Table 17.3. The model containing three common trends, no explanatory variables and a diagonal error covariance matrix was the most optimal, as judged by the AIC. The problem with this model is that the third common trend was mainly related to one time series, and its contribution to the model was small (as could be inferred from the small factor loading). This is typically an indication that too many trends are used. The second most optimal model contained three common trends and the NAO index. It had the same problem with the third trend and *t*-values for the NAO were either not significant or only slightly larger than two in the absolute sense, which indicates only borderline significance. For these reasons, we decided to present results of the model with two common trends, no explanatory variables and a diagonal error covariance matrix.

Table 17.3.Values of AIC using a dynamic factor model with 1 to 4 common trends. The model in bold typeface is selected.

V Diagonal			**V** Non-diagonal		
Number of trends	Explanatory variables	AIC	Number of trends	Explanatory variables	AIC
1	-	850.951	1	-	778.537
2	-	**755.427**	2	-	769.933
3	-	749.387	3	-	764.764
4	-	757.197	4	-	773.499
1	NAO	858.911	1	NAO	778.139
2	NAO	757.998	2	NAO	770.916
3	NAO	750.483	3	NAO	764.483
4	NAO	757.458	4	NAO	770.519

The estimated two common trends are presented in Figure 17.14 and corresponding factor loadings in Figure 17.15. The factor loadings indicate that station S4 is driven mainly by the first common trend, stations S8, S9 and S10 are related to the second common trend, S7 and S11 are driven mainly by the second trend but with a contribution from the first, and stations S1, S2, S3, S5, S6 and S7 by the first and also a bit by the second. This grouping corresponds with the location of the stations. Eiríksson (1999) found a similar pattern in these data and argued that the distinction between the stations may be due to differences in temperature and sediment type. To illustrate the differences between the time series of the groups, fitted values for all stations are presented in Figure 17.16. Thick lines in this figure correspond to stations S1, S2, S3, S5, S6 and S7. These stations tend to

have lower CPUE values in most years. Also note the differences between the two groups of time series from 1995 onwards. The fitted curve of station 4 is the curve that had the highest fitted values from 1965 until 1995.

Just as in linear regression, one has to apply a model validation. Useful tools are graphs in which fitted lines and the observed data are plotted (Figure 17.17). They identify the extent to which the model can capture the patterns in the time series. Other tools are residuals plotted versus time (Figure 17.18), residuals versus fitted values and QQ-plots and histograms of residuals. In our experience, these graphs will always show some patterns as we are summarising N time series with only 2 or 3 common trends. The more patterns we see, the more we need to improve the model.

The dynamic factor analysis applied on the 11 Nephrops time series indicated that there are two underlying common trends. The factor loadings indicated that the distinction between the series is probably based on geographical differences. The NAO index was not significantly related to the time series. To understand which biological mechanisms are driving the two common trends, further information is needed (e.g., sediment type).

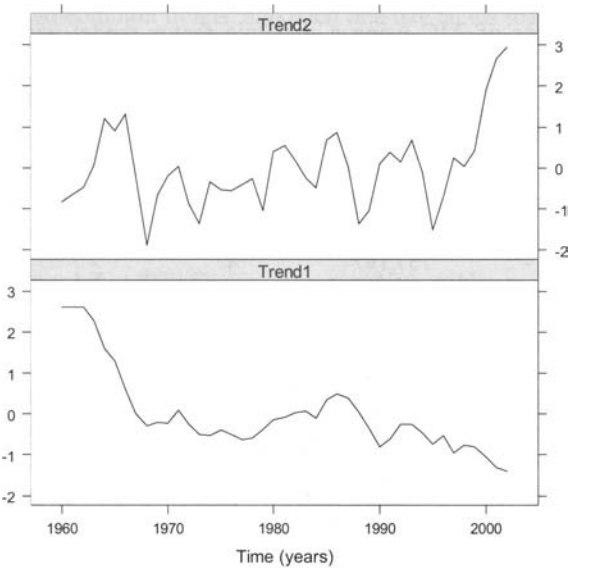

Figure 17.14. Estimated common trends obtained by the DFA model. The x-axis shows time in years and the y-axis the values of the trends (which are unitless).

