11

Multivariate Models

11.1 Purpose

Data are often collected on more than one variable. For example, in economics, daily exchange rates are available for a large range of currencies, or, in hydrological studies, both rainfall and river flow measurements may be taken at a site of interest. In Chapter 10, we considered a frequency domain approach where variables are classified as inputs or outputs to some system. In this chapter, we consider time domain models that are suitable when measurements have been made on more than one time series variable. We extend the basic autoregressive model to the vector autoregressive model, which has more than one dependent time series variable, and look at methods in `R` for fitting such models. We consider series, called cointegrated series, that share an underlying stochastic trend, and look at suitable statistical tests for detecting cointegration. Since variables measured in time often share similar properties, regression can be used to relate the variables. However, regression models of time series variables can be misleading, so we first consider this problem in more detail before moving on to suitable models for multivariate time series.

11.2 Spurious regression

It is common practice to use regression to explore the relationship between two or more variables, and we usually seek predictor variables that either directly cause the response or provide a plausible physical explanation it. For time series variables we have to be particularly careful before ascribing any causal relationship since an apparent relationship could exist due to common extraneous factors that give rise to an underlying trend or simply because both series exhibit seasonal fluctuations. For example, the Australian electricity and chocolate production series share an increasing trend (see the following code) due to an increasing Australian population, but this does not imply that changes in one variable cause changes in the other.

P.S.P. Cowpertwait and A.V. Metcalfe, *Introductory Time Series with R*, Use R, DOI 10.1007/978-0-387-88698-5_11,

```
> www <- "http://www.massey.ac.nz/~pscowper/ts/cbe.dat"
> CBE <- read.table(www, header = T)
> Elec.ts <- ts(CBE[, 3], start = 1958, freq = 12)
> Choc.ts <- ts(CBE[, 1], start = 1958, freq = 12)
> plot(as.vector(aggregate(Choc.ts)), as.vector(aggregate(Elec.ts)))
> cor(aggregate(Choc.ts), aggregate(Elec.ts))

[1] 0.958
```

The high correlation of 0.96 and the scatter plot do not imply that the electricity and chocolate production variables are causally related (Fig. 11.1). Instead, it is more plausible that the increasing Australian population accounts for the increasing trend in both series. Although we can fit a regression of one variable as a linear function of the other, with added random variation, such regression models are usually termed *spurious* because of the lack of any causal relationship. In this case, it would be far better to regress the variables on the Australian population.

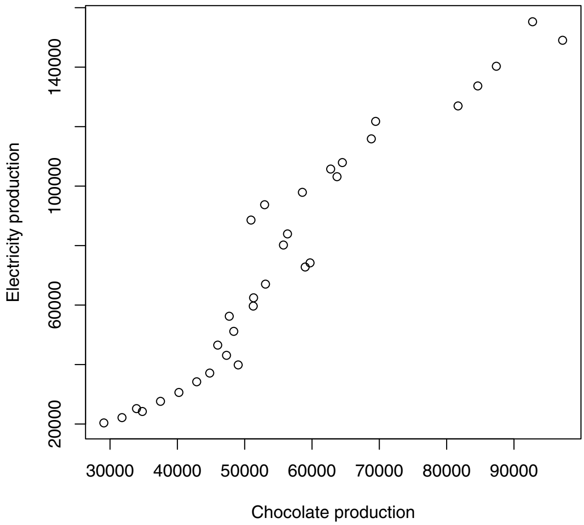

Fig. 11.1. Annual electricity and chocolate production plotted against each other.

The term spurious regression is also used when underlying stochastic trends in both series happen to be coincident, and this seems a more appropriate use of the term. Stochastic trends are a feature of an ARIMA process with a unit root (i.e., $B = 1$ is a solution of the characteristic equation). We illustrate this by simulating two independent random walks:

```
> set.seed(10); x <- rnorm(100); y <- rnorm(100)
> for(i in 2:100) {
     x[i] <- x[i-1] + rnorm(1)
     y[i] <- y[i-1] + rnorm(1) }
> plot(x, y)
> cor(x, y)
[1] 0.904
```

The code above can be repeated for different random number seeds though you will only sometimes notice spurious correlation. The seed value of 10 was selected to provide an example of a strong correlation that could have resulted by chance. The scatter plot shows how two independent time series variables might appear related when each variable is subject to stochastic trends (Fig. 11.2).

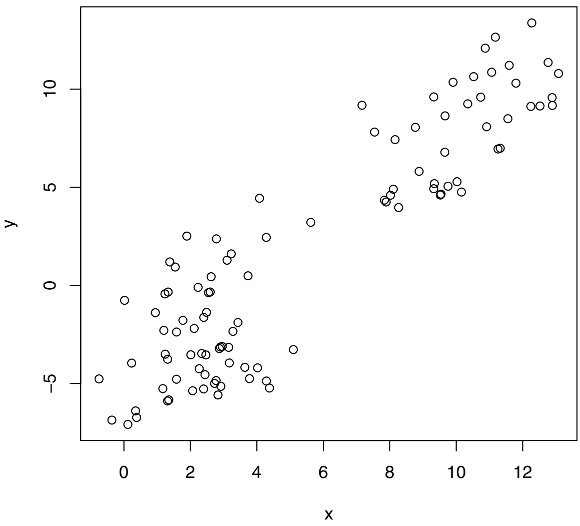

Fig. 11.2. The values of two independent simulated random walks plotted against each other. (See the code in the text.)

