4

Basic Stochastic Models

4.1 Purpose

So far, we have considered two approaches for modelling time series. The first is based on an assumption that there is a fixed seasonal pattern about a trend. We can estimate the trend by local averaging of the deseasonalised data, and this is implemented by the R function `decompose`. The second approach allows the seasonal variation and trend, described in terms of a level and slope, to change over time and estimates these features by exponentially weighted averages. We used the `HoltWinters` function to demonstrate this method.

When we fit mathematical models to time series data, we refer to the discrepancies between the fitted values, calculated from the model, and the data as a *residual error series.* If our model encapsulates most of the deterministic features of the time series, our residual error series should appear to be a realisation of independent random variables from some probability distribution. However, we often find that there is some structure in the residual error series, such as consecutive errors being positively correlated, which we can use to improve our forecasts and make our simulations more realistic. We assume that our residual error series is stationary, and in Chapter 6 we introduce models for stationary time series.

Since we judge a model to be a good fit if its residual error series appears to be a realisation of independent random variables, it seems natural to build models up from a model of independent random variation, known as discrete white noise. The name ‘white noise’ was coined in an article on heat radiation published in *Nature* in April 1922, where it was used to refer to series that contained *all* frequencies in *equal* proportions, analogous to white light. The term *purely random* is sometimes used for white noise series. In §4.3 we define a fundamental non-stationary model based on discrete white noise that is called the *random walk.* It is sometimes an adequate model for financial series and is often used as a standard against which the performance of more complicated models can be assessed.

P.S.P. Cowpertwait and A.V. Metcalfe, *Introductory Time Series with R*,
Use R, DOI 10.1007/978-0-387-88698-5_4,

4.2 White noise

4.2.1 Introduction

A residual error is the difference between the observed value and the model predicted value at time t. If we suppose the model is defined for the variable y_t and $\hat{y}_t$ is the value predicted by the model, the residual error x_t is

$$x_t = y_t - \hat{y}_t \tag{4.1}$$

As the residual errors occur in time, they form a time series: $x_1, x_2, \ldots, x_n$.

In Chapter 2, we found that features of the historical series, such as the trend or seasonal variation, are reflected in the correlogram. Thus, if a model has accounted for all the serial correlation in the data, the residual series would be serially *uncorrelated*, so that a correlogram of the residual series would exhibit no obvious patterns. This ideal motivates the following definition.

4.2.2 Definition

A time series $\{w_t : t = 1, 2, \ldots, n\}$ is *discrete white noise* (DWN) if the variables $w_1, w_2, \ldots, w_n$ are *independent* and *identically* distributed with a mean of zero. This implies that the variables all have the same variance σ^2 and $\text{Cor}(w_i, w_j) = 0$ for all $i \neq j$. If, in addition, the variables also follow a normal distribution (i.e., $w_t \sim \text{N}(0, \sigma^2)$) the series is called *Gaussian* white noise.

4.2.3 Simulation in R

A fitted time series model can be used to *simulate* data. Time series simulated using a model are sometimes called *synthetic* series to distinguish them from an observed historical series.

Simulation is useful for many reasons. For example, simulation can be used to generate plausible future scenarios and to construct confidence intervals for model parameters (sometimes called *bootstrapping*). In R, simulation is usually straightforward, and most standard statistical distributions are simulated using a function that has an abbreviated name for the distribution prefixed with an ‘r’ (for ‘random’).[1] For example, `rnorm(100)` is used to simulate 100 independent standard normal variables, which is equivalent to simulating a Gaussian white noise series of length 100 (Fig. 4.1).

```
> set.seed(1)
> w <- rnorm(100)
> plot(w, type = "l")
```

[1] Other prefixes are also available to calculate properties for standard distributions; e.g., the prefix ‘d’ is used to calculate the probability (density) function. See the R help (e.g., `?dnorm`) for more details.

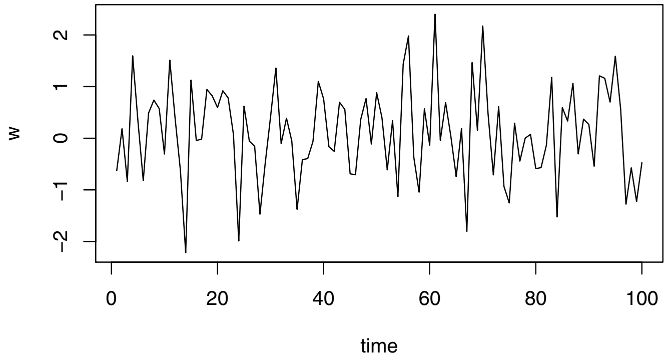


Fig. 4.1. Time plot of simulated Gaussian white noise series.

Simulation experiments in `R` can easily be repeated using the 'up' arrow on the keyboard. For this reason, it is sometimes preferable to put all the commands on one line, separated by ';', or to nest the functions; for example, a plot of a white noise series is given by `plot(rnorm(100), type="l")`.

The function `set.seed` is used to provide a starting point (or *seed*) in the simulations, thus ensuring that the simulations can be reproduced. If this function is left out, a different set of simulated data are obtained, although the underlying statistical properties remain unchanged. To see this, rerun the plot above a few times with and without `set.seed(1)`.

To illustrate by simulation how samples may differ from their underlying populations, consider the following histogram of a Gaussian white noise series. Type the following to view the plot (which is not shown in the text):

```
> x <- seq(-3,3, length = 1000)
> hist(rnorm(100), prob = T); points(x, dnorm(x), type = "l")
```

Repetitions of the last command, which can be obtained using the 'up' arrow on your keyboard, will show a range of different *sample* distributions that arise when the underlying distribution is normal. Distributions that depart from the plotted curve have arisen due to sampling variation.