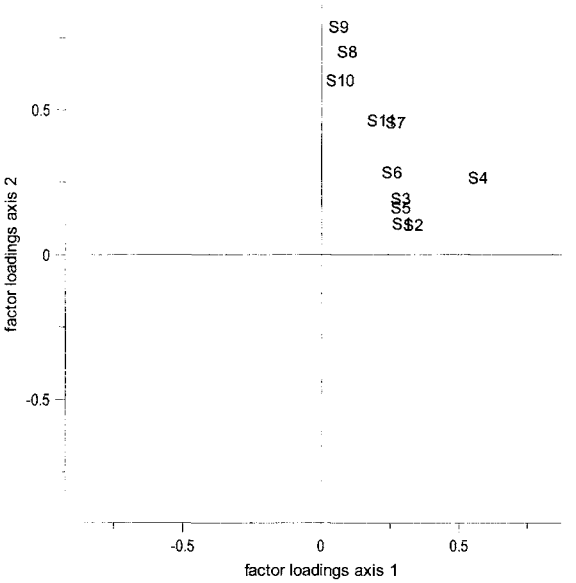

Figure 17.15. Factor loadings corresponding to the first two common trends.

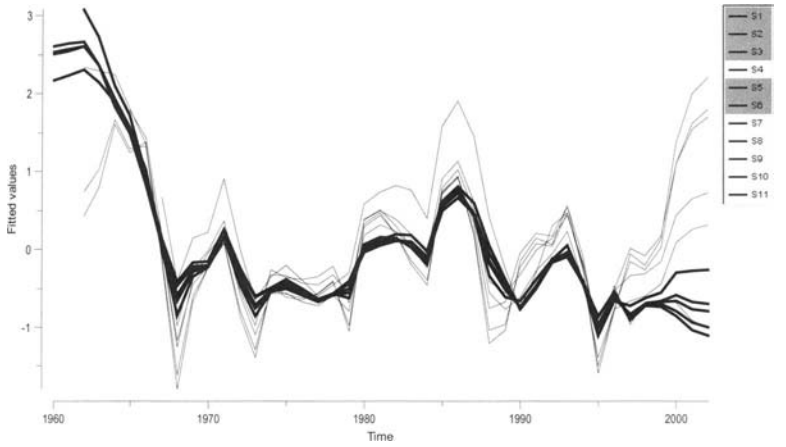

Figure 17.16. Fitted values obtained by the DFA model with two common trends. The thick lines correspond to stations S1, S2, S3, S5, S6 and S7. The x-axis shows time in years and the y-axis the values of the normalised series (which are unitless).

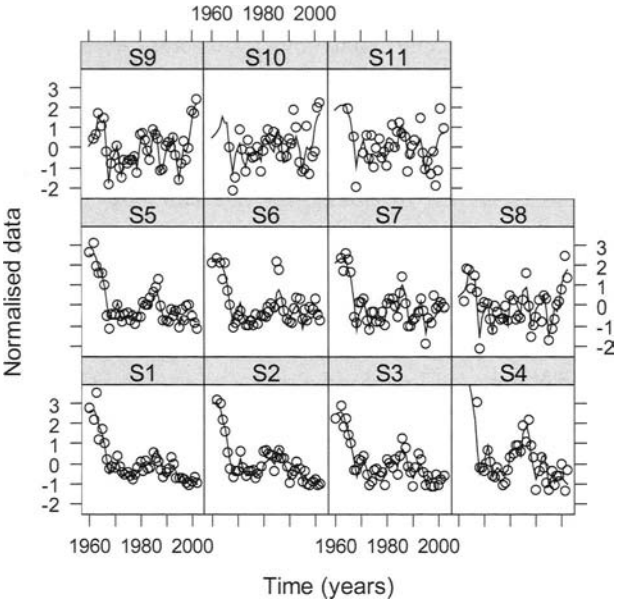

Figure 17.17. Observed data and fitted values obtained by a DFA model with two common trends. The fitted curves follow the patterns of most time series reasonably well.