Stochastic trends are common in economic series, and so considerable care is required when trying to determine any relationships between the variables in multiple economic series. It may be that an underlying relationship can be justified even when the series exhibit stochastic trends because two series may be related by a common stochastic trend.

For example, the daily exchange rate series for UK pounds, the Euro, and New Zealand dollars, given for the period January 2004 to December 2007, are all per US dollar. The correlogram plots of the differenced UK and EU series indicate that both exchange rates can be well approximated by random walks (Fig. 11.3), whilst the scatter plot of the rates shows a strong linear relationship (Fig. 11.4), which is supported by a high correlation of 0.95. Since the United Kingdom is part of the European Economic Community (EEC), any change in the Euro exchange rate is likely to be apparent in the UK pound exchange rate, so there are likely to be fluctuations common to both series; in particular, the two series may share a common stochastic trend. We will discuss this phenomenon in more detail when we look at cointegration in §11.4.

```
> www <- "http://www.massey.ac.nz/~pscowper/ts/us_rates.dat"
> xrates <- read.table(www, header = T)
> xrates[1:3, ]
```

```
     UK   NZ    EU
1 0.558 1.52 0.794
2 0.553 1.49 0.789
3 0.548 1.49 0.783
```

```
> acf( diff(xrates$UK) )
> acf( diff(xrates$EU) )
> plot(xrates$UK, xrates$EU, pch = 4)
> cor(xrates$UK, xrates$EU)
[1] 0.946
```

11.3 Tests for unit roots

When investigating any relationship between two time series variables we should check whether time series models that contain unit roots are suitable. If they are, we need to decide whether or not there is a common stochastic trend. The first step is to see how well each series can be approximated as a random walk by looking at the correlogram of the differenced series (e.g., Fig. 11.3). Whilst this may work for a simple random walk, we have seen in Chapter 7 that stochastic trends are a feature of any time series model with a unit root $B = 1$ as a solution of the characteristic equation, which would include more complex ARIMA processes.

Dickey and Fuller developed a test of the null hypothesis that $\alpha = 1$ against an alternative hypothesis that $\alpha < 1$ for the model $x_t = \alpha x_{t-1} + u_t$ in which u_t is white noise. A more general test, which is known as the augmented Dickey-Fuller test (Said and Dickey, 1984), allows the differenced series u_t to be any stationary process, rather than white noise, and approximates the stationary process with an AR model. The method is implemented in `R` by the function `adf.test` within the `tseries` library. The null hypothesis of a unit root cannot be rejected for our simulated random walk `x`:

```
> library(tseries)
> adf.test(x)
```

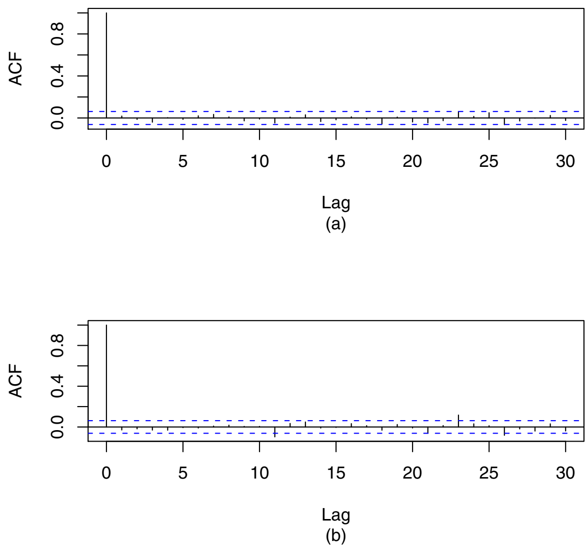

Fig. 11.3. Correlograms of the differenced exchange rate series: (a) UK rate; (b) EU rate.

```
        Augmented Dickey-Fuller Test

data:  x
Dickey-Fuller = -2.23, Lag order = 4, p-value = 0.4796
alternative hypothesis: stationary
```

This result is not surprising since we would only expect 5% of simulated random walks to provide evidence against a null hypothesis of a unit root at the 5% level. However, when we analyse physical time series rather than realisations from a known model, we should never mistake lack of evidence against a hypothesis for a demonstration that the hypothesis is true. The test result should be interpreted with careful consideration of the length of the time series, which determines the power of the test, and the general context. The null hypothesis of a unit root is favoured by economists because many financial time series are better approximated by random walks than by a stationary process, at least in the short term.