4.2.4 Second-order properties and the correlogram

The second-order properties of a white noise series $\{w_t\}$ are an immediate consequence of the definition in §4.2.2. However, as they are needed so often in the derivation of the second-order properties for more complex models, we explicitly state them here:

$$\left.\begin{aligned} \mu_w &= 0 \\ \gamma_k = \mathrm{Cov}(w_t, w_{t+k}) &= \begin{cases} \sigma^2 & \text{if} \quad k = 0 \\ 0 & \text{if} \quad k \neq 0 \end{cases} \end{aligned}\right\} \tag{4.2}$$

The autocorrelation function follows as

$$\rho_k = \begin{cases} 1 & \text{if} \quad k = 0 \\ 0 & \text{if} \quad k \neq 0 \end{cases} \tag{4.3}$$

Simulated white noise data will not have autocorrelations that are *exactly* zero (when $k \neq 0$) because of sampling variation. In particular, for a simulated white noise series, it is expected that 5% of the autocorrelations will be significantly different from zero at the 5% significance level, shown as dotted lines on the correlogram. Try repeating the following command to view a range of correlograms that could arise from an underlying white noise series. A typical plot, with one statistically significant autocorrelation, occurring at lag 7, is shown in Figure 4.2.

```
> set.seed(2)
> acf(rnorm(100))
```

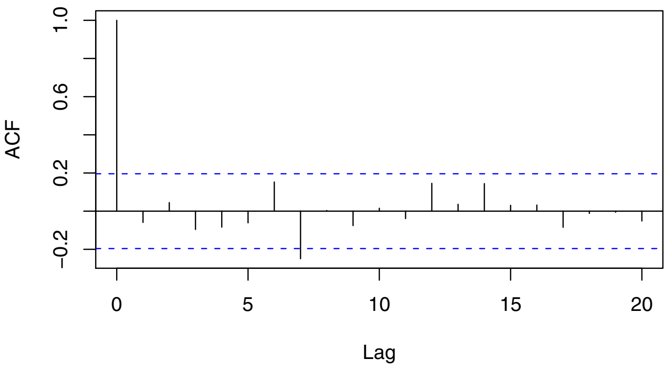


Fig. 4.2. Correlogram of a simulated white noise series. The underlying autocorrelations are all zero (except at lag 0); the statistically significant value at lag 7 is due to sampling variation.

4.2.5 Fitting a white noise model

A white noise series usually arises as a residual series after fitting an appropriate time series model. The correlogram generally provides sufficient evidence, provided the series is of a reasonable length, to support the conjecture that the residuals are well approximated by white noise.

The only parameter for a white noise series is the variance σ^2, which is estimated by the residual variance, adjusted by degrees of freedom, given in the computer output of the fitted model. If your analysis begins on data that are already approximately white noise, then only σ^2 needs to be estimated, which is readily achieved using the `var` function.

4.3 Random walks

4.3.1 Introduction

In Chapter 1, the exchange rate data were examined and found to exhibit stochastic trends. A random walk often provides a good fit to data with stochastic trends, although even better fits are usually obtained from more general model formulations, such as the ARIMA models of Chapter 7.

4.3.2 Definition

Let $\{x_t\}$ be a time series. Then $\{x_t\}$ is a random walk if

$$x_t = x_{t-1} + w_t \tag{4.4}$$

where $\{w_t\}$ is a white noise series. Substituting $x_{t-1} = x_{t-2}+w_{t-1}$ in Equation (4.4) and then substituting for x_{t-2}, followed by x_{t-3} and so on (a process known as 'back substitution') gives:

$$x_t = w_t + w_{t-1} + w_{t-2} + \dots \tag{4.5}$$

In practice, the series above will not be infinite but will start at some time $t = 1$. Hence,

$$x_t = w_1 + w_2 + \dots + w_t \tag{4.6}$$

Back substitution is used to define more complex time series models and also to derive second-order properties. The procedure occurs so frequently in the study of time series models that the following definition is needed.

4.3.3 The backward shift operator

The *backward shift* operator $\mathbf{B}$ is defined by

$$\mathbf{B}x_t = x_{t-1} \tag{4.7}$$

The backward shift operator is sometimes called the 'lag operator'. By repeatedly applying $\mathbf{B}$, it follows that

$$\mathbf{B}^n x_t = x_{t-n} \tag{4.8}$$

Using $\mathbf{B}$, Equation (4.4) can be rewritten as

$$x_t = \mathbf{B}x_t + w_t \Rightarrow (1 - \mathbf{B})x_t = w_t \Rightarrow x_t = (1 - \mathbf{B})^{-1}w_t$$

$$\Rightarrow x_t = (1 + \mathbf{B} + \mathbf{B}^2 + \ldots)w_t \Rightarrow x_t = w_t + w_{t-1} + w_{t-2} + \ldots$$

and Equation (4.5) is recovered.

4.3.4 Random walk: Second-order properties

The second-order properties of a random walk follow as

$$\left.\begin{aligned} \mu_x &= 0 \\ \gamma_k(t) &= \mathrm{Cov}(x_t, x_{t+k}) = t\sigma^2 \end{aligned}\right\} \qquad (4.9)$$

The covariance is a function of time, so the process is non-stationary. In particular, the variance is $t\sigma^2$ and so it increases without limit as t increases. It follows that a random walk is only suitable for short term predictions.

The time-varying autocorrelation function for $k > 0$ follows from Equation (4.9) as

$$\rho_k(t) = \frac{\mathrm{Cov}(x_t, x_{t+k})}{\sqrt{\mathrm{Var}(x_t)\mathrm{Var}(x_{t+k})}} = \frac{t\sigma^2}{\sqrt{t\sigma^2(t+k)\sigma^2}} = \frac{1}{\sqrt{1 + k/t}} \qquad (4.10)$$

so that, for large t with k considerably less than t, ρ_k is nearly 1. Hence, the correlogram for a random walk is characterised by positive autocorrelations that decay very slowly down from unity. This is demonstrated by simulation in §4.3.7.

4.3.5 Derivation of second-order properties*

Equation (4.6) is a finite sum of white noise terms, each with zero mean and variance σ^2. Hence, the mean of x_t is zero (Equation (4.9)). The autocovariance in Equation (4.9) can be derived using Equation (2.15) as follows:

$$\gamma_k(t) = \mathrm{Cov}(x_t, x_{t+k}) = \mathrm{Cov}\left(\sum_{i=1}^{t} w_i, \sum_{j=1}^{t+k} w_j\right) = \sum_{i=j} \mathrm{Cov}(w_i, w_j) = t\sigma^2$$

4.3.6 The difference operator

Differencing adjacent terms of a series can transform a non-stationary series to a stationary series. For example, if the series $\{x_t\}$ is a random walk, it is non-stationary. However, from Equation (4.4), the first-order differences of $\{x_t\}$ produce the stationary white noise series $\{w_t\}$ given by $x_t - x_{t-1} = w_t$.