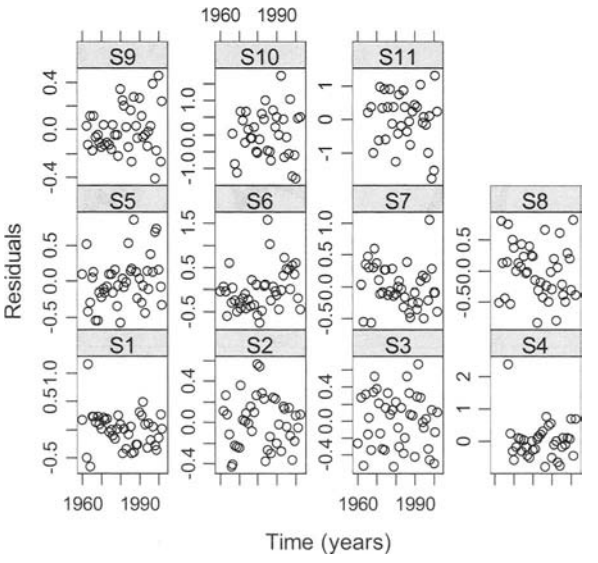

Figure 17.18. Residuals versus time for the DFA model with two common trends.

How does it work?

The reader not interested in the mathematical details underling DFA can skip the next two pages. DFA is based on multivariate structural time series models. For simplicity, we start with univariate structural time series models. The underlying principle in univariate structural time series models is that the time series are modelled in terms of a trend, seasonal component, cycle, explanatory variables and noise. Each of these components is allowed to be stochastic. This means that a trend is not restricted to being a straight line, and the cyclic and seasonal components are not necessarily smooth-looking cosine functions, but their shape can change over time. Obviously, if the time series are measured on an annual basis for a period of less than 50 years, the seasonal component can be dropped. Furthermore, using a cyclic component for such short time series is less suitable. For such data, the following univariate time series model can be used:

$$1 \text{ time series} = \text{trend} + \text{explanatory variables} + \text{noise} \tag{17.8}$$

The 'trend' component in (17.8) is based on a so-called random walk, which is mathematically defined by

$$z_t = z_{t-1} + \eta_t \quad \text{and} \quad \eta_t \sim \mathrm{N}(0,\sigma_\eta^2) \tag{17.9}$$

The trend at time t is given by z_t. This type of model is popular in econometrical fields (Harvey 1989; Durbin and Koopman 2001). They typically have the form of a slow-moving pattern with occasionally a sharp drop or sudden increase. It is also possible to add explanatory variables:

$$\begin{aligned} y_t &= z_t + \boldsymbol{\beta}\mathbf{X}_t + \varepsilon_t \\ z_t &= z_{t-1} + \eta_t \\ \varepsilon_t &\sim N(0,\sigma^2) \quad \eta_t \sim N(0,\sigma_\eta^2) \quad z_0 \sim N(a_0,\sigma_z^2) \end{aligned} \tag{17.10}$$

where the vector $\mathbf{X}_t$ contains the values of the explanatory variables at time t, and $\boldsymbol{\beta}$ the corresponding slopes. The error terms are assumed to be independent of each other. The unknown parameters are the slopes, the three variance components and the value of the trend at $t = 0$. However, the trend z_t is unknown as well! So, how does one estimate all these parameters, variances and the trend? The answer is a combination of (i) the state-space formulation, (ii) the Kalman filter and smoother and (iii) the maximum likelihood. The state-space is a special mathematical formulation that takes the form:

$$\begin{aligned} \mathbf{y}_t &= \mathbf{A}\mathbf{z}_t + \mathbf{B}\boldsymbol{\varepsilon}_t \\ \mathbf{z}_t &= \mathbf{C}\mathbf{z}_{t-1} + \mathbf{D}\boldsymbol{\eta}_t \end{aligned} \tag{17.11}$$

It turns out that a lot of time series models, including ARMAX, structural time series and DFA can be reformulated in this format. It is just a matter of choosing the right format for the matrices **A**, **B**, **C** and **D**. The Kalman filter and smoother is a mathematical procedure that estimates the so-called state vector $\mathbf{z}_t$ and its standard errors for given values of **A**, **B**, **C** and **D** and the variances of the noise com-

ponents. The EM method (Zuur et al. 2003a) is used to estimate all the unknown parameters. For example, let the response variable be the Nephrops CPUE at station 1 and let us include a trend and the NAO as an explanatory variable to give:

$$\begin{aligned} y_t &= z_t + \beta NAO_t + \varepsilon_t \\ z_t &= z_{t-1} + \eta_t \\ \varepsilon_t &\sim N(0,\sigma^2) \quad \eta_t \sim N(0,\sigma_\eta^2) \quad z_0 \sim N(a_0,\sigma_z^2) \end{aligned} \tag{17.12}$$