An alternative to the augmented Dickey-Fuller test, known as the Phillips-Perron test (Perron, 1988), is implemented in the `R` function `pp.test`. The distinction between the two tests is that the Phillips-Perron procedure estimates the autocorrelations in the stationary process u_t directly (using a kernel smoother) rather than assuming an AR approximation, and for this reason the Phillips-Perron test is described as semi-parametric. Critical values of the test statistic are either based on asymptotic theory or calculated from exten-

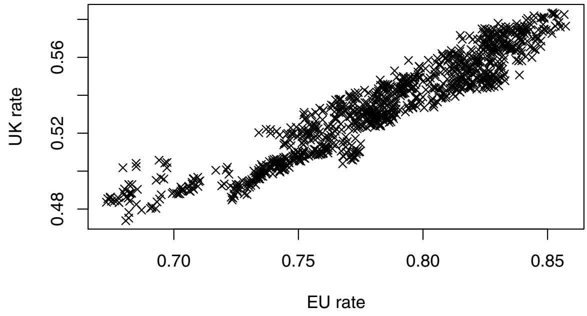

Fig. 11.4. Scatter plot of the UK and EU exchange rates. Both rates are per US dollar.

sive simulations. There is no evidence to reject the unit root hypothesis, so we conclude that the UK pound and Euro exchange rates are both likely to contain unit roots.

```
> pp.test(xrates$UK)

        Phillips-Perron Unit Root Test

data:  xrates$UK
Dickey-Fuller Z(alpha) = -10.6, Truncation lag parameter = 7,
p-value = 0.521
alternative hypothesis: stationary

> pp.test(xrates$EU)

        Phillips-Perron Unit Root Test

data:  xrates$EU
Dickey-Fuller Z(alpha) = -6.81, Truncation lag parameter = 7,
p-value = 0.7297
alternative hypothesis: stationary
```

11.4 Cointegration

11.4.1 Definition

Many multiple time series are highly correlated in time. For example, in §11.2 we found the UK pound and Euro exchange rates very highly correlated. This is explained by the similarity of the two economies relative to the US economy. Another example is the high correlation between the Australian electricity and chocolate production series, which can be reasonably attributed to an increasing Australian population rather than a causal relationship. In addition, we demonstrated that two series that are independent and contain unit roots (e.g., they follow independent random walks) can show an apparent linear relationship, due to chance similarity of the random walks over the period of the time series, and stated that such a correlation would be spurious. However, as demonstrated by the analysis of the UK pounds and Euro exchange rates, it is quite possible for two series to contain unit roots and be related. Such series are said to be *cointegrated*. In the case of the exchange rates, a stochastic trend in the US economy during a period when the European economy is relatively stable will impart a common, complementary, stochastic trend to the UK pound and Euro exchange rates. We now state the precise definition of cointegration.

> Two non-stationary time series $\{x_t\}$ and $\{y_t\}$ are cointegrated if some linear combination $ax_t + by_t$, with a and b constant, is a stationary series.

As an example consider a random walk $\{\mu_t\}$ given by $\mu_t = \mu_{t-1} + w_t$, where $\{w_t\}$ is white noise with zero mean, and two series $\{x_t\}$ and $\{y_t\}$ given by $x_t = \mu_t + w_{x,t}$ and $y_t = \mu_t + w_{y,t}$, where $\{w_{x,t}\}$ and $\{w_{y,t}\}$ are independent white noise series with zero mean. Both series are non-stationary, but their difference $\{x_t - y_t\}$ is stationary since it is a finite linear combination of independent white noise terms. Thus the linear combination of $\{x_t\}$ and $\{y_t\}$, with $a = 1$ and $b = -1$, produced a stationary series, $\{w_{x,t} - w_{y,t}\}$. Hence $\{x_t\}$ and $\{y_t\}$ are cointegrated and share the underlying stochastic trend $\{\mu_t\}$.

In `R`, two series can be tested for cointegration using the Phillips-Ouliaris test implemented in the function `po.test` within the `tseries` library. The function requires the series be given in matrix form and produces the results for a test of the null hypothesis that the two series are not cointegrated. As an example, we simulate two cointegrated series `x` and `y` that share the stochastic trend `mu` and test for cointegration using `po.test`:

```
> x <- y <- mu <- rep(0, 1000)
> for (i in 2:1000) mu[i] <- mu[i - 1] + rnorm(1)
> x <- mu + rnorm(1000)
> y <- mu + rnorm(1000)
> adf.test(x)$p.value

[1] 0.502

> adf.test(y)$p.value

[1] 0.544

> po.test(cbind(x, y))

        Phillips-Ouliaris Cointegration Test
```

```
data:  cbind(x, y)
Phillips-Ouliaris demeaned = -1020, Truncation lag parameter = 9,
p-value = 0.01
```

In the example above, the conclusion of the `adf.test` is to retain the null hypothesis that the series have unit roots. The `po.test` provides evidence that the series are cointegrated since the null hypothesis is rejected at the 1% level.