Hence, differencing turns out to be a useful 'filtering' procedure in the study of non-stationary time series. The difference operator ∇ is defined by

$$\nabla x_t = x_t - x_{t-1} \tag{4.11}$$

Note that $\nabla x_t = (1 - \mathbf{B})x_t$, so that ∇ can be expressed in terms of the backward shift operator $\mathbf{B}$. In general, higher-order differencing can be expressed as

$$\nabla^n = (1 - \mathbf{B})^n \tag{4.12}$$

The proof of the last result is left to Exercise 7.

4.3.7 Simulation

It is often helpful to study a time series model by simulation. This enables the main features of the model to be observed in plots, so that when historical data exhibit similar features, the model may be selected as a potential candidate. The following commands can be used to simulate random walk data for x:

```
> x <- w <- rnorm(1000)
> for (t in 2:1000) x[t] <- x[t - 1] + w[t]
> plot(x, type = "l")
```

The first command above places a white noise series into w and uses this series to initialise x. The 'for' loop then generates the random walk using Equation (4.4) – the correspondence between the R code above and Equation (4.4) should be noted. The series is plotted and shown in Figure 4.3.[2]

A correlogram of the series is obtained from acf(x) and is shown in Figure 4.4 – a gradual decay in the correlations is evident in the figure, thus supporting the theoretical results in §4.3.4.

Throughout this book, we will often fit models to data that we have simulated and attempt to recover the underlying model parameters. At first sight, this might seem odd, given that the parameters are used to simulate the data so that we already know at the outset the values the parameters should take. However, the procedure is useful for a number of reasons. In particular, to be able to simulate data using a model requires that the model formulation be correctly understood. If the model is understood but incorrectly implemented, then the parameter estimates from the fitted model may deviate significantly from the underlying model values used in the simulation. Simulation can therefore help ensure that the model is both correctly understood and correctly implemented.

[2] To obtain the same simulation and plot, it is necessary to have run the previous code in §4.2.4 first, which sets the random number seed.

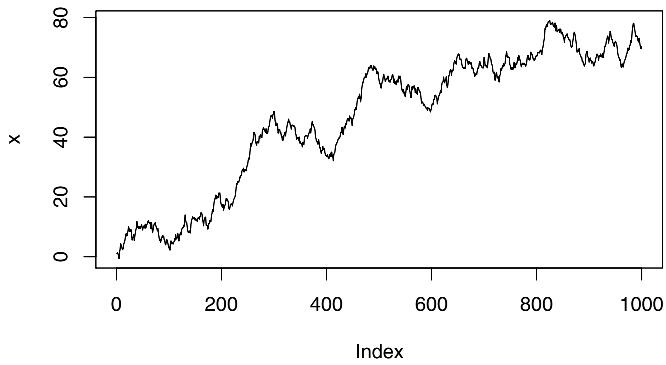


Fig. 4.3. Time plot of a simulated random walk. The series exhibits an increasing trend. However, this is purely stochastic and due to the high serial correlation.

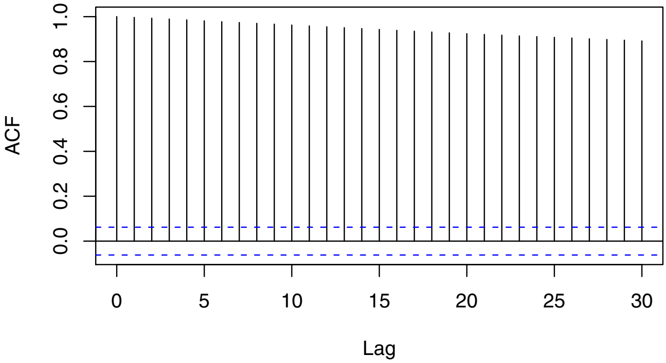


Fig. 4.4. The correlogram for the simulated random walk. A gradual decay from a high serial correlation is a notable feature of a random walk series.

4.4 Fitted models and diagnostic plots

4.4.1 Simulated random walk series

The first-order differences of a random walk are a white noise series, so the correlogram of the series of differences can be used to assess whether a given series is reasonably modelled as a random walk.

```
> acf(diff(x))
```

As can be seen in Figure 4.5, there are no obvious patterns in the correlogram, with only a couple of marginally statistically significant values. These significant values can be ignored because they are small in magnitude and about 5% of the values are expected to be statistically significant even when the underlying values are zero (§2.3). Thus, as expected, there is good evidence that the simulated series in x follows a random walk.

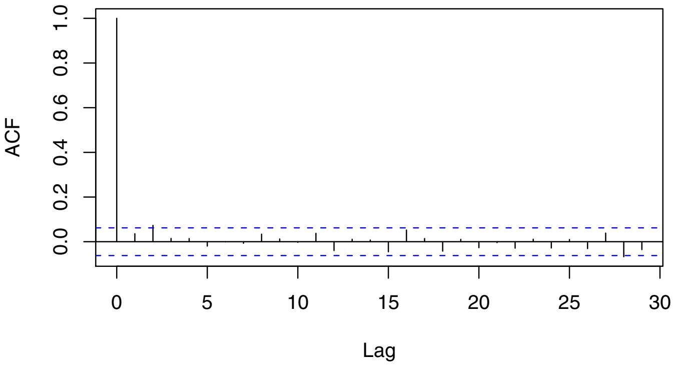


Fig. 4.5. Correlogram of differenced series. If a series follows a random walk, the differenced series will be white noise.

4.4.2 Exchange rate series

The correlogram of the first-order differences of the exchange rate data from §1.4.4 can be obtained from `acf(diff(Z.ts))` and is shown in Figure 4.6.