The first step is to rewrite this model in state-space format:

$$\begin{aligned} y_t &= \begin{pmatrix} 1 & NAO_t \end{pmatrix} \begin{pmatrix} z_t \\ \beta \end{pmatrix} + \varepsilon_t \\ \begin{pmatrix} z_t \\ \beta \end{pmatrix} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z_{t-1} \\ \beta \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \eta_{1t} \\ \eta_{2t} \end{pmatrix} \end{aligned} \tag{17.13}$$

η_{2t} is a new error term but it has no influence. The unknown parameters are the state vector and the variances. For given values of the variances, the Kalman smoother estimates the state vector and its standard errors. The construction of the design matrix **D** is such that β is constant over time. Using maximum likelihood, the unknown variance components can be estimated. The trend z_t is given in Figure 17.19 and $\beta = -0.21$, and its t-value is –3.55, indicating a significant negative effect of the NAO at the 5% level.

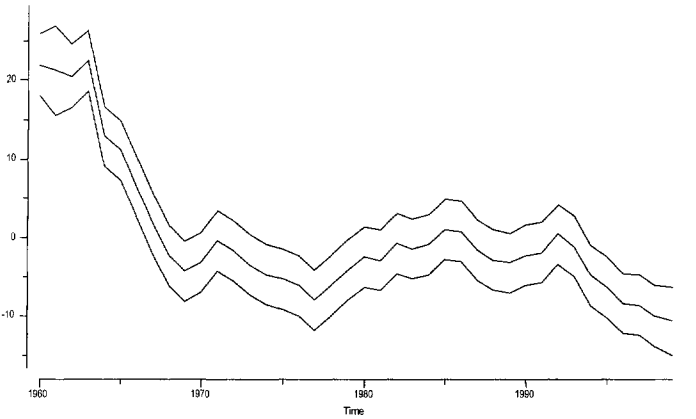

Figure 17.19. Estimated trend obtained by Kalman smoothing and 95% confidence bands for Nephrops CPUE at station 1. The x-axis shows time (in years) and the y-axis the values of the trend.

The framework in equation (17.11) allows for all kinds of exotic models, for example seemingly unrelated time series, dynamic regression models and also the DFA. Let us start with the first. Suppose we have 11 CPUE time series, which we assume are unrelated so that each has its own trend and dependency on the explanatory variables but where the error components are interacting:

$$\begin{aligned} \mathbf{y}_t &= \mathbf{z}_t + \mathbf{B}\mathbf{X}_t + \boldsymbol{\varepsilon}_t \\ \mathbf{z}_t &= \mathbf{z}_{t-1} + \boldsymbol{\eta}_t \\ \boldsymbol{\varepsilon}_t &\sim N(0, \sigma^2 \mathbf{V}) \end{aligned} \tag{17.14}$$

So, we have 11 trends ($\mathbf{z}_t$ is of dimension 11-by-1), and effects of explanatory variables, but the noise components are related to each other using a non-diagonal error covariance matrix **V**. Indeed, we have seen this before within the context of generalised least squares!

Another interesting extension is to allow the regression parameters for the explanatory variables to change over time. For example, the NAO may only have an effect on some years. For a univariate structural time series, this can be modelled as

$$\begin{aligned} y_t &= \begin{pmatrix} 1 & NAO_t \end{pmatrix} \begin{pmatrix} z_t \\ \beta_t \end{pmatrix} + \varepsilon_t \\ \begin{pmatrix} z_t \\ \beta_t \end{pmatrix} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z_{t-1} \\ \beta_{t-1} \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \eta_{1t} \\ \eta_{2t} \end{pmatrix} \end{aligned} \tag{17.15}$$

Note the difference with equation (17.13). In this model the slope β is allowed to change over time. The estimated smoother and the slope are presented in Figure 17.20. Note that the NAO is only significant during the first 8 years!