11.4.2 Exchange rate series

The code below is an analysis of the UK pound and Euro exchange rate series. The Phillips-Ouliaris test shows there is evidence that the series are cointegrated, which justifies the use of a regression model. An ARIMA model is then fitted to the residuals of the regression model. The `ar` function is used to determine the best order of an AR process. We can investigate the adequacy of our cointegrated model by using `R` to fit a more general ARIMA process to the residuals. The best-fitting ARIMA model has $d = 0$, which is consistent with the residuals being a realisation of a stationary process and hence the series being cointegrated.

```
> po.test(cbind(xrates$UK, xrates$EU))

        Phillips-Ouliaris Cointegration Test

data:  cbind(xrates$UK, xrates$EU)
Phillips-Ouliaris demeaned = -21.7, Truncation lag parameter = 10,
p-value = 0.04118

> ukeu.lm <- lm(xrates$UK ~ xrates$EU)
> ukeu.res <- resid(ukeu.lm)
> ukeu.res.ar <- ar(ukeu.res)
> ukeu.res.ar$order

[1] 3

> AIC(arima(ukeu.res, order = c(3, 0, 0)))

[1] -9886

> AIC(arima(ukeu.res, order = c(2, 0, 0)))

[1] -9886

> AIC(arima(ukeu.res, order = c(1, 0, 0)))

[1] -9880

> AIC(arima(ukeu.res, order = c(1, 1, 0)))

[1] -9876
```

Comparing the AICs for the AR(2) and AR(3) models, it is clear there is little difference and that the AR(2) model would be satisfactory. The example above also shows that the AR models provide a better fit to the residual series than the ARIMA(1, 1, 0) model, so the residual series may be treated as stationary. This supports the result of the Phillips-Ouliaris test since a linear combination of the two exchange rates, obtained from the regression model, has produced a residual series that appears to be a realisation of a stationary process.

11.5 Bivariate and multivariate white noise

Two series $\{w_{x,t}\}$ and $\{w_{y,t}\}$ are *bivariate* white noise if they are stationary and their cross-covariance $\gamma_{xy}(k) = \text{Cov}(w_{x,t}, w_{y,t+k})$ satisfies

$$\gamma_{xx}(k) = \gamma_{yy}(k) = \gamma_{xy}(k) = 0 \qquad \text{for all } k \neq 0 \tag{11.1}$$

In the equation above, $\gamma_{xx}(0) = \gamma_{yy}(0) = 1$ and $\gamma_{xy}(0)$ may be zero or non-zero. Hence, bivariate white noise series $\{w_{x,t}\}$ and $\{w_{y,t}\}$ may be regarded as white noise when considered individually but when considered as a pair may be cross-correlated at lag 0.

The definition of bivariate white noise readily extends to *multivariate* white noise. Let $\gamma_{ij}(k) = \text{Cov}(w_{i,t}, w_{j,t+k})$ be the cross-correlation between the series $\{w_{i,t}\}$ and $\{w_{j,t}\}$ $(i, j = 1, \ldots n)$. Then stationary series $\{w_{1,t}\}$, $\{w_{2,t}\}$, ..., $\{w_{n,t}\}$ are multivariate white noise if each individual series is white noise and, for each pair of series $(i \neq j)$, $\gamma_{ij}(k) = 0$ for all $k \neq 0$. In other words, multivariate white noise is a sequence of independent draws from some multivariate distribution.

Multivariate Gaussian white noise can be simulated with the `rmvnorm` function in the `mvtnorm` library. The function may take a mean and covariance matrix as a parameter input, and the dimensions of these determine the dimension of the output matrix. In the following example, the covariance matrix is 2×2, so the output variable `x` is bivariate with 1000 simulated white noise values in each of two columns. An arbitrary value of 0.8 is chosen for the correlation to illustrate the use of the function.

```
> library(mvtnorm)
> cov.mat <- matrix(c(1, 0.8, 0.8, 1), nr = 2)
> w <- rmvnorm(1000, sigma = cov.mat)
> cov(w)

      [,1]  [,2]
[1,] 1.073 0.862
[2,] 0.862 1.057

> wx <- w[, 1]
> wy <- w[, 2]
> ccf(wx, wy, main = "")
```

The `ccf` function verifies that the cross-correlations are approximately zero for all non-zero lags (Fig. 11.5). As an exercise, check that the series in each column of `x` are approximately white noise using the `acf` function.

One simple use of bivariate or multivariate white noise is in the method of *prewhitening*. Separate SARIMA models are fitted to multiple time series variables so that the residuals of the fitted models appear to be a realisation of multivariate white noise. The SARIMA models can then be used to forecast the expected values of each time series variable, and multivariate simulations can be produced by adding multivariate white noise terms to the forecasts. The method works well provided the multiple time series have no common stochastic trends and the cross-correlation structure is restricted to the error process.