A significant value occurs at lag 1, suggesting that a more complex model may be needed, although the lack of any other significant values in the correlogram does suggest that the random walk provides a good approximation for the series (Fig. 4.6). An additional term can be added to the random walk model using the Holt-Winters procedure, allowing the parameter β to be non-zero but still forcing the seasonal term γ to be zero:

```
> Z.hw <- HoltWinters(Z.ts, alpha = 1, gamma = 0)
> acf(resid(Z.hw))
```

Figure 4.7 shows the correlogram of the residuals from the fitted Holt-Winters model. This correlogram is more consistent with a hypothesis that the residual series is white noise (Fig. 4.7). Using Equation (3.21), with the parameter estimates obtained from `Z.hw$alpha` and `Z.hw$beta`, the fitted model can be expressed as

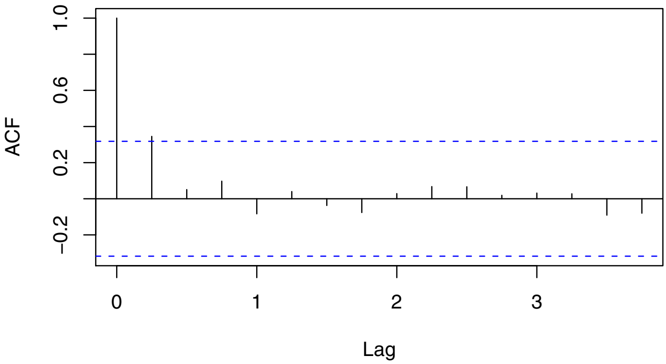


Fig. 4.6. Correlogram of first-order differences of the exchange rate series (UK pounds to NZ dollars, 1991–2000). The significant value at lag 1 indicates that an extension of the random walk model is needed for this series.

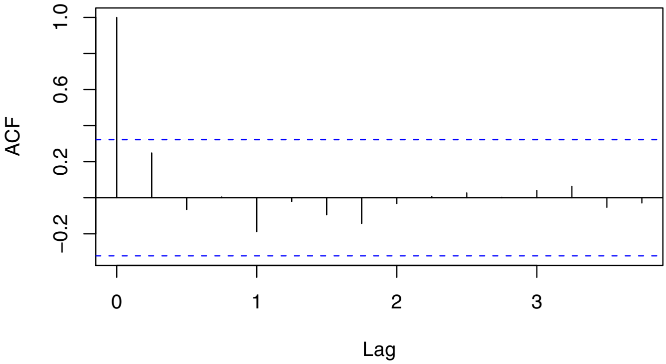


Fig. 4.7. The correlogram of the residuals from the fitted Holt-Winters model for the exchange rate series (UK pounds to NZ dollars, 1991–2000). There are no significant correlations in the residual series, so the model provides a reasonable approximation to the exchange rate data.

$$\left.\begin{aligned} x_t &= x_{t-1} + b_{t-1} + w_t \\ b_{t-1} &= 0.167(x_{t-1} - x_{t-2}) + 0.833 b_{t-2} \end{aligned}\right\} \tag{4.13}$$

where $\{w_t\}$ is white noise with zero mean.

After some algebra, Equations (4.13) can be expressed as one equation in terms of the backward shift operator:

$$(1 - 0.167\mathbf{B} + 0.167\mathbf{B}^2)(1 - \mathbf{B})x_t = w_t \tag{4.14}$$

Equation (4.14) is a special case – the *integrated autoregressive* model – within the important class of models known as ARIMA models (Chapter 7). The proof of Equation (4.14) is left to Exercise 8.

4.4.3 Random walk with drift

Company stockholders generally expect their investment to increase in value despite the volatility of financial markets. The random walk model can be adapted to allow for this by including a *drift* parameter δ.

$$x_t = x_{t-1} + \delta + w_t$$

Closing prices (US dollars) for Hewlett-Packard Company stock for 672 trading days up to June 7, 2007 are read into R and plotted (see the code below and Fig. 4.8). The lag 1 differences are calculated using `diff()` and plotted in Figure 4.9. The correlogram of the differences is in Figure 4.10, and they appear to be well modelled as white noise. The mean of the differences is 0.0399, and this is our estimate of the drift parameter. The standard deviation of the 671 differences is 0.460, and an approximate 95% confidence interval for the drift parameter is [0.004, 0.075]. Since this interval does not include 0, we have evidence of a positive drift over this period.

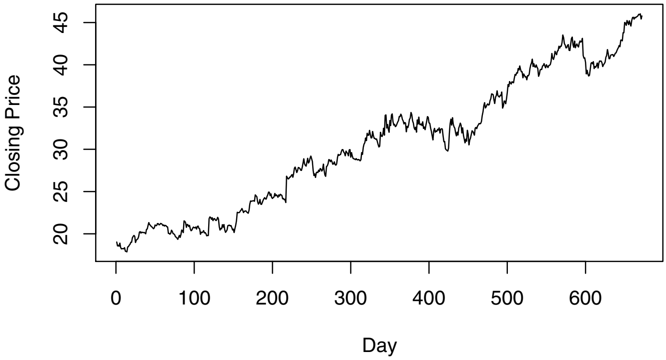


Fig. 4.8. Daily closing prices of Hewlett-Packard stock.

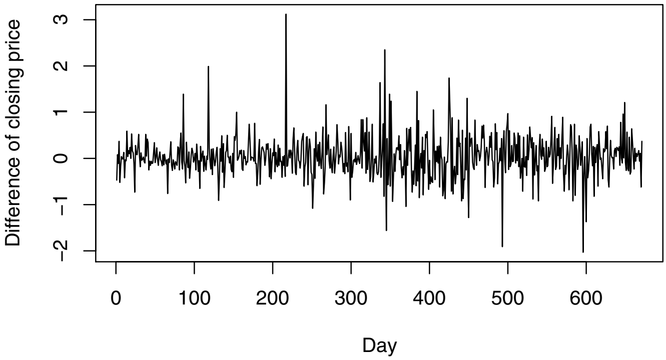


Fig. 4.9. Lag 1 differences of daily closing prices of Hewlett-Packard stock.