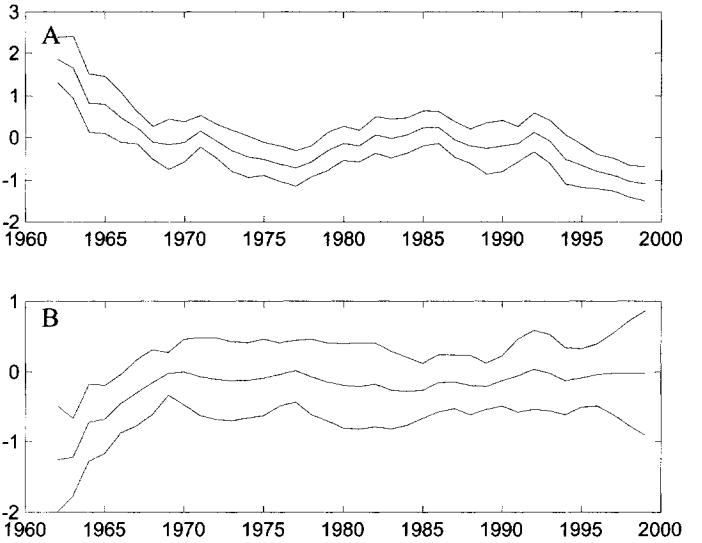

Figure 17.20. Smoothing trend (A) and effect of NAO (B) obtained by dynamic regression.

The model can easily be extended to have multiple response variables. The first part of the state-space formulation becomes:

$$\begin{pmatrix} y_{1,t} \\ y_{2,t} \\ y_{3,t} \\ \vdots \\ y_{11,t} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & & \vdots \\ \vdots & \vdots & & \ddots & 0 \\ 1 & 0 & \vdots & 0 & 1 \end{pmatrix} \begin{pmatrix} z_t \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{11} \end{pmatrix} + \varepsilon_t \tag{17.16}$$

The mathematical formulation of the dynamic factor analysis is actually much easier. Full details can be found in Zuur et al. (2003a), Harvey (1989) and Lütkepohl (1991). For the 11 Nephrops series we used a model with two common trends. The full underlying model is:

$$\begin{pmatrix} Y_{1,t} \\ Y_{2,t} \\ \vdots \\ Y_{11,t} \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{2,1} \\ a_{1,2} & a_{2,2} \\ \vdots & \\ a_{1,11} & a_{2,11} \end{pmatrix} \begin{pmatrix} z_{1,t} \\ z_{2,t} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \\ \vdots \\ \varepsilon_{11,t} \end{pmatrix}$$

$$\begin{pmatrix} z_{1,t} \\ z_{2,t} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z_{1,t-1} \\ z_{2,t-1} \end{pmatrix} + \begin{pmatrix} \eta_{1,t} \\ \eta_2, t \end{pmatrix}$$

The technical aspects of estimating the parameters, of how to deal with missing values, including explanatory variables, etc. are rather complex and are outside the scope of this book. The interested reader is referred to Zuur et al. (2003a) for these details but be prepared for some complex mathematics. There are three case study chapters that make use of DFA.

17.5 Sudden changes: Chronological clustering

MAFA and dynamic factor analysis are techniques, that can be used to estimate trends in multivariate time series. Application of these techniques on biological data assumes that the underlying ecosystem is gradually changing over time. The reason for this is that the mathematical mechanism is based on smoothing techniques, and these are less suitable to capture fast changing patterns. So, for ecosystems that change rapidly from one state into another, common long-term trend estimation is less suitable.

If an ecosystem is under study that changes rapidly from one stage to the next, then identifying years (assuming that year is the time unit) in which the ecosystem is the same, and when it is changing, becomes one of the aims of the analysis.

Standard multivariate methods like principal component analysis, correspondence analysis, etc. produce continuous gradient, and are also less suitable (i) to extract discontinuous gradients and (ii) identify breakpoints.

Cluster analysis

Cluster analysis is a multivariate method that can be used if one knows *a priori* that the observations (e.g., sites or time units) form groups. Cluster analysis can also be applied on time series, leading to groups of years in which the variables are similar. Figure 17.21 shows a so-called dendrogram for the Nephrops time series dataset. Recall that the dataset consists of Nephrops abundances measured at 11 sites south of Iceland since the early 1960s. A dendrogram shows the arrangement of years in clusters, as obtained by the cluster algorithm, and we will explain shortly how it is created. The main problem with clustering applied on time series is to explain groupings of non-sequential years. For example, in Figure 17.21 we have 1964, 1965, 1966, 1985 and 1986 in one group; hence these years are similar, but the ecological reason for this is unclear.