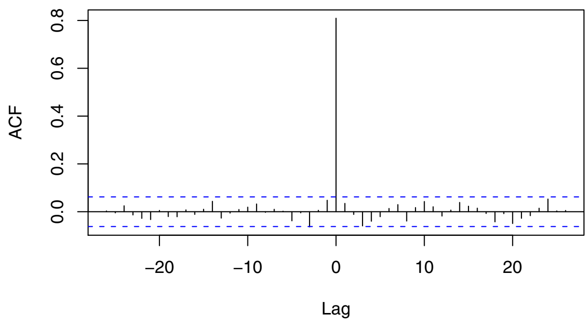

Fig. 11.5. Cross-correlation of simulated bivariate Gaussian white noise

11.6 Vector autoregressive models

Two time series, $\{x_t\}$ and $\{y_t\}$, follow a vector autoregressive process of order 1 (denoted VAR(1)) if

$$\begin{aligned} x_t &= \theta_{11} x_{t-1} + \theta_{12} y_{t-1} + w_{x,t} \\ y_t &= \theta_{21} x_{t-1} + \theta_{22} y_{t-1} + w_{y,t} \end{aligned} \qquad (11.2)$$

where $\{w_{x,t}\}$ and $\{w_{y,t}\}$ are bivariate white noise and θ_{ij} are model parameters. If the white noise sequences are defined with mean 0 and the process is stationary, both time series $\{x_t\}$ and $\{y_t\}$ have mean 0 (Exercise 1). The simplest way of incorporating a mean is to define $\{x_t\}$ and $\{y_t\}$ as deviations from mean values. Equation (11.2) can be rewritten in matrix notation as

$$\mathbf{Z}_t = \boldsymbol{\Theta} \mathbf{Z}_{t-1} + \mathbf{w}_t \qquad (11.3)$$

where

$$\mathbf{Z}_t = \begin{pmatrix} x_t \\ y_t \end{pmatrix} \quad \boldsymbol{\Theta} = \begin{pmatrix} \theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{pmatrix} \quad \mathbf{w}_t = \begin{pmatrix} w_{x,t} \\ w_{y,t} \end{pmatrix}$$

Equation (11.3) is a vector expression for an AR(1) process; i.e., the process is vector autoregressive. Using the backward shift operator, Equation (11.3) can also be written

$$(\mathbf{I} - \boldsymbol{\Theta}B)\mathbf{Z}_t = \boldsymbol{\theta}(B)\mathbf{Z}_t = \mathbf{w}_t \tag{11.4}$$

where $\boldsymbol{\theta}$ is a matrix polynomial of order 1 and $\mathbf{I}$ is the 2×2 identity matrix. A VAR(1) process can be extended to a VAR(p) process by allowing $\boldsymbol{\theta}$ to be a matrix polynomial of order p. A VAR(p) model for m time series is also defined by Equation (11.4), in which $\mathbf{I}$ is the $m \times m$ identity matrix, $\boldsymbol{\theta}$ is a polynomial of $m \times m$ matrices of parameters, $\mathbf{Z}_t$ is an $m \times 1$ matrix of time series variables, and w_t is multivariate white noise. For a VAR model, the characteristic equation is given by a determinant of a matrix. Analogous to AR models, a VAR(p) model is stationary if the roots of the determinant $|\boldsymbol{\theta}(x)|$ all exceed unity in absolute value. For the VAR(1) model, the determinant is given by

$$\begin{vmatrix} 1 - \theta_{11}x & -\theta_{12}x \\ -\theta_{21}x & 1 - \theta_{22}x \end{vmatrix} = (1 - \theta_{11}x)(1 - \theta_{22}x) - \theta_{12}\theta_{21}x^2 \tag{11.5}$$

The R functions `polyroot` and `Mod` can be used to test whether a VAR model is stationary, where the function `polyroot` just takes a vector of polynomial coefficients as an input parameter. For example, consider the VAR(1) model with parameter matrix $\boldsymbol{\Theta} = \begin{pmatrix} 0.4 & 0.3 \\ 0.2 & 0.1 \end{pmatrix}$. Then the characteristic equation is given by

$$\begin{vmatrix} 1 - 0.4x & -0.3x \\ -0.2x & 1 - 0.1x \end{vmatrix} = 1 - 0.5x - 0.02x^2 \tag{11.6}$$

The absolute value of the roots of the equation is given by

```
> Mod(polyroot(c(1, -0.5, -0.02)))

[1]  1.86 26.86
```

From this we can deduce that the VAR(1) model is stationary since both roots exceed unity in absolute value.