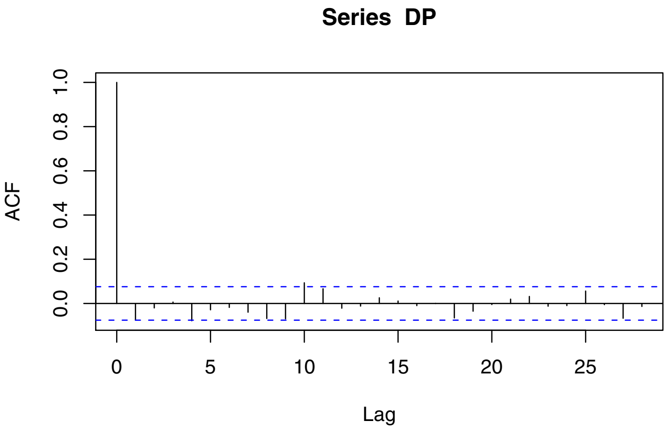


Fig. 4.10. Acf of lag 1 differences of daily closing prices of Hewlett-Packard stock.

```
> www <- "http://www.massey.ac.nz/~pscowper/ts/HP.txt"
> HP.dat <- read.table(www, header = T) ; attach(HP.dat)
> plot (as.ts(Price))
> DP <- diff(Price) ; plot (as.ts(DP)) ; acf (DP)

> mean(DP) + c(-2, 2) * sd(DP)/sqrt(length(DP))

[1] 0.004378 0.075353
```

4.5 Autoregressive models

4.5.1 Definition

The series $\{x_t\}$ is an autoregressive process of order p, abbreviated to AR(p), if

$$x_t = \alpha_1 x_{t-1} + \alpha_2 x_{t-2} + \ldots + \alpha_p x_{t-p} + w_t \tag{4.15}$$

where $\{w_t\}$ is white noise and the α_i are the model parameters with $\alpha_p \neq 0$ for an order p process. Equation (4.15) can be expressed as a polynomial of order p in terms of the backward shift operator:

$$\theta_p(\mathbf{B})x_t = (1 - \alpha_1 \mathbf{B} - \alpha_2 \mathbf{B}^2 - \ldots - \alpha_p \mathbf{B}^p)x_t = w_t \tag{4.16}$$

The following points should be noted:

(a) The random walk is the special case AR(1) with $\alpha_1 = 1$ (see Equation (4.4)).
(b) The exponential smoothing model is the special case $\alpha_i = \alpha(1-\alpha)^i$ for $i = 1, 2, \ldots$ and $p \to \infty$.
(c) The model is a regression of x_t on past terms from the same series; hence the use of the term ‘autoregressive’.
(d) A prediction at time t is given by

$$\hat{x}_t = \alpha_1 x_{t-1} + \alpha_2 x_{t-2} + \ldots + \alpha_p x_{t-p} \tag{4.17}$$

(e) The model parameters can be estimated by minimising the sum of squared errors.

4.5.2 Stationary and non-stationary AR processes

The equation $\theta_p(\mathbf{B}) = 0$, where $\mathbf{B}$ is formally treated as a number (real or complex), is called the characteristic equation. The roots of the characteristic equation (i.e., the polynomial $\theta_p(\mathbf{B})$ from Equation (4.16)) must *all* exceed unity in *absolute* value for the process to be *stationary.* Notice that the random walk has $\theta = 1 - \mathbf{B}$ with root $\mathbf{B} = 1$ and is *non-stationary.* The following four examples illustrate the procedure for determining whether an AR process is stationary or non-stationary:

1. The AR(1) model $x_t = \frac{1}{2}x_{t-1} + w_t$ is stationary because the root of $1 - \frac{1}{2}\mathbf{B} = 0$ is $\mathbf{B} = 2$, which is greater than 1.
2. The AR(2) model $x_t = x_{t-1} - \frac{1}{4}x_{t-2} + w_t$ is stationary. The proof of this result is obtained by first expressing the model in terms of the backward shift operator $\frac{1}{4}(\mathbf{B}^2 - 4\mathbf{B} + 4)x_t = w_t$; i.e., $\frac{1}{4}(\mathbf{B} - 2)^2 x_t = w_t$. The roots of the polynomial are given by solving $\theta(\mathbf{B}) = \frac{1}{4}(\mathbf{B} - 2)^2 = 0$ and are therefore obtained as $\mathbf{B} = 2$. As the roots are greater than unity this AR(2) model is stationary.

3. The model $x_t = \frac{1}{2}x_{t-1} + \frac{1}{2}x_{t-2} + w_t$ is non-stationary because one of the roots is unity. To prove this, first express the model in terms of the backward shift operator $-\frac{1}{2}(\mathbf{B}^2+\mathbf{B}-2)x_t = w_t$; i.e., $-\frac{1}{2}(\mathbf{B}-1)(\mathbf{B}+2)x_t = w_t$. The polynomial $\theta(\mathbf{B}) = -\frac{1}{2}(\mathbf{B}-1)(\mathbf{B}+2)$ has roots $\mathbf{B} = 1, -2$. As there is a *unit root* ($\mathbf{B} = 1$), the model is *non-stationary.* Note that the other root ($\mathbf{B} = -2$) exceeds unity in *absolute* value, so only the presence of the unit root makes this process non-stationary.
4. The AR(2) model $x_t = -\frac{1}{4}x_{t-2} + w_t$ is stationary because the roots of $1 + \frac{1}{4}\mathbf{B}^2 = 0$ are $\mathbf{B} = \pm 2i$, which are complex numbers with $i = \sqrt{-1}$, each having an absolute value of 2 exceeding unity.

The `R` function `polyroot` finds zeros of polynomials and can be used to find the roots of the characteristic equation to check for stationarity.

4.5.3 Second-order properties of an AR(1) model

From Equation (4.15), the AR(1) process is given by

$$x_t = \alpha x_{t-1} + w_t \tag{4.18}$$

where $\{w_t\}$ is a white noise series with mean zero and variance σ^2. It can be shown (§4.5.4) that the second-order properties follow as

$$\left.\begin{aligned} \mu_x &= 0 \\ \gamma_k &= \alpha^k \sigma^2/(1-\alpha^2) \end{aligned}\right\} \tag{4.19}$$

4.5.4 Derivation of second-order properties for an AR(1) process*

Using $\mathbf{B}$, a stable AR(1) process ($|\alpha| < 1$) can be written as

$$\left.\begin{aligned} (1-\alpha\mathbf{B})x_t &= w_t \\ \Rightarrow x_t &= (1-\alpha\mathbf{B})^{-1}w_t \\ &= w_t + \alpha w_{t-1} + \alpha^2 w_{t-2} + \ldots = \textstyle\sum_{i=0}^{\infty} \alpha^i w_{t-i} \end{aligned}\right\} \tag{4.20}$$

Hence, the mean is given by

$$E\left(x_t\right) = E\left(\sum_{i=0}^{\infty} \alpha^i w_{t-i}\right) = \sum_{i=0}^{\infty} \alpha^i E\left(w_{t-i}\right) = 0$$

and the autocovariance follows as

$$\begin{aligned} \gamma_k = \mathrm{Cov}\left(x_t, x_{t+k}\right) &= \mathrm{Cov}\left(\textstyle\sum_{i=0}^{\infty} \alpha^i w_{t-i}, \sum_{j=0}^{\infty} \alpha^j w_{t+k-j}\right) \\ &= \textstyle\sum_{j=k+i} \alpha^i \alpha^j \mathrm{Cov}\left(w_{t-i}, w_{t+k-j}\right) \\ &= \alpha^k \sigma^2 \textstyle\sum_{i=0}^{\infty} \alpha^{2i} = \alpha^k \sigma^2/(1-\alpha^2) \end{aligned}$$

using Equations (2.15) and (4.2).