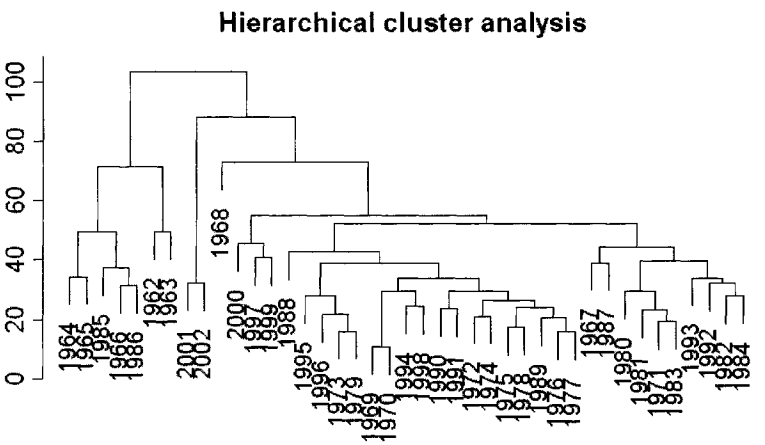

Figure 17.21. Results of hierarchical cluster analysis presented as a dendrogram. The time series were normalised prior to the cluster analysis, and hierarchical clustering using the Euclidean distance function was used with the average linkage. The vertical axis shows at which value of the Euclidean distance groups were fused.

Chronological clustering, as the name already suggests, is specially designed for clustering of time series. The method is fully described in Legendre et al. (1985), Bell and Legendre (1987), and Legendre and Legendre (1998), and a short description is given here. Before we can explain chronological clustering, we need to explain clustering, as this method has not been discussed in this book yet. A short explanation of the basic underlying principle of clustering is given here. A full discussion can be found in, for example, Legendre and Legendre (1998), Krzanowsky (1988), among many other books. The subject deserves more space than the two pages we dedicate to it. Clustering is probably the most misused technique in multivariate analysis. First of all, one should only apply it if there is prior knowledge that the observations (or variables) form groups. Formulated slightly stronger, a justification for why clustering is applied should be given! Possible justifications are that sampling took place in different areas or countries, or in different habitats.

Cluster analysis will always produce a grouping structure. The main problem with clustering methods is that one not only has to choose a measure of association, but also the grouping rule, and the type of clustering method itself. It is possible to produce a large number (>50) of different clustering structures by changing the settings of the software and choose the one that suits the report; it

has a high cheating potential. Basically, if you apply clustering methods, you need to know exactly what all options in the software mean.

Standard cluster analysis methods usually start with a similarity or dissimilarity matrix. Table 17.4 shows an artificial dissimilarity matrix for four sites, and we will use these data to explain the principle of agglomorative clustering. A high value in the dissimilarity matrix means that two sites are not similar. Cluster analysis applied on this matrix forms groups of similar sites. In the first step, it will compare all dissimilarities and it chooses the two sites that are the most similar. These are sites B and C, and they are fused in one group. So, we now have group 1 (formed by sites B and C), site A and site D. The next step is visualised in Figure 17.22. We need to know the dissimilarity between group 1 and A, and also between group 1 and D. Once we know all three dissimilarities, we can choose the smallest value and fuse the corresponding sites/group. This process can then be repeated until all sites/groups are fused into one single group. The problem is now, how do we get the dissimilarity between a group and a site? And a relevant question (for the next stage) is how to calculate the dissimilarity between two groups.

Table 17.4. An artificial dissimilarity matrix for four sites. The numbers in the cells are dissimilarity values among four sites, labeled A to D. The smaller a value is, the more similar are the two sites.

	A	B	C	D
A	–	0.4	0.7	0.9
B		–	0.3	0.5
C			–	0.6
D				–

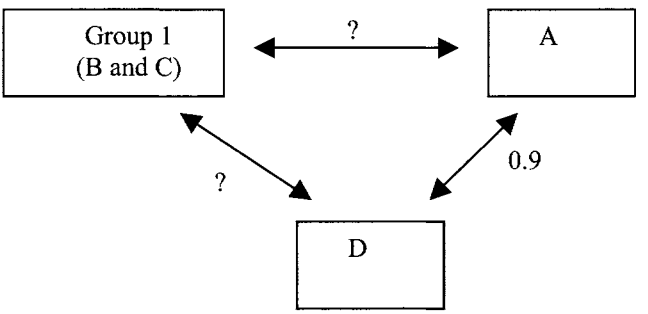

Figure 17.22. Visualisation of the second step in the clustering process. In the first step, sites B and C were fused. In the next step, we need to define the dissimilarity between group 1 and A, and between group 1 and D, and fuse the groups/sites with the smallest value.