The parameters of a VAR(p) model can be estimated using the `ar` function in R, which selects a best-fitting order p based on the smallest AIC. Using the simulated bivariate white noise process of §11.5 and the parameters from the stationary VAR(1) model given above, a VAR(1) process is simulated below and the parameters from the simulated series estimated using `ar`.

```
> x <- y <- rep(0, 1000)
> x[1] <- wx[1]
> y[1] <- wy[1]
> for (i in 2:1000) {
```

```
    x[i] <- 0.4 * x[i - 1] + 0.3 * y[i - 1] + wx[i]
    y[i] <- 0.2 * x[i - 1] + 0.1 * y[i - 1] + wy[i]
}
> xy.ar <- ar(cbind(x, y))
> xy.ar$ar[, , ]

      x     y
x 0.399 0.321
y 0.208 0.104
```

As expected, the parameter estimates are close to the underlying model values. If the simulation is repeated many times with different realisations of the bivariate white noise, the sampling distribution of the estimators of the parameters in the model can be approximated by the histograms of the estimates together with the correlations between estimates. This is the principle used to construct bootstrap confidence intervals for model parameters when they have been estimated from time series.

The bootstrap simulation is set up using point estimates of the parameters in the model, including the variance of the white noise terms. Then time series of the same length as the historical records are simulated and the parameters estimated. A $(1-\alpha)\times 100\%$ confidence interval for a parameter is between the lower and upper $\alpha/2$ quantiles of the empirical sampling distribution of its estimates.

11.6.1 VAR model fitted to US economic series

A quarterly US economic series (1954–1987) is available within the `tseries` library. A best-fitting VAR model is fitted to the (mean-adjusted) gross national product (GNP) and real money (M1) in the following example.[1] Ordinary least squares is used to fit the model to the mean adjusted series – with `dmean` set to `TRUE` and `intercept` set to `FALSE` since the latter parameter will not be required.

```
> library(tseries)
> data(USeconomic)
> US.ar <- ar(cbind(GNP, M1), method="ols", dmean=T, intercept=F)
> US.ar$ar

, , GNP

       GNP      M1
1  1.27181 -0.0338
2 -0.00423  0.0635
3 -0.26715 -0.0286

, , M1
```

[1] *Real money* means income adjusted by inflation.

```
     GNP     M1
1  1.167  1.588
2 -0.694 -0.484
3 -0.510 -0.129

> acf(US.ar$res[-c(1:3), 1])
> acf(US.ar$res[-c(1:3), 2])
```

From the code above, we see that the best-fitting VAR model is of order 3. The correlogram of the residual series indicates that the residuals are approximately bivariate white noise, thus validating the assumptions for a VAR model (Fig. 11.6).

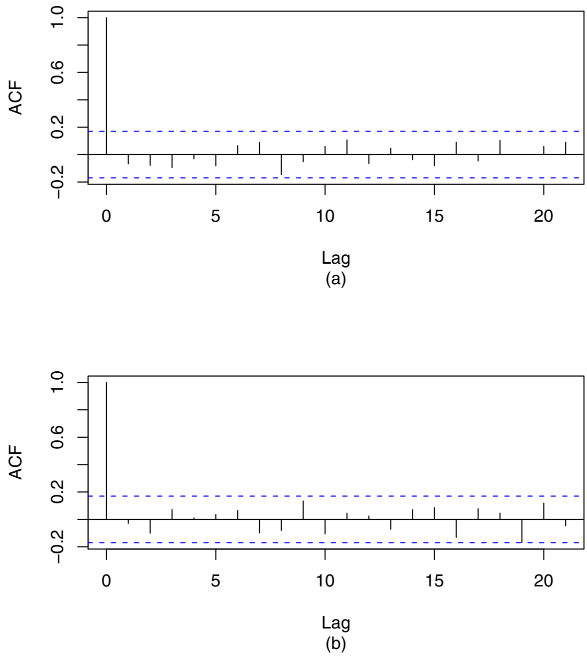

Fig. 11.6. Residual correlograms for the VAR(3) model fitted to the US economic series: (a) residuals for GNP; (b) residuals for M1.

To check for stationarity, the characteristic function can be evaluated using the determinant:

$$\left| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} 1.272 & -0.03383 \\ 1.167 & 1.588 \end{pmatrix} x - \begin{pmatrix} -0.004230 & 0.06354 \\ -0.6942 & -0.4839 \end{pmatrix} x^2 \right.$$

$$- \begin{pmatrix} -0.2672 & -0.02859 \\ -0.5103 & -0.1295 \end{pmatrix} x^3 \Bigg|$$

$$= 1 - 2.859x + 2.547x^2 - 0.3232x^3 - 0.5265x^4 + 0.1424x^5 + 0.01999x^6$$

From this it can be verified that the fitted VAR(3) model is stationary since all the roots exceed unity in absolute value:

```
> Mod( polyroot(c(1,-2.859,2.547,-0.3232, -0.5265, 0.1424, 0.01999)) )
[1] 1.025269 1.025269 1.257038 1.598381 2.482308 9.541736
```