4.5.5 Correlogram of an AR(1) process

From Equation (4.19), the autocorrelation function follows as

$$\rho_k = \alpha^k \qquad (k \geq 0) \tag{4.21}$$

where $|\alpha| < 1$. Thus, the correlogram decays to zero more rapidly for small α. The following example gives two correlograms for positive and negative values of α, respectively (Fig. 4.11):

```
> rho <- function(k, alpha) alpha^k
> layout(1:2)
> plot(0:10, rho(0:10, 0.7), type = "b")
> plot(0:10, rho(0:10, -0.7), type = "b")
```

Try experimenting using other values for α. For example, use a small value of α to observe a more rapid decay to zero in the correlogram.

4.5.6 Partial autocorrelation

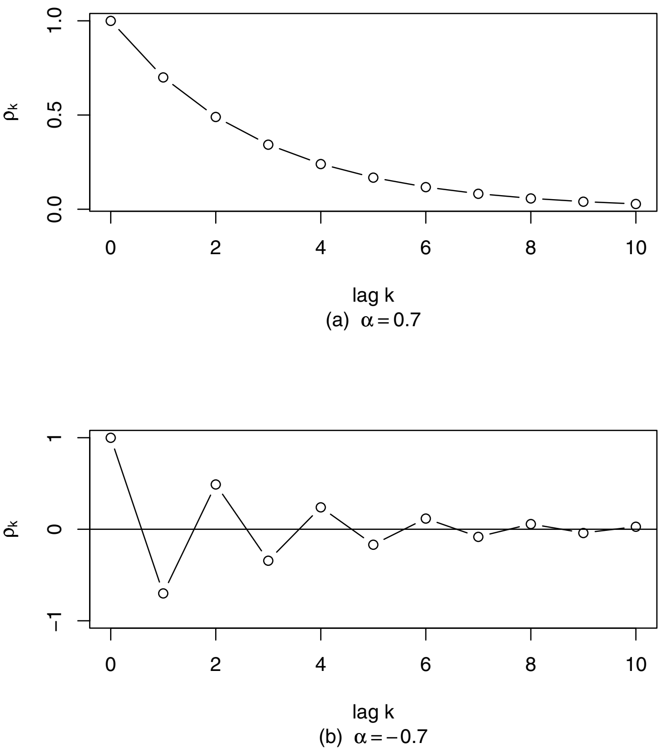
From Equation (4.21), the autocorrelations are non-zero for all lags even though in the underlying model x_t only depends on the previous value x_{t-1} (Equation (4.18)). The *partial autocorrelation* at lag k is the correlation that results after removing the effect of any correlations due to the terms at shorter lags. For example, the partial autocorrelation of an AR(1) process will be zero for all lags greater than 1. In general, the partial autocorrelation at lag k is the kth coefficient of a fitted AR(k) model; if the underlying process is AR(p), then the coefficients α_k will be zero for all $k > p$. Thus, an AR(p) process has a correlogram of partial autocorrelations that is zero after lag p. Hence, a plot of the estimated partial autocorrelations can be useful when determining the order of a suitable AR process for a time series. In R, the function pacf can be used to calculate the partial autocorrelations of a time series and produce a plot of the partial autocorrelations against lag (the 'partial correlogram').

4.5.7 Simulation

An AR(1) process can be simulated in R as follows:

```
> set.seed(1)
> x <- w <- rnorm(100)
> for (t in 2:100) x[t] <- 0.7 * x[t - 1] + w[t]
> plot(x, type = "l")
> acf(x)
> pacf(x)
```

The resulting plots of the simulated data are shown in Figure 4.12 and give one possible realisation of the model. The partial correlogram has no significant correlations except the value at lag 1, as expected (Fig. 4.12c – note that the



(a) α = 0.7

(b) α = −0.7

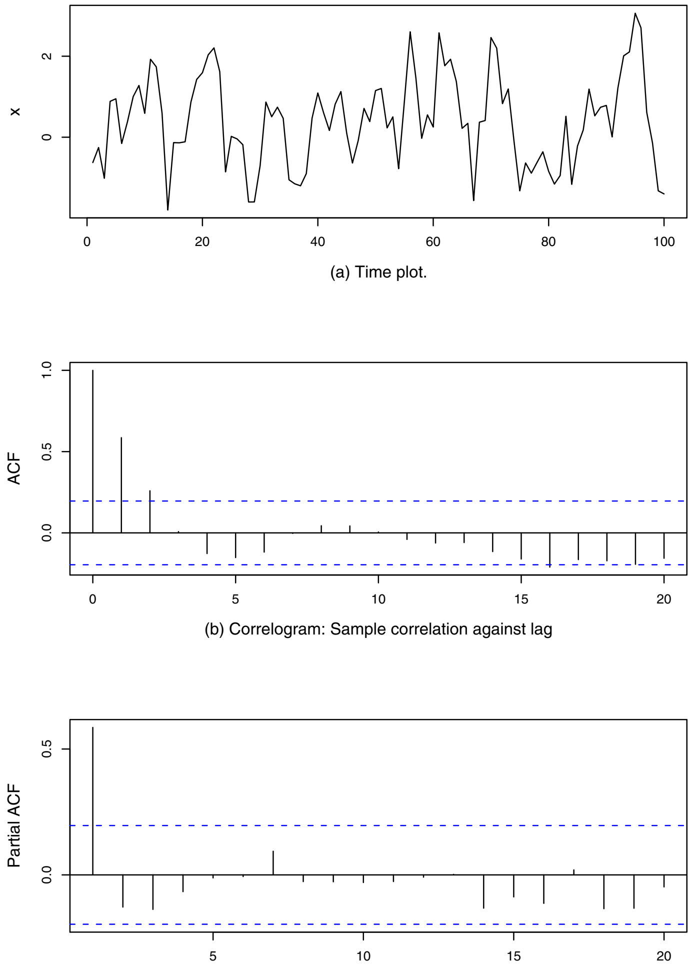
Fig. 4.11. Example correlograms for two autoregressive models: (a) $x_t = 0.7x_{t-1} + w_t$; (b) $x_t = -0.7x_{t-1} + w_t$.