There are different ways of quantifying this dissimilarity. The easiest (but not best) way is called single linkage. For group 1 and site A it works as follows. Within group 1, we have sites B and C. The dissimilarity between B and A was 0.4, and between C and A it was 0.7. Single linkage will choose the smallest of these, 0.4. Hence, the question mark between group 1 and A will be replaced by 0.4. The same process will give a dissimilarity of 0.5 between group 1 and D (minimum of 0.5 and 0.6). Therefore, the algorithm will fuse group 1 and A. Instead of taking the minimum value of the dissimilarities between two groups (or one group and a site), we could use the average (average linkage) or the maximum (maximum linkage), among many other options. The choice for linkage will greatly determine the outcome! The average linkage is the default value of most software packages.

The results of the clustering process (as described above) can visually be presented in a dendrogram, and we have already seen an example (Figure 17.21). The first two sites (B and C) were fused at the value 0.3. Site C was fused into this group at the value 0.4. So, we can visualise this in a vertical tree structure in which B and C are group first, then B and C with A, and then B, C, A with D. A vertical axis can be added that shows at which value groups fuse.

Chronological clustering

Chronological clustering works in a similar way as ordinary clustering. In the first step of the algorithm, a dissimilarity matrix between the time units is calculated. For simplicity, let us assume that the time unit is years. The only difference with ordinary clustering is that (i) the fusing rule is slightly different, and (ii) candidate groups (or years) for fusing are restricted to be sequential in time. Table 17.5 shows the same artificial data as in Table 17.4, except that we have replaced the site labels A–D by the years 2002–2003. Due to the restriction of sequential years, we can only fuse 2000–2001, 2001–2002 or 2002–2003. In this case, we would fuse 2001 and 2002 in one group as it has the lowest dissimilarities. In the next step, we only consider fusing 2000 with group 1 (=2001 and 2002) and group 1 with 2003. Fusing 2000 with 2003 is not an option as these years are not sequential. The fusing process itself is based on a permutation test similar to the Mantel test (Chapter 10), and further details can be found in Legendre and Legendre (1998). An example is given next.

Table 17.5. An artificial dissimilarity matrix for four years.

	2000	2001	2002	2003
2000	–	0.4	0.7	0.9
2001		–	0.3	0.5
2002			–	0.6
2003				–

Example chronological clustering

Explaining chronological clustering is best done with an example. Hare and Mantua (2000) used 100 biological and physical time series from the North Pacific Ocean. These were variables like atmospheric indices (teleconnection index, North Pacific index, Southern oscillation index, Arctic oscillation, etc.), terrestrial indices, oceanic indices and biological indices (zooplankton biomass, Coastal Washington oyster condition index, etc.). Most of the time series were available annually from 1965 onwards. Hare and Mantua (2000) concluded that there were two major shifts in these time series, namely in 1977 and 1989.

Here, we show that chronological clustering identifies the same regimes. Chronological clustering requires two parameters to be set, namely the connectedness and the fusion level alpha, which is a clustering sensitivity parameter. Legendre et al. (1985) suggested to use different values of alpha and to keep the connectedness constant. The effect of alpha is as follows. Small values (0.05, 0.01, 0.1) provide a birds-eye overview, and the most important breaks in the time series are visualised. Higher values of alpha (0.2, 0.3, 0.4) give more detailed information and therefore show more breaks in the time series.

The results for chronological clustering are given in Figure 17.23. Vertical lines identify the breaks, and the numbers along the lines represent the groups. Figure 17.23 shows that at the birds-eye view, there are two major breaks, namely in 1977 (a bar represents the first year of a new group) and 1989. For larger values of alpha, we can see that the 1980s were reasonably stable, and that more (short term) variation occurred during the 1970s.