At the time of writing, an algorithm was not available for extracting standard errors of VAR parameter estimates from an `ar` object. Estimates of these errors could be obtained using a bootstrap method or a function from another library. In the `vars` package (Pfaff, 2008), available on the `R` website, the `VAR` function can be used to estimate standard errors of fitted VAR parameters. Hence, this package was downloaded and installed and is used to extract the standard errors in the code below. Those estimates that are not significantly different from zero are removed before making a prediction for the following year. The `vars` package can also allow for any trends in the data, so we also include a trend term for the GNP series since US GNP will tend to increase with time due to an expanding population and increased productivity.

```
> library(vars)
> US.var <- VAR(cbind(GNP, M1), p = 3, type = "trend")
> coef(US.var)

$GNP
       Estimate Std. Error t value Pr(>|t|)
GNP.l1  1.07537     0.0884 12.1607 5.48e-23
M1.l1   1.03615     0.4103  2.5254 1.28e-02
GNP.l2 -0.00678     0.1328 -0.0511 9.59e-01
M1.l2  -0.30038     0.7543 -0.3982 6.91e-01
GNP.l3 -0.12724     0.0851 -1.4954 1.37e-01
M1.l3  -0.56370     0.4457 -1.2648 2.08e-01
trend   1.03503     0.4352  2.3783 1.89e-02

$M1
       Estimate Std. Error t value Pr(>|t|)
GNP.l1  -0.0439     0.0191  -2.298 2.32e-02
M1.l1    1.5923     0.0887  17.961 1.51e-36
GNP.l2   0.0616     0.0287   2.148 3.36e-02
M1.l2   -0.4891     0.1630  -3.001 3.25e-03
GNP.l3  -0.0175     0.0184  -0.954 3.42e-01
M1.l3   -0.1041     0.0963  -1.081 2.82e-01
trend    0.0116     0.0940   0.123 9.02e-01

> US.var <- VAR(cbind(GNP, M1), p = 2, type = "trend")
> coef(US.var)
```

```
$GNP
        Estimate Std. Error t value Pr(>|t|)
GNP.l1     1.141     0.0845   13.51 1.83e-26
M1.l1      1.330     0.3391    3.92 1.41e-04
GNP.l2    -0.200     0.0823   -2.43 1.67e-02
M1.l2     -1.157     0.3488   -3.32 1.19e-03
trend      1.032     0.4230    2.44 1.61e-02

$M1
        Estimate Std. Error t value Pr(>|t|)
GNP.l1 -0.03372     0.0181 -1.8623 6.48e-02
M1.l1   1.64898     0.0727 22.6877 7.33e-47
GNP.l2  0.03419     0.0176  1.9384 5.48e-02
M1.l2  -0.65016     0.0748 -8.6978 1.35e-14
trend   0.00654     0.0906  0.0722 9.43e-01

> acf(resid(US.var)[, 1])
> acf(resid(US.var)[, 2])
```

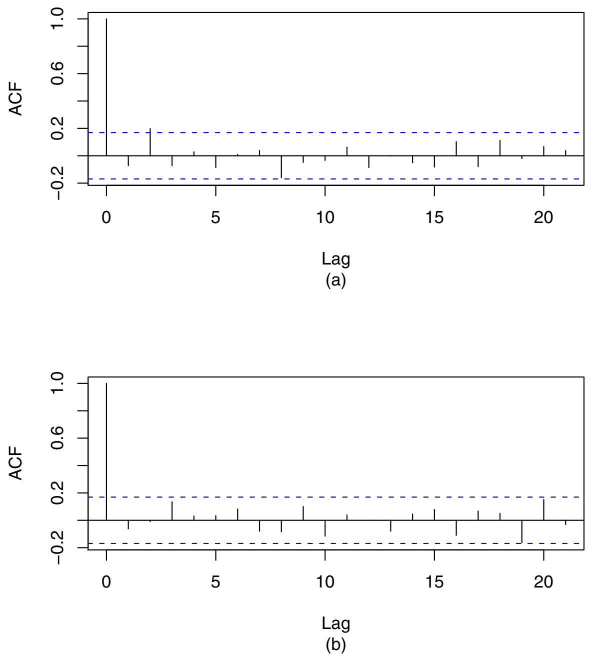

Fig. 11.7. Residual correlograms for the VAR(2) model fitted to the US economic series: (a) residuals for GNP; (b) residuals for M1.

Below we give the predicted values for the next year of the series, which are then added to a time series plot for each variable (Fig. 11.8).