pacf starts at lag 1, whilst the acf starts at lag 0). The difference between the correlogram of the underlying model (Fig. 4.11a) and the sample correlogram of the simulated series (Fig. 4.12b) shows discrepancies that have arisen due to sampling variation. Try repeating the commands above several times to obtain a range of possible sample correlograms for an AR(1) process with underlying parameter $\alpha = 0.7$. You are asked to investigate an AR(2) process in Exercise 4.

4.6 Fitted models

4.6.1 Model fitted to simulated series

An AR(p) model can be fitted to data in `R` using the `ar` function. In the code below, the autoregressive model `x.ar` is fitted to the simulated series of the last section and an approximate 95% confidence interval for the underlying



(a) Time plot.

(b) Correlogram: Sample correlation against lag

(c) Partial correlogram: Sample partial correlation against lag

Fig. 4.12. A simulated AR(1) process, $x_t = 0.7x_{t-1} + w_t$. Note that in the partial correlogram (c) only the first lag is significant, which is usually the case when the underlying process is AR(1).

parameter is given, where the (asymptotic) variance of the parameter estimate is extracted using `x.ar$asy.var`:

```
> x.ar <- ar(x, method = "mle")
> x.ar$order

[1] 1

> x.ar$ar
```

```
[1] 0.601
```

```
> x.ar$ar + c(-2, 2) * sqrt(x.ar$asy.var)
```

```
[1] 0.4404 0.7615
```

The method "mle" used in the fitting procedure above is based on maximising the likelihood function (the probability of obtaining the data given the model) with respect to the unknown parameters. The order p of the process is chosen using the Akaike Information Criterion (AIC; Akaike, 1974), which penalises models with too many parameters:

$$\text{AIC} = -2 \times \text{log-likelihood} + 2 \times \text{number of parameters} \tag{4.22}$$

In the function `ar`, the model with the smallest AIC is selected as the best-fitting AR model. Note that, in the code above, the correct order ($p = 1$) of the underlying process is recovered. The parameter estimate for the fitted AR(1) model is $\hat{\alpha} = 0.60$. Whilst this is smaller than the underlying model value of $\alpha = 0.7$, the approximate 95% confidence interval does contain the value of the model parameter as expected, giving us no reason to doubt the implementation of the model.

4.6.2 Exchange rate series: Fitted AR model

An AR(1) model is fitted to the exchange rate series, and the upper bound of the confidence interval for the parameter includes 1. This indicates that there would not be sufficient evidence to reject the hypothesis $\alpha = 1$, which is consistent with the earlier conclusion that a random walk provides a good approximation for this series. However, simulated data from models with values of $\alpha > 1$, formally included in the confidence interval below, exhibit exponentially unstable behaviour and are not credible models for the New Zealand exchange rate.

```
> Z.ar <- ar(Z.ts)
> mean(Z.ts)
```

```
[1] 2.823
```

```
> Z.ar$order
```

```
[1] 1
```

```
> Z.ar$ar
```

```
[1] 0.8903
```

```
> Z.ar$ar + c(-2, 2) * sqrt(Z.ar$asy.var)
```

```
[1] 0.7405 1.0400
```

```
> acf(Z.ar$res[-1])
```

In the code above, a "-1" is used in the vector of residuals to remove the first item from the residual series (Fig. 4.13). (For a fitted AR(1) model, the first item has no predicted value because there is no observation at $t = 0$; in general, the first p values will be 'not available' (NA) in the residual series of a fitted AR(p) model.)

By default, the mean is subtracted before the parameters are estimated, so a predicted value $\hat{z}_t$ at time t based on the output above is given by

$$\hat{z}_t = 2.8 + 0.89(z_{t-1} - 2.8) \tag{4.23}$$

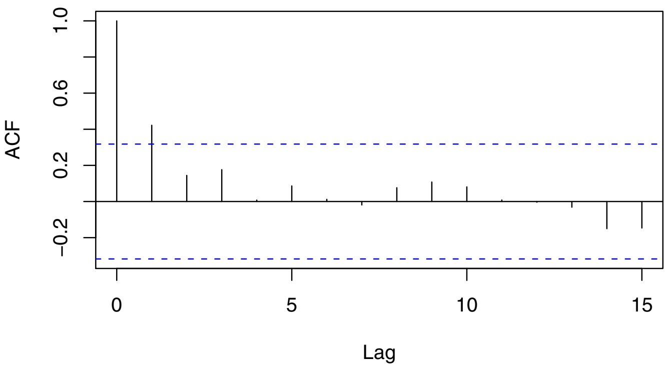


Fig. 4.13. The correlogram of residual series for the AR(1) model fitted to the exchange rate data.

4.6.3 Global temperature series: Fitted AR model

The global temperature series was introduced in §1.4.5, where it was apparent that the data exhibited an increasing trend after 1970, which may be due to the 'greenhouse effect'. Sceptics may claim that the apparent increasing trend can be dismissed as a transient stochastic phenomenon. For their claim to be consistent with the time series data, it should be possible to model the trend without the use of deterministic functions.