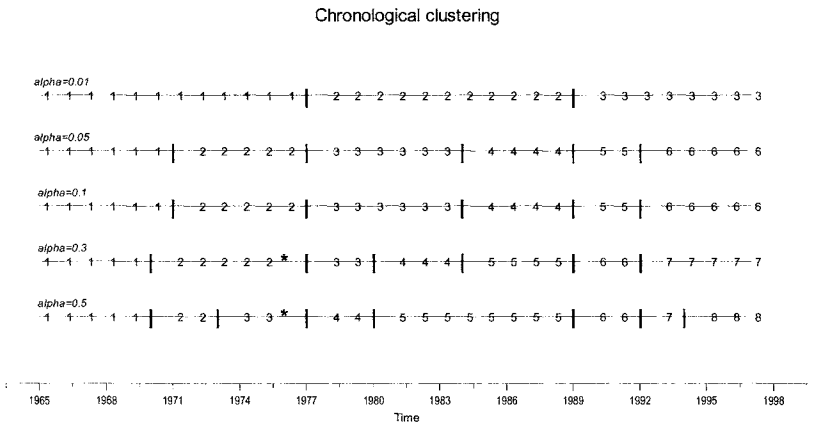

Figure 17.23. Results of chronological clustering applied to the 100 time series analysed in Hare and Mantua (2000). A vertical line corresponds to the start of a new group. Numbers refer to groups. The smallest alpha value (0.01) gives the most important breakpoints.

Legendre et al. (1985) developed a posterior test. It can be used to test if, for example of groups 1 and 3 (for alpha=0.01) belong to the same group. Formulated differently, one might ask the question whether the ecosystem changes back to its original state. If this is indeed the case, the posterior test would indicate that groups 1 and 3 were similar. The stars in the figure are so-called singletons. This is a point that does not belong to the group immediately before and after it. See Legendre et al. (1985) for a detailed interpretation of singletons.

Just as in PCoA and NMDS (Chapter 15), there is one other important point we need to discuss, namely the measure of similarity between time points (years in this case). Suppose that the time-by-variable matrix is in the following format:

	$\mathbf{Y_1}$	$\mathbf{Y_2}$	$\mathbf{Y_3}$	$\mathbf{Y_4}$
$\mathbf{T_1}$	Y_{11}	Y_{12}	Y_{13}	Y_{14}
$\mathbf{T_2}$	Y_{21}	Y_{22}	Y_{23}	Y_{24}
$\mathbf{T_3}$	Y_{31}	Y_{32}	Y_{33}	Y_{34}
$\mathbf{T_4}$	Y_{41}	Y_{42}	Y_{43}	Y_{44}
$\mathbf{T_5}$	Y_{51}	Y_{52}	Y_{53}	Y_{54}
$\mathbf{T_6}$	Y_{61}	Y_{62}	Y_{63}	Y_{64}
$\mathbf{T_7}$	Y_{71}	Y_{72}	Y_{73}	Y_{74}
$\mathbf{T_8}$	Y_{81}	Y_{82}	Y_{83}	Y_{84}
$\mathbf{T_9}$	Y_{91}	Y_{92}	Y_{93}	Y_{94}

Chronological clustering calculates the association between the rows T_1 and T_2, between the rows T_2 and T_3, etc. Legendre et al. (1985), Bell and Legendre (1987) and Legendre and Legendre (1998) used Whittaker's index of association in combination with chronological clustering. Its mathematical formulation is given by:

$$D(T_1,T_2)=0.5\times\sum_{j=1}^{p}\left|\frac{Y_{1j}}{Y_{1+}}-\frac{Y_{2j}}{Y_{2+}}\right|$$

The '+' stands for row totals and p is the number of variables (species). This index transfers a row in the table into a row of fractions (of the row total), and then it compares two rows by taking the sum of the absolute differences of the fractions. The index should only be used if the values are non-negative. If the variables are on a different scale (e.g., Y_1 goes from 0 to 1 and Y_2 from 1 to 1000), or if they are different types of variables (e.g., Y_1 is species abundance, Y_2 is the NAO index, Y_3 is temperature and Y_4 is wind speed), a sensible approach is to standardise the variables (Y_j's) first, and then use the Euclidean or absolute difference metric. Figure 17.23 was obtained by standardising the 100 time series and using the Euclidean distance function.