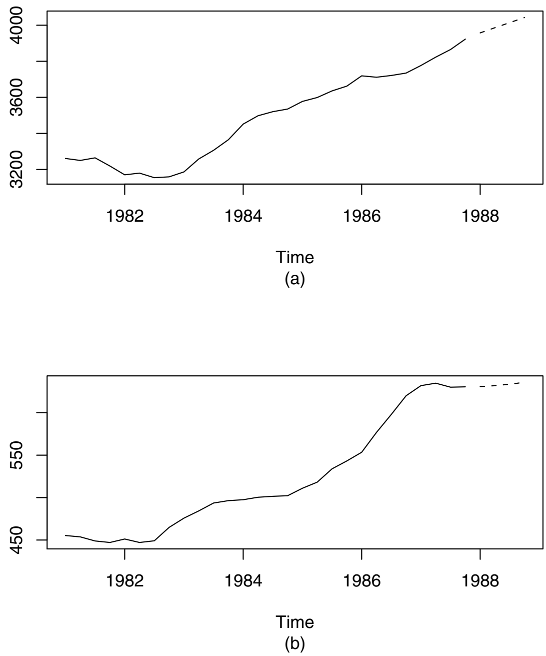

Fig. 11.8. US economic series: (a) time plot for GNP (from 1981) with added predicted values (dotted) for the next year; (b) time plot for M1 (from 1981) with added predicted values (dotted) for the next year.

```
> US.pred <- predict(US.var, n.ahead = 4)
> US.pred

$GNP
     fcst lower upper    CI
[1,] 3958  3911  4004  46.2
[2,] 3986  3914  4059  72.6
[3,] 4014  3921  4107  93.0
[4,] 4043  3933  4153 109.9

$M1
     fcst lower upper   CI
[1,]  631   621   641  9.9
[2,]  632   613   651 19.0
[3,]  634   606   661 27.5
[4,]  636   601   671 35.1
```

```
> GNP.pred <- ts(US.pred$fcst$GNP[, 1], st = 1988, fr = 4)
> M1.pred <- ts(US.pred$fcst$M1[, 1], st = 1988, fr = 4)
> ts.plot(cbind(window(GNP, start = 1981), GNP.pred), lty = 1:2)
> ts.plot(cbind(window(M1, start = 1981), M1.pred), lty = 1:2)
```

11.7 Summary of R commands

`adf.test`	Dickey-Fuller test for unit roots
`pp.test`	Phillips-Perron test for unit roots
`rmvnorm`	multivariate white noise simulation
`po.test`	Phillips-Ouliaris cointegration test
`ar`	Fits the VAR model based on the smallest AIC
`VAR`	Fits the VAR model based on least squares (`vars` package required)

11.8 Exercises

1. Show that if a VAR(1) process driven by white noise with mean 0, as defined in Equation 11.5, is stationary, then it has a mean of 0. Deduce that if a VAR(p) process driven by white noise with mean 0 is stationary, then it has a mean of 0. [Hint: Take expected values of both sides of Equation 11.5 and explain why the inverse of $\mathbf{I} - \boldsymbol{\Theta}$ exists.]

2. For what values of a is the model below stationary?

$$\begin{aligned} x_t &= 0.9x_{t-1} + ay_{t-1} + w_{x,t} \\ y_t &= ax_{t-1} + 0.9y_{t-1} + w_{y,t} \end{aligned}$$

3. This question uses the data in `stockmarket.dat`, which contains stock market data for seven cities for the period January 6, 1986 to December 31, 1997. Download the data via the book website and put the data into a variable in R.
   a) Use an appropriate statistical test to test whether the London and/or the New York series have unit roots. Does the evidence from the statistical tests suggest the series are stationary or non-stationary?
   b) Let $\{x_t\}$ represent the London series (Lond) and $\{y_t\}$ the New York series (NY). Fit the following VAR(1) model, giving a summary output containing the fitted parameters and any appropriate statistical tests:

$$\begin{aligned} x_t &= a_0 + a_1x_{t-1} + a_2y_{t-1} + w_{x,t} \\ y_t &= b_0 + b_1x_{t-1} + b_2y_{t-1} + w_{y,t} \end{aligned}$$

c) Which series influences the other the most? Why might this happen?
d) Test the London and New York series for cointegration.
e) Fit the model below, giving a summary of the model parameters and any appropriate statistical tests.

$$x_t = a_0 + a_1 y_t + w_t$$

f) Test the residual series for the previous fitted model for unit roots. Does this support or contradict the result in part (d)? Explain your answer.

4. a) Using the `VAR` function in the `vars` package, fit a multivariate VAR model to the four economic variables in the Canadian data (which can be loaded from within the `vars` package with the command `data(Canada)`).
   b) Using the fitted VAR model, make predictions for the next year. Add these predictions to a time series plot of each variable.

5. a) Fit an ARIMA(1, 1, 0)(1, 1, 1)$_{12}$ model to the logarithm of the electricity production series. Verify that the residuals are approximately white noise.
   b) Fit the same model as in (a) to the logarithm of the chocolate production series. Again, verify that the residuals are approximately white noise.
   c) Plot the cross-correlogram of the residuals of the two fitted ARIMA models, and verify that the lag 0 correlation is significantly different from zero. Give a possible reason why this may happen.
   d) Forecast values for the next month for each series, and add a simulated bivariate white noise term to each forecast. This gives one possible realisation. Repeat the process ten times to give ten possible future scenarios for the next month's production for each series.