Consider the following AR model fitted to the mean annual temperature series:

```
> www = "http://www.massey.ac.nz/~pscowper/ts/global.dat"
> Global = scan(www)
> Global.ts = ts(Global, st = c(1856, 1), end = c(2005, 12),
     fr = 12)
```

```
> Global.ar <- ar(aggregate(Global.ts, FUN = mean), method = "mle")
> mean(aggregate(Global.ts, FUN = mean))

[1] -0.1383

> Global.ar$order

[1] 4

> Global.ar$ar

[1] 0.58762 0.01260 0.11117 0.26764

> acf(Global.ar$res[-(1:Global.ar$order)], lag = 50)
```

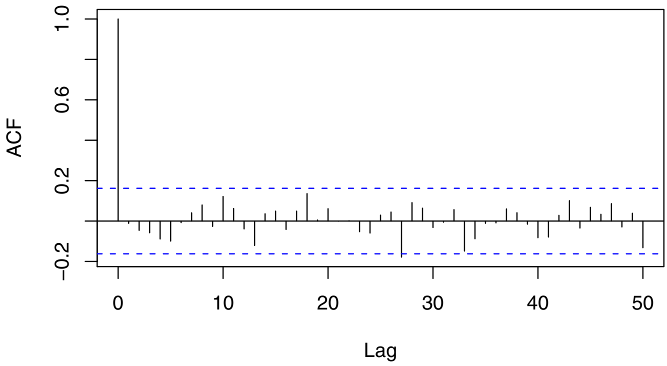


Fig. 4.14. The correlogram of the residual series for the AR(4) model fitted to the annual global temperature series. The correlogram is approximately white noise so that, in the absence of further information, a simple stochastic model can 'explain' the correlation and trends in the series.

Based on the output above a predicted mean annual temperature $\hat{x}_t$ at time t is given by

$$\begin{aligned}\hat{x}_t = & -0.14 + 0.59(x_{t-1} + 0.14) + 0.013(x_{t-2} + 0.14) \\ & +0.11(x_{t-3} + 0.14) + 0.27(x_{t-4} + 0.14)\end{aligned} \tag{4.24}$$

The correlogram of the residuals has only one (marginally) significant value at lag 27, so the underlying residual series could be white noise (Fig. 4.14). Thus the fitted AR(4) model (Equation (4.24)) provides a good fit to the data. As the AR model has no deterministic trend component, the trends in the data can be explained by serial correlation and random variation, implying that it is possible that these trends are stochastic (or could arise from a purely stochastic process). Again we emphasise that this does not imply that there is no underlying reason for the trends. If a valid scientific explanation is known, such as a link with the increased use of fossil fuels, then this information would clearly need to be included in any future forecasts of the series.

4.7 Summary of R commands

`set.seed`	sets a seed for the random number generator enabling a simulation to be reproduced
`rnorm`	simulates Gaussian white noise series
`diff`	creates a series of first-order differences
`ar`	gets the best fitting AR(p) model
`pacf`	extracts partial autocorrelations and partial correlogram
`polyroot`	extracts the roots of a polynomial
`resid`	extracts the residuals from a fitted model

4.8 Exercises

1. Simulate discrete white noise from an exponential distribution and plot the histogram and the correlogram. For example, you can use the R command `w <- rexp(1000)-1` for exponential white noise. Comment on the plots.

2. a) Simulate time series of length 100 from an AR(1) model with α equal to -0.9, -0.5, 0.5, and 0.9. Estimate the parameter of each model and make predictions for 1 to 10 steps ahead.
   b) Simulate time series of length 100 from an AR(1) model with α equal to 1.01, 1.02, and 1.05. Estimate the parameters of these models.

3. An AR(1) model with a non-zero mean μ can be expressed by either $x_t - \mu = \alpha(x_{t-1} - \mu) + w_t$ or $x_t = \alpha_0 + \alpha_1 x_{t-1} + w_t$.
   a) What is the relationship between the parameters μ and α and the parameters α_0 and α_1?
   b) Deduce a similar relationship for an AR(2) process with mean μ.

4. a) Simulate a time series of length 1000 for the following model, giving appropriate R code and placing the simulated data in a vector `x`:

$$x_t = \frac{5}{6}x_{t-1} - \frac{1}{6}x_{t-2} + w_t \tag{4.25}$$

   b) Plot the correlogram and partial correlogram for the simulated data. Comment on the plots.

c) Fit an AR model to the data in `x` giving the parameter estimates and order of the fitted AR process.

d) Construct 95% confidence intervals for the parameter estimates of the fitted model. Do the model parameters fall within the confidence intervals? Explain your results.

e) Is the model in Equation (4.25) stationary or non-stationary? Justify your answer.

f) Plot the correlogram of the residuals of the fitted model, and comment on the plot.

5. a) Show that the series $\{x_t\}$ given by $x_t = \frac{3}{2}x_{t-1} - \frac{1}{2}x_{t-2} + w_t$ is non-stationary.

b) Write down the model for $\{y_t\}$, where $y_t = \nabla x_t$. Show that $\{y_t\}$ is stationary.

c) Simulate a series of 1000 values for $\{x_t\}$, placing the simulated data in `x`, and use these simulated values to produce a series of 999 values for $\{y_t\}$, placing this series in the vector `y`.

d) Fit an AR model to `y`. Give the fitted model parameter estimates and a 95% confidence interval for the underlying model parameters based on these estimates. Compare the confidence intervals to the parameters used to simulate the data and explain the results.

e) Plot the correlogram of the residuals of the fitted model and comment.

6. a) Refit the AR(4) model of §4.6.3 to the annual mean global temperature series, and using the fitted model create a series of predicted values from $t = 2$ to the last value in the series (using Equation (4.24)). Create a residual series from the difference between the predicted value and the observed value, and verify that within machine accuracy your residual series is identical to the series extracted from the fitted model in `R`.

b) Plot a correlogram and partial correlogram for the mean annual temperature series. Comment on the plots.

c) Use the `predict` function in `R` to forecast 100 years of future values for the annual global temperature series using the fitted AR(4) model (Equation (4.24)) of §4.6.3.

d) Create a time plot of the mean annual temperature series and add the 100-year forecasts to the plot (use a different colour or symbol for the forecasts).

e) Add a line representing the overall mean global temperature. Comment on the final plot and any potential inadequacies in the fitted model.

7. Prove Equation (4.12) by mathematical induction as follows. (*i*) First, show that if Equation (4.12) holds for $n = k$, then it also holds for $n =$

$k + 1$. (*ii*) Next, show that Equation (4.12) holds for the case $n = 2$ and hence (from i) holds for all n.

8. Prove Equation (4.14). [Hint: Express the two equations in (4.13) in terms of the backward shift operator and then substitute for b_n.]